Modelling protein-protein interaction networks

Luis Eduardo Ospina-Forero

Linacre College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy in Statistics

Trinity to Michaelmas term 2017

Abstract

Proteins, the main motors of the cell, are in charge of performing a diverse array of biological functions. They rarely perform those functions alone, but generally work as groups of proteins that through a complex array of interactions perform a single biological function. These complex interactions between different proteins are often analysed via network theory, where a protein-protein interaction (PPI) network is created considering each protein as a node and each of their interactions as edges.

Different approaches from the perspective of network analysis have been proposed to describe, analyse, and predict PPI networks. Some methods focus on the use of network summary statistics, community detection, random graph models, and machine learning procedures. However, despite the large effort invested in PPI network research, current models fail to describe well the structure of PPI networks. Small overrepresented subgraphs, which have been thought as the building blocks of networks, have been shown to be important patterns in gene regulatory networks, and there is evidence that suggests they may be evolutionarily conserved across the PPI networks of different organisms. Hence, a first step to better understand the structure of protein-protein interaction networks, is to describe how the local structure of these networks, accounted by the occurrence of small connected subgraphs, is created.

We approach this problem in two stages. In the first stage, we provide a framework to statistically assess if a random graph model can describe the occurrence of different small connected subgraphs observed in PPI networks. Then, by applying this framework we find that state-of-the-art network comparison methods based on subgraph counts struggle at finding similarities between networks that have different numbers of nodes or edges. Hence, in joint work with Dr. Anatol Wegner, Dr. Robert Gaunt, Professor Gesine Reinert, and Professor Charlotte M. Deane, we propose a novel network comparison method, *NetEmd*, that tackles this problem indirectly by proposing a method that is invariant to translations and rescalings of subgraph count distributions, and which is better able to detect similarities across networks with different number of nodes or edges.

In the second stage, we use NetEmd, along with three other state-of-the-art network comparison methods, to test the ability of several random graph models to describe the occurrence of subgraphs counts in the PPI networks of six organisms, and in multiple smaller sections of these networks. We find that the overall occurrence of small connected subgraphs could potentially be described by two network generation mechanisms operating in complementary sections of the PPI networks. In addition we find that cellular compartment-specific PPI networks can be potentially described by a single model that captures, with only two parameters, both, the common properties between the different cellular compartment networks, and their individual structural features.

Acknowledgements

I would like to thank both of my supervisors Gesine Reinert and Charlotte Deane for their guidance throughout my DPhil, and the patience they had regarding my writing. To my transfer and confirmation examiner Caleb Webber, for helping me obtain a better global view of the project and suggesting different working directions such as the study of cellular compartment networks.

Thanks are also due to Jan Boylan and Beverly Lane, for helping me find peace of mind while dealing with forms. Thanks to Alistair, Nick, Jin and, particularly Bernhard, Michelle, Luke and Ishita, for the fun social activities that helped me preserve my work-life balance. To Malte Lucken for the entertaining and instructive discussions we had in an almost daily basis.

I would also like to thank my grandmother Lola, my aunt Aura Leonor, and my mother Marcela for their immense support. Without them I wouldn't have been able to reach this stage.

Lastly, I would like to thank Alejandra, my bride to be, for all the smiles she gave me, which kept me sane throughout these years.

To all of you, thanks.

Contents

List of Figures

List of Tables

Introduction

Most functions in biological systems cannot be attributed to single molecules but rather to the complex interaction between several entities. Proteins are the primary functional molecules in biological systems and they are involved in a diverse range of functions (De Las Rivas and Fontanillo, 2010). These functions are thought to be structured in a modular fashion, where the modules are composed of groups of proteins that perform a single biological function through a complex array of interactions (Rives and Galitski, 2003; Pinkert et al., 2010; Lewis et al., 2012). Interactions between proteins can be represented as networks, where each protein is a node and each of their interactions is an edge.

Over the past decade the amount of protein interaction data has increased rapidly due to the appearance of high throughput techniques such as Yeast-two-Hybrid (Y2H) and Tandem affinity purification with mass spectrometry (TAP-MS) (Keskin et al., 2016; De Las Rivas and Fontanillo, 2010). However, despite the rapid increase in protein interaction data, the protein-protein interaction (PPI) networks of all major model organisms are still incomplete (Stumpf et al., 2008; Rajagopala et al., 2014), and contain a large proportion of false positive and false negative interactions. Protein interaction data is also biased, as it has been shown that different methods to detect protein-protein interactions tend to detect them more often in some specific cellular compartments rather than uniformly across the cell (Deane et al., 2002; Keskin et al., 2016; Mehla et al., 2017).

Despite the problems found in the data, PPI networks are used to investigate many biological problems (e.g. Zoraghi and Reiner, 2013; Sarajlić et al., 2013; Noh et al.,

2013; West et al., 2013). One particular problem of interest, and the one concerning this dissertation, is to understand the underlying mechanisms that shape the way in which PPI networks are structured. Due to the shared evolutionary history between all organisms, it is thought that there could be a common network structure present in their respective PPI networks, where individual differences can be explained at least partly by a stochastic component in the network generation process (Pereira-Leal et al., 2007; Vázquez et al., 2003). Based on this idea, past studies have proposed several possible models for PPI networks that aim to generate networks that represent the typical structure of PPI networks (e.g. Gibson and Goldberg, 2011; Vázquez et al., 2003; Pržulj et al., 2004). However, no model for PPI networks that is able to generate networks with the overall same structure of PPI networks has yet been found (Ali et al., 2014; Rito et al., 2012).

A common model that can recreate the expected structure of a PPI network is of considerable interest (Barabási and Oltvai, 2004; Ali et al., 2014; Rito et al., 2012); such a model could be used as a quality control mechanism in the detection of erroneously reported protein-protein interactions, as it would provide the baseline from which to judge if the reported PPI network deviates from the typical structure of a PPI network. A model that reproduces the expected structure could also aid in the reconstruction of phylogenetic trees, as it could provide a common point of reference for all networks to compare to irrespectively of their size (number of nodes) (Ali et al., 2014). This model could also aid in the identification of small scale elements within the PPI networks. For example, it could be used to detect interaction patterns between groups of proteins that are over/under represented within a single organism, or interaction patterns that are conserved across different organisms (Ali et al., 2014; Pereira-Leal et al., 2007; Wuchty et al., 2003; Milo et al., 2002).

Finding a model for PPI networks is a challenging task (Barabási and Oltvai, 2004). Beyond the errors and experimental biases in the data, the major challenge to find a general model for PPI networks is the large heterogeneity in their structure. PPI networks often contain densely connected groups of nodes, a large proportion of transitive interactions (interactions a-b and b-c that imply interaction a-c), yet an

overall small number of edges across the whole network.

In this dissertation we aim to model the local structure of PPI networks. The local structure of a network can be studied by the occurrence of small connected subgraphs (subgraph counts), which in addition to being proposed as essential building blocks of networks, could also be conserved across the PPI networks of different organisms (Wuchty et al., 2003; Milo et al., 2002; Pereira-Leal et al., 2007). Hence, in this dissertation focus on the occurrence of small connected subgraphs.

In order to test whether a random graph model is a suitable model for the local structure of PPI networks, there are three ingredients. Firstly, the parameters of the model need to be estimated. Secondly, a network comparison statistic S is needed, where S is thought to capture the common local network structure between networks obtained from the same network generation mechanism, even when the networks vary in their number of nodes and edges. Thirdly, a statistical framework that can assess the model fit to the observed PPI network, using the statistic S, is required. In this dissertation we provide such a framework and carry out the testing.

Given the overall heterogeneity in PPI networks, instead of finding a model for the whole PPI network, it could be better to consider smaller sections of the network which can be extracted via additional biological information, such as the location of proteins within the cell. This approach may be simpler as it might imply modelling sections of the network that contain a more homogeneous structure individually. In this dissertation we also pursue this avenue, which turns out to be fairly successful.

Thesis overview

We start in Chapter 1 by describing protein-protein interaction data, how a protein-protein interaction is detected and where this data can be found. We also provide common network terminology and definitions as well as description of the random graph models used in this dissertation to model PPI networks.

Then in Chapters 2 to 5 we approach the modelling problem by splitting it into two stages. In the first stage, composed of Chapters 2 and 3, we showed how we can statistically assess if a random graph model can describe the occurrence of different

small connected subgraphs observed in protein-protein interaction networks. Then, in the second stage, composed of Chapters 4 and 5, we tested the ability of seven random graph models to describe the occurrence of subgraph counts in the whole PPI network of different organisms and in smaller sections of these networks. In detail:

In Chapter 2, we described different network comparison methods based on subgraph counts and a baseline comparison method based on network alignment (matching). We proposed a statistical test to assess if an observed network can be considered as a realisation of a random graph model, based on a network comparison statistic. We used this test to illustrate differences between the different network comparison statistics. Here we found that the network comparison statistics based on subgraph counts were not invariant to changes in the number of nodes and edges of the networks being compared.

Chapter 3, which was the result of joint work with Dr. Anatol Wegner, Dr. Robert Gaunt, Professor Gesine Reinert and Professor Charlotte M. Deane, we proposed a network comparison statistic, *NetEmd*, which can detect similarities across networks with different number of nodes and edges more reliably than the other network comparison methods based on subgraph counts used in Chapter 2.

In Chapter 4 we used the methods from the previous two Chapters to assess if the PPI networks of six organisms, or some of the local neighbourhoods in their networks, can be considered as networks, or neighbourhoods, generated by different random graph models. Due to the different types of errors introduced by the two main types of experimental methods to detect protein-protein interactions, binary and co-complex methods, here we also considered PPI networks composed of interactions reported only by binary methods, and only by co-complex methods. We found that the co-complex Yeast PPI network exhibited a block structure that was able to be captured by the Erdös-Rényi Mixture Graph (ERMG) model, and that the subgraph counts of PPI networks of several organisms formed by binary experiments could often be described by a single duplication divergence (DD) model with the same two parameter values. We also found evidence suggesting that different neighbourhoods of proteins in the PPI networks could be formed via

complementary generation mechanisms.

In Chapter 5, we further tested the ability of the DD model to capture the structure of binary and co-complex PPI networks corresponding to different cellular compartments in Yeast and Human. The DD model has two parameters, one that relates to the the divergence between node duplicates (q), and one which relates to the propensity of node duplicates to interact (p). We found that the DD model is able to describe the structure present in the PPI networks of several cellular compartments. All cellular compartment networks exhibited a similar divergence parameter q, which suggested that the heterogeneity in the structure of different cellular compartment networks can be explained by their differences in the propensity of interaction between node duplicates, which is controlled by the parameter p. Based on results found in Chapter 4 we provided a first exploration of a model for the whole PPI network that combined the DD model, and features of the ERMG model. However, despite the additional information used by our proposed model, we found that it does not outperform the results of a single DD model.

In Chapter 6 we discuss and summarise our findings.

In this dissertation we assessed random graph models with respect to their ability to represent the local structure of PPI networks. We provide a statistical framework to assess the fit of a random graph model to PPI networks. With this framework, differences between network comparison statistics can be easily found. We provided an extensive comparison of state-of-the-art network comparison methods based on subgraph counts, finding that they all were sensitive to changes in the number of nodes and edges. We have also provided a new network comparison methodology, NetEmd, which outperforms existing network comparison methods in their ability to detect similarities between networks of different number of nodes and edges. Based on these tools, we tested several models, some of which were previously suggested for PPI networks. We found that only the ERMG model and the DD model were not rejected as models that are able to describe the overall occurrence of small subgraphs for some PPI networks. However, as we did not observe a complete agreement among the different network comparison methods,

we considered analysing smaller sections of the PPI networks given by protein neighbourhoods and cellular compartments networks. From our analysis of the protein neighbourhoods we found that the occurrence of small subgraphs could potentially be generated by two network generation mechanisms (ERMG and DD models), each describing protein neighbourhoods in complementary regions of the edge-density. In our analysis of the cellular compartments, we found that the DD model was able to describe the occurrence of small subgraphs in several cellular compartment networks, and that there is evidence to suggest there is a common universal rate of divergence (q) between node duplicates across all organisms and cellular compartments.

Chapter 1

Background

Proteins, the main motors of the cell, are in charge of performing a diverse array of biological functions. They rarely perform those functions alone, but generally work as groups of proteins that through a complex array of interactions perform a single biological function (De Las Rivas and Fontanillo, 2010). These complex interactions between different proteins are often analysed via network theory, where a protein-protein interaction (PPI) network is created considering each protein as a node and each of their interactions as edges.

Different approaches from the perspective of network theory have been proposed to analyse, describe and predict PPI networks including: network summary statistics, clustering methods, random graph models, and machine learning approaches. Despite the large interest in PPI networks, current models insufficiently capture their complexity (Winterbach et al., 2013; Ali et al., 2014).

Small overrepresented subgraphs were shown to be important patterns in gene regulatory networks; and there is evidence that they may be evolutionarily preserved in PPI networks (Wuchty et al., 2003; Shen-Orr et al., 2002; Milo et al., 2002). Hence, a first step to better understand the structure of PPI networks, is to describe how the local structure of these networks is created.

In this chapter we describe the type of protein interaction data that we use, the network theory terminology and the methods frequently used throughout this dissertation.

1.1 Protein-protein interaction networks

In this dissertation we focused on protein-protein interaction (PPI), networks. We considered a protein-protein interaction as the physical contact (binding) between a pair of proteins in a particular biological context. The physical binding between proteins is not static, not all interactions occur at the same time or in the same place (e.g. different cell types, or cell cycles, etc.). However, due to a lack of experimental data that accounts for such factors, standard procedures to analyse PPI networks consider all reported protein-protein interactions simultaneously and without direction (e.g. Ali et al., 2014; Lewis et al., 2012; Shao et al., 2013; Vázquez et al., 2003). In this dissertation we used the same simplification.

We considered a PPI network as a simple undirected network with node set V given by the different proteins and edge set E given by the interactions between proteins. Figure 1.1 (a) gives an example of a pair of proteins interacting (binding), and (b) a graphical representation of a PPI network of Yeast.

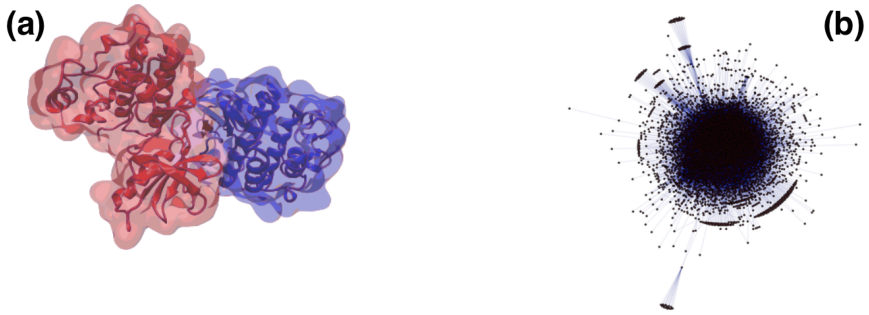


Figure 1.1: (a) Representation of the physical binding between a pair of proteins. (b) A simple undirected protein-protein interaction of Yeast with 3383 proteins (nodes) and 11161 protein-protein interactions (edges). Data obtained from BioGRID, dataset version v.3.2.100 downloaded in July 2013.

1.1.1 Detection of protein interactions

PPIs are obtained by a variety of experimental methods which can be classified into two main technology types; binary methods, which measure physical interactions between pairs of proteins and, co-complex methods which measure physical interactions among groups of proteins. Table 1.1 lists some of the binary and co-complex methods used to detect protein interactions.

Binary methods	Co-complex methods
Yeast-Two-Hybrid	**Affinity Capture-MS**
PCA	Affinity Capture-Luminescence
Reconstituted Complex	Affinity Capture-Western
FRET	Co-fractionation
Co-crystal Structure	Co-purification
Far Western	Affinity Capture-RNA
Biochemical Activity	

Table 1.1: Binary methods consider experiments that test the presence of an interaction in a one-to-one fashion, e.g. Yeast-Two-Hybrid. Co-complex methods, on the other hand, report one-to-many interactions. We based this classification on the BioGRID description of these experimental methods, which can be found in the BioGRID webpage `https://wiki.thebiogrid.org/doku.php/experimental_systems` (Stark et al., 2011), and which we accessed in November 2015.

The most commonly used techniques for large scale experiments are yeast two-hybrid (Y2H) and tandem affinity purifications with mass spectrometry (TAP-MS), for binary and co-complex methods, respectively (De Las Rivas and Fontanillo, 2010). The Y2H binary method measures the direct interaction between two proteins A and B. The idea behind this method is the use of a third protein C that triggers a signal (e.g. fluorescence) when the proteins interact. This protein (C) is split into two fragments and the one is physically bound to protein A while the other is bound to protein B. Thus, if protein A and B interact physically, the two fragments of C are close enough to trigger the signal (Mehla et al., 2017).

Co-complex methods work differently, capturing both, direct and indirect interactions; they can be understood as "fishing" methods. A bait protein is used to "fish out" a group of prey proteins; some of these proteins could bind to the prey proteins instead of directly binding to the bait protein. Co-complex experiments cannot distinguish direct from indirect interactions, hence, an algorithm (e.g. spoke or matrix model) is used to try to minimise the number of false physical interactions (false-positives) inferred (De Las Rivas and Fontanillo, 2010). The main difference between the spoke and matrix models is that the spoke model considers only interactions between the bait protein and the 'fished' proteins, whilst the matrix model also considers all possible interactions between the 'fished' proteins. Figure 1.2, reproduced from De Las Rivas and Fontanillo (2010), illustrates the ideal working of Y2H and TAP-MS techniques.

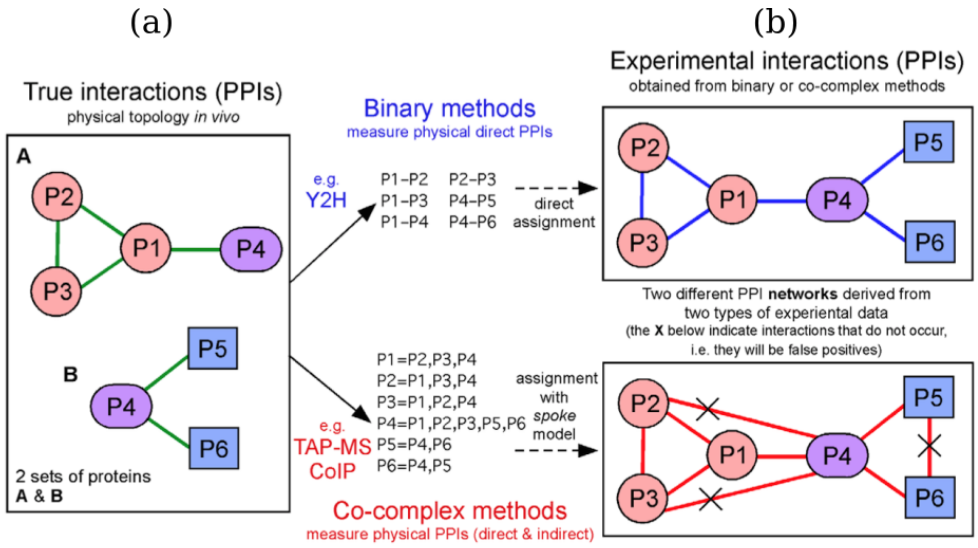


Figure 1.2: (a) True PPI network composed of two sets of proteins (A and B). (b) Resulting PPI network from a binary and co-complex experiments. Co-complex experiments do not distinguish direct from indirect interactions, leading to false physical interactions in the inferred PPI network. This figure, reproduced from De Las Rivas and Fontanillo (2010), only illustrates the ideal workings of the binary and co-complex methods. If the matrix model was used in this example all red edges would be considered as interactions, hence increasing the number of false positives.

In real life applications, both binary and co-complex methods will report error or bias (Deane et al., 2002). The main cause for **error** in binary experiments, Y2H in particular, is the larger rate of false-negative (FN) interactions when compared to co-complex experiments. However, co-complex techniques, are prone to report indirect interactions which do not account for physical interactions, thus increasing their rate of false-positives (FP) (Mehla et al., 2017; Keskin et al., 2016; Wodak et al., 2013). In general, both binary and co-complex techniques come with large false-positive and large false-negative rates (Keskin et al., 2016; Huang and Bader, 2009; Hart et al., 2006). For example Hart et al. (2006) estimated that over 50% of the reported interactions in Y2H experiments are false positives, although more recent estimates for Y2H studies have decreased (Keskin et al., 2016; Mehla et al., 2017); for example Huang and Bader (2009) obtained false positive estimates of 26-44%. For tandem-affinity purification methods, 35% of interactions are thought as false positives (Ali et al., 2010). Regarding false negatives, estimates range from 28%-51% for Y2H methods (Huang and Bader, 2009) and around 15% for co-complex methods (Ali et al., 2010).

In terms of **bias**, Wodak et al. (2013) inspected the PPI networks of Yeast, Fly, Human and others, using curated databases and in particular the BioGRID database, and concluded that Y2H techniques preferentially detect interactions between nuclear proteins involved in cell cycle, cell division or stress response; and TAP-MS interactions are enriched with proteins involved in translation, transcription and chromosome organisation. Comparing the experimental techniques to each other, Wodak et al. (2013) concluded: Y2H methods exhibited the least bias, protein complementation assays (PCA) exhibited the most bias and TAP-MS methods had intermediate levels of bias.

Lastly, and in addition to errors present in PPI networks, it should also be noted that most PPI networks are believed to be incomplete. However, there is no certainty about how complete the data is, as in addition to false positives and false negatives present, estimates of the total number of interactions in different PPI networks largely vary. For example, Stumpf et al. (2008), based on the Database of Interacting Proteins (Xenarios et al., 2000), (DIP), estimated that the total number of protein interactions in Yeast was 25229 and in Human 672918, however, Hart et al. (2006) estimated a total number of interactions of 52500 for Yeast and 225000 for Human. Currently, the total number of interactions reported in the BioGRID database (Stark et al., 2006) (January 2017) for Yeast and Human are 85586 and 216800, respectively.

In order to improve the quality of PPI data, studies have proposed "high quality" (HC) datasets by modelling the error using restrictions, or validations, of the interactions present in the raw PPI datasets. However, these procedures may lead to HC datasets that carry larger problems than the ones present in the initial raw PPI datasets (Hakes et al., 2008). HC datasets are often obtained via thresholding over a confidence score, or by taking all interactions that have been reported by at least 2 different studies, or by posing other filters over the initial set of interactions. These approaches lead to HC datasets that, despite having a higher proportion of true positive interactions, increase certain types of biases such as, scientific interest bias or experimental bias. For example, if a "HC" dataset is created by taking interactions reported in at least two different studies, the experimental

bias could be increased, as interactions detected by easily applicable experimental techniques, (which may preferentially detect interactions in particular cellular locations or stages of the cell cycle), will tend to be overrepresented. Hakes et al. (2008) showed that even for one of the most high-quality datasets at the time of his study, that dataset led to a network structure that was not necessarily representative of the true underlying biological network. In his study Hakes et al. (2008) showed that HC datasets led to changes in global summary statistics that contradict some expected biological behaviours of the PPI networks. For example, instead of hub proteins being more likely to interact with other hub proteins, they would tend to be isolated from one another and to form sparser disconnected structures.

Despite all the difficulties present in PPI data, there are still ways to extract relevant biological information, although viewed with caution. This can be achieved by controlling the type of study where the PPI data is obtained from (e.g. Yeast-Two-Hybrid experiments) (Hakes et al., 2008; Venkatesan et al., 2009), and by replicating results across different PPI datasets. Hence in this dissertation we carried out an in depth analysis of the PPI networks of six organisms as well as using PPI networks formed by interactions obtained from binary experiments alone, co-complex experiments alone and with both binary-&-co-complex experiments. We also analysed PPI networks with three different time stamps, 2013, 2015 and 2017.

In addition, due to the modular structure present in PPI networks (Hartwell et al., 1999; Barabási and Oltvai, 2004), where single biological functions are performed by a complex array of interactions in a group of proteins (module), we were also interested in analysing the network structure present in network modules. However, identification of network modules is an ongoing problem (Lewis et al., 2010), and therefore we were not able to analyse the module structure directly, as individual modules could not be identified. Instead, we considered other groups of proteins that, similarly to functional modules, described different local regions of the PPI networks, such as protein neighbourhoods formed by a protein and all other proteins that interact directly with it or through another protein that interacts directly with it. We also considered groups of proteins that are located at specific cellu-

lar locations such as nucleus and membrane, as several modules are composed of proteins that are colocalized (Rives and Galitski, 2003; Barabási and Oltvai, 2004; Tarassov et al., 2008).

1.1.2 Protein-protein interactions databases

There are a wide variety of databases available that contain protein-protein interaction data. Some of these databases contain curated protein-protein interaction data that can be specific to a particular organism, e.g. the HPRD *Human proteome database* (Keshava Prasad et al., 2009), or that contain information for several organisms such as, IntAct (Hermjakob et al., 2004), iRefIndex(Razick et al., 2008), HINT *High-quality INTeractomes* (Das and Yu, 2012), DIP *database of interacting proteins* (Xenarios et al., 2000) and BioGRID *general repository for interaction datasets* (Stark et al., 2006). Other databases report additional information regarding the confidence of the reported interaction, such as STRING *search tool for recurring instances of neighbouring genes* (Snel et al., 2000).
In this dissertation we have focused on protein-protein interactions extracted from the BioGRID database, as BioGRID provides continuous updates and maintains a large and well documented archive of previous data releases. In addition, PPI networks obtained from this database are often used for PPI network analysis (e.g. Hayes et al., 2013; Malod-Dognin and Pržulj, 2015; Shao et al., 2015), which allows for a straightforward comparison of results to other studies.

1.2 Networks

Given a set of nodes $V = \{v_1, v_2, ..., v_{n_v}\}$ and a set of edges $E = \{e_1, e_2, ..., e_{n_e}\}$ representing connections among nodes and composed of tuples of nodes (so that $E \subset V \times V$), the structure $G := (V, E)$ is called a graph or network (used interchangeably in this dissertation) with node set V and edge set E. This network structure can be represented by an adjacency matrix A with $n_v := |V|$ number of rows and columns, and where $A_{ij} := 1$, if $(v_i, v_j) \in E$ and $A_{ij} := 0$ otherwise. Throughout all our analyses we used undirected graphs, that is $(v_i, v_j) = (v_j, v_i)$

and (Kolaczyk, 2009, ch. 2)

$$A_{ij} = 1 \iff A_{ji} = 1, \forall i, j \in \{1, 2, ..., n_v\}.$$

We also assumed that edges are unweighted and that there is no more than one edge between any two nodes. Additionally we removed all self-loops from the networks considered, i.e. $A_{ii} = 0$, $\forall i \in \{1, 2, ..., n_v\}$.

Nodes can be connected through *paths*. A *path* between two nodes v_1, $v_{n+1} \in V$ is an alternating sequence of nodes and edges $\{v_1, e_1, v_2, e_2, ..., e_n, v_{n+1}\}$ where the end points of edge e_i are v_i and v_{i+1}, $i \in \{1, 2, ..., n\}$ (Kolaczyk, 2009, ch. 2).

The *shortest path length*, $l(.,.)$, between two nodes v_i and v_j, $i, j \in \{1, 2, ..., n_v\}$, is the minimum number of edges found across all paths that start in v_i and end in v_j. The *shortest path length* is also known as the *graph distance* or *geodesic distance*. We say nodes v_i, v_j are at k steps, or k hops apart if $l(v_i, v_j) = k$. In particular, all nodes v_j such that $l(v_i, v_j) = 1$ are called the nearest *neighbours* of v_i. The generalisation to k-step neighbours follows naturally (Kolaczyk, 2009, ch. 2).

Networks which do not have paths connecting every node to every other node are called *disconnected* (Kolaczyk, 2009). In this dissertation, we referred to the disconnected parts of a network as *components*. The largest connected component of a network (LCC) is the connected component with the largest proportion of nodes from the network.

1.3 Summary statistics of networks

Several network summary statistics have been proposed as means to understand the global characteristics of networks. Here we list some well-known network summary statistics (Kolaczyk, 2009, ch. 4) and which are often mentioned throughout this dissertation. For a graph $G = (V, E)$:

- n_v **and** n_e**:** The number of nodes and the number of edges in the network.
- **Global clustering coefficient or transitivity (**C**):** The global clustering coefficient is the ratio of triangles to connected triplets in the graph. Here,

a *triangle* ($\triangle$) is any complete subgraph on three nodes, and a *triplet* ($\wedge$) is a connected subgraph with three nodes and two edges. In adjacency matrix notation

$$C := \frac{\sum_{l\neq m, m\neq k, k\neq l} A_{kl}A_{lm}A_{mk}}{\sum_{l\neq m, m\neq k, k\neq l} A_{kl}A_{mk}},$$

where $v_k, v_l, v_m \in V$. The global clustering coefficient is always between 0 and 1.

- **Local clustering coefficient ($\bar{C}$):** For a given node v_k the quantity $C(v_k)$, also known as the Watts-Strogatz local clustering coefficient of node v_k, is the ratio of the 'number of edges between neighbours of node v_k' and the 'number of all possible edges between its neighbours'. The average of all $C(v_k)$ is known as the Watts-Strogatz local clustering coefficient:

$$\bar{C} := \sum_{k=1}^{n_v} C(v_k)/n_v \quad \text{where} \quad C(v_k) := \frac{\sum_{l\neq m, m\neq k, k\neq m} A_{kl}A_{lm}A_{mk}}{\sum_{l\neq m, m\neq k, k\neq m} A_{kl}A_{mk}}.$$

- **Edge-density (ρ):** The edge-density or network density is the proportion of edges present in the network over all possible edges:

$$\rho := \frac{2n_e}{n_v(n_v-1)} = \frac{n_e}{\binom{n_v}{2}}.$$

Hence, the density is always between 0 and 1.

- **Average shortest path length (L):** If $G = (V, E)$ is a connected graph with n_v nodes, the average shortest path length of G is:

$$L := \frac{\sum_{k\neq m} l(v_k, v_m)}{n_v(n_v-1)} = \frac{\sum_{k<m} l(v_k, v_m)}{n_v(n_v-1)/2}.$$

- **Diameter ($Diam$):** The diameter of a graph is the maximum of the shortest

path lengths:

$$Diam := \max_{\substack{k \neq m \\ k,m \in \{1,2,...,n_v\}}} \{l(v_k, v_m)\}.$$

- **Average degree ($\bar{d}$), degree sequence and degree distribution:** The node degree (d_k) of a node is

$$d_k = degree(v_k) = \sum_{j \neq k} A_{kj},$$

and the average degree of a network is

$$\bar{d} = \sum_k d_k / n_v.$$

The sequence $(d_1, d_2, ..., d_{n_v})$ is the *degree sequence* of G. A sequence $(d_1, d_2, ..., d_{n_v})$ is called a *feasible* degree sequence if there exist a graph which has $(d_1, d_2, ..., d_{n_v})$ as degree sequence. The *degree distribution* of a graph is then a function that provides the frequency, or the number of times, a given degree appears in the degree sequence of a graph G.

1.4 Random graph models and parameter estimation

In order to assess significance of patterns in networks, it is useful to consider different network generation mechanisms. In this section we described some well-known random graph models for undirected graphs that have been used to model PPI networks (e.g. Daudin et al., 2008; Hayes et al., 2013; Shao et al., 2013). We also describe a static power law model as it has been suggested that this type of model is suitable to capture some properties of PPI networks (Kolaczyk, 2009).

Along with the description of each model we give the parameter estimation used in this dissertation.

Erdös-Rényi random graph models

An Erdös-Rényi model $ER(n_v, p)$ (Gilbert, 1959) is a random graph on n_v nodes where edges are present independently at random with probability p. The number of edges is thus a random variable that follows a $Binomial(\binom{n_v}{2}, p)$ distribution. The ER model is one of the few models for which the maximum likelihood estimator of the parameters, in this case p, is easy to calculate. It suffices to notice that all edges of the random graph from a sample of i.i.d. random variables such that $X_{ij} \sim Bernoulli(p)$ $i, j = 1, 2, ..., n_v$ and $i < j$. Thus leading to the maximum likelihood estimate of p, the sample mean:

$$\hat{p} = \frac{n'_e}{\binom{n_v}{2}} = \frac{2n'_e}{n_v(n_v - 1)},$$

where n'_e is the observed number of edges in the graph.

Erdös-Rényi Mixture Graphs

The ERMG model, also known as stochastic block model, proposed by Holland et al. (1983), is based on a classification of nodes into Q classes with prior probabilities $\{\alpha_1, \alpha_2, ..., \alpha_Q\}$, with $\sum_{k=1}^{Q} \alpha_k = 1$. Let $Z_i = (Z_{i1}, Z_{i2}, ..., Z_{iQ})$ be the (often) unobservable class membership of node v_i, where $Z_{iq} = 1$ if node v_i is in class q, and 0 otherwise. Lastly, if edges $\{X_{ij}\}$, $i, j \in \{1, 2, ..., n_v\}$, are assumed conditionally independent given the class memberships; then

$$X_{ij}|\{v_i \in q,\, v_j \in l\} = X_{ij}|\{Z_{iq} = 1,\, Z_{jl} = 1\} \sim Bernoulli(\pi_{ql}); \quad \text{for } i \neq j,$$
$$X_{ii} = 0, \text{ otherwise.}$$

As we are considering undirected networks we take $\pi_{ql} = \pi_{lq}$. Here the parameters of this model are: Q, α_q and π_{ql}, where $q, l \in \{1, 2, ..., Q\}$.

The likelihood of the model is then (Daudin et al., 2008; Mariadassou et al., 2010)

$$p(X, Z|\pi, \alpha) = p(X|Z, \pi) \times p(Z|\alpha) = \prod_{i<j} \prod_{q,l} (\pi_{ql}^{X_{ij}} (1 - \pi_{ql})^{1 - X_{ij}})^{Z_{iq} Z_{jl}} \times \prod_{i=1}^{n_v} \prod_{q=1}^{Q} \alpha_q^{Z_{iq}}.$$

Parameter estimation of ERMG graphs often use approximate maximum likelihood strategies like variational approaches (e.g. Mariadassou et al., 2010; Latouche et al., 2012), where a lower bound of the data log-likelihood $\log p(X|\pi,\alpha) = \log \sum_Z p(Z,X|\pi,\alpha)$ is used. Based on Jordan et al. (1999) and Dempster et al. (1977), Mariadassou et al. (2010) used the following lower bound:

$$J(R_X,\pi,\alpha) = H(R_X) + \sum_Z R_X(Z)\log p(X,Z|\alpha,\pi),$$

where $H(.)$ denotes the entropy of a distribution and $R_X(.)$ is a probability distribution of Z, such that

$$R_X(Z) = \prod_i M(Z_i; 1, \tau_{i1}, \tau_{i2}, ..., \tau_{iQ}),$$

where $\tau_{iq} \in [0,1]$, $q = 1,2,...,Q$, and $\sum_q \tau_{iq} = 1$ are the variational parameters (to be optimised).

For such R_X, $J(R_X,\pi,\alpha)$ can be written as

$$\begin{aligned}
J(R_X,\pi,\alpha) &= H(R_X) + \sum_Z R_X(Z)\log p(X,Z|\alpha,\pi) \\
&= H(R_X) + \sum_{i<j}\sum_{q,l} E_{R_X}[Z_{iq}Z_{jl}]\log(\pi_{ql}^{x_{ij}}(1-\pi_{ql})^{1-x_{ij}}) \\
&+ \sum_{i=1}^{n_v}\sum_{q=1}^{Q} E_{R_X}[Z_{iq}]\log\alpha_q \\
&= H(R_X) + \sum_{i<j}\sum_{q,l} \tau_{iq}\tau_{jl}\log(\pi_{ql}^{x_{ij}}(1-\pi_{ql})^{1-x_{ij}}) \\
&+ \sum_{i=1}^{n_v}\sum_{q=1}^{Q} \tau_{iq}\log\alpha_q \\
&= -\sum_i\sum_q \tau_{iq}\log\tau_{iq} + \sum_{i<j}\sum_{q,l} \tau_{iq}\tau_{jl}\log(\pi_{ql}^{x_{ij}}(1-\pi_{ql})^{1-x_{ij}}) \\
&+ \sum_{i=1}^{n_v}\sum_{q=1}^{Q} \tau_{iq}\log\alpha_q
\end{aligned}$$

The variational estimates of π, α are then obtained by optimising $J(R_X,\pi,\alpha)$ with

respect to π, α and $\tau_{i,j}$, $i, j = 1, 2, ..., Q$. This methodology is implemented in the R packages *blockmodel* and *mixer* from which we obtained the parameter estimates for the ERMG model. Detailed reviews of variational methods can be found in Jordan et al. (1999), or Beal (2003).

The configuration model (Molloy and Reed (1995) construction)

Given a feasible degree sequence $(d_1, d_2, ..., d_{n_v})$, node v_i is assigned the value d_i, $i = 1, 2, ..., n_v$. A network is constructed as follows:

Each node i is assigned d_i 'stubs', then the stubs across all nodes are connected uniformly at random.

This construction leads to a network that has the pre-established degree sequence. Here self-loops and multiple edges are possible.

A sequence of numbers $(d_1, d_2, ..., d_{n_v})$ is called a *feasible* degree sequence if there is a graph which has $(d_1, d_2, ..., d_{n_v})$ as degree sequence.

The degree sequence of the observed graph is often taken as the estimate of the degree sequence of the model (e.g. Rito et al., 2010; Pržulj, 2007; Pržulj et al., 2004). In this dissertation we followed the same approach.

The Chung-Lu model

A special case of the stochastic block model (Holland et al., 1983) but with self-loops and when prior knowledge about node classes and edge probabilities is available was proposed by Chung and Lu (2002). This model is also known as the Sticky model (Pržulj and Higham, 2006) and the Newman-Girvan model (Newman, 2006). Given a sequence $\{d_1, d_2, ..., d_{n_v}\}$ such that $max_i\, d_i^2 < \sum_k d_k$ and $d_i > 0$, $\forall i$, the model assigns a weight (θ_i) to each node, where $\theta_i \propto d_i$. Then, an edge is placed between any two nodes with probability proportional to the product of their weights. Explicitly, the model is:

$$P((i, j) \in E) = \theta_i \theta_j,$$

where

$$\theta_i := \frac{d_i}{\sqrt{\sum_j d_j}}.$$

Then

$$\mathbb{E}\left(degree(i)\right) = d_i, \qquad i = 1, 2, ..., n_v.$$

By construction, the model is based on a sequence $\{d_1, d_2, ..., d_{n_v}\}$, which is the 'only' parameter of this model. As in the configuration model, the estimation of this parameter is usually made by taking the degree sequence of the observed graph as the parameter for this model. This estimate leads to random graphs with an expected degree sequence equal to the one of the observed graph (e.g. Newman, 2010; Pržulj and Higham, 2006; Shao et al., 2013; Hayes et al., 2013).

Goh's power law model

The following static power law model creates a network with a degree distribution that follows a power law distribution. Proposed by Goh et al. (2001), this model is described by the number of edges, n_e, and the exponent of the power law distribution, $\alpha = (1+\gamma)/\gamma$, where $\gamma \in (0, 1)$.

For a fixed $\gamma \in (0, 1)$, the generation of the graph starts by considering n_v nodes $v_1, v_2, ..., v_i, ..., v_{n_v}$. Two nodes, v_i, v_j are selected with probabilities $p_i / \sum_{k=1}^{n_v} p_k$ and $p_j / \sum_{k=1}^{n_v} p_k$ respectively, where $p_i = i^{-\gamma}$, $i = 1, 2, ..., n_v$. Then an edge is placed between v_i and v_j if one does not already exist. This process is repeated until n_e edges are obtained. Goh et al. (2001) stated that the proportion of nodes with degree k is asymptotically proportional to $k^{-\alpha}$, where $\alpha = (1+\gamma)/\gamma \in (2, \infty)$. There are different approaches to estimate the exponent of a power law distribution, but in this dissertation we used a least squares approach related to log-log plots (Kolaczyk, 2009). This approach is based on a linear relation between the degrees and the frequency of appearance of those degrees. For this least squares estimation consider a random variable X with probability mass function $f(x) = Cx^{-\alpha}$, $\forall x \geq x_{min}$ and 0 otherwise, where $C, \alpha, x_{min} > 0$, then

$$f(x) = Cx^{-\alpha} \implies \log(f(x)) = \log(C) - \alpha \log(x).$$

Hence, $log(f(x))$ behaves linearly with respect to $log(x)$ with change rate $-\alpha$. Given this relation the exponent α can be estimated by a least squares regression of the points $(\log(f(x)); \log(x))$. However as $f(x)$ is unknown, it is estimated by using the observed relative frequencies which are then used to estimate α by a least squares regression model to the points $log(\hat{f}(x))$ and $log(x)$ (Newman, 2005).
Due to the heavy tail behaviour of a power law distribution, random samples from this distribution could have a small number of values at the tail of the distribution. This feature could add noise to the estimation of $f(x)$ and therefore to the estimation of α as well. For this reason, Newman (2005) reviewed different ways to reduce this noise and concluded that a better way to deal with the noise present is to estimate α via a linear model for the following relation

$$\log(1 - \hat{F}(x)) = C' - (\alpha - 1)\log(x),$$

where $\hat{F}$ is the observed cumulative distribution function and C' is a real constant. The observed cumulative distribution of a sample $X_1, X_2, ..., X_n$ is
$\hat{F}(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{X_i \leq x\}}$.
More details about estimation of power law exponents can be found in A.1.

Geometric random graph models

Geometric random graphs on k dimensions, Geo-kD, were proposed by Gilbert (1961), as a way to capture the spatial representation of communication networks. Geometric random graphs consider two parameters a radius or threshold distance (r) and the dimensions (k) of the space in which the nodes are embedded. Given the parameters r and k, n_v nodes are placed uniformly at random in a k-dimensional cube $[0,1]^k$, where each node v_i is identified by its position in $[0,1]^k$. Then, edges are drawn among all nodes v_i, v_j $i \neq j$ for which the distance between them is less or equal than r, this is: $d(v_i, v_j) \leq r$, (in this dissertation we used the Euclidean distance as it is a standard metric (Penrose, 2003)).
Several studies have used geometric random graph models to fit PPI networks, and the dimension is usually taken as 2 or 3 (e.g. Pržulj, 2007; Rito et al., 2010).

Similarly to these studies we used $k = 3$ dimensions in our implementations.

Regarding the parameter r, the value $\hat{r}$ that leads to graphs with an expected number of edges, $\mathbb{E}(n_e)$, equal to the observed number of edges, n'_e, is frequently used (e.g. Shao et al., 2013; Pržulj et al., 2004; Pržulj, 2007; Janjič et al., 2014). This method of moments estimation leads to a value $\hat{r}$ that satisfies the following equation, taking the observed edge density (ρ') and the expected edge density (P_r) for Geo-kD random graphs. Note that if both sides of the equation are multiplied by $n_v(n_v - 1)/2$, then the equation relates the observed number of edges to the expected number of edges:

$$\rho' = P_r(X_{12} = 1), \quad \text{or equivalently,} \quad \frac{2n'_e}{n_v(n_v - 1)} = P_r(X_{12} = 1),$$

where n_v is the observed number of nodes, ρ' is the observed density of the graph, and $P_r(X_{12} = 1)$ is the probability that there is an edge between nodes 1 and 2. Following Philip (2007), if $k = 3$ then $P_r(X_{12} = 1)$ is

$$P_r(X_{12} = 1) = \begin{cases} p_1(r) & \text{if } 0 < r \leq 1, \\ p_1(1) + p_{\sqrt{2}}(r) & \text{if } 1 < r \leq \sqrt{2}, \\ p_1(1) + p_{\sqrt{2}}(\sqrt{2}) + p_{\sqrt{3}}(r) & \text{if } \sqrt{2} < r \leq \sqrt{3}, \end{cases}$$

where $p_1(r)$, $p_{\sqrt{2}}(r)$ and $p_{\sqrt{3}}(r)$ are

$$p_1(r) = 4/3\pi r^3 - 3/2\pi r^4 + 8/5r^5 - r^6/6,$$

$$\begin{aligned} p_{\sqrt{2}}(r) = {} & \frac{(6\pi - 1)}{2}(r^2 - 1) - 6(r^2 - 1)^{1/2} - 10(r^2 - 1)^{3/2} - \frac{16}{5}(r^2 - 1)^{5/2} \\ & - \frac{8\pi}{3}(r^3 - 1) + \frac{3}{2}(r^4 - 1) + 6r^4 \arctan(\sqrt{r^2 - 1}) + \frac{1}{3}(r^6 - 1), \\ p_{\sqrt{3}}(r) = {} & \frac{(6\pi - 5)}{2}(r^2 - 2) + 10(r^2 - 2)^{3/2} + \frac{8}{5}(r^2 - 2)^{5/2} + 14(r^2 - 2)^{1/2} \\ & - \frac{8\pi}{3}(r^3 - 2^{3/2}) + \frac{3}{2}(\pi - 1)(r^4 - 2^2) - \frac{1}{6}(r^6 - 2^3) \\ & + \arctan(\sqrt{r^2 - 2})(2 - 12r^2 - 6r^4) + 8r^3 \arctan(\sqrt{r^4 - 2r^2}) \end{aligned}$$

with $-\pi/2 < \arctan(x) < \pi/2$, $x \in \mathbb{R}$.
To obtain an estimate of the parameter r such that $\frac{2n'_e}{n_v(n_v-1)} = P_r(X_{12} = 1)$, we performed a grid search on the interval $[0, \sqrt{3}]$, as we used $k = 3$.

Duplication-divergence models

Duplication-divergence (DD) models were one of the first random graph models to take an explicitly biological viewpoint in the generation of networks. Here, a network is considered as a result of an 'evolutionary' process. One of the first duplication-divergence models was proposed by Vázquez et al. (2003). This model addresses the evolutionary process of a network by two biologically related steps: 'duplication' and 'divergence' of proteins (nodes). As proposed by Vázquez et al. (2003) the model is given as follows:

- Initialise the network with two connected nodes and then alternate between the next duplication and divergence steps, until the desired number of nodes is reached.
- Duplication: A node v_i is randomly selected and a new node v_j is created which inherits all interactions as the node previously selected (v_i). An edge between nodes v_i and v_j is created with probability p.
- Divergence: For all pairs of edges $\{(v_i, v_k); (v_j, v_k)\}$ from the duplication step; one of the two edges is selected uniformly at random and then deleted with probability q.

The duplication-divergence process is shown in Figure 1.3.
Several extensions to this model have been proposed (e.g. Peterson et al., 2012; Gibson and Goldberg, 2011; Ispolatov et al., 2005). In this dissertation we used the model of Vázquez et al. (2003) (described above) because of both its simplicity and potential to model protein-protein interaction networks (Shao et al., 2013).
To estimate p and q we followed the approach taken by (Vázquez et al., 2003) where values $\hat{p}$ and $\hat{q}$ are obtained by finding quantities that lead to graphs with an expected number of edges and expected clustering coefficient equal to the ones

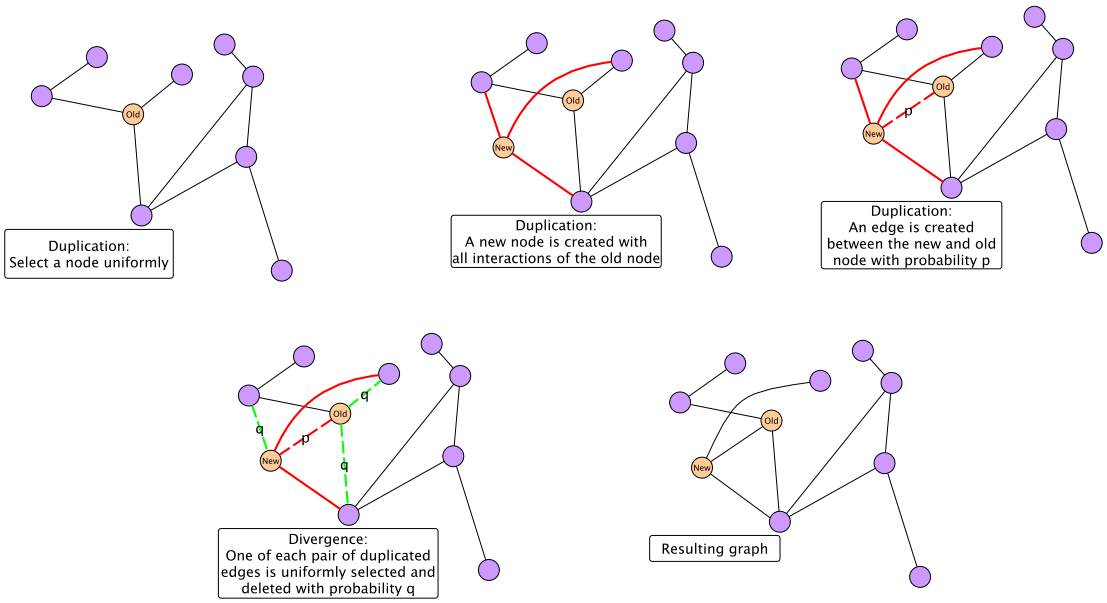


Figure 1.3: One cycle of the duplication and divergence steps. Node v_i is labelled as "Old" and v_j is labelled as "New". The red edges are created in the duplication step, and the green edges are selected and deleted (each one with probability q) in the divergence step.

observed in the data. Following Vázquez et al. (2003) and Gibson and Goldberg (2011), we used the exact derivation of the expected number of edges of a DD graph and obtained a landscape across the parameter space of p and q ($[0,1] \times [0,1]$) of the clustering coefficient, in order to find the values $\hat{p}$ and $\hat{q}$.

The expected number of edges on a DD graph with $t+1$ nodes (time step $t+1$) is (Vázquez et al., 2003):

$$\mathbb{E}\left(n_{e,t+1}\right) = \frac{1}{t!}\left[\prod_{i=2}^{t}(i+2\alpha) + \sum_{k=3}^{t}(p(k-1)!\prod_{i=k}^{t}(i+2\alpha)) + pt!\right],$$

where $\alpha = 1-q$ and $t+1 \geq 4$. For $t+1=3$

$$\begin{aligned}\mathbb{E}\left(n_{e,3}\right) &= \frac{1}{t!}\left[\prod_{i=2}^{t}(i+2\alpha) + pt!\right] \\ &= \frac{1}{2}\left[(2+2\alpha)+2p\right] \\ &= 1+\alpha+p,\end{aligned}$$

and for $t+1=2$, $\mathbb{E}\left(n_{e,2}\right) = 1$.

We obtained the clustering coefficient landscape by taking the average clustering

coefficient of a sample of 1000 duplication-divergence networks (for each pair of possible values of (p, q)) similarly to Gibson and Goldberg (2011) and Vázquez et al. (2003).

Estimates of p and q are then obtained by an exhaustive search of the parameter space using the explicit form of $\mathbb{E}(n_{e,t+1})$ and an estimate of $\mathbb{E}(C)$ given by the clustering landscape. Values $\hat{p}$ and $\hat{q}$ such that the distance from $\mathbb{E}(n_{e,t+1})$ and $\mathbb{E}(C)$ to the observed number of edges and clustering coefficient is minimum are selected.

1.4.1 Alternative parameter estimation methodologies

In this dissertation we considered several random graph models. However, only for a few random graph models, such as the ER model and the ERMG model, the parameters can be estimated via classical likelihood approaches (Daudin et al., 2008; Latouche et al., 2012; Mariadassou et al., 2010). For other random graph models, such as the geometric random graph model or the Duplication Divergence model, parameter estimation via likelihood-free approaches could be more appropriate; as the likelihood is often analytically or computationally intractable in these models (Ratmann et al., 2009; Thorne and Stumpf, 2012; Ali et al., 2010). One of the most well-known likelihood-free approaches is Approximate Bayesian Computation (ABC) (Pritchard et al., 1999; Didelot et al., 2011; Ratmann et al., 2009), and it is often used when the likelihood is not available.

One of the most basic ABC algorithms, which requires the specification of a prior distribution for the parameters of the model, $\pi(.)$, a distance, $\rho(.,.)$, between summary statistics of the data, $\eta(.)$, and a tolerance threshold, ϵ; consist of the following steps (Mengersen et al., 2013): (1) Generate parameters, θ', for the random graph model from a prior distribution $\pi(.)$. (2) Generate a random network from the model using θ' as the parameter of the model. (3) Compute the distance ρ between the generated network and the network of interest. (4) If the distance between the observed network and the generated network is less than a given threshold ϵ then store θ'. (5) Repeat all previous steps until a number N of θ' values has been

found. This sample can then be used to estimate the posterior distribution of the parameters θ of the random graph model.

From the previous steps it can be seen that ABC approaches can be very simple, however, their application in the context of random graph models might not be completely straightforward as there is no clear indication on how to select the threshold ϵ, the distance metric ρ, or the summary statistics. In principle, given our interest in network structure and random graph models, network comparison statistics could be used as a proxy for the distance ρ, but the choice of ϵ could still be subjective and difficult, given the lack of interpretability and meaning of the network comparison scores provided by the network comparison methods.

In this dissertation, instead of using ABC approaches, we decided to use simpler and faster moment estimation procedures to obtain estimates of the parameters of the random graph models for which the likelihood is not readily available; as in the Duplication-Divergence model. We chose this simpler procedure over the likelihood free approach ABC as it has been shown (e.g. Thorne and Stumpf, 2012) that ABC can struggle computationally when dealing with large PPI networks, and for some random graph models, the method may fail to obtain an estimation of the posterior distribution of the parameters of the random graph models. In fact, for one of the random graph models used by Thorne and Stumpf (2012), the method failed to estimate the posterior distribution of the parameters of the random graph model for 3 out of the 4 PPI networks studied.

1.5 Subgraphs, induced subgraphs, ego-networks and other network substructures

While large networks can be difficult to analyse and understand, smaller sections of a network (sub-networks) may be more easily analysed and potentially used to characterise a network (Winterbach et al., 2013; Shen-Orr et al., 2002; Kolaczyk, 2009). In our studies, we paid particular attention to the following sub-networks and network structures (Shen-Orr et al., 2002; Kolaczyk, 2009):

- **Subgraphs:** Given a network or graph $G = (V, E)$ a subgraph G_s of G is a graph 'included' in G. A subgraph is defined as the structure $G_s = (V_s, E_s)$ where $V_s \subset V$ and $E_s \subset E \cap (V_s \times V_s)$.

- **Induced Subgraphs:** These are subgraphs $G_s = (V_s, E_s)$ for which E_s contains exactly all edges among the nodes of V_s that are present in the original graph $G = (V, E)$; i.e. $E_s = E \cap (V_s \times V_s)$.

- **Complete Subgraphs:** These are subgraphs $G_s = (V_s, E_s)$ with $E_s := (V_s \times V_s) \setminus \{(v_i, v_i) : v_i \in V_s\}$. That is, subgraphs which contain all possible edges between all different pairs of nodes of V_s. These subgraphs are referred to as k-cliques, where $k = |V_s|$.

- **Network motif:** small connected subgraphs observed in a network that appear significantly more frequently than in the configuration model of such network.

This dissertation focused partly on a particular type of induced subgraphs, called ego-networks. In general, **k-step ego-networks** are formed by considering a single node v_i (*Ego*) and all nodes that are, at most, at k-steps apart from v_i along with all interactions among them, including v_i. Thus, the k-step ego-network of v_i is the induced subgraph generated by the set of nodes $\{v_j\}_j$ for which the path lengths $l(v_i, v_j) \leq k$.

Ego-networks can be used to extract the local network of a node and its neighbours, leading to ego-networks that range from small local subgraphs to even the whole network, (when the network is connected), by considering 'close' or 'far' neighbours. Figure 1.4 shows the growth ($k = 1, 3$ and 6 steps) of a single ego-network on a random geometric 2-D graph with 1000 nodes and 14387 edges.

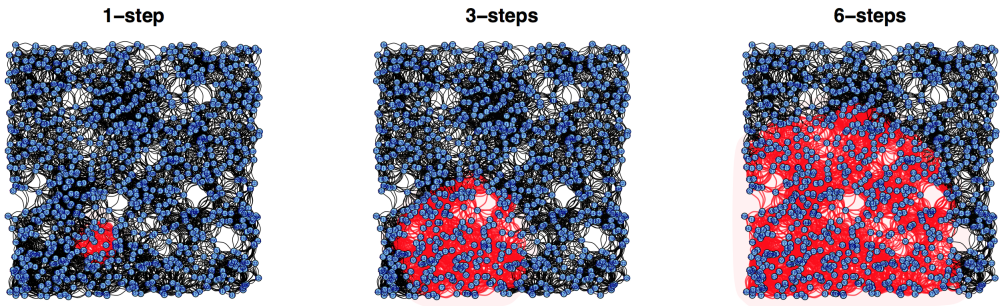


Figure 1.4: In red, edges of an ego-network of steps $k = 1, 3$ and 6, in a network of 1000 nodes and 14387 edges. Nodes are scattered uniformly at random in a unit square and edges are placed between nodes if the Euclidean distance between them is less or equal to 0.1 .

1.6 Monte Carlo method for hypothesis testing

The Monte Carlo method for hypothesis testing stems out of Monte Carlo integration (Robert and Casella, 1999; Davison, 2003). Classic Monte Carlo integration is a simulation technique to approximate the integral

$$\mathbb{E}_f(h(x)) = \int_X h(x)f(x)dx,$$

by

$$\bar{h} = \frac{1}{n}\sum_{i=1}^{n} h(x_i),$$

where $\mathbb{E}_f$ is the expected value operator calculated under the density function f and where $x_1, x_2, ..., x_n$ are independent realisations of a variable X with density function $f(.)$. This method, among other applications, can then be used to obtain an estimation of the cumulative distribution function of the random variable X by

$$\hat{P}(X \leq t) = \frac{1}{n}\sum_{i=1}^{n} I_{x_i \leq t}, \qquad t \in \mathbb{R},$$

where I is the set indicator function (Robert and Casella, 1999).

In the context of hypothesis testing, given a test statistic T computed over a sample $X^* = (X_1, X_2, ..., X_l)$, and an observed test statistic t_{obs}; the p-value, usually $P_{H_0}(T \leq t_{obs})$, is obtained from the distribution of T under the null hypothesis. However, when this distribution is unknown, the p-value can be approximated by

a Monte Carlo p-value calculated by generating n independent datasets X_i^* under the null hypothesis and obtaining their respective test statistic $T_i = T(X_i^*)$. The Monte Carlo p-value can be then computed as (Davison, 2003):

$$\hat{p}\text{-value} = \frac{1 + \sum_{i=1}^{n} I_{T_i \leq t_{obs}}}{1 + n}.$$

Note that the added $1s$ arise in the prior calculation because under H_0, t_{obs} is considered an observation obtained from the null distribution of T. Hence, the minimum Monte Carlo p-value that can be obtained is $\frac{1}{1+n}$.

Chapter 2

Random graph model selection using network comparison methods based on subgraph counts

The problem of assessing 'similarity' between networks has led to several proposals of network comparison methodologies. These methodologies range from comparing summary statistics of networks, such as average degree and clustering coefficient (e.g. Shao et al., 2013; Topirceanu et al., 2013; Berlingerio et al., 2013), to machine learning approaches (e.g. Aliakbary et al., 2015), network alignments (e.g. Neyshabur et al., 2013; Hashemifar and Xu, 2014) and comparisons of subgraph counts (e.g. Ali et al., 2014; Pržulj, 2007; Yaveroglu et al., 2014).

Some of these network comparison methods have also been used to relate an observed network to different random graph models. However, this assessment often uses the raw comparison scores alone and disregards the expected comparison score between networks truly coming from the particular random graph model of interest (e.g. Hayes et al., 2013; Shao et al., 2013; Pržulj, 2007; Pržulj et al., 2004). This method for assessing the fit of a random graph model to an observed network should be used with care, as usually there is no unique comparison score threshold that can be used to assess when a network can be considered "close" to a model.

Another difficulty when using network comparison methods is how to select the

appropriate method for a particular research question. This difficulty arises as it is often not clear what the comparison scores mean, or how the network features used for the comparison relate to the problem of interest.

In this chapter, which consists of a paper to be submitted as a journal article, we show why relating an observed network to a network generation mechanism based on a comparison score threshold can lead to misleading results. We also illustrate a statistical framework that is well suited to the task of model selection. We assess different network comparison methods using this framework in order to elucidate their properties.

In Section 2.1 we start by giving a detailed description of the different network comparison methods, as several of these methods are frequently used throughout this dissertation. Most of the methods described here focus on subgraph counts, as it has been thought that some small connected subgraphs are essential building blocks of complex networks and, in the context of biological networks, may also relate to particular biological processes (Milo et al., 2002; Alon, 2007; Sarajlić et al., 2013). Next, in Section 2.2, we introduce the paper titled "Random graph model selection using network comparison methods based on subgraph counts" using the article submission format of PLoS One. Then, in Section 2.3, we provide the Supplementary Information (SI), of the paper. In this section some redundant information regarding the description of the network comparison methods will be encountered. Finally, in Section 2.4, we discussed a particular behaviour we observed in one of the network comparison statistics used, Netdis, in the setting of model selection.

2.1 Methods

2.1.1 Network comparison methods based on subgraph counts

Graphlet degree distribution agreement

The graphlet degree distribution agreement (GDDA) proposed by Pržulj (2007) is based on a generalisation of the degree distribution (Section 1.3), to the degree

distribution of automorphism orbits of connected subgraphs (graphlets) on three to five nodes. An *automorphism* of a graph $G = (V, E)$ is a bijection $g : V \to V$ such that $(i, j) \in E$ if and only if $(g(i), g(j)) \in E$. An *automorphism orbit* of a node $i \in V$ is the set of nodes $\{x \in V | g(x) = i\}$, where g is any automorphism of G. Figure 2.1 shows the different automorphism orbits that are obtained for subgraphs on two to five nodes. For each subgraph, nodes that share the same automorphism orbit have the same grey shading. Over all the 30 subgraphs, here called graphlets, there are 73 different automorphism orbits (henceforth orbit), numbered from 0 to 72.

The generalisation of degree distributions to graphlet degree distributions consist on taking the number of nodes that 'touch' the appropriate graphlet at orbit j exactly k times as the frequency of the orbit degree, $d_G^j(k)$, $j = 0, 1, ..., 72$.

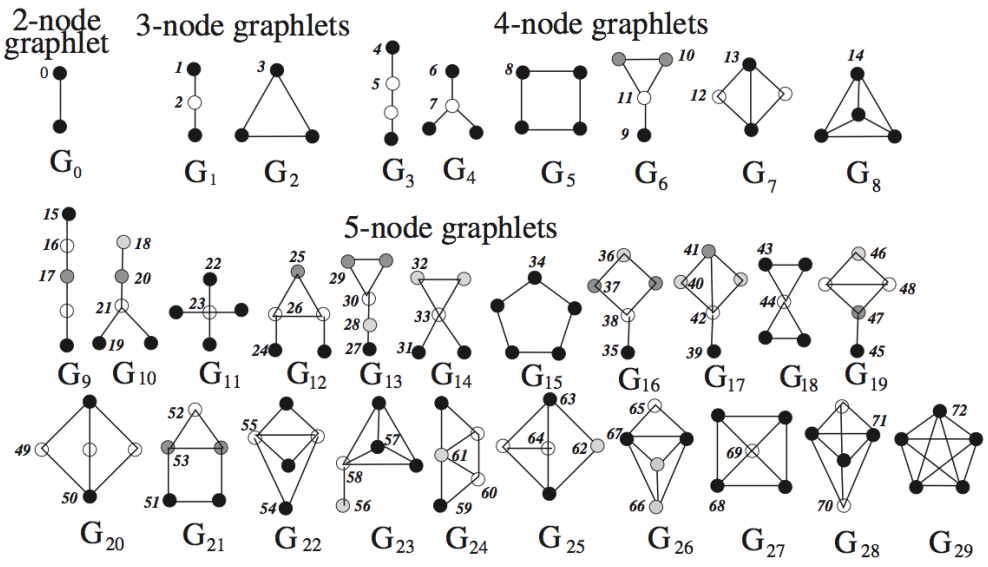


Figure 2.1: Graphlets on 2 to 5 nodes and the 73 automorphism orbits used by the GDDA. In each graphlet, nodes in the same orbit share the same grey filling. Figure obtained from Pržulj (2007).

The counting process only considers induced subgraphs (see Section 1.5), hence other non-induced subgraphs with n-nodes (n fixed) which are included in the induced subgraph on n-nodes are not taken into account in the counting process. Consider the graph in Figure 2.2 (which is repeated three times). To obtain the orbit degree distributions of orbits 1, 2 and 3 (Table 2.2), consider first counting

the number of times each node in the graph is "touching" the graphlets G_1 (2-stars) and G_2 (triangles) at orbits 1, 2 and 3 (Table 2.1). Note, in Figure 2.2, that node 5 is "touching" graphlet G_1 (in red) three times at orbit 1, and zero times at orbit 2 (zero times as well at orbit 3 in graphlet G_2), since node 5 is not in the middle of any graphlet G_1-like nor in any part of a triangle (graphlet G_2). On the other hand, node 1 only "touches" graphlet G_1 at orbit 1 two times (corresponding to the induced subgraphs 1-3-5 and 1-3-4); if nodes 1, 2 and 3 are considered, the induced subgraph corresponds to graphlet G_2 only. Hence, node 1 touches graphlet G_2 once. Once all counts of Table 2.1 are compiled, the orbit degree is the number of times a node touched that orbit exactly zero, one, two, three, etc. times (Table 2.2).

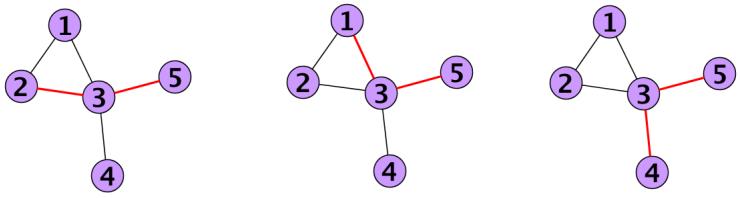


Figure 2.2: A graph on 5 nodes (repeated three times) illustrating the number of times that node 5 is "touching" graphlet G_1 (red) at orbit 1.

Node \Orbit	1	2	3
Node 5	3	0	0
Node 4	3	0	0
Node 3	0	5	1
Node 2	2	0	1
Node 1	2	0	1

Table 2.1: Number of times each node in the graph is "touching" Graphlets G_1 and G_2 at orbits 1, 2 and 3.

Orbit \ Degree	0	1	2	3	4	5
Orbit 1	1	0	2	2	0	0
Orbit 2	4	0	0	0	0	1
Orbit 3	2	3	0	0	0	0

Table 2.2: Orbit degree of the first, second and third orbit.

Given the set of 30 graphlets with their 73 orbits and the 'orbit degree distributions' of each one of these orbits $d_G^j(.)$, $j = 0, 1, ..., 72$, the GDDA is constructed as follows: For a graph G with n_v nodes, obtain for each orbit j

$$N_G^j(k) = \frac{d_G^j(k)/k}{\sum_{m=1}^{n_v} d_G^j(m)/m}, \qquad k = 1, 2, ..., n_v,$$

and similarly for any other graph. Then, to compare graphs G_1 and G_2 take

$$D^j = \frac{1}{\sqrt{2}}\big(\sum_{k=1}[N^j_{G_1}(k) - N^j_{G_2}(k)]^2\big)^{1/2}.$$

D^j is supposed to reflect 'how different' the orbit degree distributions are, and $1 - D^j$ 'how close' they are. The GDDA is obtained as the arithmetic or geometric mean of the values $1 - D^j$; in our analysis we used the arithmetic mean of all 73 orbits, which is the reference value used by the authors (e.g. Hayes et al., 2013; Kuchaiev et al., 2011; Pržulj, 2007).

As a final remark, we point out that the formulation of GDDA given by Pržulj (2007) does not take into account the degree 'zero' ($k = 0$).

Graphlet correlation distance

The graphlet correlation distance (GCD) (Yaveroglu et al., 2014) is a network comparison statistic that continues the idea of comparing small subgraph counts. Here, instead of orbit degrees, the "raw" orbit counts for each node are used. Yaveroglu et al. (2014) proposed the set of the 11 orbits (0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12) on 2-4 node subgraphs, (out of the 73 orbits shown in Figure 2.1), as the default set of orbits to use with their network comparison statistic GCD, as this set of orbits gave the best overall performance, and it was considered as a set of non-redundant counts, i.e. the counts of any of the orbits within the set cannot be derived from the counts in the other orbits in the set.

For each orbit, a count vector of the number of times each node in the network "touches" that orbit is obtained (as shown in Table 2.1). The count vectors are used to build a count matrix with a number of rows equal to the number of nodes in the network and 11 columns (like Table 2.1). Then, using Spearman's correlation, which is calculated as the Pearsons's correlation between the ranks of two variables, a correlation matrix is constructed by taking all pairwise Spearman's correlations between the columns of the former count matrix. Repeating this process for both networks leads to two correlation matrices. GCD compares these two correlation

matrices to obtain the final GCD statistic: For networks G_1, G_2

$$GCD\text{-}11 = \sqrt{\sum_{i=1}^{11}\sum_{j>i}^{11}(C_{i,j}^{G_1} - C_{i,j}^{G_2})^2},$$

where C^{G_k} is the correlation matrix of network G_k, with $k = 1, 2$. The rationale behind this comparison is based on the dependence between the counts observed in the different orbits and the assumption that networks that are 'similar' will preserve similar dependencies between their orbit counts.

Yaveroglu et al. (2014) proposed other graphlet correlation measures by using different sets of orbits, for example GCD-73 where all orbits on 2-5 node subgraphs are used and GCD-15 where all four node subgraphs are used. GCD-11 uses 11 orbits of the 15 possible orbits on subgraphs of 2-4 nodes and it is reported as the best choice among their alternatives as it achieved the best performance in a network classification task where synthetic networks were classified according to their generative models. The authors mentioned that small values of GCD suggest higher similarity (Yaveroglu et al., 2014).

Netdis

Netdis (Ali et al., 2014) also uses subgraph counts as building blocks for a network comparison statistic. In contrast to GCD and GDDA, Netdis takes a background expectation for the subgraph counts into account, which aims to give more reliable comparisons between networks of different sizes and edge-densities.

Netdis counts small subgraphs w on k nodes for all 2-step ego-networks, $k = 3, 4, 5$. These counts are centred by subtracting the expected number of counts E_w. The centred counts between the networks are compared and used to form the Netdis statistic. In detail, Netdis is constructed as follows:

Let $N_{w,i}(G)$ be the number of induced occurrences of small subgraphs w (Figure 2.1) in the 2-step ego-network of vertex i (Section 1.5). Let $E_w(\rho)$ be the expected number of occurrences of w in an ego-network of edge-density ρ.

For a given network G, compute the centred subgraph counts as

$$S_w(G) = \sum_i \left(N_{w,i}(G) - \binom{n_i}{k} E_w(\rho(i)) \right),$$

where i is a node in G, n_i is the number of nodes of the 2-step ego-network of node i with edge-density $\rho(i)$, and k is the size of subgraph w.

Now, to compare networks G_1 and G_2, set

$$netD_2^S(k) = \frac{1}{\sqrt{M(k)}} \sum_{w \in A(k)} \left(\frac{S_w(G_1) S_w(G_2)}{\sqrt{S_w(G_1)^2 + S_w(G_2)^2}} \right), \quad k = 3, 4, 5,$$

where $M(k)$ is a normalising constant equal to

$$M(k) = \sum_{w \in A(k)} \left(\frac{S_w(G_1)^2}{\sqrt{S_w(G_1)^2 + S_w(G_2)^2}} \right) \sum_{w \in A(k)} \left(\frac{S_w(G_2)^2}{\sqrt{S_w(G_1)^2 + S_w(G_2)^2}} \right),$$

so that $netD_2^S(k) \in [-1, 1]$. Finally take the Netdis statistic as

$$Netdis(k) = netd_2^S(k) = \tfrac{1}{2}(1 - netD_2^S(k)) \in [0, 1].$$

The authors mentioned that small values of Netdis suggest higher 'similarity' between the networks (Ali et al., 2014).

When $E_w(\rho)$ is not known explicitly, it is estimated through a gold-standard model. This is done by simulating one network Q from the model; or by taking a reference network, and extracting and binning all 2-step ego-networks according to their edge-density. Then, for each density bin ρ, and each subgraph w, obtain the total number of subgraphs w in ego-network i, $N_{w,i}(Q)$. Compute the empirical subgraph count expectations as:

$$E_w(Q, \rho) = \frac{1}{|\{i \in \{1, ..., q\} : d_i \approx \rho\}|} \sum_{\substack{i=1...q: \\ d_i \approx \rho}} \frac{N_{w,i}(Q)}{\binom{n_i}{k}},$$

where q is the number of ego-networks in density bin ρ, n_i the number of nodes in the 2-step ego-network of node i, k the size of the subgraph (3 to 5), and where

$d_i \approx \rho$ indicates that the edge-density of ego-network i falls in density bin ρ.

Ali et al. (2014) proposed an adaptive binning approach of the edge-density where 100 initial equally spaced bins are created in the interval $[0, 1]$. Then each ego-network is classified according to its edge-density in the respective density bin. Lastly, going from left to right, density bins with less than 5 ego-networks in them are merged with the next density bin. In our implementation of Netdis we modified the binning strategy proposed by Ali et al. (2014), so that the binning of the ego-network densities starts from the observed range of the ego-network density instead of the whole interval $[0, 1]$ as initially proposed.

2.1.2 Other network comparison methods

Other network comparison methods that are not based on subgraph counts have also been proposed. A well-known network comparison method is the Edit distance (Sanfeliu and Fu, 1983), defined as the minimum cost associated to a sequence of edition operations over one graph such as node/edge deletion or addition, in order to transform it into the graph it is compared against. Even though this method has a clear meaning and interpretation, it is often not used in practice as it is typically computationally intractable.

Network embeddings are also used to compare networks; for example Asta and Shalizi (2014) proposed an embedding of the nodes of a network in a hyperbolic space. In this space, networks are represented by their respective node clouds, which are assumed to be sampled from a density function defined over the hyperbolic space. Thus, Asta and Shalizi (2014) proposed to compare the networks by first estimating the density functions used to generate each of the node clouds, (and which represent both networks), and then compare the density functions via any suitable divergence measure. One of the disadvantages of this method is that it may perform poorly when the networks cannot be considered close to a hyperbolic networks, as this would lead to a poor embedding representation of such networks and consequently a poor comparison. Hyperbolic networks are networks whose nodes are generated by a density function over a hyperbolic space, and whose edges are

created based on the corresponding distances between those nodes.

Machine learning methods have also been used; for example NetDistance (Aliakbary et al., 2015) uses 17 network summary statistics, the average shortest path, diameter, edge-density, average degree, global clustering coefficient and average local clustering coefficient and others; in order to compare networks. NetDistance compares two networks G_1 and G_2 by

$$D_{weighted-manhatan}(G_1, G_2) = \sum_{i=1}^{17} |w_i s^i_{G_1} - w_i s^i_{G_2}|,$$

where s^i_G, $i = 1, 2, .., 17$ is the i^{th} summary statistic and w_i is a real valued weight to be learned by optimising the classification of a training set of synthetic networks via a k-nearest neighbour classifier based on $D_{weighted-manhatan}$.

Another type of methods that can be considered for comparing networks are network alignments. These methods aim to create a mapping, ('alignment'), between two networks in such a way that a particular statistic that assesses the goodness of the alignment is optimised. As with other types of network comparison methods, there are several network alignment methodologies, some well known methods are Netal (Neyshabur et al., 2013), HubAlign (Hashemifar and Xu, 2014), MAGNA (Saraph and Milenković, 2014), L-GRAAL (Malod-Dognin and Pržulj, 2015), Wave (Sun et al., 2015), Great (Crawford and Milenković, 2015), GHOST (Patro and Kingsford, 2012), NATALIE (El-Kebir et al., 2011), SPINAL (Aladağ and Erten, 2013) and others. In this chapter we considered using Netal as an additional methodology for network comparison, outside the methods based on subgraph counts. We selected an alignment methodology as this type of method, in addition to assessing similarity between networks, goes one step further away by providing a mapping between the networks, which illustrates what sections of the first network are structurally similar to other sections of the second network.

Finally, among the different network alignment methods mentioned above, we chose to use Netal as this method often provides a benchmark case for other network alignment methods. Additionally, Netal is able to assess the alignment based on structural features alone, and focuses on the comparison of local neighbourhoods

between the networks being aligned.

Netal

In contrast to the previous methods, instead of directly comparing local regions of the network through subgraph counts, Netal (Neyshabur et al., 2013) compares local neighbourhoods in order to create a "mapping" (alignment) between the node sets of two networks. The resulting network comparison statistic is based on the resulting alignment. In detail:

Given two networks G_1, G_2 such that $n_1 := |V(G_1)| \leq n_2 := |V(G_2)|$, an alignment f is an injective function $f : V_1 \to V_2$, that aims to maximise the number of conserved interactions between two networks. A conserved interaction is an edge (l, l') present in G_1 for which $(f(l), f(l')))$ is an edge present in G_2. Figure 2.3 shows an example of a possible alignment between two networks.

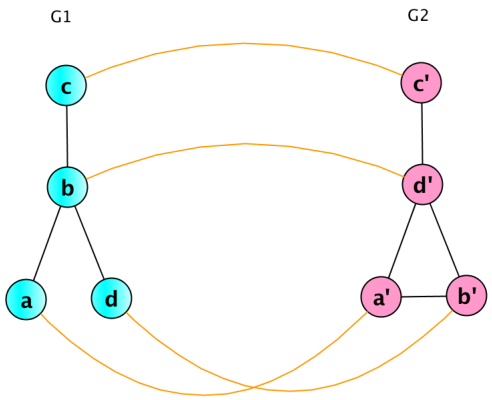


Figure 2.3: Illustration of an alignment (in orange) between network G_1 and G_2.

Netal uses two matrices, $S_{n_1 \times n_2}$ (n_1 rows and n_2 columns), and $I_{n_1 \times n_2}$ with rows representing nodes of G_1 and columns representing nodes of G_2 to obtain an alignment score matrix, $A_{n_1 \times n_2}$, whose entries give an overall similarity score between the neighbourhoods of node $i \in V(G_1)$ and node $j \in V(G_2)$. Under this scenario the alignment score matrix is a weighted sum between $S(i, j)$, which consist of topological scores between the local neighbourhoods of a node i and j, and $I(i, j)$, which is an approximation to the expected number of conserved interactions when

node i is assumed mapped to j:

$$A(i,j) = \lambda S(i,j) + (1-\lambda)I(i,j),$$

where λ is a weighting parameter between zero and one but suggested to be taken as $1/|V(G_1)|$.

Construction of the topological score matrix $S_{n_1 \times n_2}$. This topological score matrix is computed over a number T of iterations ($T = 2$ by default), here to keep track of the current iteration we used the index t, e.g. $S^t_{n_1 \times n_2}$ and $S^t(.,.)$.

At $t = 0$ the matrix S^0 is initialised with entries $S^0(i,j) = 1$, for $i \in V(G_1)$, $j \in V(G_2)$. Then to update $S^t(i,j)$ to $S^{t+1}(i,j)$, a weighted bipartite graph, G_B, is constructed with the neighbour set of i, $N(i)$, and the neighbour set of j, $N(j)$ as the two node sets. The neighbour set of a node is composed by the nodes at exactly one hop/step away from it. The edge weights of G_B are assigned as $S^t(u,v)$ for $u \in N(i)$, $v \in N(j)$. Continuing with the example of Figure 2.3; assume that $S^0(b,a')$ is going to be updated. Figure 2.4 shows the bipartite graph generated in this case.

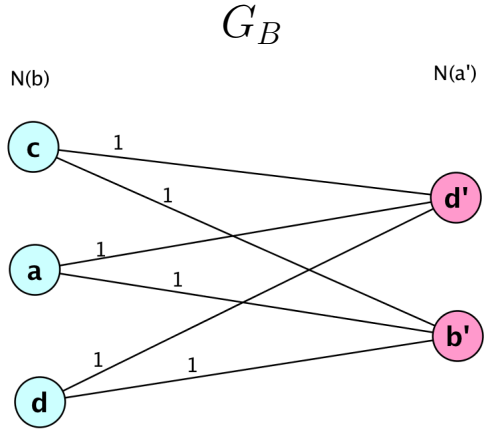


Figure 2.4: Bipartite graph, G_B, constructed with node sets given by the neighbours of $b \in V(G_1)$, denoted by $N(b)$, and the the set of neighbours of $a' \in V(G_2)$, denoted by $N(a')$. The weight of the edges are $S^0(u,v) = 1$, for $u \in N(b)$, $v \in N(a')$. Here $N(b) = \{a, c, d\}$ and $N(a') = \{b', d'\}$.

Now, to obtain the value $S^{t+1}(i,j)$, an edge (u,v) in G_B with, $u \in N(i)$, $v \in N(j)$, is selected such that for every other edge (x,y) in G_B, $S^t(u,v) \geq S^t(x,y)$ (order ties randomly). After an edge (u,v) is selected, all nodes incident to it in G_B

are removed. This process is then repeated until no edges are left. During each repetition of the process each selected edge (u, v) is placed in a set of edges M that is used next as a sum indexing set. The value $S^{t+1}(i, j)$ is then calculated as:

$$S^{t+1}(i,j) = \frac{\sum_{(u,v)\in M} S^t(u,v)}{\max\{|N(i)|, |N(j)|\}}.$$

Continuing with the example of Figure 2.3, note that only two edges need to be removed from G_B. For example, in Figure 2.4, first selecting edge (c, d') and then selecting (a, b'), thus taking $M = \{(c, d'), (a, b')\}$. Now, the updated $S^1(b, a') = \frac{1+1}{\max\{3,2\}} = 2/3$.

Note that $0 \leq S^t(u, v) \leq 1$, as the the maximum possible number of edges in M is $\min\{|N(i)|, |N(j)|\}$, and so at any iteration, $S^t(u, v) \leq \frac{\min\{|N(i)|,|N(j)|\}}{\max\{|N(i)|,|N(j)|\}} \leq 1$. Hence, the topological score would assign a higher score, initially, to similar degree nodes. In our example the updated matrix S^1 is:

	a'	b'	c'	d'
a	0.5	0.5	1	1/3
b	2/3	2/3	1/3	1
c	0.5	0.5	1	1/3
d	0.5	0.5	1	1/3

.

Construction of the interaction score matrix $I_{n_1\times n_2}$. According to Neyshabur et al. (2013), $I(i, j)$ is supposed to reflect the approximate number of interactions incident to $i \in V(G_1)$ that would be conserved, (edges in G_1 mapped to edges in G_2), when a node $i' \in N(i)$ is aligned to a random node $j' \in V(G_2)$. The interaction score matrix is computed as:

$$I(i,j) = \frac{min\left\{(\sum_{i'\in N(i)} 1/|N(i')|) - C_1(i), (\sum_{j'\in N(j)} 1/|N(j')|) - C_2(j)\right\}}{max_{k\in V_1\cup V_2}\{|N(k)|\}},$$

where $C_1(i)$ and $C_2(j)$ are the number of conserved interactions, accounted by the nodes aligned up to the current step, that are incident to nodes i and j, respectively. ($C_1(i)$ and $C_2(j)$ are updated once a pair of nodes is aligned). Thus $C_1(i)$ and $C_2(j)$

give the number of interactions that i, j make part of and, which up to current iteration, are known to be conserved interactions.

Following the example in Figure 2.3, assume that none of the nodes have been aligned. To obtain $I(b, c')$ then

$$\begin{aligned} I(b,c') =& \frac{min\left\{(\sum_{i' \in N(b)} 1/|N(i')|) - C_1(b), (\sum_{j' \in N(c')} 1/|N(j')|) - C_2(c')\right\}}{max_{k \in V_1 \cup V_2}\{|N(k)|\}} \\ =& \frac{min\left\{(1/1 + 1/1 + 1/1) - 0, (1/3) - 0\right\}}{3} \\ =& 1/9. \end{aligned}$$

Alignment algorithm. The alignment $f(.)$, and the construction of A follows an iterative algorithm that selects pairs (i, j) for which $A(i, j)$ is largest at the current iteration, and for which $f(i)$ is set to j. The following pseudo-code illustrates the alignment algorithm:

Result: Alignment $f(.)$
Set $t = 0$;
Initialise matrix S^t;
while $t < T$ **do**
    Update matrix S to S^{t+1};
    Set $t = t + 1$;
end
Take $S = S^T$;
Compute interaction score matrix I;
Set counter $k = 0$;
while $k < |V(G_1)|$ **do**
    Compute alignment score matrix A;
    Select nodes i, j for which $A(i, j) = max_{xy}\{A(x, y)\}$;
    Set $f(i) = j$ and take $k = k + 1$;
    Update matrix I;
end

Comparison statistic. Once the alignment is created, the percentage of conserved edges, also called Edge Correctness, (EC), in the first network is used as a

network comparison statistic:

$$EC = \frac{|\{(u,v) \in E(G_1) : (f(u), f(v)) \in E(G_2)\}|}{|E(G_1)|}.$$

EC is a standard measure used to judge network alignments (e.g. Neyshabur et al., 2013; Saraph and Milenković, 2014; Clark and Kalita, 2014). Larger values of EC would suggest higher similarity between the two networks. However, this measure is not perfect, for example, if G_2 is a complete graph, then $EC = 1$ for any G_1.
We note that other network alignment methods could be used, such as MAGNA (Saraph and Milenković, 2014) or HubAlign (Hashemifar and Xu, 2014), which in addition to EC implement other alignment measures such as *symmetric substructure score* (S^3) or *largest common connected subgraph* (LCCS) (Saraph and Milenković, 2014; Hashemifar and Xu, 2014). However, the fact that we are primarily interested in adding another methodology that does not use subgraph counts to make a comparison and, the fact that Netal is frequently used as a benchmarking method (e.g. Malod-Dognin and Pržulj, 2015; Sun et al., 2015; Hashemifar and Xu, 2014; Crawford and Milenković, 2015; Clark and Kalita, 2014), makes Netal a good method to consider in this dissertation.

2.1.3 Model selection

There are several approaches that can be used for statistical model selection. Some of these approaches such as the Akaike Information Criterion (AIC), the Bayesian Information Criterion (BIC), or the likelihood ratio test are based on maximum likelihood techniques (Akaike, 1973; Schwarz, 1978; Hooten and Hobbs, 2015). Bayesian model selection methods are also available, ranging from approaches based on closed forms of the prior and posterior distributions, to likelihood-free approaches such as Approximate Bayesian Computation (ABC), (see Section 1.4.1) (Chipman et al., 2001; Ratmann et al., 2009; Hooten and Hobbs, 2015).
Most of these model selection methods are mostly used when there are multiple models from which a best model needs be selected. However, in this dissertation, rather than being focused on selecting a "best" model among a pool of models, we

are interested in a simple methodology that allows to test whether a given individual random graph model can be used to recreate the structure of PPI networks in relation to the occurrence of different subgraph on 2 to 5 nodes. Thus, given the considerable analytical and/or computational complexity in proposing a closed form for all the different stochastic processes inherent to the different random graph models considered in this dissertation; we propose to use a simple classical Monte Carlo test (Robert and Casella, 1999) based on network comparison statistics, based on subgraph counts.

We proposed this method over the other more standard procedures because the Monte Carlo test approach, as will be soon described in Section 2.2, directly compares and assesses the ability of an individual random graph model to reproduce the structure of a given network, in relation to different network structure comparison statistics. And, in contrast to most standard approaches, a Monte Carlo test does not require the formulation or estimation of the likelihood or posterior distributions of the model, which in some cases may require greater computational effort and may still risk failing their computation. For example, Thorne and Stumpf (2012) proposed a Bayesian model selection approach via an Approximate Bayesian Computation methodology, to estimate parameters of different duplication divergence models and to estimate their respective posterior model probability distributions. However, only for 1 out of the 4 PPI networks considered, the methodology was able to make inference of all the posterior distributions required for the different random graph models considered.

2.2 Random graph model selection using network comparison methods based on subgraph counts

(Here starts the paper which is to be submitted as a journal article)

While the number of network comparison methods is increasing, benchmarking of these methods is still in its origins. The lack of understanding of complex

dependencies among network characteristics makes it difficult to fully understand the meaning of the different network comparison methodologies and the relations among them.

In this paper we give a Monte Carlo test that provides a statistical framework for network model selection via network comparison statistics. We use this framework to show that, despite the network comparison methods aiming to show "closeness" between networks, they are still not able to identify when networks share the same network generation mechanisms. In this work we consider four network comparison statistics GCD, GDDA, Netal and Netdis.

We found that these network comparison statistics are not invariant to the number of nodes and edges, making the identification of common generation mechanisms between networks that have different number of nodes and/or edges a challenge. Furthermore, we found that GCD and Netdis focus on a fine-grained distinction, sometimes missing their large scale similarity, while Netal and GDDA show a better compromise between fine scale and coarse scale comparison.

The methodology proposed for model selection is further illustrated by testing the fit of ER, Chung-Lu and of a duplication divergence model to protein-protein interaction networks of Yeast, Fly, Worm, Human, E. Coli, five herpes virus networks and five social networks. In contrast to previous claims in the literature, the large protein-protein interaction networks are not well modelled by the Chung-Lu model.

2.2.1 Introduction

Networks appear in a large number of applications across linguistics, sociology, engineering, biology and politics, among others (Newman, 2003). Network comparison and network generation methods are some frequently used tools of particular interest in network analysis (e.g. Newman, 2003; Shao et al., 2013).

Different methodologies for network comparison have been proposed during recent years; ranging from comparing summary statistics of networks, such as average degree and clustering coefficient (e.g. Shao et al., 2013; Topirceanu et al., 2013; Berlingerio et al., 2013), using network alignment methodologies (e.g. Neyshabur

et al., 2013; Hashemifar and Xu, 2014), subgraph counts (e.g. Ali et al., 2014; Pržulj, 2007; Yaveroglu et al., 2014), machine learning procedures (e.g. Aliakbary et al., 2015) and latent spaces (e.g. Asta and Shalizi, 2014). The results obtained from these methods may not always point towards the same conclusions, as they can follow different premises, and could be accounting for characteristics not explicitly considered in the initial methodology. This is due to the inherent dependence between characteristics of network data. Hence, given a particular research question, it is not always clear what network comparison measure should be used. For example, what network comparison statistic is best suited to compare the core-periphery structure in networks? Would the same statistic detect that two networks have similar community structure? A similar problem arises in model selection, where the interest lies in identifying networks that share the same generation mechanisms. In this paper we used this setting to provide a statistical framework that is well suited to this task, and that can be further used to obtain a better understanding of the different network comparison methods.

We apply our statistical framework to more practical settings, such as protein-protein interaction (PPI) networks and Facebook networks. PPI networks have been used in many biological studies, including the discovery of disease risk pathways, the investigation of genes undergoing age expression changes, and the identification of single druggable targets (e.g. Sarajlić et al., 2013; Noh et al., 2013; Zoraghi and Reiner, 2013; Higueruelo et al., 2013; West et al., 2013). The pursuit of a network model that can describe the modular structure of protein-protein interaction networks has been a focus of much research, leading to several proposals of models, such as, the Chung-Lu model (Chung and Lu, 2002; Pržulj and Higham, 2006) and duplication divergence models (Vázquez et al., 2003; Gibson and Goldberg, 2011; Ispolatov et al., 2005) among others.

Several studies have assessed the fit of a random graph model to PPI networks by using network summary statistics and other network comparison statistics in an empirical fashion (e.g. Shao et al., 2013; Janjiċ et al., 2014; Przulj et al., 2010). This type of procedure can leave room to misleading results, as asserting that two networks are close when their comparison statistic falls below a given universal

threshold might not always hold true for all cases of interest.

Comparing networks through small connected subgraphs is a methodology of particular interest, as small overrepresented subgraphs are thought to be building blocks of complex networks (Milo et al., 2002). These small subgraphs have been shown to be important patterns in gene regulatory networks; and there is evidence that they may be preserved in biological networks (Shen-Orr et al., 2002; Wuchty et al., 2003; Alon, 2007; Pereira-Leal et al., 2007). In this study we focus on three main network comparison methods based on connected subgraphs: "graphlet correlation distance" (GCD) (Yaveroglu et al., 2014), the "graphlet degree distribution agreement" (GDDA) (Pržulj, 2007) and Netdis (Ali et al., 2014). We also considered a reference method (Sun et al., 2015; Crawford and Milenković, 2015; Hashemifar and Xu, 2014; Malod-Dognin and Pržulj, 2015) based on network alignments, Netal (Neyshabur et al., 2013), as this type of method aims to find an exact mapping of one network to the other.

Here, we propose an easy to apply framework for the statistical evaluation of a network model. To evaluate the fit of a network model, B, to data G, through a statistic S, there are two challenges. (1) Finding the null distribution of the statistic S, i.e. if the data comes from B, how should the distribution of S look like? (2) comparing the value of S which is observed in the data G to the distribution of S under the null model. Our framework addresses these challenges through straightforward simulation of model-vs-model and data-vs-model comparisons, followed by a Monte Carlo test (a related procedure was proposed by Rito et al. (2010) for GDDA). This procedure is then used to compare the performance of the different network comparison methods in the setting of model selection.

By means of the GDDA we find that the use of a universal threshold, for the task of model selection, may lead to unreliable results, as not all comparisons between networks that achieve values smaller than the threshold share a similar structure, and not all networks that come from the same network generation mechanism achieve comparison values smaller than the given threshold. However, by means of the proposed Monte Carlo test, rigorous model selection is possible. In addition, this statistical framework enables accurate evaluation of the different network compar-

ison methods. For example, by considering the type two error of the Monte Carlo test, we found that GCD and Netdis tend to focus on fine-grained distinctions at the expense of missing large scale similarities while Netal and GDDA portray a better balance between fine grain and global structure. However, none of the comparison statistics is able to perfectly detect both, the model a network comes from and the models the network does not come from. This suggests that there is still a need for a network comparison statistic that is better equipped to detect the similarities left by the network generation mechanisms across networks with different numbers of nodes and/or edges.

By applying this framework to assess the fit of the Chung-Lu and DD models to five large PPI networks and five small (virus) PPI networks we found that the suggestion of Hayes et al. (2013) that the Chung-Lu model is a good model for PPI networks does not hold for the large PPI networks; the Chung-Lu model was rejected as a null model for the five large PPI networks by all four network comparison statistics.

This paper has four main results. 1) A Monte Carlo framework for hypothesis testing which is well suited to the task of model selection. 2) By considering the type two errors of the Monte Carlo tests, we gain a better understanding of the practical differences between network comparison statistics. 3) The use of a threshold in the raw scores from a network comparison statistic, as a way to define when two networks share the same network generation mechanism, may lead to unreliable results. 4) Given the crucial importance of a model for PPI networks, we assessed the fit of the Chung-Lu model, previously claimed as a good null model for PPI networks (Hayes et al., 2013). Here, we found that although this claim might be true for the small virus PPI networks, all four network comparison statistics rejected the Chung-Lu model for all five large PPI networks.

2.2.2 Materials and Methods

Network data

We used the large PPI networks of Yeast, Human, Fly, Worm and E. coli, downloaded from BioGRID (Stark et al., 2006) in October 2015. The E. coli PPI network was obtained from Rajagopala et al. (2014). We also used the datasets analysed by Hayes et al. (2013), for which, according to Hayes et al. (2013), "STICKY, SF-GD and GEO-GD (in that order) are the best fitting models" (see Section 2.3.2, SI). This claim was based on the network comparison statistic GDDA (described in the following section). In addition we considered the smaller PPI networks of the viruses EBV, VZV, mCMV, HSV-1 and KSHV obtained from Fossum et al. (2009) which were also analysed by Hayes et al. (2013). All self-loops, multiple edges and interspecies interactions are removed from the networks, leaving simple undirected networks for the analysis. All degree 0 nodes were also removed from the analysed networks. Table 2.3 describes these networks; the column *density* in the table is the number of edges divided by the possible number of edges and is given here to aid comparisons.

	Nodes	Edges	Density	Avg d.	Extracted from
Worm	3189	5556	0.00109	3.49	BioGRID ver 3.4.130
Fly	7958	36322	0.00115	9.13	BioGRID ver 3.4.130
Human	15590	182701	0.00150	23.44	BioGRID ver 3.4.130
Yeast	5862	79537	0.00463	27.14	BioGRID ver 3.4.130
E. coli	2002	3574	0.00178	3.57	(Rajagopala et al., 2014)
mCMV	111	393	0.06437	7.08	(Fossum et al., 2009)
KSHV	50	115	0.09388	4.60	(Fossum et al., 2009)
VZV	57	160	0.10025	5.61	(Fossum et al., 2009)
HSV-1	47	100	0.09251	4.26	(Fossum et al., 2009)
EBV	60	208	0.11751	6.93	(Fossum et al., 2009)

Table 2.3: Number of nodes, edges, density, average degree (Avg d.) and source of recent Yeast, Human, Fly, Worm PPI networks (downloaded in October 2015); and previously studied mCMV, KSHV, VZV, HSV-1, EBV virus PPI networks.

Five social network are also used in this work. These networks correspond to the Facebook networks of Caltech, Reed, Haverford, Simmons and Swarthmore universities. The nodes in each network represent users of Facebook who were members of the same university at September 2005. The undirected edges represent reciprocated friendship between the users. These five Facebook networks are the

smallest networks of a larger set of 100 universities (Traud et al., 2012; Onnela et al., 2012). Table 2.4 describes these networks.

All PPI and Facebook networks are sparse, in the sense that the average node degree is much smaller than the number of nodes.

	Nodes	Edges	Density	Avg d.
Caltech	769	16656	0.05640	43.32
Reed	962	18812	0.04070	39.11
Haverford	1446	59589	0.05704	82.42
Simmons	1518	32988	0.02865	43.46
Swarthmore	1659	61050	0.04439	73.60

Table 2.4: Number of nodes, edges, density, average degree and source of Facebook social networks of five universities previously studied by Traud et al. (2012).

Network Comparison methods

Graphlet correlation distance The graphlet correlation distance, (GCD) (Yaveroglu et al., 2014) is a network comparison statistic based on a comparison of small subgraph counts. The construction of GCD considers a particular set of 11 automorphism orbits of connected subgraph on 2-4 nodes for which a count vector of the number of times each node in the network "touches" that orbit is obtained. Then, a correlation matrix formed by all pairwise Spearman's correlations between the 11 count vectors is constructed. Repeating this process for both networks leads to two correlation matrices. GCD compares these two correlation matrices to obtain the final GCD statistic.

Smaller values of GCD are thought to suggest higher similarity between the networks (Yaveroglu et al., 2014).

Graphlet degree distribution agreement The graphlet degree distribution agreement, (GDDA) (Pržulj, 2007) is based on a generalisation of the degree distribution to the degree distribution of automorphism orbits of connected subgraphs (graphlets) on two to five nodes. GDDA is formed by the comparisons of each individual graphlet degree distribution between two networks.

The GDDA is supposed to reflect 'how similar' the orbit degree distributions are. Smaller values of $1 - GDDA$ are thought to imply 'similarity' between the com-

pared graphs (Pržulj, 2007; Hayes et al., 2013).

Netal Netal (Neyshabur et al., 2013) is an algorithm that creates a "mapping" (alignment) between the node sets of two networks. Given two networks G_1, G_2 such that $|V(G_1)| \leq |V(G_2)|$, an alignment f is an injective function $f : V_1 \to V_2$, that aims to maximise the number of conserved interactions. A conserved interaction is an edge (l, l') present in G_1 for which $(f(l), f(l')))$ is an edge present in G_2. After the alignment is created, we use the proportion of conserved edges in the first network as a network comparison statistic, also known as Edge Correctness (EC) (Neyshabur et al., 2013; Saraph and Milenković, 2014).
Smaller values of $1 - EC$ would suggest similarity between the two networks. Note however that, if G_2 is a complete graph, then $1 - EC = 0$ for any G_1.

Netdis Netdis (Ali et al., 2014) uses subgraph counts as building blocks for a network comparison statistic. In contrast to GCD and GDDA, Netdis uses an ensemble approach and a background expectation for the subgraph counts. Netdis counts small subgraphs w on k nodes for all 2-step ego-networks, $k = 3, 4, 5$. These counts can be centred by subtracting the expected number of counts, E_w, from the null model when a null model is available. The centred counts between the networks are compared to form the Netdis statistic.
Smaller Netdis values suggest higher 'similarity' between the networks (Ali et al., 2014). In this paper we consider no null model for the subgraph counts as this provides a level playing field to compare the performance of Netdis under different scenarios.

Random graph models

The Erdös-Rényi model: An Erdös-Rényi model $ER(n_v, \rho)$ (Gilbert, 1959) is a random graph on n_v nodes where single undirected edges are present independently at random, each with probability ρ.

The Chung-Lu model: In the Chung-Lu model (CL) (Chung and Lu, 2002),

also called Sticky model (Pržulj and Higham, 2006), positive weights are assigned to nodes. Then, an edge is established between any two nodes with probability proportional to the product of their weights. The weights are often taken as the observed degrees in the data.

A duplication divergence model: One of the first duplication divergence models was proposed by Vazquez et al. (Vázquez et al., 2003). This model addresses the evolutionary process of PPI networks by two biologically related steps: 'duplication' of nodes and their respective edges and 'divergence', where some of the duplicated edges are deleted with a given probability.

Monte-Carlo test

Consider testing the null hypothesis "H_0: Network G_0 is a realisation the fully specified model B", based on a network comparison statistic S; against the alternative hypothesis "H_1: Network G_0 is not a realisation of the fully specified model B". We use a Monte Carlo test which compares data-vs-model and model-vs-model to assess such hypothesis, relative to S; see also (Rito et al., 2010) for a precursor. For simplicity assume that the model is rejected when S is large. The test is given by the following steps:

- Generate M random graphs from the given model. For each of them generate another N random graphs and obtain the average of the comparison statistics between each of the M random graphs and the respectively generated N random graphs. This leads to a sample of M averages $\bar{S}_1, \bar{S}_2, ..., \bar{S}_M$.

- Generate N random graphs from the given model, and for each of these random graphs calculate S comparing the random graph to the data. Take the average of this sample ($\bar{S}_0$).

- If $\bar{S}_0$ is the k^{th} value on the ordered sample (break ties randomly) $\bar{S}_{(0)} \geq \bar{S}_{(1)} \geq ... \geq \bar{S}_{(M)}$, (note $\bar{S}_0 = \bar{S}_(k)$), then the p-value of the test is $\frac{k}{M+1}$. We reject the null hypothesis when the p-value is small (typically p-value ≤ 0.05).

The smallest possible p-value in this test is hence $\frac{1}{M+1}$.

To allow for a p-value of $\frac{1}{100}$ we take $M = 99$, and following Hayes et al. (2013) and Rito et al. (2010), we take $N = 30$. We show in section 2.2.3 that $N = 30$ is sufficiently large to obtain consistent results from the Monte Carlo tests.
Using this framework, we perform a right hand side test with the network comparison statistics GCD, $1 - GDDA$, $1 - EC$ (Netal), and Netdis.

2.2.3 Results and discussion

Model selection and assessment of network comparison statistics

In this section we illustrate a reliable Monte Carlo test as a way to assess if a given network can be considered as a realisation of a particular random graph model. Here we considered two simulation scenarios where the following general questions can be addressed: (1) Can the network comparison statistics differentiate between networks generated from different network generation mechanisms? (2) Are the number of nodes or average degree, confounding factors for the comparison of networks? (3) Do all four network comparison statistics lead to the same conclusions? The setup of the two simulation scenarios is as follows:

(a) ER, Chung-Lu and DD networks generated with 1000 nodes and with approximate average degrees, (d), of 20, 15 and 11, (see SI for details).

(b) ER, Chung-Lu and DD networks generated with 1500, 1000 and 500 nodes and with an approximate average degree of 15, (see SI for details).

Consider a network G_0 generated from a model A, and then use the Monte-Carlo test to evaluate the null hypothesis "H_0 : G_0 is a realisation of the fully specified model B" against the alternative hypothesis "H_1: Network G_0 is not a realisation of the fully specified model B", with a significance level α of 5%. The quantity $P(\text{Not reject } H_0 | H_0 \text{ is false})$, is known as the type two error of a hypothesis test, and it is the probability of not rejecting H_0 when it is false. This probability is particularly useful as it provides an intuitive and rigorous framework to compare and understand the behaviour of the tests using different network comparison statistics, and under deviations of the assumptions placed in the null hypothesis.

Here we estimated this quantity by obtaining the frequency of tests for which the null hypothesis is not rejected, i.e. the percentage of networks A found not to be different from networks generated from a fully specified null model (B); ($A \neq B$).

For each of the previous simulation scenarios, Figure 2.5 shows 9×9 different test cases of null hypotheses of the type "H_0 : Network G_0 is a realisation of model B", for each of the network comparison statistics. Note that there are 9 possible cases for data G_0: It can be drawn from one of three different models, and with one of three possible average degrees for scenario (a); (or for scenario (b), three possible network sizes). Model B can also represent any of the three different models and with any of the three average degrees for scenario (a) (or network sizes for scenario (b)). This led to a total of 9×9 possible test cases for each of the four network comparison statistics. Figure 2.5 arranges these cases in a grid with 36×9 blocks for each simulation scenario.

For each block, 100 hypothesis tests with significance level $\alpha = 5\%$ were conducted. Then, each block in Figure 2.5 was coloured according to the percentage of tests where the null hypothesis was not rejected. See SI Tables 2.7 and 2.8 for the exact percentage of rejection.

In Figure 2.5, we grouped the test cases for each network comparison statistic into larger blocks of 3×3 each. The purpose of these larger blocks is to focus on comparisons of the type "Data A" vs. "Model B" regardless of the average degree or network size. The columns of the figure represent Models and the rows represent actual network realisations from those models (Data). Hence, the arrays displayed in Figure 2.5 are not symmetric, as a test of "Data A" vs. "Model B" is not the same as a test of "Data B" vs. "Model A".

The outcome of comparisons of "Data A" vs. "Model B" using an ideal network comparison statistic that is only able to capture the network generation process is: 0%, (shown in blue), for $A \neq B$, i.e. the null hypothesis is always rejected. And, for comparisons of "Data A" vs. "Model A", the expected outcome is 95%, as the significance level of the test is 5% (shown in red). This ideal case would show that the network comparison statistic would be able to detect similarities that are independent of network size or edge density. This ideal case is portrayed in Figure

2.6.

Note that in Figure 2.5 (or SI Tables 2.7 and 2.8), the cases where a network drawn from a fully specified model A is tested as a realisation of such fully specified model A, behave as it is expected from a hypothesis test with a confidence level of 5%; i.e. the number of not rejected tests mostly fall within the expected number of not rejected tests plus/minus two standard deviations ($95 \pm 2 \times 2.18$). Thus, this results shows that taking $N = 30$, as done by (Hayes et al., 2013), leads to consistent results from the Monte Carlo tests performed in the scenarios here considered.

In scenario (a) of Figure 2.5, two broad patterns created by the network comparison statistics can be observed. In the first pattern, composed by GCD and Netdis, the network comparison statistics rejected the null hypothesis in most cases. However, they did not detect, for example, that an ER network (Data A) with an average degree 15 shared the same network generation mechanism as an ER network with an average degree of 11 (Model B). The second pattern, composed by Netal and GDDA, showed that these two methods could better state what networks shared similar generation mechanisms. However, neither Netal nor GDDA performed perfectly. On one hand GDDA struggled at rejecting the DD model for Chung-Lu networks, and on the other, Netal struggled at rejecting ER and Chung-Lu models for DD networks.

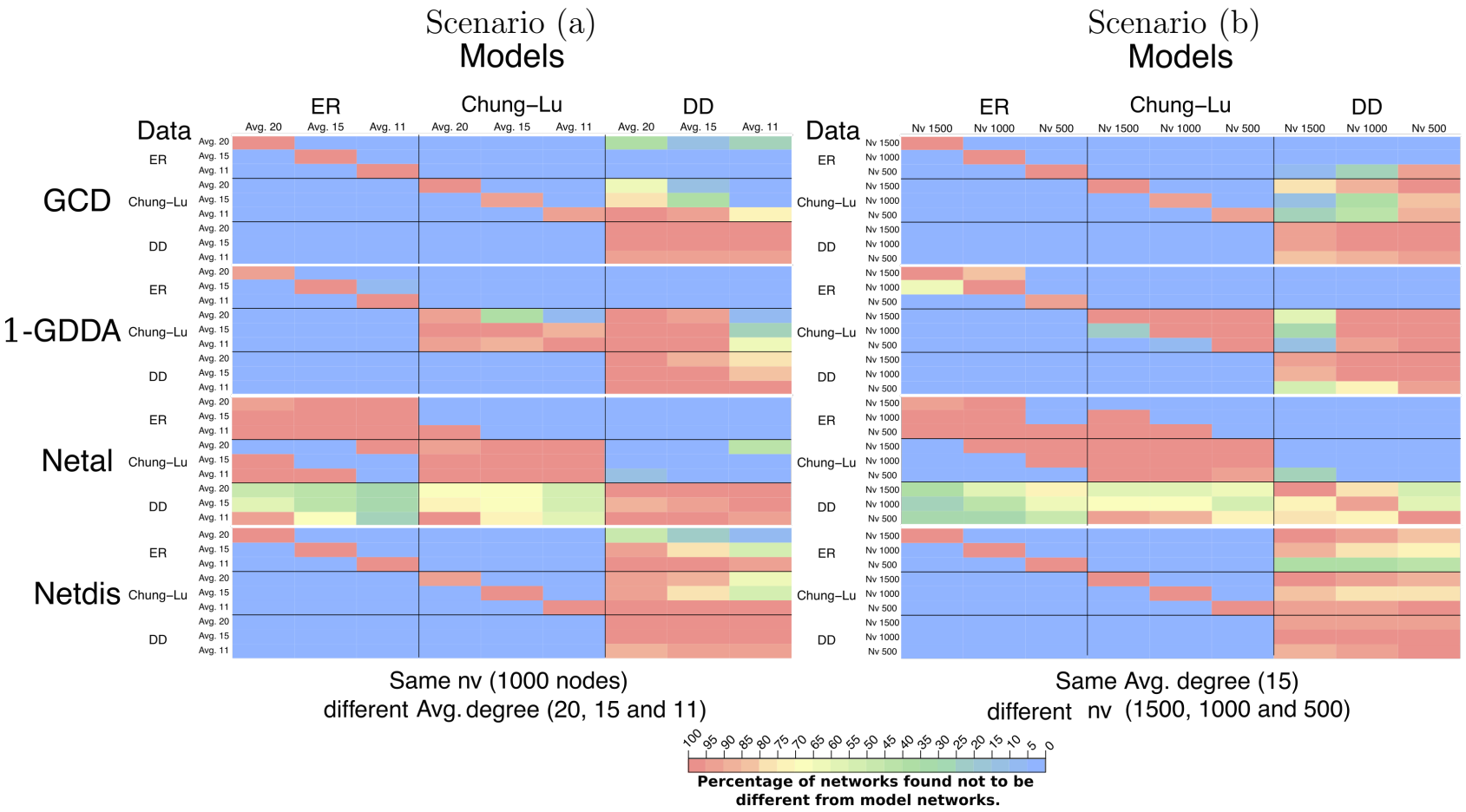


Figure 2.5: This Figure shows the results for scenario (a) where all networks generated are set to have the 1000 nodes and expected average degrees of 20, 15 and 11; and results for scenario (b), where all networks generated are set to have an expected average degree of 15 and node sizes of 1500, 1000 and 500. The colour scale shows the percentage of times the null hypothesis "$H_0 : G_0$ is a realisation of model B" is not rejected, at $\alpha = 5\%$, from 100 realisations of the Monte-Carlo test using the four network comparison statistics GCD, GDDA, Netal and Netdis. An ideal result of a network comparison statistic, relative to model selection, is shown in Figure 2.6. Despite the fact all methods aim to show how 'close' or 'far' two networks are from one another, they are not always able to tell when the networks come from the same or different network generation mechanism. Over the two scenarios the following conclusions can be made: GCD and Netdis perform better at telling fine grained differences between networks, but struggle at detecting when networks share the same network generation mechanism. GDDA and Netal show a better compromise between fine grain differences and the broader scale similarities.

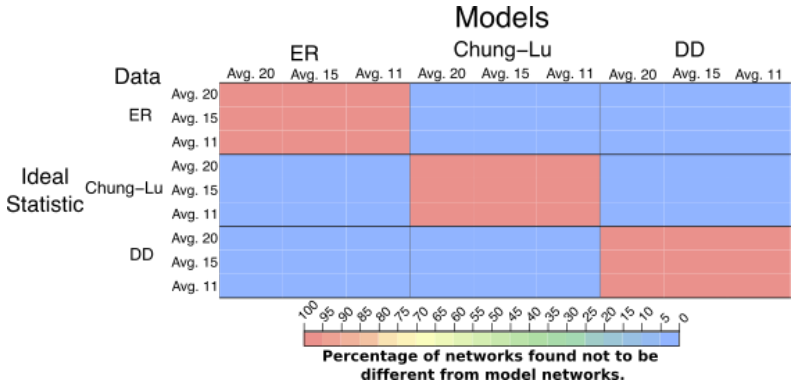


Figure 2.6: This Figure shows the result of using an ideal network comparison statistic in the Monte Carlo test for scenario (a) where all networks generated are set to have the 1000 nodes and expected average degrees of 20, 15 and 11. The colour scale shows the percentage of times the null hypothesis "$H_0 : G_0$ is a realisation of model B" is not rejected, at $\alpha = 5\%$. An equivalent figure would be obtained for scenario (b).

Results for scenario (b) had a similar overall behaviour as the one observed in scenario (a). Here the network comparison statistics GCD and Netdis rejected ER networks on 1500, 1000 and 500 nodes as networks with the same generation mechanisms, despite all networks having the same expected average degree, and being generated from the same model. The same result was observed for the Chung-Lu model, GCD nor Netdis were able to identify the Chung-Lu networks with different node sizes, as coming from the same network generation mechanism.

The results shown in scenarios (a) and (b) also illustrate clear differences between the network comparison statistics, that are not evident from their formulation. GCD, GDDA and Netdis, all aim to measure "closeness" between networks and they all use similar inputs (subgraph counts). However, from Figure 2.5, it can be seen that GCD and Netdis focus more on fine grained differences, thus finding discrepancies more easily within networks of the ER and Chung-Lu models. In contrast, Netal and GDDA, seem to focus on coarser differences, thus detecting similarities between networks of the same model despite having different number of nodes or edges. It can also be seen that all network comparison methods, particularly GCD, GDDA and Netdis, are not invariant under the number of nodes or network density, as a different number of nodes and/or average degree can suffice for the methods to avoid detecting that networks share the same network generation mechanism. Thus a network comparison statistic that is invariant to changes

in the number of nodes and network density is still required.

Reliability of thresholds in model selection

In this section we show that the use of universal thresholds in network comparison statistics as a way to establish which networks share the same network generation mechanisms may lead to unreliable results. Here we continue using the two previous simulation scenarios. Pržulj (2007) proposed to consider the threshold 0.86 for GDDA values as a cut off to indicate that the networks being compared are closely related. Here we used this threshold to classify networks into a given network generation mechanism. Whenever the comparison between a pair of networks passes the cut-off, they are considered as coming from the same network generation mechanism. We show that, even with a small number of models considered (ER, Chung-Lu and DD), this type of procedure is not always able to correctly identify networks coming from the same network generation mechanism, even when the networks are generated from the same model with the same number of nodes and parameter values.

In detail, for all combinations of model and average degree (scenario (a)), or model and number of nodes (scenario (b)), we performed pairwise GDDA comparisons among networks generated from each of the models, (including a comparison of the models against themselves), thus leading to 9×9 GDDA pairwise comparisons. We repeated this procedure 100 times an recorded the percentage of network comparisons such that one minus their GDDA comparison (1-GDDA) was smaller than the threshold proposed in Pržulj (2007), $0.14 = 1 - 0.86$. Figure 2.7 shows the results. Similarly to the plots shown for the Monte Carlo test (Figures 2.5 and 2.6), here the plots are arranged by a grid of 9×9 blocks, which correspond to all pairwise comparisons among networks from the different models along with their respective average degrees, or number of nodes.

Note that if the threshold is used to state that two networks share the same network generation mechanism, the ideal result would show that all comparisons between networks from the same model would be smaller than the threshold considered for 1-GDDA, and all comparisons between networks from different models would be

larger than such threshold.

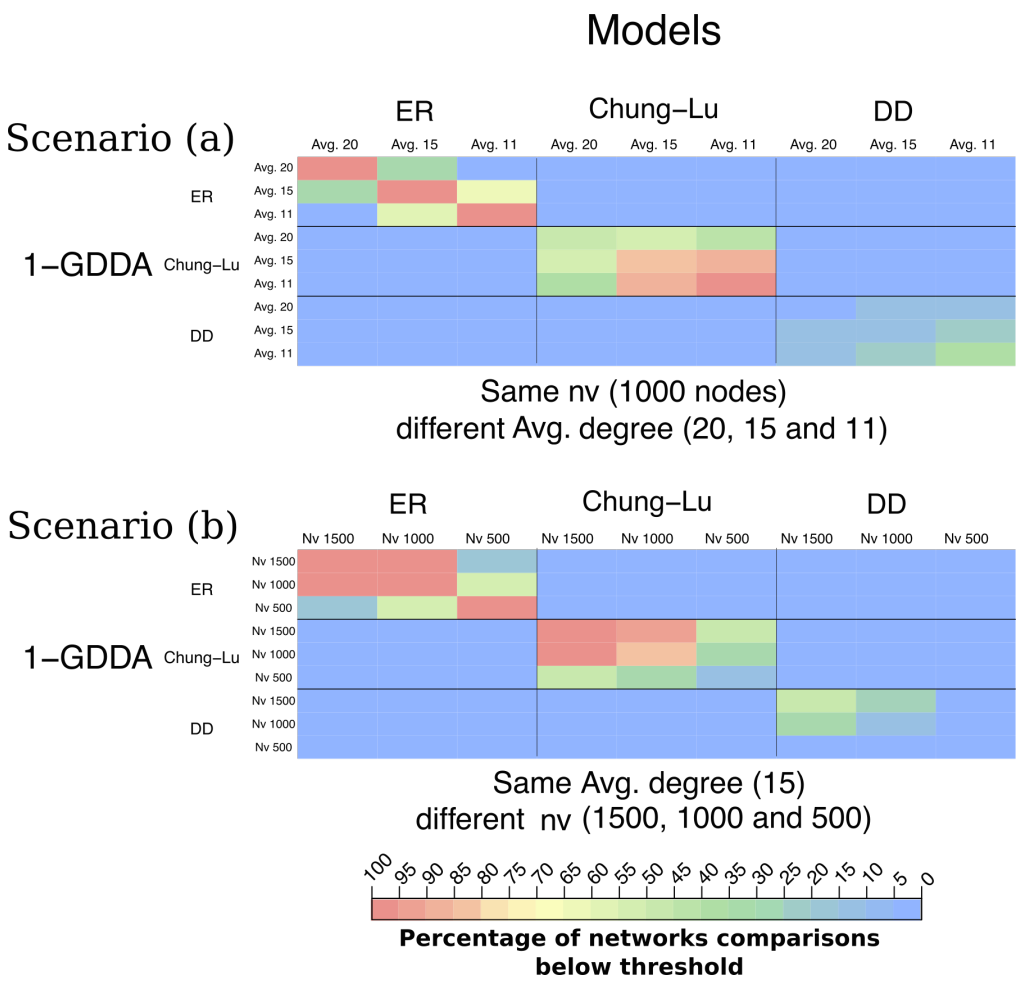


Figure 2.7: Percentage of 1-GDDA pairwise network comparisons with values smaller than $0.14 = 1 - 0.86$.

The results shown in Figure 2.7 illustrate that considering the 0.14 threshold as a way to state whether two networks share similar generation mechanisms, may lead to unreliable results. For example, for both scenarios (a) and (b), the GDDA was in most cases not able to detect that DD networks generated from the same model, and even with the same number of nodes and same parameters, did come from the same model. This behaviour is also observed for networks coming from the Chung-Lu model with 500 nodes and average degree 15, (scenario (b)), where less than 20% of the comparisons detected that the networks came from the same model. Similarly, for Chung-Lu networks with 1000 nodes and average degree 20, approximately 50% of the comparisons did not detect that the networks came from the same model, even when those networks were generated with the same number

of nodes and the same degree sequence.

In Figure 2.7 we do not observe network comparisons with a comparison value smaller than the threshold (0.14) for networks coming from different models. However this does not mean that such behaviour cannot occur. Take for example 1-GDDA comparisons between a complete graph and a line-like network on 10, 20, 30,...,90 and 100 nodes (Figure 2.8). It can be noted that all comparison values are below the threshold $0.14 = 1 - 0.86$, despite line-like networks and complete networks being largely different.

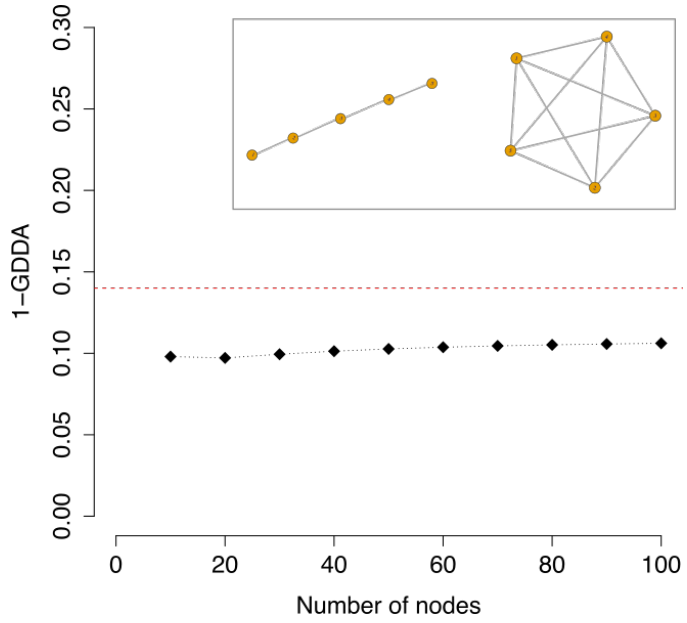


Figure 2.8: 1-GDDA comparisons of line-like networks vs. complete graphs on 10, 20, 30,...,90 and 100 nodes. The insert shows a sketch of these type of networks on five nodes. All comparison values are below the threshold $0.14 = 1 - 0.86$ (shown in red).

The results shown in Figures 2.7 and 2.8 illustrate that considering a threshold as a way to state that two networks share the same generation mechanisms or, are similar to one another, can lead to a wide array of results, some of which are misleading and inconsistent. In addition, there is no telling when the threshold used will be able to correctly assess the similarity between networks. Thus, for a task of model selection, a more reliable method should be used.

Testing the Chung-Lu and Duplication divergence models as null models for protein-protein interaction networks

Using GDDA Hayes et al. (2013) suggested that the Chung-Lu model was a good model for the Human, Yeast, Worm, Fly and E. coli PPI networks, but provided evidence only for the five virus networks EBV, HSV-1, KSHV, mCMV and VZV. In this section we tested, via the proposed Monte Carlo test, whether the large and updated PPI networks of Human, Yeast, Worm, Fly and E. coli could be viewed as a realisation of a Chung-Lu model with the same degree sequence of the respective PPI network. In addition to the Chung-Lu model, we also considered the biologically inspired duplication divergence model, which aims to reproduce protein-protein interaction networks. We performed a Monte-Carlo test, as proposed in Section 2.2.2, with each of the four network comparison methods to test whether the PPI networks could be thought as realisations of the Chung-Lu or DD models.

Table 2.5 shows the results of the Monte-Carlo test using each of the four network comparison statistics. All four statistics rejected the Chung-Lu model for the five larger PPI networks. However, apart from the EBV virus, there was no unanimous rejection of the Chung-Lu model among the four network comparison statistics for all other virus networks. In addition, the DD model was never rejected for these small virus networks. Thus, both the Chung-Lu and the DD model could generate networks "similar" to the virus PPI networks. Most of the results for the five large PPI networks (Yeast, Human, Worm, Fly and E. coli) rejected both the Chung-Lu and the DD model. Among the large PPI networks, only for Worm and E. coli the DD model was not rejected by all four network comparison statistics.

From the Monte Carlo tests we found no evidence for the claim that the Chung-Lu model is as a "good" model for the large PPI networks, even with the same statistic used by Hayes et al. (2013), GDDA. We note that Hayes et al. (2013) showed no data for their claim that large PPI networks can be modelled well by a Chung-Lu model. Hence, to understand whether our results differed because of the different data sets used, we used the GDDA and performed the Monte Carlo test with the

Model	PPI	GCD	GDDA	Netal	Netdis
	Worm	**0.01**	**0.01**	**0.01**	**0.01**
	Fly	**0.01**	**0.01**	**0.01**	**0.01**
CH-L	Human	**0.01**	**0.01**	**0.01**	**0.01**
	Yeast	**0.01**	**0.01**	**0.01**	**0.01**
	E. coli	**0.01**	**0.01**	**0.01**	**0.01**
	mCMV	**0.04**	0.35	**0.01**	0.10
	KSHV	**0.05**	**0.01**	0.14	0.21
CH-L	VZV	**0.05**	0.08	0.11	**0.04**
	HSV-1	0.38	0.50	**0.04**	0.12
	EBV	**0.02**	**0.03**	**0.02**	**0.05**
	Worm	**0.01**	0.35	**0.01**	**0.01**
	Fly	**0.01**	**0.01**	**0.01**	**0.03**
DD	Human	**0.01**	**0.02**	**0.01**	**0.01**
	Yeast	**0.01**	**0.03**	**0.01**	**0.01**
	E. coli	**0.02**	0.41	**0.01**	**0.01**
	mCMV	0.77	0.77	0.17	0.41
	KSHV	0.76	0.64	0.13	0.65
DD	VZV	0.64	0.78	0.20	0.17
	HSV-1	0.77	0.76	0.38	0.97
	EBV	0.64	0.24	0.23	0.74

Table 2.5: P-values of the Monte-Carlo test using the network comparison statistics GCD, GDDA, Netal and Netdis. The test is performed for the updated E. coli, Worm, Fly, Yeast and Human PPI networks, and the small virus PPI networks. P-values smaller or equal to 0.05 are in bold. The smallest possible p-value is equal to $\frac{1}{99+1} = 0.01$. The DD and Chung-Lu models are rejected as models for the large PPI networks for most of the network comparison statistics. In contrast for the smaller virus PPI networks the Chung-Lu model, and in particular the DD model are not rejected by most of the network comparison statistics.

same PPI networks used by Hayes et al. (2013), see SI (Section 2.3.2). We found the same results as the ones obtained for the updated versions of the Yeast, Human, Worm, Fly and E. coli PPI networks, (given in Table 2.5). However, for some of the virus PPI networks we did obtain evidence to support the conclusion proposed by Hayes et al. (2013).

One explanation why the virus networks performed differently could be due the idea that viruses are fundamentally different to other living organisms. For example, viruses are not able to reproduce on their own, as they require of living cells to replicate (Moreira and Lopez-Garcia, 2009). In fact, such are the differences between viruses and other living organisms that there is an ongoing debate to whether viruses are actually living organism (Moreira and Lopez-Garcia, 2009).

Another possible explanation for the difference of results between the virus datasets and the other larger PPI networks is that many virus-host interactions are highly relevant for the virus life cycle. However, these type of interactions are often not taken into account when analysing PPI networks, as there are no clear ways to select which hosts to consider, e.g. (Hayes et al., 2013). Due to this difficulty, in this study we did not consider such interactions either.

Lastly, another factor that could have led to this difference in results could be the small size (number of nodes) of the virus PPI networks. These virus PPI networks are small networks (the largest has 111 nodes), in comparison to the Worm, Fly, Human, Yeast or E. coli PPI networks (the smallest having 2002 nodes). With a lower number of nodes there are fewer options for connections available, which could lead to a lower level of complexity than the one observed in other larger PPI networks. Hence, it could be easier for random graph models to reproduce network structures which are similar to the ones observed in the virus PPI networks.

Testing the ER and Chung-Lu models as null models for the Facebook networks

The Chung-Lu model is a standard null model used in community detection, for both PPI networks and social networks (e.g. Newman, 2006; Porter et al., 2009; Lewis et al., 2010; Onnela et al., 2012; Traud et al., 2012). Here we tested if the Facebook networks of five USA universities from 2005 could be considered as realisations of the ER or Chung-Lu models. The p-values of the Monte-Carlo test using the four network comparison methods are given in Table 2.6.

The results obtained by GCD, GDDA, Netal and Netdis reject the null hypothesis in all cases. Hence, while the Chung-Lu model is often suitable for community detection, we cannot recommend it as a good null network generation model for these Facebook networks, particularly when there is interest in the small scale structure.

Model	FB	GCD	GDDA	Netal	Netdis
ER	Caltech	**0.01**	**0.01**	**0.01**	**0.01**
	Reed	**0.01**	**0.01**	**0.01**	**0.01**
	Haverford	**0.01**	**0.01**	**0.01**	**0.01**
	Simmons	**0.01**	**0.01**	**0.01**	**0.01**
	Swarthmore	**0.01**	**0.01**	**0.01**	**0.01**
CH-L	Caltech	**0.01**	**0.01**	**0.01**	**0.01**
	Reed	**0.01**	**0.01**	**0.01**	**0.01**
	Haverford	**0.01**	**0.01**	**0.01**	**0.01**
	Simmons	**0.01**	**0.01**	**0.01**	**0.01**
	Swarthmore	**0.01**	**0.01**	**0.01**	**0.01**

Table 2.6: P-values of the Monte-Carlo test using the network comparison statistics GCD, GDDA, Netal and Netdis. The test is performed for Facebook networks of five USA universities. P-values smaller or equal to 0.05 are in bold. The ER and Chung-Lu models are rejected as models for these five Facebook networks by all of the network comparison statistics.

2.2.4 Conclusions

We have shown that considering a universal threshold over the comparison values obtained from different comparison methods for the task of model selection can be inappropriate, as there is no clarity when such thresholds successfully identify networks with similar structural characteristics or that are generated from the same model. We showed, for example, DD networks with the same number of nodes and same parameters cannot often be identified as similar by means of the threshold approach suggested by Pržulj (2007) using the GDDA.

Instead of a threshold approach we provided a rigorous framework to statistically assess when a given network can be considered as a realisation of a random graph model, by means of any network comparison statistic.

We have shown how the analysis of the type two errors in this framework provides a level playing field in which different network comparison statistics can be compared and assessed with regards to standard goals, such as model selection. This framework allowed us to find differences between the four network comparison statistics used, that could otherwise be difficult to identify, particularly for GCD, GDDA and Netdis, which use similar comparison strategies (subgraph counts). We found that GCD and Netdis focus on more fine grained differences between networks while Netal and GDDA focus more on coarser differences, thus making the detection of

common generation mechanisms easier. We also found that changes in edge density or number of nodes imposes a challenge on the network comparison statistics. For GCD and Netdis, this means that they might not be able to detect networks that share the same network generation mechanism, while for Netal this means that it might not be able to differentiate between networks with different network generation mechanisms. These results show that there is still a need for a network comparison statistic that is invariant to changes in the number of nodes and edges but still able to detect networks with common generation mechanisms.

In the application of the framework to the PPI networks, we showed that recent protein-protein interaction networks of Yeast, Human, Fly and Worm are not fitted well by the Chung-Lu model nor the DD model. In contrast, we find that all of the virus data sets can be seen as a realisation of the DD model. The difference in the results obtained between the virus PPI networks and the other larger PPI networks may stem from the level of complexity of the networks, and the fundamental differences between viruses and the other organisms.

The methodology is also applied to five small Facebook networks. Here, all network comparison methods reject both the Chung-Lu and the ER model. Hence, while the Chung-Lu model might be well suited for the task of community detection it does not provide a good representation of the small structure accounted by the occurrence of small subgraphs.

2.3 Supplementary Information

2.3.1 Setup and results of the simulation study

We used two simulation scenarios to perform a study of the type two error at $\alpha = 0.05$ for the Monte Carlo test via different network comparison methods. These scenarios are:

(a) All model networks are generated with 1000 nodes and with approximate average degrees of 20, 15 and 11.

(b) All model networks are generated with a different number of nodes (1500, 1000 and 500), and with an approximate average degree of 15.

The aim of these scenarios is to better understand the performance of the different network comparison methods when networks have similar number of nodes and edges but come from different models. We are also interested in the capacity of the network comparison methods to realise that two networks do (or do not) come from the same model, despite having different number of nodes and/or different average degrees.

To carry out the simulation we first set the parameters of the duplication -divergence model $DD(p, q)$ of Vázquez et al. (2003) on 1000 nodes (n_v) to $p = 0.2$ and $q = 0.3$. Similar parameters have been used to model some Yeast PPI networks in the past, e.g. ($p = 0.26$, $q = 0.33$) and ($p = 0.22$, $q = 0.54$) (Shao et al., 2013). This parameter selection leads to DD networks with an expected average degree of 20.274. Then we considered the average degrees that would be obtained if the number of nodes is reduced in half (500) and to a quarter (250) (still maintaining $p = 0.2$ and $q = 0.3$). The resulting average degrees are 15.12 and 11.226 for 500 and 250 nodes, respectively. We used these expected degrees in order to set the parameters for scenarios (a) and (b) in the following manner.

In scenario (a) all networks are generated with 1000 nodes ($n_v = 1000$). Hence, in order to fix the expected average degrees of the DD models used in scenario (a) to 20.274, 15.12 and 11.226, we maintained $p = 0.2$ and vary q. This procedure lead to $q = 0.3$, $q = 0.32996$ and $q = 0.36122$, respectively for the expected average degrees 20.274, 15.12 and 11.226. The parameter of a Chung-Lu model is a complete degree sequence of a graph. To that purpose, we used the degree sequence of a realisation of a DD model with parameters ($p = 0.2$, $q = 0.3$); ($p = 0.2$, $q = 0.32996$) and ($p = 0.2$, $q = 0.36122$). We selected degree sequences such that their average degree was 20.274, 15.12 and 11.226, respectively. For the $ER(n_v, \rho)$ model, where ρ is the probability of connecting any two nodes, we used $\rho = 0.02029$, $\rho = 0.01513$ and $\rho = 0.01124$, respectively for each of the expected average degrees 20.274, 15.12 and 11.226.

For scenario (b), networks are generated with a fixed expected average degree of 15.12 but with different number of nodes (1500, 1000 and 500). For the DD model, as in scenario (a), we fixed $p = 0.2$ and varied q in order to obtain the desired degree with the different number of nodes. This procedure led to q values of 0.3442998, 0.3299654 and 0.3, for networks of sizes 1500, 1000 and 500, respectively. The parameters of the Chung-Lu model were obtained from degree sequences of realisations of DD graphs, as in scenario (a). We selected degree sequences such that their average degree was 15.12. For the $ER(n_v, \rho)$ model, where ρ is the probability of connecting any two nodes, we used $\rho = 0.01009$, $\rho = 0.01514$ and $\rho = 0.03031$, respectively for each of the network sizes.

In both simulation scenarios, we take a network G_0 generated from model A and used the Monte Carlo test to evaluate the null hypothesis "$H_0 : G_0$ is a realisation of model B" against the general alternative, with a significance level α of 5%. Models A and B can be any of the three random graph models, eg. ER, DD and Chung-Lu. We repeated this test 100 times and recorded the number of times the test rejected network G_0 as a realisation of model B, for each corresponding network comparison statistic. Tables 2.7 and 2.8 show the results of these simulations for scenarios (a) and (b), respectively.

2.3.2 Inspecting claims of good fit

Hayes et al. (2013) wrote "... Examining the biological reasons for the good fit of the sticky model in 80% of the viral networks, and why it is a less good fit for KSHV, is a subject of future research. Similar plots (not shown because of space limitations) of three bacterial PPI networks [MZL (Shimoda et al., 2008), SPP (Sato et al., 2007) and CJJ (Parrish et al., 2007)], the functional interaction network of E. coli (Peregrin - Alvarez et al., 2009), as well as the Arabidopsis thaliana PPI network (Arabidopsis Interactome Mapping Consortium, 2011), and PPI networks of Yeast, Worm, Fly and Human from BioGRID, all indicate that STICKY, SF-GD and GEO-GD (in that order) are the best fitting models for these networks."

		Model	ER			CHL			DD		
	Data	Avg d.	20	15	11	20	15	11	20	15	11
GCD	ER	20	**0.97**	0.00	0.00	0.00	0.00	0.00	0.35	0.12	0.25
		15	0.00	**0.96**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		11	0.00	0.00	**0.95**	0.00	0.00	0.00	0.00	0.00	0.00
	CHL	20	0.00	0.00	0.00	**0.95**	0.00	0.00	**0.6**	0.13	0.00
		15	0.00	0.00	0.00	0.00	**0.94**	0.00	**0.78**	0.37	0.04
		11	0.00	0.00	0.00	0.00	0.00	**0.92**	**0.99**	**0.94**	**0.73**
	DD	20	0.00	0.00	0.00	0.00	0.00	0.00	**0.96**	**0.95**	**0.95**
		15	0.00	0.00	0.00	0.00	0.00	0.00	**0.96**	**0.95**	**0.95**
		11	0.00	0.00	0.00	0.00	0.00	0.00	**0.92**	**0.92**	**0.93**
GDDA	ER	20	**0.93**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		15	0.00	**0.95**	0.05	0.00	0.00	0.00	0.00	0.00	0.00
		11	0.00	0.00	**0.95**	0.00	0.00	0.00	0.00	0.00	0.00
	CHL	20	0.00	0.00	0.00	**0.93**	0.36	0.05	**1.00**	**0.92**	0.05
		15	0.00	0.00	0.00	**0.99**	**0.98**	**0.86**	**1.00**	**1.00**	0.28
		11	0.00	0.00	0.00	**0.91**	**0.87**	**0.96**	**1.00**	**1.00**	**0.64**
	DD	20	0.00	0.00	0.00	0.00	0.00	0.00	**0.97**	**0.89**	**0.75**
		15	0.00	0.00	0.00	0.00	0.00	0.00	**1.00**	**0.95**	**0.83**
		11	0.00	0.00	0.00	0.00	0.00	0.00	**1.00**	**1.00**	**0.98**
Netal	ER	20	**0.92**	**1.00**	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
		15	**1.00**	**0.96**	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
		11	**1.00**	**1.00**	**0.95**	**1.00**	0.00	0.00	0.00	0.00	0.00
	CHL	20	0.00	0.00	**1.00**	**0.94**	**1.00**	**1.00**	0.00	0.00	0.42
		15	**1.00**	0.00	0.00	**1.00**	**0.97**	**1.00**	0.00	0.00	0.02
		11	**1.00**	**1.00**	0.00	**1.00**	**1.00**	**0.97**	0.11	0.00	0.00
	DD	20	0.46	0.43	0.32	**0.65**	**0.65**	**0.53**	**0.96**	**0.96**	**1.00**
		15	**0.59**	0.42	0.3	**0.72**	**0.66**	**0.52**	**0.89**	**0.94**	**0.97**
		11	**0.9**	**0.66**	0.25	**0.95**	**0.73**	**0.55**	**0.97**	**0.95**	**0.94**
Netdis	ER	20	**0.97**	0.00	0.00	0.00	0.00	0.00	0.46	0.23	0.05
		15	0.00	**0.98**	0.00	0.00	0.00	0.00	**0.92**	**0.78**	**0.52**
		11	0.00	0.00	**0.97**	0.00	0.00	0.00	**0.97**	**0.96**	**0.94**
	CHL	20	0.00	0.00	0.00	**0.93**	0.00	0.00	**0.94**	**0.85**	**0.64**
		15	0.00	0.00	0.00	0.00	**0.99**	0.00	**0.91**	**0.76**	**0.54**
		11	0.00	0.00	0.00	0.00	0.00	**0.96**	**1.00**	**1.00**	**0.99**
	DD	20	0.00	0.00	0.00	0.00	0.00	0.00	**0.96**	**0.96**	**0.96**
		15	0.00	0.00	0.00	0.00	0.00	0.00	**0.95**	**0.97**	**0.96**
		11	0.00	0.00	0.00	0.00	0.00	0.00	**0.86**	**0.91**	**0.91**

Table 2.7: Results of scenario (a) where all networks generated are set to have the 1000 nodes and approximate average degrees of 20, 15 and 11. The value in the table shows the percentage of times the null hypothesis "H_0 : G_0 is a realisation of model B" is not rejected against the general alternative, at $\alpha = 5\%$, from 100 realisations of the Monte Carlo test using the four network comparison statistics GCD, GDDA, Netal and Netdis. Despite the fact all methods aim to show how 'close' or 'far' two networks are from one another, they may not detect when the networks come from the same network generation mechanism. GCD and Netdis perform better at telling fine grained differences between networks, but may not detect when the networks share the same network generation mechanism. Netal and GDDA compromise in fine grain differences to compare more broad scale similarities. Values larger than 0.50 are shown in bold.

In this section we used PPI datasets analysed by Hayes et al. (2013) that were publicly available, and for which "STICKY, SF-GD and GEO-GD (in that order) are the best fitting models" (Hayes et al., 2013). These datasets are shown in Table 2.9. All self-loops, multiple edges and interspecies interactions are removed from PPI networks, leaving simple undirected networks for the analysis. All degree 0 nodes were also removed from the analysed PPI networks. Three of the BioGRID datasets used by Hayes et al. (2013) have a slightly different number of nodes and

		Model	ER			CHL			DD		
	Data	N. of nodes	1500	1000	500	1500	1000	500	1500	1000	500
GCD	ER	1500	**0.95**	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1000	0.04	**0.96**	0.00	0.00	0.00	0.00	0.00	0.00	0.02
		500	0.00	0.00	**0.95**	0.00	0.00	0.00	0.06	0.28	**0.90**
	CHL	1500	0.00	0.00	0.00	**0.99**	0.00	0.00	**0.77**	**0.85**	**0.98**
		1000	0.00	0.00	0.00	0.00	**0.94**	0.00	0.14	0.37	**0.83**
		500	0.00	0.00	0.00	0.00	0.00	**0.94**	0.29	0.43	**0.87**
	DD	1500	0.00	0.00	0.00	0.00	0.00	0.00	**0.94**	**0.98**	**0.99**
		1000	0.00	0.00	0.00	0.00	0.00	0.00	**0.92**	**0.95**	**1.00**
		500	0.00	0.00	0.00	0.00	0.00	0.00	**0.82**	**0.85**	**0.93**
GDDA	ER	1500	**0.95**	**0.83**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1000	**0.64**	**0.95**	0.02	0.00	0.00	0.00	0.00	0.00	0.00
		500	0.00	0.01	**0.92**	0.00	0.00	0.00	0.00	0.00	0.04
	CHL	1500	0.00	0.00	0.00	**0.96**	**1.00**	**1.00**	**0.59**	**1.00**	**1.00**
		1000	0.00	0.00	0.00	0.22	**0.98**	**0.99**	0.33	**1.00**	**1.00**
		500	0.00	0.00	0.00	0.00	0.08	**0.95**	0.12	**0.92**	**1.00**
	DD	1500	0.00	0.00	0.00	0.00	0.00	0.00	**0.92**	**0.95**	**1.00**
		1000	0.00	0.00	0.00	0.00	0.00	0.00	**0.88**	**0.95**	**1.00**
		500	0.00	0.00	0.00	0.00	0.00	0.00	**0.53**	**0.73**	**0.92**
Netal	ER	1500	**0.93**	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		1000	**1.00**	**0.96**	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
		500	**1.00**	**1.00**	**0.96**	**1.00**	**1.00**	0.00	0.00	0.00	0.00
	CHL	1500	0.00	**1.00**	**1.00**	**0.95**	**1.00**	**1.00**	0.00	0.00	0.01
		1000	0.00	0.00	**1.00**	**1.00**	**0.97**	**1.00**	0.00	0.00	0.00
		500	0.00	0.00	0.00	**1.00**	**1.00**	**0.92**	**0.26**	0.00	0.00
	DD	1500	0.35	**0.57**	**0.71**	**0.57**	**0.55**	**0.62**	**0.96**	**0.77**	**0.53**
		1000	0.27	0.42	**0.64**	**0.65**	**0.66**	**0.54**	**0.73**	**0.94**	**0.56**
		500	0.31	0.32	0.47	**0.90**	**0.86**	**0.70**	**0.75**	**0.70**	**0.96**
Netdis	ER	1500	**0.98**	0.00	0.00	0.00	0.00	0.00	**0.96**	**0.92**	**0.82**
		1000	0.00	**0.98**	0.00	0.00	0.00	0.00	**0.86**	**0.78**	**0.73**
		500	0.00	0.00	**0.96**	0.00	0.00	0.00	0.37	0.38	0.44
	CHL	1500	0.00	0.00	0.00	**0.97**	0.00	0.00	**0.96**	**0.94**	**0.86**
		1000	0.00	0.00	0.00	0.00	**0.99**	0.00	**0.80**	**0.76**	**0.77**
		500	0.00	0.00	0.00	0.00	0.00	**0.96**	**0.90**	**0.90**	**0.98**
	DD	1500	0.00	0.00	0.00	0.00	0.00	0.00	**0.93**	**0.91**	**0.92**
		1000	0.00	0.00	0.00	0.00	0.00	0.00	**0.95**	**0.97**	**0.95**
		500	0.00	0.00	0.00	0.00	0.00	0.00	**0.89**	**0.90**	**0.96**

Table 2.8: Results of scenario (b) where all networks generated are set to have expected average degrees of 15 but with different number of nodes 1500, 1000 and 500. The value in the table shows the percentage of times the null hypothesis "H_0 : G_0 is a realisation of model B" is not rejected against the general alternative, at $\alpha = 5\%$, from 100 realisations of the Monte Carlo test using the four network comparison statistics GCD, GDDA, Netal and Netdis. Despite the fact all methods aim to show how 'close' or 'far' two networks are from one another, they may not detect when the networks come from the same network generation mechanism. GCD and Netdis perform better at telling fine grained differences between networks, but may not detect when the networks share the same network generation mechanism. Netal and GDDA compromise in fine grain differences to compare more broad scale similarities. Values larger than 0.50 are shown in bold.

edges. Table 2.9 describe these networks; the column *density* in the tables is the number of edges divided by the possible number of edges and is given here to aid comparisons.

	Nodes	Edges	Density	Extracted from
Human	8920	35386	0.000890	BioGRID ver 3.1.74
Worm	2831*	4527	0.001130	BioGRID ver 3.1.74
Fly	7373*	24063	0.000885	BioGRID ver 3.1.74
Yeast	5608*	57143	0.003635	BioGRID ver 3.1.74
AT	2634	5529	0.001594	(Dreze et al., 2011)
E. coli	1941	3989	0.002119	(Peregrín-Alvarez et al., 2009)
EBV	60	208	0.117514	(Fossum et al., 2009)
HSV-1	47	100	0.092507	(Fossum et al., 2009)
KSHV	50	115	0.093878	(Fossum et al., 2009)
mCMV	111	393	0.064373	(Fossum et al., 2009)
VZV	57	160	0.100251	(Fossum et al., 2009)

Table 2.9: Number of nodes, edges, density and source of the PPI networks used by Hayes et al. (2013). *The number of nodes found is slightly different from those reported in (Hayes et al., 2013). Only the Worm PPI network has a difference greater than 1 node (Worm +14.)

The sticky model is not generally a background model for local structure in protein-protein interaction networks

Using the GDDA, Hayes et al. (2013) indicated that the Chung-Lu model was in most cases the best model for the PPI networks in Table 2.9, although they only showed results for the virus PPI networks. As we do not obtain the claimed results for the more recent PPI networks of Human, Yeast, Fly and Worm, we performed the Monte Carlo test for the data used by Hayes et al. (2013). Table 2.10 shows the p-values obtained. Except for the virus datasets EBV, HSV-1, mCMV and VZV, all p-values take the smallest possible value of the Monte Carlo test. Hence, for 8 of the 11 datasets used by Hayes et al. (2013) we reject the hypothesis, at 5% significance level, that the Chung-Lu model is a good null model by means of the GDDA.

Issues when using histograms for comparing distributions

Hayes et al. (2013) used comparisons of 30 networks from the random graph model to create two samples; a first sample of comparisons of PPI data vs. model networks and a second sample of GDDA comparisons of model vs. model networks. From these samples two histograms are created and the overlap between them is used to assess the goodness of fit between the model and the PPI networks.

Data	p-value (GDDA)
Networks used by Hayes et al. (2013) ($M = 30$ $N = 30$)	
Yeast	0.0322
Human	0.0322
Worm	0.0322
Fly	0.0322
AT	0.0322
ECL	0.0322
EBV	0.0400*
HSV-1	0.4516
KSHV	0.0322
mCMV	0.3548
VZV	0.1100*

Table 2.10: P-values of the Monte Carlo test using the data vs. model and model vs. model GDDA comparisons shown in Figure 2.9. Note that for this particular test the smallest p-value is equal to $\frac{1}{30+1} = 0.0322$ ($M = 30$). We reject the null hypothesis that the Chung-Lu model fits when the p-value < 0.05. Hence, except for HSV-1, mCMV and VZV, all null hypotheses are rejected. p-values marked with '*' were obtained using $M = 99$ since $M = 30$ was inconclusive in this two cases. The p-values obtained with $M = 30$ for VZV and EBV were 0.0645 and 0.0967 respectively.

Figure 2.9 shows the histograms we obtained from the comparisons created in the application of the Monte Carlo test, where we take $N = M = 30$. Histograms with considerable overlap are seen for three of the five virus PPI networks (as expected, since a similar figure is presented by Hayes et al. (2013)). In contrast, for the other larger PPI networks the amount of overlap is very low or null (no figure was shown by Hayes et al. (2013) for these large networks).

To assess the histogram method used by Hayes et al. (2013) we use the Sturges rule, a standard binning procedure (Venables and Ripley, 2002, p. 112). However, as the overlap between the histograms can vary depending on the number of bins, we recommend using the Monte Carlo test procedure proposed in the main text instead of considering the overlap between histograms.

The Sturges rule formula is:

$$k = \lceil 1 + \log_2 N \rceil,$$

where k is the number of classes, $\lceil . \rceil$ is the ceiling function, and N is the size of the

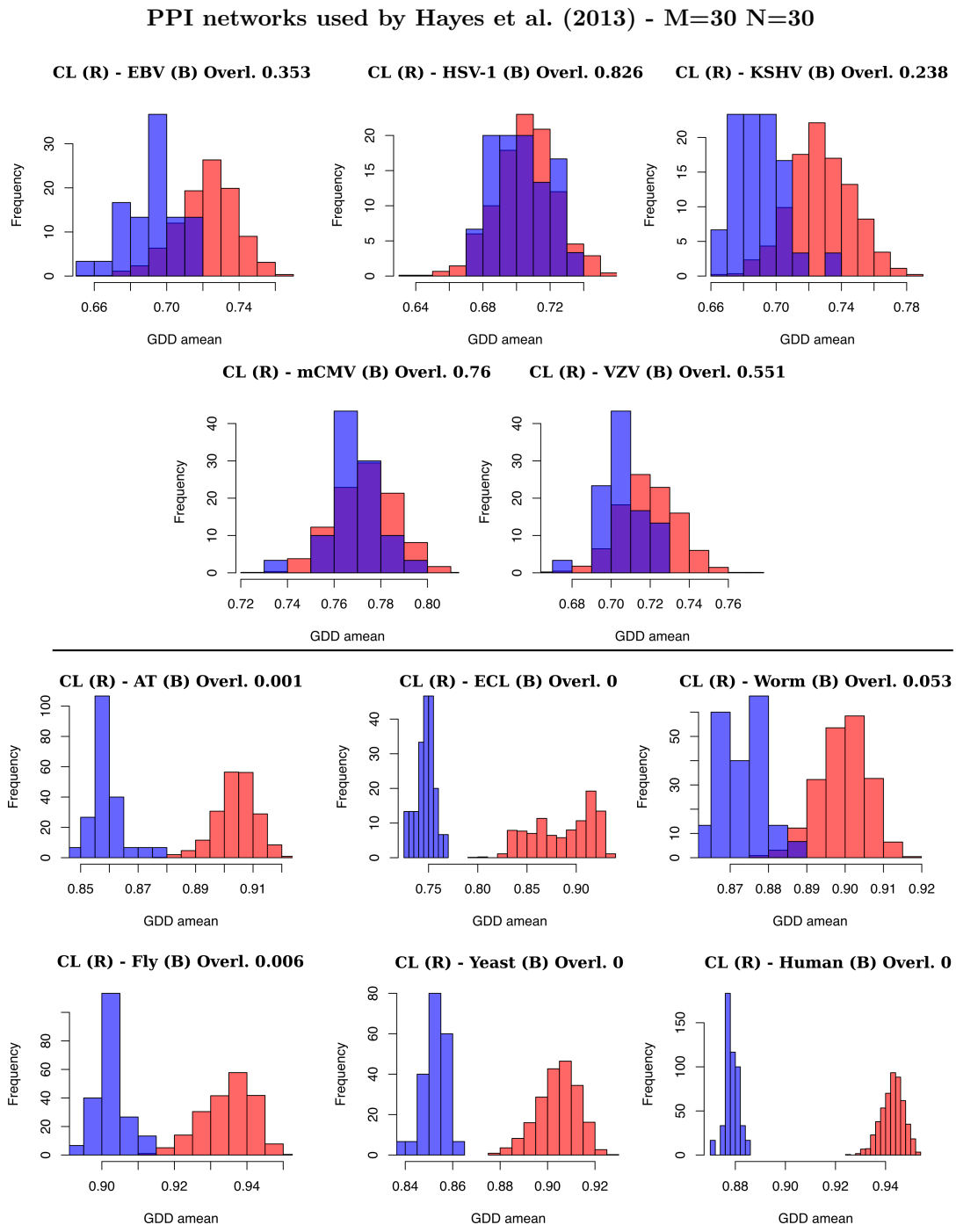


Figure 2.9: Histograms (Sturges rule) overlap of the GDDA values – data vs. model (blue) and model vs. model (red) (Chung-Lu model, CL)– obtained for the Monte Carlo test ($M = 30$, $N = 30$) of PPI networks shown in Table 2.9.

sample (Sturges, 1926) (Venables and Ripley, 2002, p. 112).

2.4 Unexpected behaviour of Monte Carlo test using background counts in Netdis

Among the different network comparison statistics considered in this chapter, and throughout this dissertation, Netdis is the only comparison method which uses background expectations of subgraph counts in order to provide more reliable comparisons among networks with different numbers of nodes and/or edges. However, applications of the Monte Carlo test via Netdis with background expectations led to unexpected results. For example, Figure 2.10 shows the percentage of times the null hypothesis "H_0 : G_0 is a realisation of model B" is not rejected, at $\alpha = 5\%$, from 100 realisations of the Monte Carlo test for the simulation scenarios mentioned in the previous Section 2.2, and which are:

(a) ER, Chung-Lu and DD networks generated with 1000 nodes and with approximate average degrees, (d), of 20, 15 and 11;

(b) ER, Chung-Lu and DD networks generated with 1500, 1000 and 500 nodes and with an approximate average degree of 15.

Contrary to the results obtained for the Netdis version without expectations, the version of Netdis using background expectations showed that in most of the test cases the probability of not rejecting the null hypothesis when it is false was at least 30%.

We found these results surprising, as we expected the results from Netdis using background expectations to have a better performance than without background expectations, as the background expectations should adjust the observed subgraph counts according to the disparity between the number of nodes and/or edges observed in the two networks.

Specifically, in each Monte Carlo test the background expectations were estimated by simulating one network Q from the model being tested, then extracting and binning all its 2-step ego-networks according to their edge density, and finally

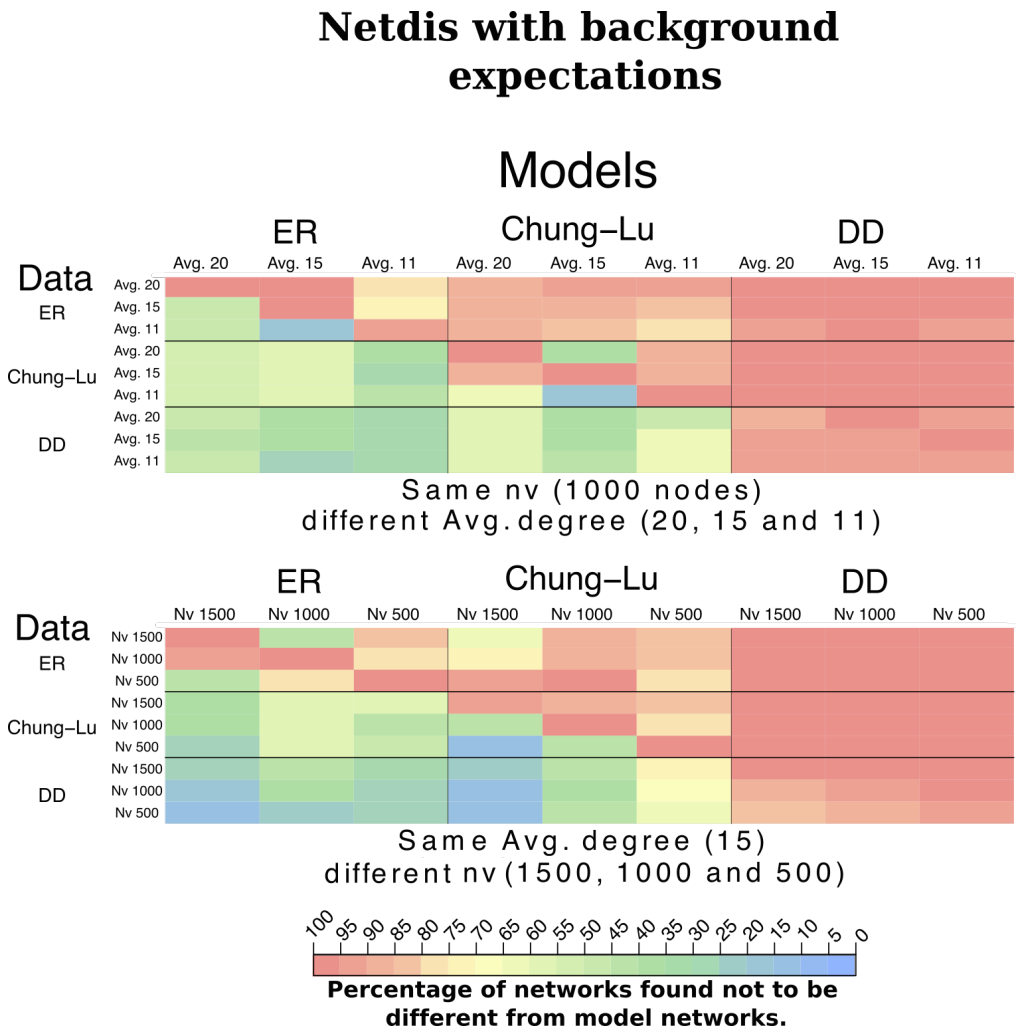


Figure 2.10: Following the simulation scenarios used in Section 2.2, here we show the percentage of times the null hypothesis "H_0 : G_0 is a realisation of model B" is not rejected, at $\alpha = 5\%$, from 100 realisations of the Monte Carlo test using Netdis with background expectations.

obtaining an estimate of the background counts by

$$E_w(Q, \rho) = \frac{1}{|\{i \in \{1, ..., q\} : d_i \approx \rho\}|} \sum_{\substack{i=1...q: \\ d_i \approx \rho}} \frac{N_{w,i}(Q)}{\binom{n_i}{k}},$$

where q is the number of ego-networks in density bin ρ, n_i the number of nodes in the 2-step ego-network of node i, k the size of the subgraph (3 to 5), and where $d_i \approx \rho$ means that the network density of ego-network i falls in density bin ρ.

We note that the results for the Monte Carlo test are not symmetric, as testing if a network drawn from an ER model can be considered as a realisation of the

Chung-Lu model; is not the same as testing if a network drawn from the Chung-Lu model can be considered as a realisation the ER model. This is particularly evident for the case where Netdis with background expectations is used, as on the first case a Chung-Lu network is used to estimate the background expectations, whereas in the second case an ER network is used to estimate the background expectations. We found that some Netdis comparisons between networks obtained from the same model from which the background expectations were obtained, were close to the maximum possible comparison value, 1. This particular behaviour of the comparison values consequently led to a broad and heavy-tailed null distribution of the test statistic used by the Monte Carlo test, $\bar{S}$, thus making the null hypothesis less likely to be rejected when it is false. Figure 2.11 shows the histogram of a sample of 100 values of the test statistic, $\bar{S} = \sum_{i=1}^{30} Netdis(G^0_{ER}, G^i_{ER})/30$, under the null hypothesis for ER networks, (G_{ER}), on 1000 nodes and with expected average degree 15.

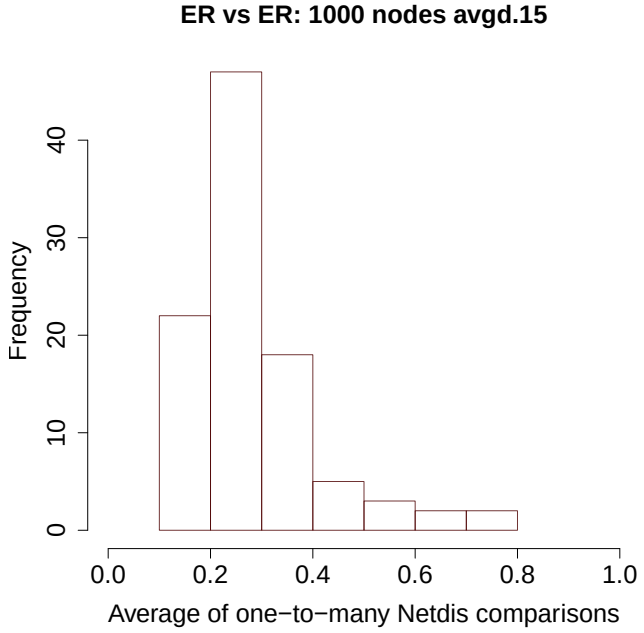


Figure 2.11: Histogram of a sample of the test statistic under the null hypothesis for ER networks on 1000 nodes and with expected average degree 15 i.e. a sample of 100 realisations of $\bar{S}$, where $\bar{S} = \sum_{i=1}^{30} Netdis(G^0_{ER}, G^i_{ER})/30$.

We would expect S to be small in order to detect that G^0_{ER} and G^i_{ER} are generated

from the same model. Figure 2.12 shows the histograms of independent one-to-one Netdis comparisons, $Netdis(G'_{ER}, G''_{ER})$, for ER networks with 1000, 10000 and 20000 nodes and with average degrees 10, 20, 40 and 80, and found that large values occur more frequently in networks with higher edge-densities.

We note that values larger than 0.5 can only occur when the expected subgraph counts, $(E_w(.))$, underestimate the subgraph counts, $(N_{w,.}(G))$, of one of the observed networks and, at the same time, overestimates the subgraph counts of the other network. Such scenario would lead to a positive $S_w(G)$ for one network and a negative $S_w(G)$ for the other network, and hence to a positive contribution in

$$Netdis(k) = 0.5 - \frac{1}{2\times\sqrt{M(k)}} \sum_{w\in A(k)} \left(\frac{S_w(G)S_w(H)}{\sqrt{S_w(G)^2+S_w(H)^2}} \right),$$

where

$$S_w(G) = \sum_i \left(N_{w,i}(G) - \binom{n_i}{k} E_w(\rho(i)) \right).$$

In Figure 2.12 we observed that larger comparison values occurred more frequently as the average degree of the networks increased. We conjecture that a pair of sparse networks generated with a larger average degree are more likely to contain ego-networks with subgraph counts at opposite sides of the expected number of subgraph counts $E_w(.)$, than networks generated with smaller average degrees (for which some subgraphs have a low probability of occurrence), thus increasing the likelihood of obtaining a negative value of

$$\sum_{w\in A(k)} \left(\frac{S_w(G)S_w(H)}{\sqrt{S_w(G)^2+S_w(H)^2}} \right),$$

and, therefore, leading to a Netdis value greater than 0.5, (see Appendix B for more details). However, if we consider that there is no interest in differentiating networks with the same absolute difference between their subgraph counts and their expected subgraph counts, Netdis could instead be constructed by taking

$$S_w(G) = \sum_i \left| N_{w,i}(G) - \binom{n_i}{k} E_w(\rho(i)) \right|.$$

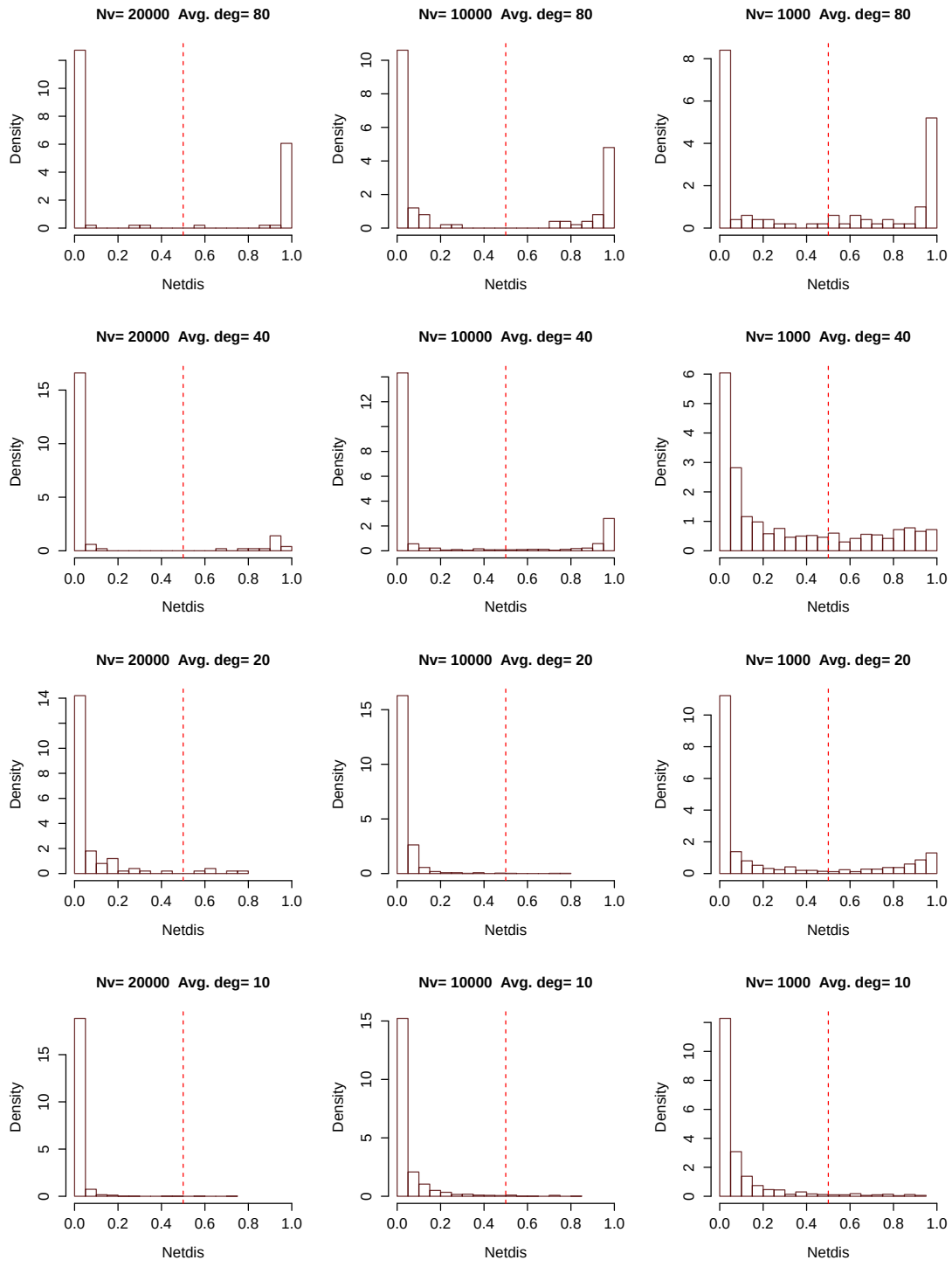


Figure 2.12: Histograms of independent one-to-one Netdis comparisons for ER networks with 1000, 10000 and 20000 nodes and with average degrees, $\bar{d}$, 10, 20, 40 and 80. It can be seen that as the edge-density increases, larger Netdis values occur more frequently.

Figure 2.13 shows the results obtained for this variation of Netdis for scenarios (a) and (b). It can be seen that the results perform better than the ones obtained for

the Netdis version that considered zero count expectations (Section 2.2). However, this version still displays dependence on the number of nodes and edges.

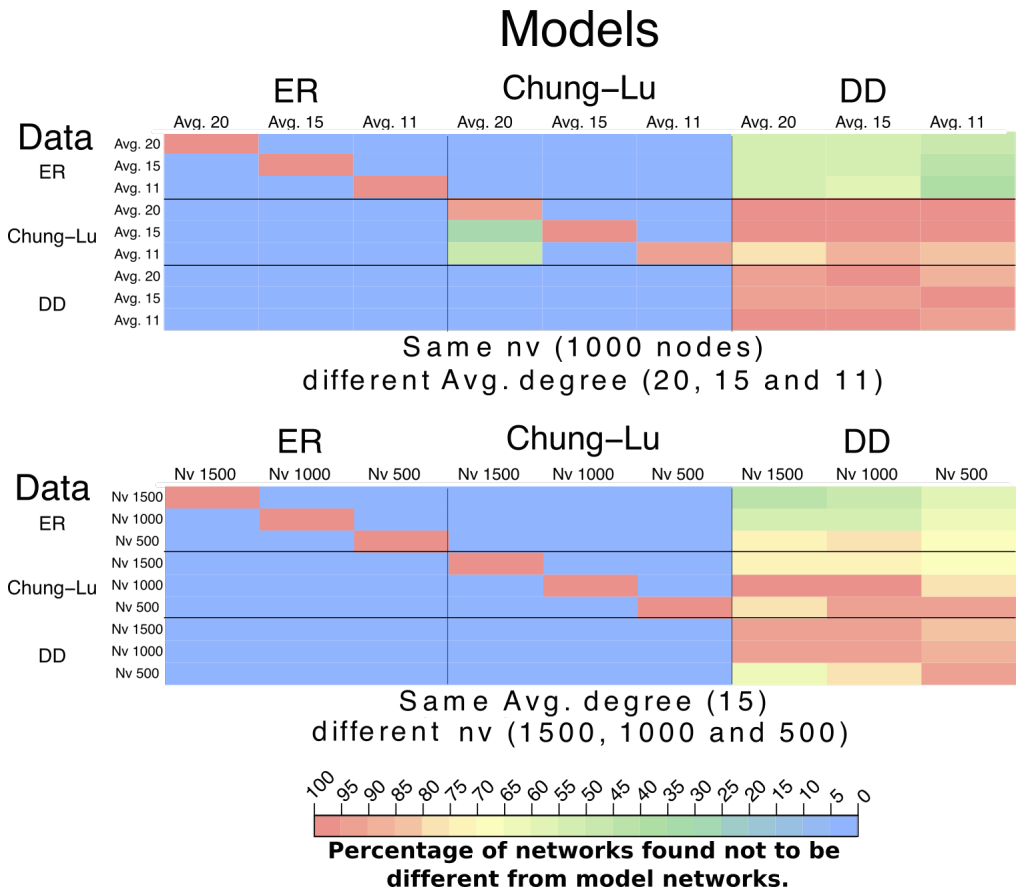


Figure 2.13: Results of Monte Carlo tests for Netdis, using $S_w(G) = \sum_i |N_{w,i}(G) - \binom{n_i}{k} E_w(\rho(i))|$.

2.5 Discussion

In this chapter, we have described three network comparison methodologies based on subgraph counts and one benchmark methodology based on network alignments. We have shown that performing a model selection task based on a threshold of the network comparison statistic GDDA, as suggested by Pržulj (2007), led to unreliable results. Instead we proposed to use a Monte Carlo framework that is well suited to the task of model selection.

Analysis of the type two error of the Monte Carlo test under different scenarios showed differences between the network comparison methods that could not have

been easily inferred from their definition or that were previously known. We found that GCD and Netdis focus on fine-grained differences while GDDA and Netal seem to focus on more coarser differences which allowed them to identify similar generation mechanisms between networks with a different number of nodes or edges. However, all methods showed not to be invariant to the number of nodes and edges present in networks. These results showed that there is still a need for a network comparison method that can better pick up similarities between networks based on their generation mechanisms or their building blocks, and which is invariant to the number of nodes and edges.

Lastly, we showed an unexpected behaviour from the Netdis version that implements background count expectations. Despite aiming to account for changes in the number of nodes and edges, Netdis performed badly in our analysis of the type two error of the Monte Carlo test. We also showed that this behaviour was linked to the occurrence of large comparison values, which came from comparisons of networks coming from the same model used to obtain the subgraph count expectations, and whose subgraph counts were on opposite sides of the expected subgraph counts.

Chapter 3

NetEmd

In the previous chapter we analysed different network comparison methods based on subgraph counts. Based on a model selection framework we found that these methods, despite aiming to measure 'closeness' between networks and using the same or similar inputs, provided different and in some cases contradictory conclusions. These results illustrated differences between the different network comparison methods that were not previously known, as for example their sensitivity to changes in the number of nodes and/or number of edges of the networks being compared.

Due to the dependence of the previous network comparison measures on the number of nodes and edges, in this chapter we propose a new network comparison method which can detect similarities across networks with different number of nodes and edges more reliably than the other network comparison methods based on subgraph counts.

This chapter contains the draft of an article titled "Identifying networks with common organisational principles" to be submitted as a journal article, and which is the result of collaborative work between myself, Dr. Anatol Wegner, Dr. Robert Gaunt, Professor Gesine Reinert and Professor Charlotte M. Deane. We worked together to develop the novel measure *NetEmd*. My contribution in terms of calculations was the development and implementation of the code necessary to provide the distributions of subgraph counts over networks, the distribution of the total subgraph counts over collections of ego-networks, and the code of the other net-

work comparison measures based on subgraph counts. Major contributions that I made jointly with the other authors of the paper are: Setup of computational experiments, i.e. selection of synthetic and real world datasets, selection of benchmarking network comparison methodologies, selection of statistics for the evaluation of a classification tasks, and the development of *NetEmd*. I had no major input in: the proof that NetEmd is a pseudo-metric (Section 3.2.2); the sub-sampling analysis of NetEmd (Section 3.2.4) and the implementation of C-support vector machines (Section 3.2.6). Dr. Anatol Wegner carried out the computations and comparisons using my code and his implementation of NetEmd and the implementation of C-support vector machines.

The remainder of this chapter is arranged as follows: Section 3.1 contains the main text of the paper following the submission guidelines of the Journal of Complex Networks, and Section 3.2 contains the appendix of the paper.

There is some overlap between the content of Sections 3.1 and 3.2, and the content of the previous Chapters 1 and 2 regarding the description of network comparison methods and random graph models. This overlap is intended to make clear what are the pages that form the paper to be submitted as a journal article.

A copy of the paper can be found in the arXiv at `https://arxiv.org/abs/1704.00387`.

3.1 Identifying networks with common organizational principles

Many complex systems can be represented as networks, and the problem of network comparison is becoming increasingly relevant. There are many techniques for network comparison, from simply comparing network summary statistics to sophisticated but computationally costly alignment-based approaches. Yet it remains challenging to accurately cluster networks that are of a different size and density, but hypothesized to be structurally similar. In this paper, we address this problem by introducing a new network comparison methodology that is aimed at identi-

fying common organizational principles in networks. The methodology is simple, intuitive and applicable in a wide variety of settings ranging from the functional classification of proteins to tracking the evolution of a world trade network.

Keywords: networks | network comparison | machine learning | earth mover's distance | network topology

3.1.1 Introduction

Many complex systems can be represented as networks, including friendships, the World Wide Web, global trade flows and protein-protein interactions (Newman, 2010). The study of networks has been a very active area of research in recent years, and in particular, network comparison has become increasingly relevant (e.g. Wilson and Zhu, 2008; Neyshabur et al., 2013; Ali et al., 2014; Yaveroglu et al., 2014). Network comparison itself has many wide-ranging applications, for example, comparing protein-protein interaction networks could lead to increased understanding of underlying biological processes (Ali et al., 2014). Network comparison can also be used to study the evolution of networks over time and for identifying sudden changes and shocks.

Network comparison methods have attracted increasing attention in the field of machine learning, where they are mostly referred to as graph kernels, and have numerous applications in personalised medicine (e.g. Borgwardt et al., 2007), computer vision and drug discovery (e.g. Wale et al., 2008). In the machine learning setting, the problem of interest is to obtain classifiers that can accurately predict the class membership of graphs.

Methods for comparing networks range from comparison of summary statistics to sophisticated but computationally expensive alignment-based approaches (Kuchaiev and Pržulj, 2011; Neyshabur et al., 2013; Mamano and Hayes, 2017). Real-world networks can be very large and are often inhomogeneous, which makes the problem of network comparison challenging, especially when networks differ significantly in terms of size and density. In this paper, we address this problem by introducing a new network comparison methodology that is aimed at comparing networks

according to their common organizational principles.

The observation that the degree distribution of many real world networks is highly right skewed and in many cases approximately follows a power law has been very influential in the development of network science (Barabási and Albert, 1999). Consequently, it has become widely accepted that the shape of the degree distribution (for example, binomial vs. power law) is indicative of the generating mechanism underlying the network. In this paper, we formalize this idea by introducing a measure that captures the shape of distributions. The measure emerges from the requirement that a metric between forms of distributions should be invariant under rescalings and translations of the observables. Based on this measure, we then introduce a new network comparison methodology, which we call *NetEmd*.

Although our methodology is applicable to almost any type of feature that can be associated to nodes or edges of a graph, we focus mainly on distributions of small connected subgraphs, also known as graphlets. Graphlets form the basis of many of the state-of-the-art network comparison methods (Pržulj, 2007; Ali et al., 2014; Yaveroglu et al., 2014) and hence using graphlet based features allows for a comparative assessment of the presented methodology. Moreover, certain graphlets, called network motifs (Milo et al., 2002), occur much more frequently in many real world networks than is expected on the basis of pure chance. Network motifs are considered to be basic building blocks of networks that contribute to the function of the network by performing modular tasks and have therefore been conjectured to be favoured by natural selection. This is supported by the observation that network motifs are largely conserved within classes of networks (Milo et al., 2004; Wegner, 2014).

Our methodology provides an effective tool for comparing networks even when networks differ significantly in size and density, which is the case in most applications. The methodology performs well on a wide variety of networks ranging from chemical compounds having as few as 10 nodes to tens of thousands of nodes in internet networks. The method achieves state-of-the-art performance even when it is based on rather restricted sets of inputs that can be computed efficiently and hence scales favourably to networks with millions and even billions of nodes. The method also

behaves well under network sub-sampling as described in Ali et al. (2016). The methodology further meets the needs of researchers from a variety of fields, from the social sciences to the biological and life sciences, by being computationally efficient and simple to implement.

We test the presented methodology in a large number of settings, starting with clustering synthetic and real world networks, where we find that the presented methodology outperforms state-of-the-art graphlet-based network comparison methods in clustering networks of different sizes and densities. We then test the more fine grained properties of *NetEmd* using data sets that represent evolving networks at different points in time. Finally, we test whether *NetEmd* can predict functional categories of networks by exploring machine learning applications and find that classifiers based on *NetEmd* outperform state-of-the-art graph classifiers on several benchmark data sets.

3.1.2 A measure for comparing forms of distributions

Here we build on the idea that the information encapsulated in the shape of the degree distribution and other network properties reflects the topological organization of the network. From an abstract point of view we think of the shape of a distribution as a property that is invariant under linear deformations i.e. translations and re-scalings of the axis. For example, a Gaussian distribution always has its characteristic bell curve shape regardless of its mean and standard deviation. Consequently, we postulate that any metric that aims to capture the similarity of shapes should be invariant under linear transformations of its inputs.

Based on these ideas we define the following measure between distributions p and q that are supported on $\mathbb{R}$ and have non-zero, finite variances:

$$EMD^*(p,q) = \inf_{c\in\mathbb{R}} \left(EMD\big(\tilde{p}(\cdot + c), \tilde{q}(\cdot)\big)\right), \tag{3.1}$$

where EMD is the earth mover's distance and $\tilde{p}$ and $\tilde{q}$ are the distributions obtained by rescaling p and q to have variance 1. More precisely, $\tilde{p}$ is the distribution obtained from p by the transformation $x \to \frac{x}{\sigma(p)}$, where $\sigma(p)$ is the standard devi-

ation of p. Intuitively, EMD (also known as the 1st Wasserstein metric (Runber et al., 1998) can be thought of as the minimal work, i.e. mass times distance, needed to "transport" the mass of one distribution onto the other. For probability distributions p and q with support in $\mathbb{R}$ and bounded absolute first moment, the EMD between p and q is given by $EMD(p,q) = \int_{-\infty}^{\infty} |F(x) - G(x)| \, \mathrm{d}x$, where F and G are the cumulative distribution functions of p and q respectively.

In principle, EMD in Equation (3.1) can be replaced by almost any other probability metric d to obtain a corresponding metric d^*. Here we choose EMD because it is well suited to comparing shapes, as shown by its many applications in the area of pattern recognition and image retrieval (Runber et al., 1998). Moreover, we found that EMD produces superior results to classical L^1 and Kolmogorov distances, especially for highly irregular distributions that one frequently encounters in real world networks.

For two networks G and G' and given network feature t, we define the corresponding $NetEmd_t$ measure by:

$$NetEmd_t(G, G') = EMD^*(p_t(G), p_t(G')), \tag{3.2}$$

where $p_t(G)$ and $p_t(G')$ are the distributions of t on G and G' respectively. $NetEmd_t$ can be shown to be a pseudometric between graphs for any feature t (see Section 3.2.2), that is it is non-negative, symmetric and satisfies the triangle inequality. Figure 3.1 gives examples where t is taken to be the degree distribution, and $p_t(G)$ is the degree distribution of G.

Measures that are based on the comparison of multiple features can be expected to be more effective at identifying structural differences between networks than measures that are based on a single feature t, because for two networks to be considered similar they must show similarity across multiple features. Hence, for a given set $T = \{t_1, t_2, ..., t_m\}$ of network features, we define the $NetEmd$ measure corresponding to T simply as:

$$NetEmd_T(G, G') = \frac{1}{m} \sum_{j=1}^{m} NetEmd_{t_j}(G, G'). \tag{3.3}$$

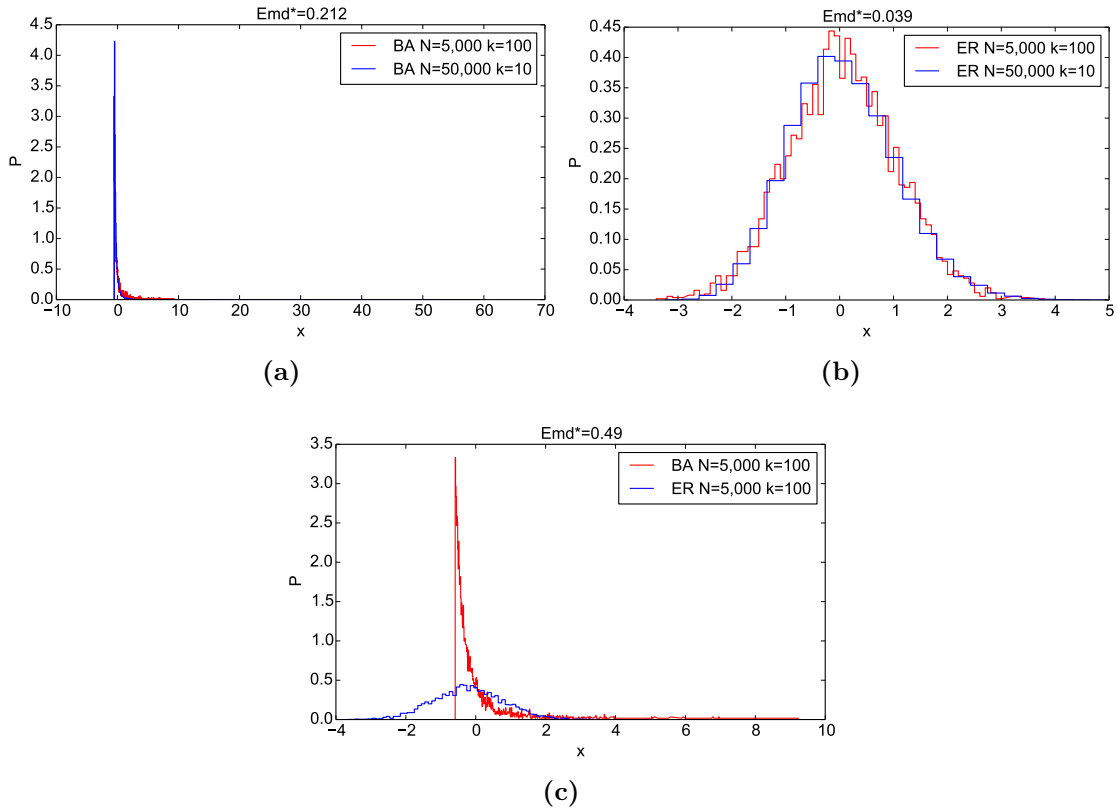


Figure 3.1: Plots of rescaled and translated degree distributions for Barabasi-Albert (BA) and Erdős-Rényi (ER) models with N nodes and average degree k: (a) BA $N = 5,000$, $k = 100$ vs BA $N = 50,000$, $k = 10$. (b) ER $N = 5,000$, $k = 100$ vs ER $N = 50,000$, $k = 10$. (c) BA $N = 5,000$, $k = 100$ vs ER $N = 5,000$, $k = 100$. The EMD^* distances between the degree distribution of two BA or ER models with quite different values of N and k are smaller than the EMD^* distance between the degree distribution of a BA and ER model when the number of nodes and average degree are equal.

Although *NetEmd* can in principle be based on any set T of network features to which one can associate distributions, we initially consider only features that are based on distributions of small connected subgraphs, also known as graphlets. Graphlets form the basis of many state-of-the-art network comparison methods and hence allow for a comparative assessment of the proposed methodology.

First, we consider graphlet degree distributions (GDDs) (Pržulj, 2007) as our set of features. For a given graphlet m, the graphlet degree of a node is the number of graphlet-m induced subgraphs that are attached to the node. One can distinguish between the different positions the node can have in m, which correspond to the automorphism orbits of m, see Figure 3.2. For graphlets up to size 5 there are 73 such orbits. We initially take the set of 73 GDDs corresponding to graphlets up to

size 5 to be the default set of inputs, for which we denote the metric as $NetEmd_{G5}$. Later we also explore alternative definitions of subgraph distributions based on ego networks, as well as the effect of varying the size of subgraphs considered in the input. Finally, we consider the eigenvalue spectra of the graph Laplacian and the normalized graph Laplacian as inputs.

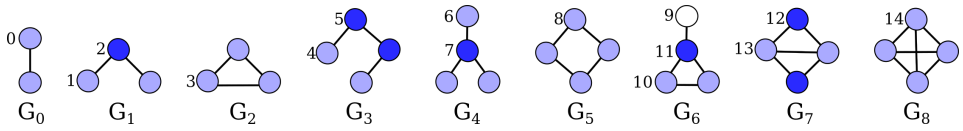


Figure 3.2: Graphlets on two to four nodes. The different shades in each graphlet represent different automorphism orbits, numbered from 0 to 14.

3.1.3 Results

In order to give a comparative assessment of *NetEmd*, we consider other graphlet based network comparison methods, namely *GDDA* (Pržulj, 2007), *GCD* (Yaveroglu et al., 2014) and Netdis (Ali et al., 2014). These represent the most effective alignment-free network comparison methodologies in the existing literature. While *GDDA* directly compares distributions of graphlets up to size 5 in a pairwise fashion, *GCD* is based on comparing rank correlations between graphlet degrees. Here we consider both default settings of GCD (Yaveroglu et al., 2014), namely *GCD*11, which is based on a non-redundant subset of 11 graphlets up to size 4, and *GCD*73 which uses all graphlets up to size 5. *Netdis* differs from *GDDA* and *GCD* in that it is based on subgraph counts in ego-networks of nodes. Another important distinction is that *Netdis* first centers these raw counts by comparing them to the counts that could be expected under a particular null model before computing the final statistics. In our analysis, we consider two null models: an Erdös-Rényi random graph and a duplication divergence (Vázquez et al., 2003) graph which has a scale-free degree distribution as well as a high clustering coefficient. We denote these two variants as $Netdis_{ER}$ and $Netdis_{SF}$ respectively.

Clustering synthetic and real world networks

We start with the classical setting of network comparison where the task is to identify groups of structurally similar networks. The main challenge in this setting is to identify structurally similar networks even though they might differ substantially in terms of size and density.

Given a set $S = \{G_1, G_2, ..., G_n\}$ of networks consisting of disjoint classes $C = \{c_1, c_2, ..., c_m\}$ one would like a network comparison measure d to position networks from the same class closer to each other when compared to networks from other classes. Given a network G, this can be measured in terms of the empirical probability $P(G)$ that $d(G, G_1) < d(G, G_2)$ where G_1 is a randomly selected network from the same class as G (excluding itself) and G_2 is a randomly selected network from outside the class of G and d is the network comparison statistic. Consequently, the performance over the whole data set is measured in terms of the quantity $\overline{P} = \frac{1}{|S|} \sum_{G \in S} P(G)$. It can be shown that $\overline{P}$ is related to the average area under the receiver operator characteristic curve of a classifier that for a given network G classifies the k nearest neighbours of G with respect to d as being similar to G. This relation is more clear when considering that the area under the ROC curve (AUC) is equivalent to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance (Fawcett, 2006). Hence, a measure that positions networks randomly has an expected $\overline{P}$ of 0.5 whereas $\overline{P} = 1$ corresponds to perfect separation between classes. Other measures are discussed in the Appendix. Conclusions reached in this paper hold regardless of which performance measure one uses.

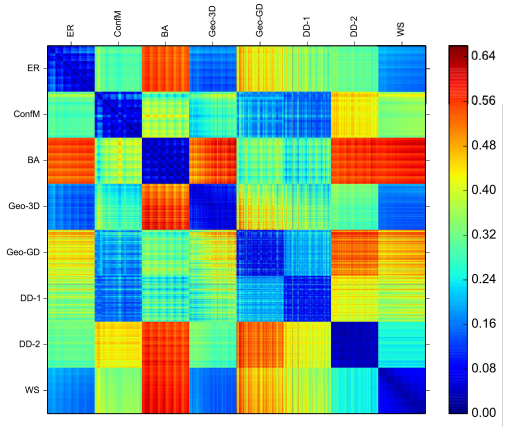
We first test *NetEmd* on synthetic networks corresponding to realizations of eight random graph models, namely the Erdős-Rényi random graphs (Gilbert, 1959), the Barabasi Albert preferential attachment model (Barabási and Albert, 1999), two duplication divergence models (Vázquez et al., 2003; Ispolatov et al., 2005), the geometric gene duplication model (Higham et al., 2008), 3D geometric random graphs (Penrose, 2003), the configuration model (Molloy and Reed, 1995), and Watts-Strogatz small world networks (Watts and Strogatz, 1998) (see Section 3.2.7

in the Appendix for details).

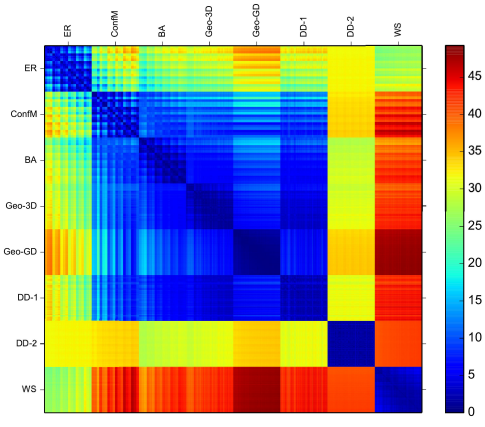
For synthetic networks we consider three experimental settings of increasing difficulty, starting with the task of clustering networks that have same size N and average degree k according to generating mechanism - a task that is relevant in a model selection setting. For this we generate 16 data sets, which collectively we call RG_1, corresponding to combinations of $N \in \{1250, 2500, 5000, 10000\}$ and $k \in \{10, 20, 40, 80\}$, each containing 10 realizations per model, i.e. 80 networks. This is an easier problem than clustering networks of different sizes and densities, and in this setting we find that the $\overline{P}$ scores (see Table 3.3c) of top performing measures tend to be within one standard deviation of each other. We find that $NetEmd_{G5}$ and $GCD73$ achieve the highest scores, followed by $GCD11$ and $Netdis_{SF}$.

Having established that $NetEmd$ is able to differentiate networks according to generating mechanism, we move on to the task of clustering networks of different sizes and densities. For this we generate two data sets: RG_2 in which the size N and average degree k are increased independently in linear steps to twice their initial value ($N \in \{2000, 3000, 4000\}$ and $k \in \{20, 24, 28, 32, 36, 40\}$) and RG_3 in which the size and average degree are increased independently in multiples of 2 to 8 times their initial value ($N \in \{1250, 2500, 5000, 10000\}$ and $k \in \{10, 20, 40, 80\}$). In RG_3, the number of nodes and average degrees of the networks both vary by one order of magnitude, and therefore clustering according to model type is challenging. Both RG_2 and RG_3 contain 10 realizations per model parameter i.e. contain $3 \times 6 \times 8 \times 10 = 1440$ and $4 \times 4 \times 8 \times 10 = 1280$ networks, respectively. Finally, we consider a data set consisting of networks from 10 different classes of real world networks (RWN) as well as a data set from (Ali et al., 2014) that consists of real world and synthetic networks from the larger collection compiled by Onnela *et al.* (Onnela et al., 2012).

We find that $NetEmd_{G5}$ outperforms all of the other three methods at clustering networks of different sizes and densities on all data sets. The difference can also be seen in the heatmaps of $NetEmd_{G5}$ and $GCD73$, the second best performing method for RG_2, given in Figures 3.3a and 3.3b. While the heatmap of $NetEmd_{G5}$



(a) Heatmap of $NetEmd_{G5}$ for RG_2.



(b) Heatmap of $GCD73$ for RG_2.

Dataset	$NetEmd_{G5}$	$Netdis_{ER}$	$Netdis_{SF}$	$GCD11$	$GCD73$	$GDDA$
Synthetic Networks						
RG_1	**0.997±0.003**	0.981±0.013	0.986±0.011	0.992±0.012	0.996±0.005	0.952±0.056
RG_2	**0.988**	0.897	0.919	0.976	0.976	0.956
RG_3	**0.925**	0.790	0.800	0.872	0.861	0.812
RWN	**0.942**	0.898	0.866	0.898	0.906	0.745
Onnela et al.	**0.890**	0.832	0.809	0.789	0.819	0.783

(c) $\overline{P}$ values for different network measures on data sets of synthetic and real world networks.

Figure 3.3: (a) and (b) show the heatmaps of pairwise distances on RG_2 ($N \in \{2000, 3000, 4000\}$ and $k \in \{20, 24, 28, 32, 36, 40\}$) according to $NetEmd_{G5}$ and $GCD73$, respectively. In the heat map, networks are ordered from top to bottom in the following order: model, average degree and node count. The heatmap of $NetEmd$ shows eight clearly identifiable blocks on the diagonal corresponding to different generative models while the heatmap of $GCD73$ shows signs of off-diagonal mixing. (c) $\overline{P}$ values for various comparison measures for data sets of synthetic and real world networks. For RG_1 we calculated the value of $\overline{P}$ for each of the 16 sub-data sets. The table shows the average and standard deviation of the $\overline{P}$ values obtained over these 16 sub-data sets.

shows eight clearly identifiable blocks on the diagonal corresponding to different generative models, the heatmap of $GCD73$ shows signs of off-diagonal mixing. The difference in performance becomes even more pronounced on more challenging data sets, i.e. on RG_3 (see Figure 3.6 in the Appendix) and the Onnela *et al.* data set.

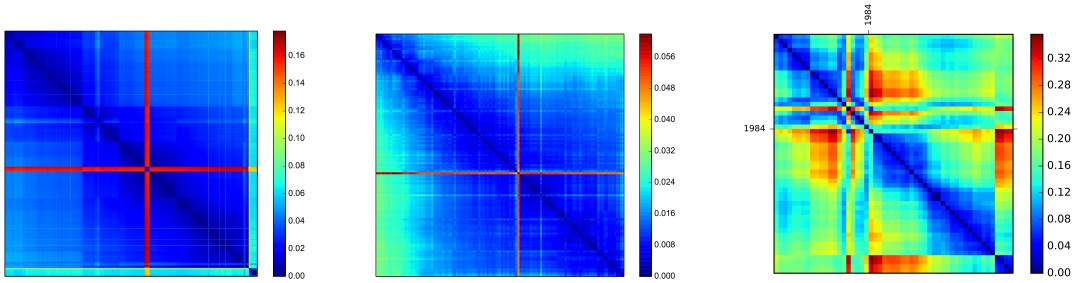
Time ordered networks

A network comparison measure should ideally not only be able to identify groups of similar networks but should also be able to capture structural similarity at a finer local scale. To study the behavior of $NetEmd$ at a more local level, we consider data sets that represent a system measured at different points in time. Since such

networks can be assumed to evolve gradually over time they offer an ideal setting for testing the local properties of network comparison methodologies.

We consider two data sets, named AS-caida and AS-733 (Leskovec et al., 2005), that represent the topology of the internet at the level of autonomous systems and a third data set that consists of bilateral trade flows between countries for the years 1962–2014 (Feenstra et al., 2005; Division, 2015). Both edges and nodes are added and deleted over time in all three data sets. As was noted in (Leskovec et al., 2005) the time ranking in evolving networks is reflected to a certain degree in simple summary statistics. Hence, recovering the time ranking of evolving networks should be regarded as a test of consistency rather than an evaluation of performance.

In order to minimize the dependence of our results on the algorithm that is used to rank networks, we consider four different ways of ranking networks based on their pairwise distances as follows. We assume that either the first or last network in the time series is given. Rankings are then constructed in a step-wise fashion. At each step one either adds the network that is closest to the last added network (Algorithm 1), or adds the network that has smallest average distance to all the networks in the ranking constructed so far (Algorithm 2). The performance of a measure in ranking networks is then measured in terms of Kendall's rank correlation coefficient τ between the true time ranking and the best ranking obtained by any of the 4 methods. We find that $NetEmd_{G5}$ successfully recovers the time ordering for all three data sets, as can be seen in the time ordered heatmaps given in Figure 3.4 which all show clear groupings along the diagonal. The red regions in the two internet data sets correspond to outliers which can also be identified as sudden jumps in summary statistics e.g. the number of nodes. The two large clusters in the heatmap of world trade networks (Figure 3.4c) coincide with a change in the data gathering methodology in 1984 (Feenstra et al., 2005). Although $NetEmd_{G5}$ comes second to $Netdis_{SF}$ on AS-733 and to $GCD11$ on AS-caida, $NetEmd_{G5}$ has the highest overall score and is the only measure that achieves consistently high scores on all three data sets.



(a) Heatmap of $NetEmd_{G5}$ for AS-733

(b) Heatmap of $NetEmd_{G5}$ for AS-caida

(c) Heatmap of $NetEmd_{G5}$ for world trade networks.

Dataset	$NetEmd_{G5}$	$Netdis_{ER}$	$Netdis_{SF}$	$GCD11$	$GCD73$	$GDDA$
AS-733	0.874	0.867	**0.933**	0.763	0.770	0.740
AS-caida	0.890	0.844	0.849	**0.897**	0.878	0.870
World Trade	**0.821**	0.666	0.388	0.380	0.567	0.649

(d) Kendall's τ between the true time ranking and rankings inferred from network comparison methodologies.

Figure 3.4: (a), (b) & (c) Heatmaps of $NetEmd_{G5}$ for networks representing the internet at the level of autonomous systems networks and world trade networks. The date of measurement increases from left to right/ top to bottom. $NetEmd_{G5}$ accurately captures the evolution over time in all three data sets by positioning networks that are close in time closer to each other resulting in a clear signal along the diagonal. (d) Kendall's rank correlation coefficient between the true time ranking and rankings inferred from different network comparison measures.

NetEmd based on different sets of inputs

We examine the effect of reducing the size of graphlets considered in the input of $NetEmd$, which is also relevant from a computational point of view, since enumerating graphlets up to size 5 can be challenging for very large networks. We consider variants based on the graphlet degree distributions of graphlets up to size 3 and 4, which we denote as $NetEmd_{G3}$ and $NetEmd_{G4}$. We also consider $NetEmd_{DD}$ which is based only on the degree distribution as a baseline. Results are given in Table 3.1.

We find that reducing the size of graphlets from 5 to 4 does not significantly decrease the performance of $NetEmd$ and actually produces better results on three data sets (RG_3, Real world and Onnela et *al.*). Even when based on only graphlets up to size 3, i.e. just edges, 2-paths and triangles, $NetEmd$ outperforms all other non-$NetEmd$ methods that we tested on at least 6 out of 8 data sets.

Given that the complexity of enumerating graphlets up to size s in a network on

N nodes having maximum degree k_{mak} is $O(Nk_{max}^{s-1})$, $NetEmd_{G4}$ offers an optimal combination of performance and computational efficiency in most cases. The even less computationally costly $NetEmd_{G3}$ scales favourably even to networks of billions of edges for which enumerating graphlets of size 4 can be computationally prohibitive. This opens the door for comparing very large networks which are outside the reach of current methods while still retaining state-of-the-art performance. Furthermore, the $NetEmd$ measures perform well under sub-sampling of nodes (Ali et al., 2016) (see Section 3.2.4 in the Appendix) which can be leveraged to further improve computational efficiency.

We find that in some cases restricting the set of inputs actually leads to an increase in the performance of $NetEmd$. This indicates that not all graphlet distributions are equally informative in all settings (Maugis et al., 2017). Consequently, identifying (learning) which graphlet distributions contain the most pertinent information for a given task might lead to significant improvements in performance. Such generalizations can be incorporated into $NetEmd$ in a straightforward manner, for instance by modifying the sum in Equation 3.3 to incorporate weights. $NetEmd$ is ideally suited for such metric learning (Xing et al., 2002) type generalizations since it constructs an individual distance for each graphlet distribution. Moreover, such single feature $NetEmd$ measures are in many cases highly informative even on their own. For instance $NetEmd_{DD}$, which only uses the degree distribution, outperforms the non-$NetEmd$ measures we tested individually on more than half the data sets we considered.

We also considered counts of graphlets up to size 4 in 1-step ego networks of nodes ($NetEmd_{E4}$) (Ali et al., 2014) as an alternative way of capturing subgraph distributions, for which we denote the measure as $NetEmd_{E4}$. Although we find that $NetEmd_{E4}$ achieves consistently high scores, we find that variants based on graphlet degree distributions tend to perform better on most data sets.

Finally, we consider spectral distributions of graphs as a possible alternative to graphlet based features. The spectra of various graph operators are closely related to topological properties of graphs (Chung, 1997; Mohar et al., 1991; Banerjee and Jost, 2008) and have been widely used to characterize and compare graphs (Gu

et al., 2016; Wilson and Zhu, 2008). We used the spectra of the graph Laplacian and normalized graph Laplacian as inputs for $NetEmd$ for which we denote the measure as $NetEmd_S$. For a given graph the Laplacian is defined as $L = D - A$ where A is the adjacency matrix of the graph and D is the diagonal matrix whose diagonal entries are the node degrees. The normalized Laplacian $\hat{L}$ is defined as $D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$. Given the eigenvalue distributions $S(L)$ and $S(\hat{L})$ of L and $\hat{L}$ we define $NetEmd_S$ to be $\frac{1}{2}(NetEmd_{S(L)} + NetEmd_{S(\hat{L})})$.

We find that in general $NetEmd_S$ performs better in clustering random graphs of different sizes and densities when compared to graphlet based network comparison measures. However, on the RWN and Onnela et al. data sets graphlet based $NetEmd$ measures tend to perform better than the spectral variant which can be attributed to the prevalence of network motifs in real world networks, giving graphlet based measures an advantage. The spectral variant is also outperformed on the time ordering of data sets which in turn might be a result of the sensitivity of graph spectra to small changes in the underlying graph (Wilson and Zhu, 2008).

Data set	$NetEmd_{G3}$	$NetEmd_{G4}$	$NetEmd_{E4}$	$NetEmd_S$	$NetEmd_{DD}$
RG_1	0.989±0.008	0.995±0.005	0.993±0.004	0.992±0.007	0.957±0.024
RG_2	0.982	0.987	0.983	**0.992**	0.944
RG_3	0.940	0.941	0.947	**0.972**	0.902
RWN	**0.952**	0.950	0.933	0.933	0.907
Onnela et al.	0.892	**0.898**	0.892	0.858	0.867
AS-733	0.808	0.874	0.922	0.855	0.928
AS-caida	**0.898**	0.892	0.820	0.780	0.821
World Trade	0.697	0.785	0.665	0.430	0.358

Table 3.1: Results for different variants of $NetEmd$ based on distributions of graphlets up to size 3 and 4 ($NetEmd_{G3}$ and $NetEmd_{G4}$), counts of graphlets up to size 4 in 1-step ego networks of nodes ($NetEmd_{E4}$), eigenvalue spectra of Laplacian operators ($NetEmd_s$) and the degree distribution ($NetEmd_{DD}$). Values in bold indicate that a measure achieves the highest score among all measures considered in the manuscript. For RG_1 we calculate the value of $\overline{P}$ for each of the 16 sub-data sets. The table shows the average and standard deviation of the $\overline{P}$ values obtained over these 16 sub-data sets.

Functional classification of networks

One of the primary motivations in studying the structure of networks is to identify topological features that can be related to the function of a network. In the context

of network comparison this translates into the problem of finding metrics that can identify functionally similar networks based on their topological structure.

In order to test whether *NetEmd* can be used to identify functionally similar networks, we use several benchmarks from the machine learning literature where graph similarity measures, called graph kernels, have been intensively studied over the past decade. In the context of machine learning the goal is to construct classifiers that can accurately predict the class membership of unknown graphs.

We test *NetEmd* on benchmark data sets representing social networks (Yanardag and Vishwanathan, 2015) consisting of Reddit posts, scientific collaborations and ego networks in the Internet Movie Database (IMDB). The Reddit data sets Reddit-Binary, Reddit-Multi-5k and Reddit-Multi-12k consist of networks representing Reddit threads where nodes correspond to users and two users are connected whenever one responded to the other's comments. While for the Reddit-Binary data sets the task is to classify networks into discussion based and question/answer based communities, in the data sets Reddit-Multi-5k and Reddit-Multi-12k the task is to classify networks according to their subreddit categories. COLLAB is a data set consisting of ego-networks of scientists from the fields High Energy Physics, Condensed Matter Physics and Astro Physics and the task is to determine which of these fields a given researcher belongs to. Similarly, the data sets IMDB-Binary and IMDB-Multi represent collaborations between film actors derived from the IMDB and the task is to classify ego-networks into different genres i.e. action and romance in the case of IMDB-Binary and comedy, action and Sci-Fi genres in the case of IMDB-Multi.

We use C - support vector machine (C-SVM) (Cortes and Vapnik, 1995) classifiers with a Gaussian kernel $K(G, G') = \exp(-\alpha NetEmd(G, G')^2)$, where α is a free parameter to be learned during training. Performance evaluation is carried out by 10 fold cross validation, where at each step of the validation 9 folds are used for training and 1 fold for evaluation. Free parameters of classifiers are learned via 10 fold cross validation on the training data only. Finally, every experiment is repeated 10 fold and average prediction accuracy and standard deviation are reported.

Kernel	Reddit-Binary	Reddit-Multi-5k	Reddit-Multi-12k	COLLAB	IMDB-Binary	IMDB-Multi
$NetEmd_{G5}$	**92.67 ±0.30**	**54.61±0.18**	**48.09±0.21**	**79.32±0.27**	66.99 ±1.19	41.45±0.70
$NetEmd_S$	88.59 ±0.35	53.05±0.34	44.45±0.18	**79.05±0.20**	**71.68±0.88**	**46.06±0.50**
DGK	78.04±0.39	41.27±0.18	32.22±0.10	73.09 ±0.25	66.96 ±0.56	44.55 ±0.52
GK	77.34±0.18	41.01±0.17	31.82±0.08	72.84 ±0.28	65.87 ±0.98	43.89 ±0.38
RF	88.7± 1.99	50.9 ±2.07	42.7 ±1.28	76.5±1.68	**72.4 ±4.68**	**47.8 ±3.55**
PCSN	86.30±1.58	49.10±0.70	41.32±0.42	72.60±2.15	**71.00±2.29**	**45.23±2.84**

Table 3.2: 10 fold cross validation accuracies of Gaussian kernels based on $NetEmd$ measures using the distributions of graphlets up to size 5 ($NetEmd_{G5}$) and Laplacian spectra ($NetEmd_S$) and other graph kernels, namely the deep graphlet kernels (DGK)(Yanardag and Vishwanathan, 2015) and the graphlet kernel (GK) (Shervashidze et al., 2009). We also consider alternatives to support vector machines classifiers, namely the random forest classifiers (RF) introduced in (Barnett et al., 2016) and convolutional neural networks (PCSN) (Niepert et al., 2016). Values in bold correspond to significantly higher scores, which are scores with t-test p-values less than 0.05 when compared to the highest score.

Table 3.2 gives classification accuracies obtained using $NetEmd$ measures based on graphlets up to size five ($NetEmd_{G5}$) and spectra of Laplacian operators ($NetEmd_S$) on the data sets representing social networks. We compare $NetEmd$ based kernels to graphlet kernels (Shervashidze et al., 2009) and deep graphlet kernels (Yanardag and Vishwanathan, 2015) as well as two non-SVM classifiers namely the random forest classifier introduced in (Barnett et al., 2016) and the convolutional neural network based classifier introduced in (Niepert et al., 2016).

On the Reddit data sets and the COLLAB data set, $NetEmd_{G5}$ significantly outperforms other state-of-the-art graph classifiers. On the other hand, we find that $NetEmd_{G5}$ performs poorly on the IMDB data sets. This can be traced back to the large number of complete graphs present in the IMDB data sets: 139 out of the 1000 graphs in IMDB-Binary and 789 out of 1500 graphs in IMDB-Multi are complete graphs which correspond to ego-networks of actors having acted only in a single film. By definition, $NetEmd_{G5}$ cannot distinguish between complete graphs of different sizes since all graphlet degree distributions are concentrated on a single value in complete graphs. The spectral variant $NetEmd_S$ is not affected by this and we find that $NetEmd_S$ is either on par with or outperforms the other non-$NetEmd$ graph classifiers on all six data sets.

We also tested $NetEmd$ on benchmark data sets representing chemical compounds and protein structures. Unlike the social network data sets, in these data sets nodes and edges are labeled to reflect domain specific knowledge such as atomic

number, amino acid type and bond type. Although *NetEmd*, in contrast to the other graph kernels, does not rely on domain specific knowledge in the form of node or edge labels, we found that *NetEmd* outperforms many of the considered graph kernels coming only second to the Weisfeiler-Lehman (Shervashidze et al., 2011) type kernels in terms of overall performance (see Section 3.2.5 in the Appendix).

3.1.4 Discussion

Starting from basic principles, we have introduced a general network comparison methodology, *NetEmd*, that is aimed at capturing common generating processes in networks. We tested *NetEmd* in a large variety of experimental settings and found that *NetEmd* successfully identifies similar networks at multiple scales even when networks differ significantly in terms of size and density, generally outperforming other graphlet based network comparison measures. Even when based only on graphlets up to size 3 (i.e. edges, 2-paths and triangles), *NetEmd* has performance comparable to the state of the art, making *NetEmd* feasible even for networks containing billions of edges and nodes.

By exploring machine learning applications we showed that *NetEmd* captures topological similarity in a way that relates to the function of networks and outperforms state-of-the art graph classifiers on several graph classification benchmarks.

Although we only considered variants of *NetEmd* that are based on distributions of graphlets and spectra of Laplacian operators in this paper, *NetEmd* can also be applied to other graph features in a straightforward manner. For instance, distributions of paths and centrality measures might capture larger scale properties of networks and their inclusion into *NetEmd* might lead to a more refined measure.

Data availability The source code for *NetEmd* is freely available at:

`www.opig.ox.ac.uk/resources`

3.2 Appendix

3.2.1 Implementation

Graphlet distributions.

In the main paper, both the graphlet degree distribution and graphlet counts in 1-step ego networks were used as inputs for $NetEmd$.

Graphlet degree distributions The graphlet degree (Pržulj, 2007) of a node specifies the number of graphlets (small induced subgraphs) of a certain type the node appears in, while distinguishing between different positions the node can have in a graphlet. Different positions within a graphlet correspond to the orbits of the automorphism group of the graphlet. Among graphs on two to four nodes, there are 9 possible graphs and 15 possible orbits. Among graphs on two to five nodes there are 30 possible graphs and 73 possible orbits.

Graphlet distributions based on ego-networks. Another way of obtaining graphlet distributions is to consider graphlet counts in ego-networks (Ali et al., 2014). The k-step ego-network of a node i is defined as the subgraph induced on all the nodes that can be reached from i (including i) in less than k steps. For a given k, the distribution of a graphlet m in a network G is then simply obtained by counting the occurrence of m as an induced subgraph in the k-step ego-networks of each individual node.

Step-wise implementation

In this paper, for integer valued network features such as graphlet based distributions, we base our implementation on the probability distribution that corresponds to the histogram of feature t with bin width 1 as $p_t(G)$. $NetEmd$ can also be defined on the basis of discrete empirical distributions i.e. distributions consisting of point masses (see Section 3.2.3).

Here we summarise the calculation of the $NetEmd_T(G, G')$ distance between networks G and G' (with N and N' nodes respectively), based on the comparison of

the set of local network features $T = \{t_1, \ldots, t_m\}$ of graphlet degrees corresponding to graphlets up to size k.

1. First one computes the graphlet degree sequences corresponding to graphlets up to size k for networks G and G'. This can be done efficiently using the algorithm ORCA (Hočevar and Demšar, 2014). For the graphlet degree t_1 compute a histogram across all N nodes of G having bins of width 1 of which the centers are at their respective values. This histogram is then normalized to have total mass 1. We then interpret the histogram as the (piecewise continuous) probability density function of a random variable. This probability density function is denoted by $p_{t_1}(G)$. The standard deviation of $p_{t_1}(G)$ is then computed, and is used to rescale the distribution so that it has variance 1. This distribution is denoted by $\widetilde{p_{t_1}(G)}$.

2. Repeat the above step for network G', and denote the resulting distribution by $\widetilde{p_{t_1}(G')}$. Now compute

$$NetEmd^*_{t_1}(G, G') = \inf_{c \in \mathbb{R}}\big(EMD\big(\widetilde{p_{t_1}(G)}(\cdot + c), \widetilde{p_{t_1}(G')}(\cdot)\big)\big).$$

   In practice, this minimisation over c is computed using a suitable optimization algorithm. In our implementation we use the Brent-Dekker algorithm (Brent, 1971) with an error tolerance of 0.00001 and with the number of iterations upper bounded by 150.

3. Repeat the above two steps for the network features $t_2, \ldots, t_m$ and compute

$$NetEmd_T(G, G') = \frac{1}{m}\sum_{j=1}^{m} NetEmd^*_{t_j}(G, G').$$

Alternatively to perform the optimisation step over c (step 2), an approximation to $NetEmd^*_{t_1}(G, G')$ could be used instead by setting $\widetilde{p_{t_1}(G)}$ and $\widetilde{p_{t_1}(G)}$ to have mean zero, or median zero. Thus aligning the two scaled distributions in relation to their mean, (or median for a more robust approach). This type of approximation would lead to an improvement in the computation time of the $NetEmd_T(G, G')$ statistic,

and would provide similar results for smooth unimodal distributions. However, for multimodal distributions or non-smooth distributions, which could arise from the heavy tail behaviour of the subgraph count distributions, or their threshold behaviour (Rito et al., 2010), the optimisation step would ensure a better alignment of the shape of the distributions.

It should also be noted that although the final comparison statistic $NetEmd_T(G, G')$ (step 3) is a linear combination of the previous $NetEmd^*_{t_j}$ scores over different properties t_j; the scale of the different properties does not bias the resulting $NetEmd_T(G, G')$ score, as all comparisons are carried out in the same scale, and the distributions $\widetilde{p_{t_j}(G)}$ have been rescaled by the variance of the respective $p_{t_j}(G)$ distribution. However, when considering a large set of properties and when the interest lies in capturing the joint differences across all the properties considered, the $\log(NetEmd^*_{t_j}(G, G'))$ could be used, as this transformation would avoid bias towards the contribution of individual properties.

Example: EMD^* for Gaussian distributions

Suppose that p and q are $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$ distributions, respectively. Then

$$\begin{aligned} EMD^*(p, q) &= \inf_{c \in \mathbb{R}} \Big(EMD\big(\tilde{p}(\cdot + c), \tilde{q}(\cdot)\big) \Big) \\ &= EMD\Big(\tilde{p}(\cdot - \frac{\mu_1}{\sigma_1} + \frac{\mu_2}{\sigma_2}), \tilde{q}(\cdot)\Big) \\ &= EMD\big(\tilde{q}(\cdot), \tilde{q}(\cdot)\big) = 0. \end{aligned}$$

Here we used that if $X \sim N(\mu_1, \sigma_1^2)$ and $Y \sim N(\mu_2, \sigma_2^2)$, then $\frac{X}{\sigma_1} + c \sim N(\frac{\mu_1}{\sigma_1} - c, 1)$ and $\frac{Y}{\sigma_2} \sim N(\frac{\mu_2}{\sigma_2}, 1)$, and these two distributions are equal if $c = \frac{\mu_1}{\sigma_1} - \frac{\mu_2}{\sigma_2}$.

Spectral NetEmd

When using spectra of graph operators, which take real values instead of the integer values one has in the case of graphlet distributions, we use the empirical distribution consisting of point masses for computing $NetEmd$. For more details see Section

3.2.3 of this appendix.

Computational complexity

The computational complexity of graphlet based comparison methods is dominated by the complexity of enumerating graphlets. For a network of size N and maximum degree d, enumerating all connected graphlets up to size m has complexity $O(Nd^{m-1})$, while counting all graphlets up to size m in all k-step ego-networks has complexity $O(Nd^{k+m-1})$. Because most real world networks are sparse, graphlet enumeration algorithms tends to scale more favourably in practice than the worst case upper bounds given above.

In the case of spectral measures, the most commonly used algorithms for computing the eigenvalue spectrum have complexity $O(N^3)$. Recent results show that the spectra of graph operators can be approximated efficiently in $O(N^2)$ time (Thüne, 2013).

Given the distribution of a feature t, computing $EMD_t^*(G, G')$ has complexity $O(k(s+s')\log(s+s'))$, where s and s' are the number of different values t takes in G and G' respectively and k is the maximum number function calls of the optimization algorithm used to align the distributions. For node based features such as motif distributions, the worst case complexity is $O(k(N(G)+N(G'))\log(N(G)+N(G')))$, where $N(G)$ is the number of nodes of G, since the number of different values t can take is bounded by the number of nodes.

3.2.2 Proof that *NetEmd* is a distance measure

We begin by stating a definition. A ***pseudometric*** on a set X is a non-negative real-valued function $d : X \times X \to [0, \infty)$ such that, for all $x, y, z \in X$,

1. $d(x, x) = 0$;
2. $d(x, y) = d(y, x)$ (symmetry);
3. $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality).

If Condition 1 is replaced by the condition that $d(x, y) = 0 \iff x = y$ then d defines a ***metric***. Note that this requirement can only be satisfied by a network comparison measure that is based on a complete set of graph invariants and hence network comparison measures in general will not satisfy this requirement.

Proposition Let M denote the space of all real-valued probability measures supported on $\mathbb{R}$ with finite, non-zero variance. Then the EMD^* distance between probability measures, μ_X and μ_Y in M defined by

$$EMD^*(\mu_X, \mu_Y) = \inf_{c \in \mathbb{R}} EMD(\tilde{\mu}_X(\cdot), \tilde{\mu}_Y(\cdot + c)),$$

defines a pseudometric on the space of probability measures M.

Proof We first note that if $\mu_X \in M$ then $\tilde{\mu}_X(\cdot + c) \in M$ for any $c \in \mathbb{R}$. Let us now verify that EMD^* satisfies all properties of a pseudometric. Clearly, for any $\mu_X \in M$, we have $0 \leq EMD^*(\mu_X, \mu_X) \leq EMD(\tilde{\mu}_X(\cdot), \tilde{\mu}_X(\cdot)) = 0$, and so $EMD^*(\mu_X, \mu_X) = 0$. Symmetry holds, since for, any μ_X and μ_Y in M,

$$\begin{aligned} EMD^*(\mu_X, \mu_Y) &= \inf_{c \in \mathbb{R}} EMD(\tilde{\mu}_X(\cdot), \tilde{\mu}_Y(\cdot + c)) \\ &= \inf_{c \in \mathbb{R}} EMD(\tilde{\mu}_Y(\cdot + c), \tilde{\mu}_X(\cdot)) \\ &= \inf_{c \in \mathbb{R}} EMD(\tilde{\mu}_Y(\cdot), \tilde{\mu}_X(\cdot + c)) \\ &= EMD^*(\mu_Y, \mu_X). \end{aligned}$$

Finally, we verify that EMD^* satisfies the triangle inequality. Suppose μ_X, μ_Y and μ_Z are probability measures from the space M, then so are $\tilde{\mu}_X(\cdot + a)$, $\tilde{\mu}_Y(\cdot + b)$ for any $a, b \in \mathbb{R}$. Since EMD satisfies the triangle inequality, we have, for any $a, b \in \mathbb{R}$,

$$EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Y(\cdot + b)) \leq EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Z(\cdot)) + EMD(\tilde{\mu}_Y(\cdot + b), \tilde{\mu}_Z(\cdot)).$$

Since the above inequality holds for all $a, b \in \mathbb{R}$, we have that

$$\begin{aligned}
EMD^*(\mu_X, \mu_Y) &= \inf_{c\in\mathbb{R}} EMD(\tilde{\mu}_X(\cdot + c), \tilde{\mu}_Y(\cdot)) \\
&= \inf_{a,b\in\mathbb{R}} EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Y(\cdot + b)) \\
&\leq \inf_{a,b\in\mathbb{R}} \big[EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Z(\cdot)) + EMD(\tilde{\mu}_Y(\cdot + b), \mu_Z(\cdot))\big] \\
&= \inf_{a\in\mathbb{R}} \big[EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Z(\cdot)) + \inf_{b\in\mathbb{R}} EMD(\tilde{\mu}_Y(\cdot + b), \tilde{\mu}_Z(\cdot))\big] \\
&= \inf_{a\in\mathbb{R}} EMD(\tilde{\mu}_X(\cdot + a), \tilde{\mu}_Z(\cdot)) + \inf_{b\in\mathbb{R}} EMD(\tilde{\mu}_Y(\cdot + b), \tilde{\mu}_Z(\cdot)) \\
&= EMD^*(\mu_X, \mu_Z) + EMD^*(\mu_Y, \mu_Z),
\end{aligned}$$

as required. We have thus verified that EMD^* satisfies all properties of a pseudo-metric. □

3.2.3 Generalization of EMD^* to point masses

Although in the case of graphlet based features we based our implementation of *NetEmd* on probability distribution functions that correspond to normalized histograms having bin width 1 *NetEmd* can also be based on empirical distributions consisting of collections of point masses located at the observed values.

The definition of EMD^* can be generalized to include distributions of zero variance, i.e. unit point masses. Mathematically, the distribution of a point mass at x_0 is given by the Dirac measure $\delta_x(x_0)$. Such distributions are frequently encountered in practice since some graphlets do not occur in certain networks.

First, we note that unit point masses are always mapped onto unit point masses under rescaling operations. Moreover, for a unit point mass $\delta_x(x_0)$ we have that $\inf_{c\in\mathbb{R}}(EMD(\tilde{p}(\cdot+c), \delta_x(x_0))) = \inf_{c\in\mathbb{R}}(EMD(\tilde{p}(\cdot + c), \delta_x(kx_0)))$ for all $p \in M$ and $k > 0$. Consequently, EMD^* can be generalized to include unit point masses in a consistent fashion by always rescaling them by 1:

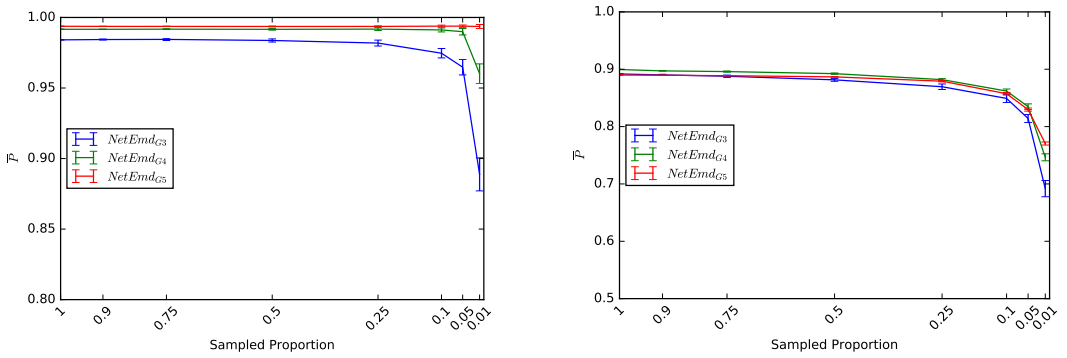
$$EMD^*(p, q) = \inf_{c\in\mathbb{R}}\big(EMD(\hat{p}(\cdot + c), \hat{q})\big),$$

where $\hat{p} = \tilde{p}$ (as in Eq. 3.1) if p has a non-zero variance, and $\hat{p} = p$ if p has variance zero.

3.2.4 Sub-sampling

NetEmd is well suited for the sub-sampling procedure from (Ali et al., 2016). Following this procedure we base the graphlet distributions used as an input of *NetEmd* on a sample of nodes rather than the whole network.

Figure 3.5 shows the $\overline{P}$ scores for variants of *NetEmd* on a set of synthetic networks and the Onnela et al. data set. We find that the performance of *NetEmd* is stable under sub-sampling and that in general using a sample of only 10% of the nodes produces results comparable to the case where all nodes are used.



(a) Synthetic networks **(b)** Onnela et al.

Figure 3.5: The $\overline{P}$ values for different variants of *NetEmd* under sub-sampling for (a) a set of 80 synthetic networks coming from eight different random graph models with 2500 nodes and average degree 20, (b) for the Onnela et al. data set showing the average and standard deviation over 50 experiments for each sampled proportion. Note that the performance of *NetEmd* under sub-sampling is remarkably stable and is close to optimal even when only 10% of nodes are sampled. For synthetic networks we find that the stability of *NetEmd* increases as the size of the graphlets used in the input is increased.

3.2.5 Results for data sets of chemical compounds and proteins

We also tested *NetEmd* on benchmark data sets representing chemical compounds (MUTAG, NCI1 and NCI109) and protein structures (ENZYMES and D&D). MUTAG (Debnath et al., 1991) is a data set of 188 chemical compounds that are la-

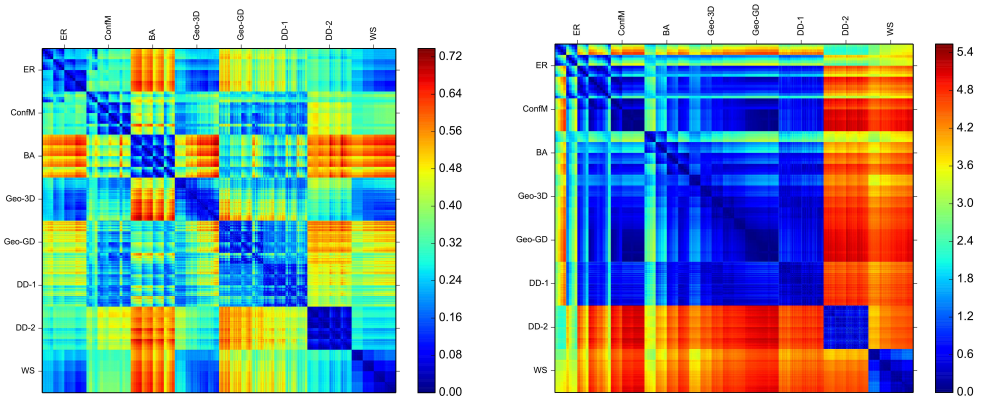
belled according to their mutagenic effect on Salmonella typhimurium. NCI1 and NCI109 represent sets of chemical compounds which are labelled for their activity against non-small cell lung cancer and ovarian cancer cell lines, respectively (Wale et al., 2008). Nodes and edges in MUTAG, NCI1 and NCI109 are labeled by atomic number and bound type, respectively. ENZYMES and D&D (Borgwardt et al., 2005) consist of networks representing protein structures at the level of tertiary structure and amino acids respectively. While networks in ENZYMES are classified into six different enzyme classes, networks in D&D are classified according to whether or not they correspond to an enzyme. Nodes in ENZYMES are labelled according to structural element type and according to amino acid types in D&D.

Classification accuracies obtained using $NetEmd$ on the data sets of chemical compounds and protein structures are given in Table 3.3, along with results for other graph kernels reported in (Shervashidze et al., 2011). For a detailed description of these kernels we refer to (Shervashidze et al., 2011) and the references therein. Note that, in contrast to all other kernels in Table 3.3, $NetEmd$ does not use any domain specific knowledge in the form of node or edge labels. Node and edge labels are highly informative for all five classification tasks - as shown in (Sugiyama and Borgwardt, 2015).

Kernel	MUTAG	NCI1	NCI109	ENZYMES	D & D
$NetEmd_{G5}$	83.71 ±1.16	78.59±0.28	76.71±0.34	46.55±1.25	78.01 ±0.38
$NetEmd_{S}$	83.30 ±1.20	77.36±0.38	76.14±0.27	42.75±0.78	76.74 ±0.43
WL subtree	82.05±0.36	82.19 ±0.18	82.46 ±0.24	52.22±1.26	79.78 ±0.36
WL edge	81.06±1.95	84.37±0.30	84.49±0.20	53.17±2.04	77.95±0.70
WL shortest path	83.78±1.46	84.55±0.36	83.53±0.30	59.05±1.05	79.43±0.55
Ramon & Gärtner	85.72±0.49	61.86±0.27	61.67±0.21	13.35±0.87	57.27±0.07
p-random walk	79.19±1.09	58.66±0.28	58.36±0.94	27.67±0.95	66.64±0.83
Random Walk	80.72±0.38	64.34±0.27	63.51±0.18	21.68±0.94	71.70±0.47
Graphlet count	75.61±0.49	66.00±0.07	66.59±0.08	32.70±1.20	78.59±0.12
Shortest path	87.28±0.55	73.47±0.11	73.07±0.11	41.68±1.79	78.45±0.26

Table 3.3: 10 fold cross validation accuracies of Gaussian kernels based on $NetEmd_{G5}$ and $NetEmd_{S}$ and other kernels reported in (Shervashidze et al., 2011).

On MUTAG, $NetEmd$ achieves an accuracy that is comparable to the Weisfeiler-Lehman (WL) shortest path kernel, but is outperformed by the shortest path kernel



(a) Heatmap of $NetEmd_{G5}$ for RG_3. **(b)** Heatmap of $GCD11$ for RG_3.

Figure 3.6: (a) and (b) show the heatmaps of pairwise distances on RG_3 ($N \in \{1250, 2500, 5000, 10000\}$ and $k \in \{10, 20, 40, 80\}$) according to $NetEmd_{G5}$ and next best performing measure $GCD11$, respectively. In the heat map, networks are ordered from top to bottom in the following order: model, average degree and node count. Although we observe some degree of off diagonal mixing the heatmap of $NetEmd$ still shows 8 diagonal blocks corresponding to different generative models in contrast to the heat map of $GCD11$.

and the kernel by Ramon & Gärtner. While on NCI1, NCI109 and ENZYMES, $NetEmd$ is outperformed only by WL kernels, on D&D $NetEmd$ achieves a classification accuracy that is comparable to the best performing kernels. Notably, on D&D $NetEmd$ also outperforms the vector model by Dobson and Doig (Dobson and Doig, 2003) (classification accuracy: 76.86±1.23) which is based on 52 physical and chemical features without using domain specific knowledge i.e. solely based on graph topology.

3.2.6 Implementation of C-SVMs

Following the procedure in (Shervashidze et al., 2011) we use 10-fold cross validation with a C-SVM (Cortes and Vapnik, 1995) to test classification performance. We use the python package scikit-learn (Pedregosa et al., 2011) which is based is build on libsvm implementation (Chang and Lin, 2011). The $C - value$ of the C-SVM and the α for the Gaussian kernel is tuned independently for each fold using training data from that fold only. Each experiment is repeated 10 times, and average prediction accuracies and their standard deviations are reported.

We also note that note for all values of α is the Gaussian NetEmd kernel is positive

semidefinite (psd) (Jayasumana et al., 2015). The implication is that the C-SVM converges to a stationary point that is not always guaranteed to be global optimum. Although there exist alternative algorithms (Luss and d'Aspremont, 2008) for training C-SVMs with indefinite kernels which might result in better classification accuracy, here we chose to use the standard libsvm-algorithm in order to ensure a fair comparison between kernels. For a discussion of support vector machines with indefinite kernels see (Haasdonk, 2005).

3.2.7 Detailed description of data sets

Synthetic networks and random graph models

RG_1 consists of 16 sub data sets corresponding to combinations of $N \in \{1250, 2500, 5000, 10000\}$ and $k \in \{10, 20, 40, 80\}$ containing 10 realizations for each model i.e. contain 80 networks each.

In RG_2 the size N and average degree k are increased independently in linear steps to twice their initial value ($N \in \{2000, 3000, 4000\}$ and $k \in \{20, 24, 28, 32, 36, 40\}$) and contains 10 realizations per model parameter combination, resulting in a data set of $3 \times 6 \times 8 \times 10 = 1440$ networks.

In RG_3 the size N and average degree k are increased independently in multiples of 2 to 8 times their initial value ($N \in \{1250, 2500, 5000, 10000\}$ and $k \in \{10, 20, 40, 80\}$) and again contains 10 realizations per model parameter combination, resulting in a data set of $4 \times 4 \times 8 \times 10 = 1280$ networks. The models are as follows.

The Erdős-Rényi model We consider the Erdős-Rényi (ER) model (Erdös and Rényi, 1960) $G(N, m)$ where N is the number of nodes and m is the number of edges. The edges are chosen uniformly at random without replacement from the $\binom{N}{2}$ possible edges.

The configuration model Given a graphical degree sequence, the configuration model creates a random graph that is drawn uniformly at random from the space of all graphs with the given degree sequence. The degree sequence of the configuration

models used in the paper is taken to be degree sequence of a duplication divergence model that has the desired average degree.

The Barabási Albert preferential attachment model In the Barabási-Albert model (Barabási and Albert, 1999) a network is generated starting from a small initial network to which nodes of degree m are added iteratively and the probability of connecting the new node to an existing node is proportional to the degree of the existing node.

Geometric random graphs Geometric random graphs (Gilbert, 1961) are constructed under the assumption that the nodes in the network are embedded into a D dimensional space, and the presence of an edge depends only on the distance between the nodes and a given threshold r. The model is constructed by placing N nodes uniformly at random in an D-dimensional square $[0, 1]^D$. Then edges are placed between any pair of nodes for which the distance between them is less or equal to the threshold r. We use $D = 3$ and set r to be the threshold that results in a network with the desired average degree, while the distance is the Euclidean distance.

The geometric gene duplication model The geometric gene duplication model is a geometric model (Higham et al., 2008) in which the nodes are distributed in 3 dimensional Euclidean space $\mathbb{R}^3$ according to the following rule. Starting from an small initial set of nodes in three dimensions, at each step a randomly chosen node is selected and a new node is placed at random within a Euclidean distance d of this node. The process is repeated until the desired number of nodes is reached. Nodes within a certain distance r are then connected. We fix r to obtain the desired average degree.

The duplication divergence model of Vázquez et al. The duplication divergence model of Vázquez et al. (Vázquez et al., 2003) is defined by the following growing rules: (1) Duplication: A node v_i is randomly selected and duplicated (v_i')

along with all of its interactions. An edge between v_i and v_i' is placed with probability p. (2) Divergence: For each pair of duplicated edges $\{(v_i, v_k); (v_i', v_k)\}$; one of the duplicated edges is selected uniformly at random and then deleted with probability q. This process is followed until the desired number of nodes is reached. In our case we fix p to be 0.05 and adjust q through a grid search to obtain a network that on average has the desired average degree.

The duplication divergence of Ispolatov et al. The duplication divergence model of Ispolatov et al. (Ispolatov et al., 2005) starts with an initial network consisting of a single edge and then at each step a random node is chosen for duplication and the duplicate is connected to each of the neighbours of its parent with probability p. We adjust p to obtain networks that have on average the desired average degree.

The Watts-Strogatz model The Watts-Strogatz model, (Watts and Strogatz, 1998) creates graphs that interpolate between regular graphs and ER graphs. The model starts with a ring of n nodes in which each node is connected to its k-nearest neighbours in both directions of the ring. Each edges is rewired with probability p to a node which is selected uniformly at random. While k is adjusted to obtain networks having the desired average degree we take p to be 0.05.

Real world data sets

Summary statistics of the data sets are given in Table 3.4.

Real world networks from different classes (RWN) We compiled a data set consisting of 10 different classes of real world networks: social networks, metabolic networks, protein interaction networks, protein structure networks, food webs, autonomous systems networks of the internet, world trade networks, airline networks, peer to peer file sharing networks and scientific collaboration networks. Although in some instances larger versions of these data sets are available, we restrict the maximum number of networks in a certain class to 20 by taking random samples of

Data set	#Networks	N_{min}	Median(N)	N_{max}	E_{min}	Median(E)	E_{max}	d_{min}	Median(d)	d_{max}
RWN	167	24	351	62586	76	2595	824617	7.55e-05	0.0163	0.625
Onnela et al.	151	30	918	11586	62	2436	232794	4.26e-5	0.0147	0.499
AS-caida	122	8020	22883	26475	18203	46290	53601	1.48e-4	1.78e-4	5.66e-4
AS-733	732	493	4180.5	6474	1234	8380.5	13895	6.63e-4	9.71e-4	1.01e-2
World Trade Networks	53	156	195	242	5132	7675	18083	0.333	0.515	0.625
Reddit-Binary	2000	6	304.5	3782	4	379	4071	5.69e-4	8.25e-3	0.286
Reddit-Multi-5k	4999	22	374	3648	21	422	4783	6.55e-4	6.03e-3	0.091
Reddit-Multi-12k	11929	2	280	3782	1	323	5171	5.69e-4	8.27e-3	1.0
COLLAB	5000	32	52	492	60	654.5	40120	0.029	0.424	1.0
IMDB-Binary	1000	12	17	136	26	65	1249	0.095	0.462	1.0
IMDB-Multi	1500	7	10	89	12	36	1467	0.127	1.0	1.0
MUTAG	188	10	17.5	28	10	19	33	0.082	0.132	0.222
NCI1	4110	3	27	111	2	29	119	0.0192	0.0855	0.667
NCI109	4127	4	26	111	3	29	119	0.0192	0.0862	0.5
ENZYMES	600	2	32	125	1	60	149	0.0182	0.130	1.0
D&D	1178	30	241	5748	63	610.5	14267	8.64e-4	0.0207	0.2

Table 3.4: Summary statistics of data sets N, E and d stand for the number of nodes, number of edges and edge density, respectively.

larger data sets in order to avoid scores being dominated by larger network classes.

The class of social networks consists of 10 social networks from the Pajek data set which can be found at `http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm` (June 12th 2015) (Networks: 'bkfrat', 'bkham', 'bkoff', 'bktec', 'dolphins', 'kaptailS1', 'kaptailS2', 'kaptailT1', 'kaptailT2', 'karate', 'lesmis', 'prison') and a sample of 10 Facebook networks from (Traud et al., 2012) (Networks:'Auburn71', 'Bucknell39', 'Caltech36', 'Duke14', 'Harvard1', 'JMU79', 'MU78', 'Maine59', 'Maryland58', 'Rice31', 'Rutgers89', 'Santa74', 'UC61', 'UC64', 'UCLA26', 'UPenn7', 'UVA16', 'Vassar85', 'WashU32', 'Yale4'). The class of metabolic networks consists of 20 networks taken (Jeong et al., 2000) (Networks: 'AB', 'AG', 'AP', 'AT', 'BS', 'CE', 'CT', 'EF', 'HI', 'MG', 'MJ', 'ML', 'NG', 'OS', 'PA', 'PN', 'RP', 'TH', 'TM', 'YP'). The class of protein interaction networks consists of 6 networks from BIOGRID (Stark et al., 2006) (Arabidopsis thaliana, Caenorhabditis elegans, Drosophila melanogaster, Homo sapiens, Mus musculus and Saccharomyces cerevisiae downloaded: October 2015) and 5 networks from HINT (Das and Yu, 2012) (Arabidopsis thaliana, Caenorhabditis elegans, Drosophila melanogaster, Homo sapiens and Mus musculus (Version: June 1 2014)) and the protein interaction network of Echeria coli by Rajagopala et al. (Rajagopala et al., 2014). The class of protein structure networks consists of a sample of 20 networks from the data set D&D (Networks: 20, 119, 231, 279, 335, 354, 355, 369, 386, 462, 523, 529, 597, 748, 833,

866, 990, 1043, 1113, 1157). The class of food webs consists of 20 food webs from the Pajek data set: http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm (June 10th 2015) (Networks: 'ChesLower', 'ChesMiddle', 'ChesUpper', 'Chesapeake', 'CrystalC', 'CrystalD', 'Everglades', 'Florida', 'Michigan', 'Mondego', 'Narragan', 'StMarks', 'baydry', 'baywet', 'cypdry', 'cypwet', 'gramdry', 'gramwet', 'mangdry', 'mangwet'). The class of internet networks consists of 10 randomly chosen networks from AS-733 (Leskovec et al., 2005) (Networks:'1997/11/12', '1997/12/28', '1998/01/01', '1998/06/06', '1998/08/13', '1998/12/04', '1999/03/30', '1999/04/17', '1999 /06/18', '1999/08/30') and 10 randomly chosen networks from AS-caida (Leskovec et al., 2005) (Networks: '2004/10/04', '2006/01/23', '2006/03/27', '2006/07/10', '2006/09/25', '2006/11/27', '2007/01/15', '2007/04/30', '2007/05/28', '2007/09/24'). Both datasets are from SNAP (Jure and Krevl, 2014)(June 1 2016). The class of world trade networks is a sample of 20 networks of the larger data set considered in (Feenstra et al., 2005; Division, 2015) (Networks: 1968, 1971, 1974, 1975, 1976, 1978, 1980, 1984, 1989, 1992, 1993, 1996, 1998, 2001, 2003, 2005, 2007, 2010, 2011, 2012). The airline networks were derived from the data available at: http://openflights.org/ (June 12 2015). For this we considered the 50 largest airlines from the database in terms of the number of destinations that the airline serves. For each airline a network is obtained by the considering all airports that are serviced by the airlines which are connected whenever there is direct flight between a pair of nodes. We then took a sample of 20 networks from this larger data set (Airline codes of the networks: 'AD', 'AF', 'AM', 'BA', 'DY', 'FL', 'FR', 'JJ', 'JL', 'MH', 'MU', 'NH', 'QF', 'SU', 'SV', 'U2', 'UA', 'US', 'VY', 'ZH'). The class of peer to peer networks consist of 9 networks of the Gnutella file sharing platform measured at different dates which are available at (Jure and Krevl, 2014). The scientific collaboration networks consists of 5 networks representing different scientific disciplines which were obtained from (Jure and Krevl, 2014) (June 1 2015).

Onnela et al. data set The Onnela et al. data set consists of all undirected and unweighted networks from the larger collection analysed in (Onnela et al., 2012). A complete list of networks and class membership can be found in the supplementary

information of (Ali et al., 2014).

Time ordered data sets The data sets AS-caida and AS-733 each represent the internet measured at the level of autonomous systems at various points in time. Both data sets were downloaded from (Jure and Krevl, 2014)(June 1 2015).
The World Trade Networks data set is based on the data set (Feenstra et al., 2005) for the years 1962-2000 and on UN COMTRADE (Division, 2015) for the years 2001-2015. Two countries are connected in the network whenever they import or export a commodity from a each other within the given calendar year. The complete data set was downloaded from: `http://atlas.media.mit.edu/en/resources/data/` on July 12 2015.

Machine learning benchmarks A short description of the social networks datasets was given in the main text. A more detailed description can be found in (Yanardag and Vishwanathan, 2015). The social network data sets were downloaded from `https://ls11-www.cs.tu-dortmund.de/staff/morris/graphkerneldatasets` on September 2 2016.
A short short description of the chemical compound and protein structure data sets was given in Section 3.2.5. A more detailed description of the data set can be found in (Shervashidze et al., 2011). These data sets were downloaded from: `https://www.bsse.ethz.ch/mlcb/research/machine-learning/graph-kernels.html` on June 12 2016.

3.2.8 Performance evaluation via area under precision recall curve

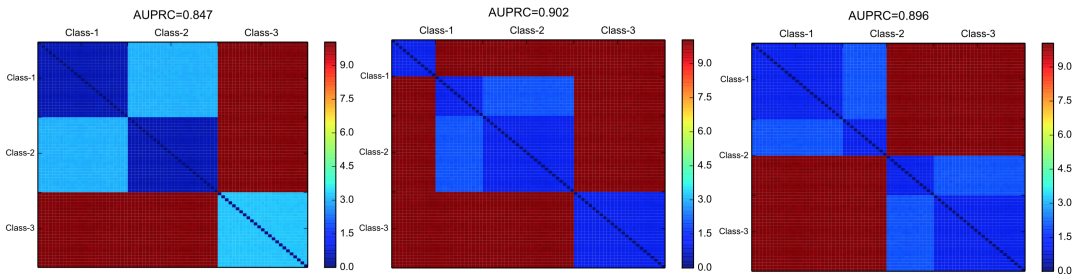
The area under precision recall curve (AUPRC) was used as a performance metric for network comparison measures by Yaveroglu et al. (Yaveroglu et al., 2014). The AUPRC is based on a classifier that for a given distance threshold ϵ classifies pairs of networks to be similar whenever $d(G, G') < \epsilon$.
A pair satisfying $d(G, G') < \epsilon$ is taken to be a true positive whenever G and G' are from the same class. The AUPRC is then defined to be the area under the

precision recall curve obtained by varying ϵ in small increments. However, AUPRC is problematic, especially in settings where one has more than two classes and when classes are separated at different scales.

Figure 3.7 gives three examples of metrics for a problem that has three classes: (a) shows a metric d_1 (AUPRC=0.847) that clearly separates the 3-classes which, however, has a lower AUPRC than the metrics given in (b) (AUPRC=0.902) which confuses half of Class-1 with Class-2 and (c) (PRC=0.896) which shows 2 rather than 3 classes. The colour scale in the figure represents the magnitude of a comparison between a pair of individuals according to the corresponding metric.

Some of the problems of AUPRC are the following. First, AUPRC is based on a classifier that identifies pairs of similar networks and hence is only indirectly related to the problem of separating classes. Moreover, the classifier uses a single global threshold ϵ for all networks and classes, and hence implicitly assumes that all classes are separated on the same scale. The AUPRC further lacks a clear statistical interpretation, which complicates its use especially when one has multiple classes and when precision recall curves of different measures intersect.

Despite its problems we give AUPRC values for all measures we considered in the main text in Table 3.5 for the sake of completeness. Note that *NetEmd* measures achieve the highest AUPRC on all data sets.



(a) d_1 **(b)** d_2 **(c)** d_3

Figure 3.7: Heat maps of three measures for in an example of 3 equally sized classes. (a) Metric d_1 shows clear separation between the 3 classes. (b) d_2 shows 3 classes with half of Class-1 positioned closer to Class-2. (c) d_3 identifies 2 rather than 3 classes. Note that d_1 has lower AUPRC than d_2 and d_3 despite being best at identifying the 3 classes whereas $\overline{P}$ values for the metrics are $\overline{P}(d_1)$=1.0, $\overline{P}(d_2)$=0.887 and $\overline{P}(d_3)$=0.869.

	RG_1	RG_2	RG_3	RWN	Onnela et al.
$NetEmd_{G3}$	0.917± 0.039	0.869	0.702	**0.800**	0.756
$NetEmd_{G4}$	0.959± 0.030	0.930	0.759	0.774	**0.786**
$NetEmd_{G5}$	**0.981± 0.018**	0.957	0.766	0.722	0.757
$NetEmd_{S}$	0.967±0.015	**0.958**	**0.833**	0.702	0.672
$NetEmd_{E4}$	0.966±0.030	0.945	0.801	0.777	0.739
$NetEmd_{DD}$	0.756±0.044	0.708	0.516	0.655	0.612
$Netdis_{ER}$	0.867 ±0.044	0.579	0.396	0.607	0.621
$Netdis_{SF}$	0.852±0.028	0.657	0.437	0.522	0.592
$GCD11$	0.888±0.084	0.709	0.478	0.713	0.693
$GCD73$	0.966±0.052	0.858	0.571	0.736	0.743
$GGDA$	0.815±0.176	0.740	0.481	0.500	0.625

Table 3.5: AUPRC scores for measures and data sets considered in the main text. $NetEmd$ measures have the highest AUPRC score (given in bold) on all data sets. For RG_1 we calculated the value of the AUPRC score for each of the 16 sub-data sets. The table shows the average and standard deviation of the AUPRC values obtained over these 16 sub-data sets.

Chapter 4

Fit of random graph models to protein-protein interaction networks

Protein-protein interaction (PPI) networks summarise physical interactions between proteins. They can be used to aid our understanding of the individual roles of proteins (Sarajlić et al., 2013), the co-functioning properties of sets of proteins (West et al., 2013) and even the operation of the complete system (Janowski et al., 2014). It is commonly accepted that proteins perform functions usually in conjunction with other proteins, forming a functional module (Hartwell et al., 1999; Lewis et al., 2010). It has been thought that these functional modules are often found where there is a large density of highly interconnected nodes in the network (Barabási and Oltvai, 2004), and consequently, where some small subgraph configurations could occur more frequently than in the rest of network, or more than what could be expected from a background model. Thus, the identifications of network regions, such as protein neighbourhoods, for which small subgraphs occur more frequently than expected is of much interest, as this could aid in the identification of functional modules and protein complexes (Alon, 2007; Pereira-Leal et al., 2007; Barabási and Oltvai, 2004). However, assessing the statistical significance of the occurrence of the different small subgraph configurations depends on a suitable null model for PPI networks.

Prior studies that aimed to assess the fit of random graph models to PPI networks often operated without a clear statistical methodology (Przulj et al., 2010; Hayes

et al., 2013; Shao et al., 2013; Peterson et al., 2012), or disregarded the information provided by the different experimental methods used to detect the protein interactions, and hence considered interactions from only one class of detection mechanisms (Gibson and Goldberg, 2011; Vázquez et al., 2003). This chapter aims to statistically assess the ability of different random graph models to describe the occurrence of small connected subgraphs in PPI networks.

We performed a comprehensive analysis of a global and local fit of different random graph models to the PPI networks of six organisms. In the global fit we tested whether the random graph models were able to generate networks with similar subgraph counts as those found in the PPI networks. In the local fit, we tested whether the distribution of subgraph counts observed in groups of local neighbourhoods of proteins could be obtained in local neighbourhoods of nodes extracted from networks that aim to describe whole PPI network. These analyses are based on Monte Carlo tests using different network comparison statistics based on subgraph counts. There is evidence that protein interaction data detected by different experimental methods can portray different properties (Wuchty and Uetz, 2014). In this chapter we aimed to model PPI networks formed by all detected physical interactions, and also considered PPI networks that were formed by interactions detected by experimental methods from the same interaction detection class: binary or co-complex (see Table 1.1).

This chapter is arranged in five sections. The first section describes the PPI networks considered. The second section gives the global fit of random graph models to the PPI networks. The third section contains the local fit of random graph models to the PPI networks. The fourth section assesses the robustness of the results obtained, when including updates in the PPI networks studied. The fifth section summarises and discusses the results.

4.1 Datasets

This chapter focuses on the physical interactions between proteins within the organisms *Caenorhabditis elegans* (Worm), *Drosophila melanogaster* (Fly), *Homo sapiens*

(Human), *Saccharomyces cerevisiae* (Yeast), *Arabidopsis thaliana Columbia* (AT) and *Mus musculus* (Mouse) and that were extracted from the BioGRID database (Stark et al., 2006) in October 2015.

Protein-protein interactions are detected via a variety of experimental methods that can be classified into two main classes: binary methods and co-complex methods. These two general classes of experiments introduce different types of experimental error and bias. Co-complex methods may report all possible direct and indirect interactions between a group of proteins, thus enlarging the global clustering coefficient of the resulting network. In contrast, binary methods report direct interactions only, but they often have a larger false negative rate (Hart et al., 2006) (see Section 1.1).

For this reason, we also considered the PPI networks that corresponded to co-complex methods and binary methods alone, in addition to the whole (binary-&-cocomplex) PPI networks. We referred to interactions detected via these methods as binary interactions and/or co-complex interactions. Table 4.1 gives the network summary statistics for the largest connected components of the PPI networks considered in this chapter. Among all PPI networks, the summary statistics of the Yeast and Human PPI networks can be easily differentiated from the rest. These networks have the largest average degrees and the largest number of edges. These differences between the other PPI networks and the Yeast or Human PPI networks is mostly due to the greater effort that the scientific community has devoted to study protein-protein interactions in these organisms. Another notable statistic in Table 4.1 is the global clustering coefficient of the co-complex Fly network. This clustering coefficient is one order or magnitude higher than the one observed in the other PPI networks, apart from the binary-&-cocomplex Fly network. We found that this large clustering coefficient could be attributed to the interactions reported in the study conducted by Guruharsha et al. (2011), which was based on co-complex experiments (see Section C.1 for more details).

Lastly, it can be noted that the co-complex network of Worm only consists of a few nodes and edges. For this reason, this network was not considered in the analysis that followed.

Method	Organism	n_v	n_e	C	ρ	L	$Diam$	$\bar{d}$
Binary-&-cocomplex	Worm	2964	5426	0.0203	0.001236	4.80	13	3.66
	Fly	7862	36267	0.1560	0.001174	4.14	10	9.23
	Human	15552	182681	0.0605	0.001511	3.22	8	23.49
	Yeast	5862	79537	0.0499	0.004630	2.56	6	27.14
	AT	8864	33172	0.0227	0.000844	4.29	13	7.48
	Mouse	5156	11848	0.0105	0.000892	4.04	16	4.60
Binary	Worm	2902	5247	0.0173	0.001247	4.78	13	3.62
	Fly	7161	23642	0.0145	0.000922	4.33	12	6.60
	Human	12152	61138	0.0207	0.000828	3.68	10	10.06
	Yeast	4690	25815	0.0617	0.002348	3.63	9	11.01
	AT	6988	27412	0.0322	0.001123	4.80	15	7.85
	Mouse	1541	2459	0.0349	0.002072	6.03	16	3.19
Co-complex	Worm	23	24	0.0800	0.094862	3.53	7	2.09
	Fly	2504	12593	0.3649	0.004019	4.64	14	10.06
	Human	13280	133516	0.0711	0.001514	3.32	9	20.11
	Yeast	5616	57211	0.0445	0.003629	2.57	6	20.37
	AT	3659	6444	0.0053	0.000963	3.81	12	3.52
	Mouse	4531	9690	0.0094	0.000944	3.94	15	4.28

Table 4.1: Number of nodes (n_v), number of edges (n_e), global clustering coefficient (C), edge-density (ρ), average shortest path length (L), diameter ($Diam$) and average degree ($\bar{d}$) of the largest connected component of the PPI networks of Worm, Fly, Human, Yeast, AT and Mouse. The Yeast and Human PPI networks have larger average degrees than other networks. The clustering coefficient of the co-complex Fly network is the largest among almost all PPI networks, by one order of magnitude. The co-complex Worm network is very small as most of the interactions reported for this organism came from binary experiments.

4.2 Global fit

Several studies have aimed to assess the ability of different random graph models to describe the structure found in PPI networks (Shao et al., 2013; Pržulj et al., 2004; Pržulj, 2007; Hayes et al., 2013; Higham et al., 2008; Vázquez et al., 2003). However most of these studies lacked a statistical framework which was able to assess the fit of the random graph models to the PPI networks. Additionally, studies often disregarded the type of experimental method used to detect the protein interactions, thus concluding on datasets that had a mixture of binary and co-complex interactions (Shao et al., 2013; Thorne and Stumpf, 2012; Rhodes et al., 2005), or that considered one class of detection mechanism e.g. binary interactions (Gibson and Goldberg, 2011; Vázquez et al., 2003). This practice can be problematic, as conclusions cannot be easily extrapolated and it leaves no information regarding the possible bias present in the data from the predominant type of interactions present in the PPI networks, e.g. binary or co-complex.

In this section we considered PPI networks composed of all physical interactions reported, regardless of their detection method, as done by previous studies. However, we also considered PPI networks composed of interactions detected by binary experiments alone or by co-complex experiments alone. We statistically assessed the global fit of several random graph models to the binary, co-complex and binary-&-cocomplex networks of these 6 organisms, with regards to the appearance of small connected subgraphs. Small subgraph counts were used to reflect the interest in detection of functional modules, which may show overrepresentation of small subgraphs, particularly the densely connected modules (Barabási and Oltvai, 2004).

In this section, by global fit to an observed network G, we mean the ability of a random graph model to generate a network with, approximately, the same number of nodes and edges as G, and with similar subgraph counts, similarity being judged by means of network comparison methods.

4.2.1 Methods

Monte Carlo test Consider testing the null hypothesis "H_0: Network G_0 is a realisation of model B" against the general alternative "H_1: Network G_0 is not a realisation of model B". Here we used the Monte Carlo test proposed in Section 2.2.2, and which compares data-vs-model and model-vs-model to assess such hypotheses based on a network comparison statistic S.

Network comparisons statistics As our interest focused on generation mechanisms for the appearance of small connected subgraphs in PPI networks, we used the following network comparison statistics based in subgraph counts: GDDA (Pržulj, 2007), GCD (Yaveroglu et al., 2014), Netdis (Ali et al., 2014) and NetEmd (Wegner et al., 2017). These statistics are described in detail in Chapter 2 and 3. For the network comparison statistic Netdis we used the version with no background expectations as comparisons were made across graphs with the same number of nodes.

Random graph models We used eight common random graph models, namely, the Erdös-Rényi model (ER) (Gilbert, 1959), the Configuration model (Molloy and Reed, 1995), the Chung-Lu model (Chung and Lu, 2002), Goh's power law model (Goh et al., 2001), the Geometric-3D model (Geo-3D) (Gilbert, 1961), the Erdös-Rényi Mixture Graph model (ERMG) (Holland et al., 1983) and the Duplication-Divergence model (DD) by Vázquez et al. (2003), along with a variation of the DD model where, similarly to (Gibson and Goldberg, 2011), an ER network on 100 nodes is given as a starting point for the network generation process (DD seed). A detailed description of these models along with their parameter estimation procedures can be found in Section 1.4. We considered these models as they displayed a wide variety of features that had also been observed in the context of other real-world networks, for example the power law behaviour of the degree distribution, the presence of a block structure, biological mechanics for network growth, and preferential attachment (Kolaczyk, 2009; Pržulj et al., 2004; Pržulj, 2007; Hayes et al., 2013; Higham et al., 2008). Among the models considered, the Chung-Lu model was claimed, by Hayes et al. (2013), as a good fitting model for the large PPI networks of Yeast, Worm, Fly and Human, among others.

4.2.2 Results

We performed Monte Carlo tests to assess hypotheses of the type H_0 : "Network A is a realisation of Model B" using a statistic S that accounted for the appearance of small connected subgraphs. Network A was any of the Worm, Fly, Yeast, Human, AT or Mouse PPI networks. PPI networks composed only of binary or co-complex interactions were also considered. Figure 4.1 shows the p-values obtained from each individual Monte Carlo test using the network comparison statistics GDDA, GCD, Netdis and NetEmd, for each of the PPI networks and random graph models considered.

It can be noted from the Monte Carlo tests performed, that there was no evidence that supported the claim made by Hayes et al. (2013) that the Chung-Lu model is a good null model for PPI networks in relation to the appearance of small con-

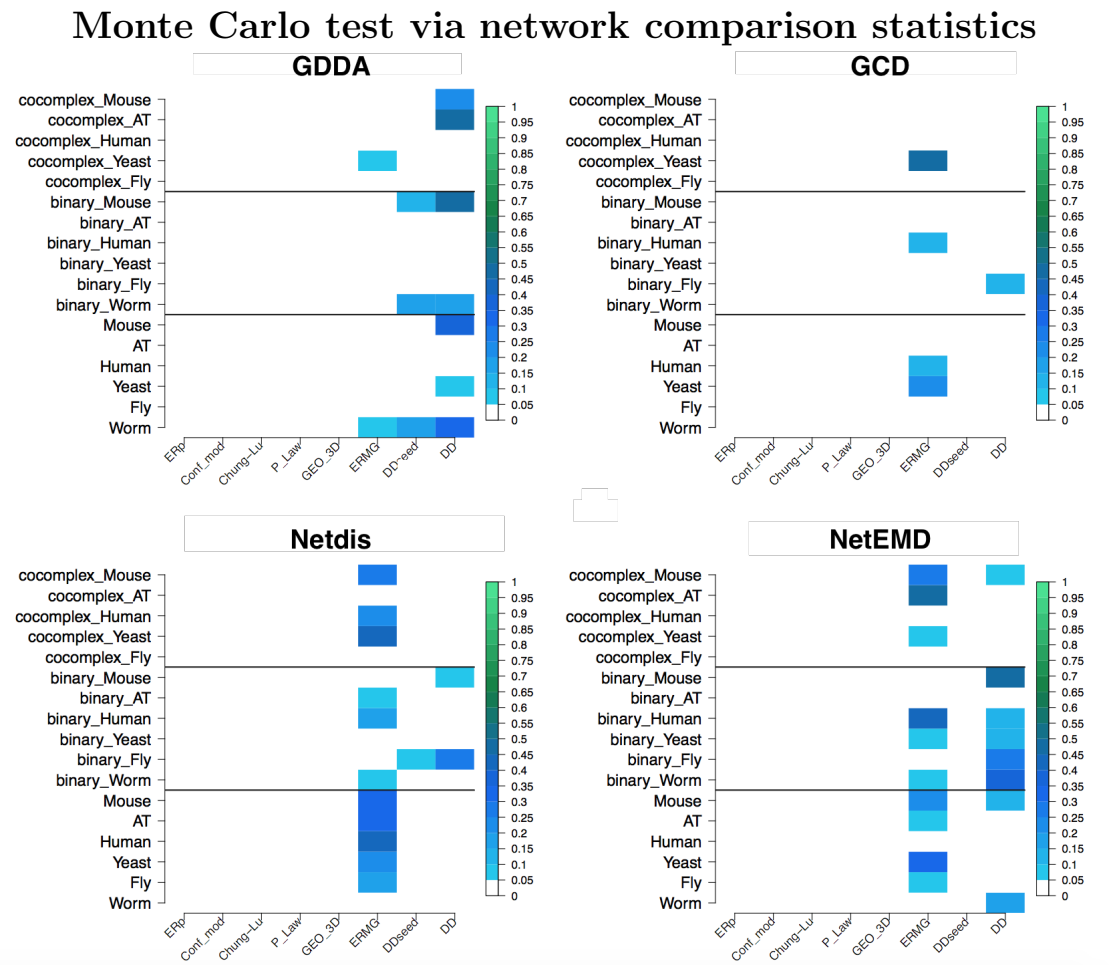


Figure 4.1: Monte Carlo test p-values obtained for each PPI network and each random graph model considered, using the four network comparison statistics. P-values are shown via a colour scale. Only the ERMG, DD seed and DD models obtained p-values larger than 0.05. Exact p-values can be found in Section C.2.

nected subgraphs. This negative result was supported in all variants of the PPI networks considered; binary networks, co-complex networks, the "whole" (binary-&-cocomplex interaction) networks, and by all four network comparison statistics. The random graph models ERMG and DD were the only random graph models for which several PPI networks were not rejected as possible realisations of the models at a 5% level. However, this result was not absolute, as among the different network comparison statistics there was no complete agreement over the rejection (or not rejection) of the null hypothesis, considering a 5% level. For example, the Monte Carlo test rejected the null hypothesis H_0: "The binary Human PPI network is a realisation of the ERMG model" with regards to the network comparison

statistic GDDA. But in contrast, this hypothesis was not rejected with the network comparison statistic GCD.

The only two networks for which all four network comparison statistics reached agreement, at a significance level of 5%, were the Yeast and Fly networks composed of protein interactions detected via co-complex methods. In the case of the co-complex Yeast network all network comparison statistics did not reject the observed network as a possible realisation of the ERMG model. On the other hand, for the co-complex Fly network, all random graph models tested were rejected by the four network comparison statistics. The result for the co-complex Fly network was not surprising, as we found that there was a study, (Guruharsha et al., 2011), which reported a large number of co-complex interactions that could have possibly change the structure of this network (see Section C.1).

Table 4.2 shows the, consensus, number of network comparison statistics for which the Monte Carlo test obtained a p-value greater than 0.05 for the ERMG model and a p-value greater than 0.10 for the DD model. We chose a larger significance value for the DD model due to the large variation of the DD model (Gibson and Goldberg, 2011), which can lead to a larger probability of not rejecting the null hypothesis when it is false.

Table 4.2 suggests that the ERMG model might be a useful model for describing the appearance of subgraph counts in the PPI networks. However, the number of parameters used by the ERMG models ranged from 28 (Worm and binary-Worm) to 1540 (co-complex-Human). (See Section C.3 for the exact number of parameters used by the ERMG models). On the other hand, the DD model considered two parameters only, but was still able to describe the appearance of small connected subgraphs in several PPI networks. This outcome supported the idea that the DD model captures some relation between the biological mechanisms of duplication and divergence and the expected occurrence of small connected subgraphs in PPI networks. Section C.4, shows an example of the fit of the ERMG and DD models to the subgraph distributions of the binary Fly network and the co-complex Yeast network.

It should be noted that in Table 4.2, Figure 4.1 and across this chapter, no multiple

	ERMG	DD
Worm	1	2
Fly	2	0
Yeast	3	0
Human	2	0
AT	2	0
Mouse	2	2
binary_Worm	2	2
binary_Fly	0	3
binary_Yeast	1	1
binary_Human	3	1
binary_AT	1	0
binary_Mouse	0	3
cocomplex_Fly	0	0
cocomplex_Yeast	4	0
cocomplex_Human	1	0
cocomplex_AT	1	1
cocomplex_Mouse	2	1

Table 4.2: Number of Monte Carlo test with a p-value larger than 0.05 across the four network comparison statistics for the random graph models ERMG and DD, (p-value larger than 0.10 for the DD model). Only for the co-complex Yeast and co-complex Fly networks the four network comparison statistics reach consensus. In the case of the co-complex Yeast network the ERMG model is not rejected. For the co-complex Fly network all models always obtained a p-value smaller than 0.05 across all four network comparison methods.

testing correction is performed as we are interested in the non-rejection of the null hypothesis H_0 : "Network A is a realisation of Model B". Hence, by not performing multiple testing corrections we are comparing the results on a stricter setting than if multiple testing correction was performed, as multiple testing correction procedures, such as the Bonferroni correction, take a conservative approach which often leads to a larger number of tests not being rejected (i.e. our result of interest). For example, under the Bonferroni correction and a significance level α, each individual test is only rejected if their respective p-value is smaller than α/m, where m is the number of tests performed. This approach would then lead to fewer test being rejected in Figure 4.1 and to larger values in Table 4.2. Thus, in order to maintain stricter results, we do not perform multiple testing correction.

Figure 4.2 shows the parameters used in the DD model for the three PPI networks conformed by binary interactions, co-complex interactions and binary-&-cocomplex interactions for the 6 organisms we considered. Each point in the graph is coloured on a scale from red to green according to the consensus among the network comparison statistics. It can be noted that the DD model achieved greater consensus

for the binary PPI networks than for co-complex or binary-&-cocomplex PPI networks. In addition, only for these binary networks the estimated parameters were not scattered across the whole parameter space but rather confined to the region $[0, 0.4] \times [0.35, 0.45]$. This result hinted to a possible common set of parameters that could describe the expected occurrence of subgraph counts in these PPI networks. Thus, we considered taking a DD model with parameters given by the average of the individual estimates of p and q of the networks with the largest consensus value in Table 4.2, for each of the three sets of PPI networks. The resulting average values for p and q were $(0.0360; 0.4232)$, $(0.0544; 0.4231)$ and $(0.0183; 0.4168)$ respectively for binary-&-cocomplex networks, the binary networks and co-complex networks. Table 4.3 shows the results of the Monte Carlo test for the DD model with the previously mentioned consensus parameters, for all four network comparison statistics. It can be seen that the use of the DD model with parameters given by the average of the individual estimates of p and q of the networks with the largest consensus value of Table 4.2, led, in most cases, to a larger consensus; particularly for the binary-&-cocomplex networks. This outcome suggested that the average parameters from the networks that obtained a global fit, better captured a common global structure present in the different organisms. We believe this common global structure was not captured by all individual parameter estimates due to a presence of biased or erroneous interactions large enough to conceal the underlaying common global structure of the PPI networks studied. This idea was particularly supported by what we observed in the Fly network, for which a large change in its structure occurred after considering the co-complex interactions reported by (Guruharsha et al., 2011) (see Table C.1). In addition, this idea could help explain the large discrepancy observed within the parameter estimates of the co-complex and binary-&-cocomplex PPI networks (Figure 4.2).

Conclusion

In this section we statistically assessed the ability of several random graph models to describe the occurrence of different small connected subgraphs in PPI networks. We made this assessment via four different network comparison statistics. Two major

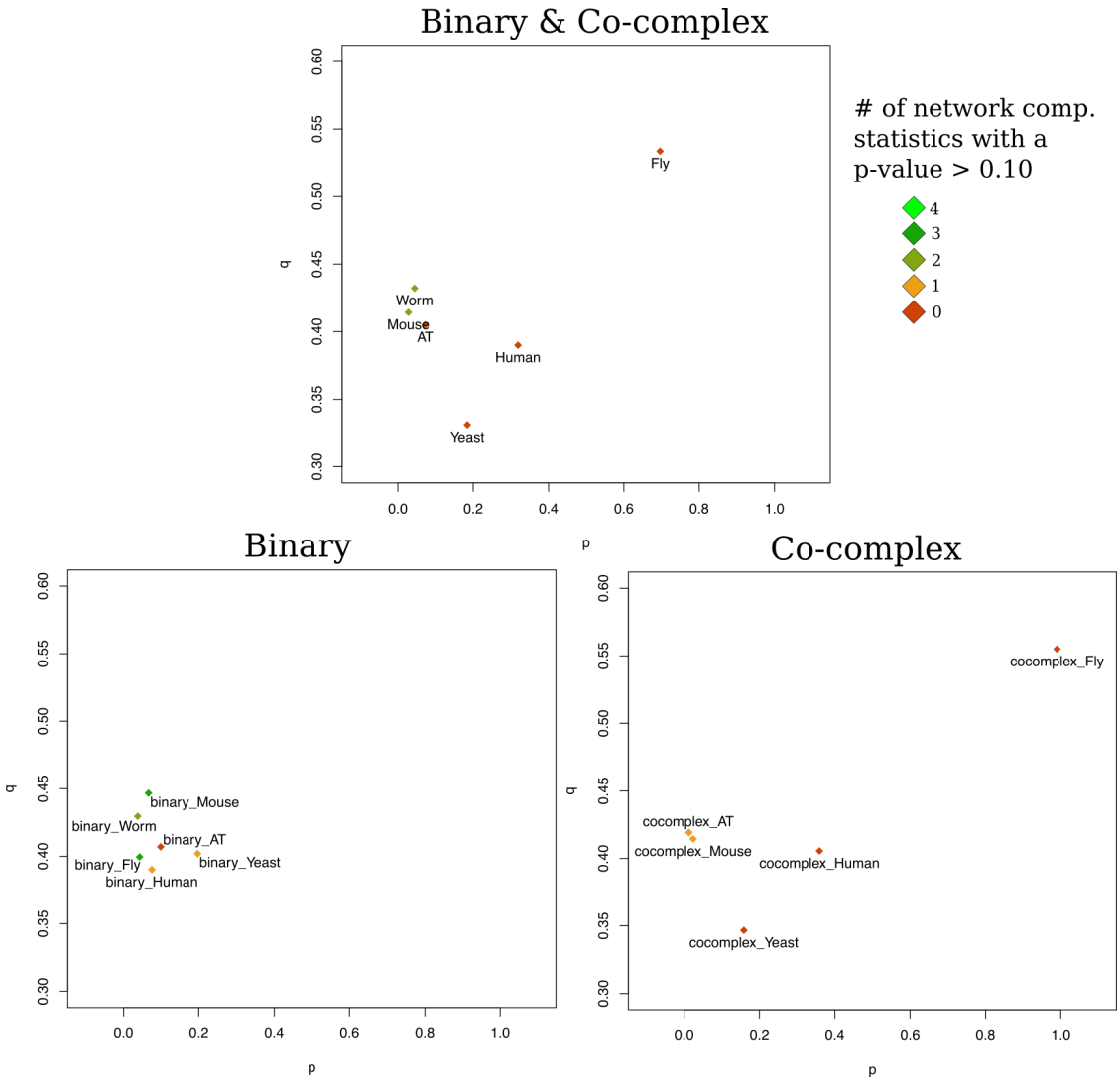


Figure 4.2: Parameters of the duplication divergence model used for all PPI networks considered. Points were coloured from red to bright green according to the number of network comparison statistics for which the Monte Carlo test obtained a p-value larger than 0.10. Note that in all three cases the parameters displayed a greater spread across the x axis in comparison to the y axis.

results from this analysis stand out. Firstly we found that the ERMG model and DD model show evidence of being able to describe the occurrence of small subgraphs of some PPI networks. In particular for the DD model, our results suggested the existence of a single value of p and q that could describe the occurrence of small subgraphs in the binary PPI networks of the different organisms considered. Secondly, that despite previous claims of the Chung-Lu model being able to describe PPI networks (Hayes et al., 2013), we did not find any evidence that supported

	GDDA	GCD	NetEmd	Netdis	Consensus	Consensus 4.2
Worm	0.30	0.03	0.39	0.03	2	2
Fly	0.13	0.03	0.25	0.04	2	0
Yeast	0.01	0.01	0.03	0.01	0	1
Human	0.01	0.01	0.11	0.30	2	0
AT	0.20	0.01	0.17	0.01	2	0
Mouse	0.47	0.01	0.04	0.01	1	2
binary_Worm	0.37	0.01	0.30	0.02	2	2
binary_Fly	0.02	0.12	0.19	0.46	3	3
binary_Yeast	0.03	0.01	0.09	0.34	1	1
binary_Human	0.02	0.02	0.12	0.01	1	1
binary_AT	0.08	0.01	0.06	0.21	1	0
binary_Mouse	0.31	0.01	0.45	0.08	2	3
cocomplex_Fly	0.48	0.02	0.08	0.05	1	0
cocomplex_Yeast	0.07	0.01	0.04	0.02	0	0
cocomplex_Human	0.10	0.01	0.11	0.29	2	0
cocomplex_AT	0.27	0.01	0.06	0.01	1	0
cocomplex_Mouse	0.34	0.02	0.03	0.01	1	1

Table 4.3: Monte Carlo p-values and consensus ($\alpha = 0.10$) across the four network comparison statistics for the DD model with parameters $p = 0.0360$ and $q = 0.4232$ for binary-&-cocomplex networks, $p = 0.0544$ and $q = 0.4231$ for binary networks and $p = 0.0183$ and $q = 0.4168$ for co-complex networks. These parameters are obtained by considering the average of the estimated parameters of the Worm and Mouse network; the binary Mouse and binary Fly networks; and the co-complex AT and co-complex Mouse networks, respectively. We selected these networks as they achieved the largest consensus across the four network comparison statistics within the binary-&-cocomplex networks, binary networks and co-complex networks, respectively. The column Consensus 4.2 shows the consensus obtained using the individual parameter estimates, given in Table 4.2.

such claim. This conclusion was observed with unanimity across the four network comparison statistics considered, including the same network comparison statistic used by Hayes et al. (2013), GDDA. In addition, in Section 2.3.2, we reached the same conclusion with the same data used by Hayes et al. (2013). Furthermore, our findings could be observed in each of the three PPI network sets considered (binary, co-complex and binary-&-cocomplex). Lastly, the results obtained from the Monte Carlo test also suggested that none of the models considered, apart from the ERMG and DD models, could explain the global occurrence of small connected subgraphs in PPI networks.

4.3 Local fit

It is commonly accepted that proteins perform functions usually in conjunction with other proteins, forming a functional module (Hartwell et al., 1999; Lewis

et al., 2010). Within these modules and across the entire network it is thought that there are small subgraph configurations which naturally occur in protein complexes (Pereira-Leal et al., 2007), and which are related to biological processes such as gene regulation and evolutionary conservation of proteins across different organisms (Wuchty et al., 2003; Alon, 2007). Therefore, understanding the process by which small subgraph configurations occur in local regions of PPI networks is a topic of interest. Here we considered the local regions of proteins by taking the local neighbourhoods of all proteins in the PPI networks.

In Section 4.2 we found that only block structures and biological mechanisms were able to globally describe the structure of some PPI networks with regards to the occurrence of small connected subgraphs. Here we tested wether the globally fitted models are suited to model some local neighbourhoods of proteins. We followed such an approach as we contemplated the possibility that different groups of local neighbourhoods could portray characteristics of different generation mechanics. In this section we tested the same random graph models used in Section 4.2.1. Similarly to section Section 4.2.1, here we follow a Monte Carlo test approach to assess the fit of these random graph models to different local regions accounted by the local neighbourhoods of proteins of the PPI networks, in relation to the occurrence of specific small connected subgraphs.

In this section we are not interested in estimating parameters for each individual local neighbourhood of the PPI networks. Instead, we are interested in using the global fit to generate a "complete" network from which to extract local regions that can describe the local neighbourhoods of PPI networks.

4.3.1 Methods

Random graph models

In this section we continue using the eight random graph models mentioned in Section 4.2.1, namely the ER model, the Configuration model, the Chung-Lu model, a power law model, the Geometric 3D model, the ERMG model and the DD model (including the DD model with an ER seed). We continued using all eight models

despite the results of Section 4.2.1, as different local regions of PPI networks could still portray different generation mechanics.

Extraction of local regions

In order to extract local regions from PPI networks we considered k-step ego-networks as this allowed the creation of regions that ranged from small structures to the whole network by considering different k-step values. Here we considered ego-networks as induced subgraphs formed by a central node (ego) and all other nodes at k-steps away from the ego (Section 1.5). For a fixed k and a network with n_v nodes, there are at most n_v k-step ego-networks. In this section we considered all n_v k-step ego-networks.

Evaluation of local fit

We evaluated the local fit of a random graph model to ego-networks of a PPI network by the ability of the model to fit the observed distribution of different connected subgraphs on 2 to 4 nodes present in a particular group of ego-networks. In detail:

1. **Grouping ego-networks:** Based on the edge density binning formed by the quantiles 5%, 10%, 15%,...,90%, 95% and 100% of the edge density of PPI ego-networks, ego-networks are classified according to their edge-density into the respective density bins. We considered this binning as it provided a large quantity of groups (20) while still comprising a large enough sample size to perform a Monte Carlo test within each bin. Other binning approaches were also considered, namely the histogram binning rule for bin lengths provided by Freedman-Diacconis (Freedman and Diaconis, 1981) ($length = 2IQRn^{-1/3}$, where IQR is the interquartile range of the edge-densities and n the number of edge-densities). However this latter method created binnings with a number of bins that ranged from 20 to 100 bins, and most of the ego-networks were placed in the first few bins, thus making interpretability of results unnecessarily confusing. Hence, we used the equal frequency binning approach

as it still achieved the same overall results but facilitating their interpretability. Figure 4.3 shows an example of this two methods applied to the 2-step ego-networks of the Human PPI network.

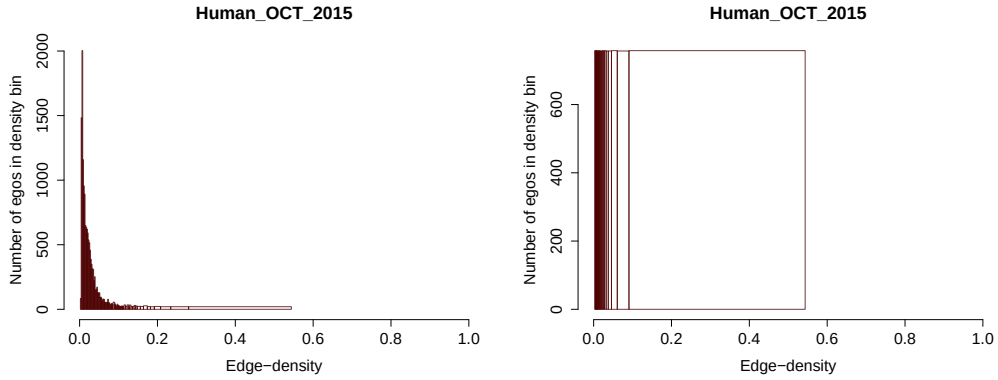


Figure 4.3: Binning of Human ego-networks according to their edge-density via the Freedman-Diacconis bin length rule (left) and equal frequency binning (right). Freedman-Diacconis gives 74 bins, with the first 5 bins accounting for 37.51% of the ego-networks. In contrast the equal frequency rule gives 20 bins, each containing approximately 5% of the ego-networks.

2. **Model ego-networks:** For a random graph model of interest, networks with the same number of nodes as the whole PPI network are generated from it. For each realisation of the random graph model, the ego-networks are classified according to the edge-density binning obtained for the PPI ego-networks. Table 4.4 shows an example of the resulting classification of Chung-Lu ego-networks using the edge-density binning obtained for the Human PPI network. Note that for the Chung-Lu ego-networks the classification does not necessarily distribute the ego-networks in equal frequencies across the binning.

Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Bin 9	Bin 10
297	251	192	278	371	573	680	681	782	869

Bin 11	Bin 12	Bin 13	Bin 14	Bin 15	Bin 16	Bin 17	Bin 18	Bin 19	Bin 20
810	861	755	728	911	1027	1271	1514	979	639

Outside of bins: 1083

Table 4.4: Binning of 2-step ego-networks of a network generated from the Chung-Lu model with the same number of nodes and edges as the Human PPI network. The bin breaks where obtained by equal frequency binning of the Human 2-step ego-networks. 1083 Chung-Lu ego-networks fell outside the binning used.

3. **Comparison statistic between groups of ego-networks:** For a given density bin, the number of subgraphs g in each ego-network in that density bin are obtained. These counts constituted a sample of the variable "number of subgraphs g in ego-network" for the selected bin. This sample of observed counts of subgraph g can then be compared against another sample of subgraph counts obtained from model ego-networks by means of the Kolmogorov-Smirnov D-statistic, i.e.

$$D = \sup_x |F_1(x) - F_2(x)|,$$

where $F_1(x)$ and $F_2(x)$, $x \in \mathbb{R}$, are the two empirical cumulative distribution functions obtained from each sample. The empirical cumulative distribution of a sample $X_1, X_2, ..., X_n$ is $F(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{X_i \leq x\}}$. Figure 4.4 gives a sketch of a single comparison between the PPI ego-networks and model ego-networks for a given density bin.

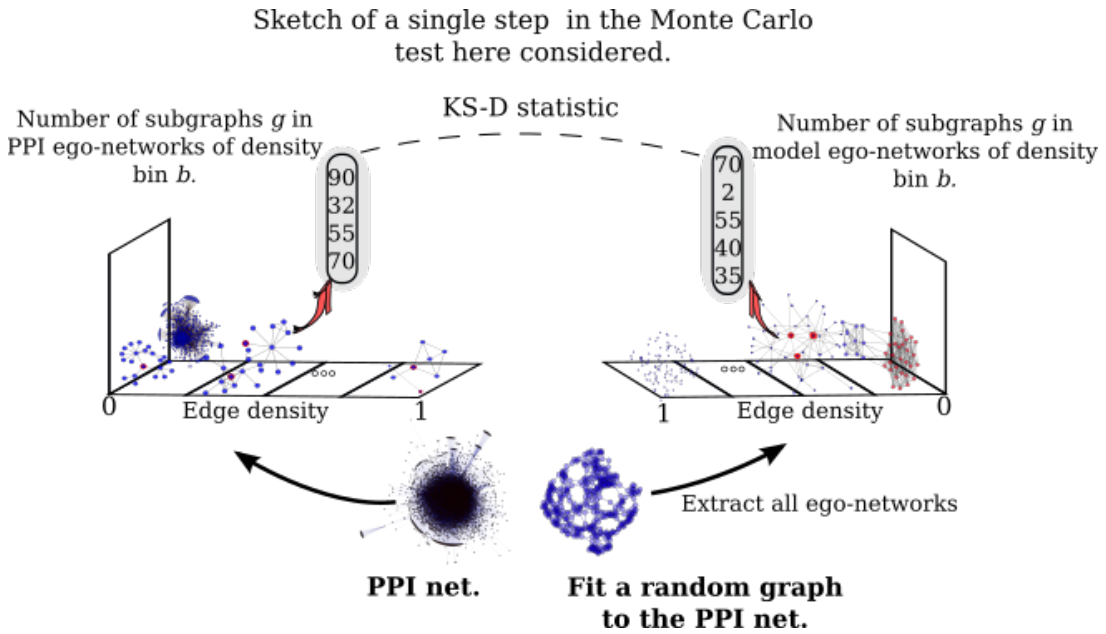


Figure 4.4: Sketch of a Monte Carlo step comparing the empirical distribution of the number of subgraphs g found in PPI ego-networks, of a given density bin, vs. the the empirical distribution of the number of subgraphs g found in ego-networks extracted from a synthetic network and which fall in the same edge-density bin. The comparison between the empirical distributions is made via the KS-D statistic. This statistic is used as statistic S in the description of the Monte Carlo test given in Section 4.2.1.

4. **Monte Carlo test:** For a given edge-density bin b, a Monte Carlo test is performed as described in Section 4.2.1; here we take S as the Kolmogorov-Smirnov D-statistic that compares the empirical distribution of the number subgraphs g in ego-networks whose edge-density falls in bin b. Here the null hypothesis is H_0: "the observed PPI ego-networks of density bin b are ego-networks extracted from a realisation of the given random graph model". Note that for each edge-density bin, a separate Monte Carlo test has to be carried out.

Subgraph counts

In this section we considered the subgraphs on 2 to 4 nodes shown in Figure 4.5. We did not consider subgraphs of size 5 for computational reasons.

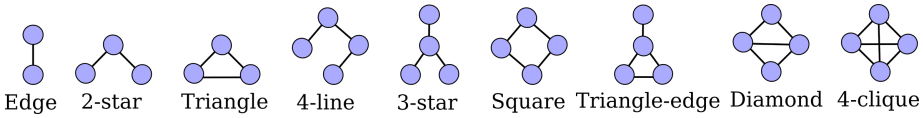


Figure 4.5: Small connected subgraphs on 2 to 4 nodes.

4.3.2 Results

In order to assess the fit of different random graph models to the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse we extracted all possible 2-step ego-networks and grouped them according to their edge-density in 20 bins, each containing approximately 5% of the total number of ego-networks. Similarly to Ali et al. (2014), we used 2-step ego-networks as this was the smallest step that created ego-networks that considered nodes that were still part of the local neighbourhood of the ego while not being directly connected to the ego itself. This step selection led to ego-networks larger than the immediate neighbourhood of a node and smaller than global sections of the network. Figure 4.6 shows the percentage of nodes included in the 2-step ego-networks of all the PPI networks studied. Most ego-networks contained less than 30% of the nodes of their respective PPI networks. However, the Yeast ego-networks covered the whole range. From ego-networks with

only a few nodes to ego-networks that covered most of the nodes in the Yeast PPI network. This behaviour might have only been observed on the Yeast PPI network, as among the 6 organism considered here, the Yeast network was the only network with a number of edges larger than some estimates of its total number of edges (Hart et al., 2006; Stumpf et al., 2008), (see Section 1.1).

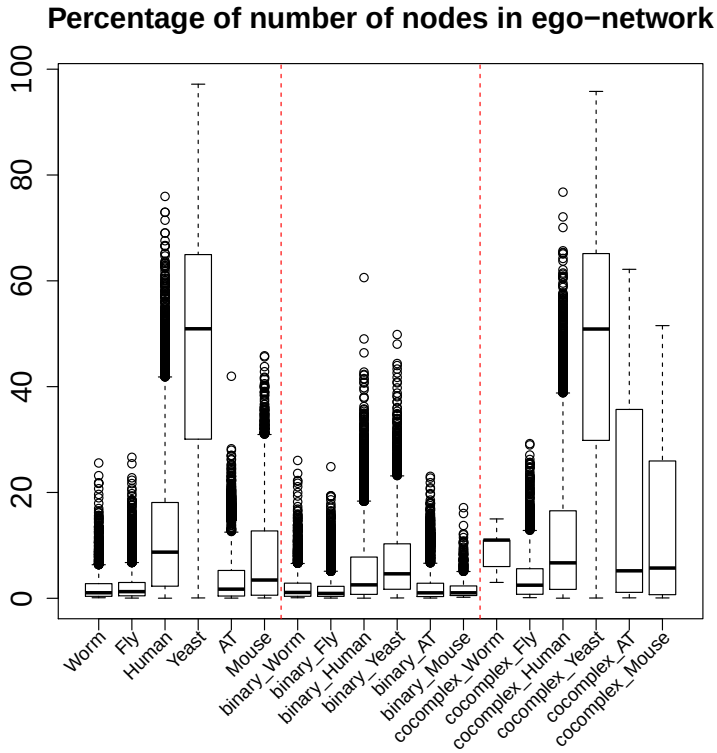


Figure 4.6: Box-plots of the proportion of nodes included in 2-step PPI ego-networks. Most Ego-networks contained less than 30% of the nodes of their respective PPI networks. A clear difference between the coverage of binary ego-networks and co-complex ego-networks can be seen, as nodes in most co-complex networks had larger node degrees than in binary networks (see Table 4.1).

Figures 4.7 and C.3 (same as 4.7 but with the x axis in log scale) show the edge-density binning used in each of the PPI networks. Each point in Figures 4.7 and C.3 represents the limits of each edge-density bin.

The binning of PPI ego-networks according to their edge-density created a classification where larger ego-networks got placed in lower density bins whereas small ego-networks were placed in the larger density bins. This occurred because the larger ego-networks were still sparse thus having a low edge-density. This phenomenon was further observed via the Spearman correlation coefficient between the edge-density and the number of nodes or number of edges. The average Spearman cor-

relation coefficient between the edge-density and the number of nodes/edges across all networks studied was -0.7291810 for the number of nodes and -0.7405458 for the number of edges. This corroborated that ego-networks in the higher edge-density bins tended to have a smaller number of nodes and/or edges. In contrast ego-networks in the lower edge-density bins tended to have an overall larger number of nodes and/or edges.

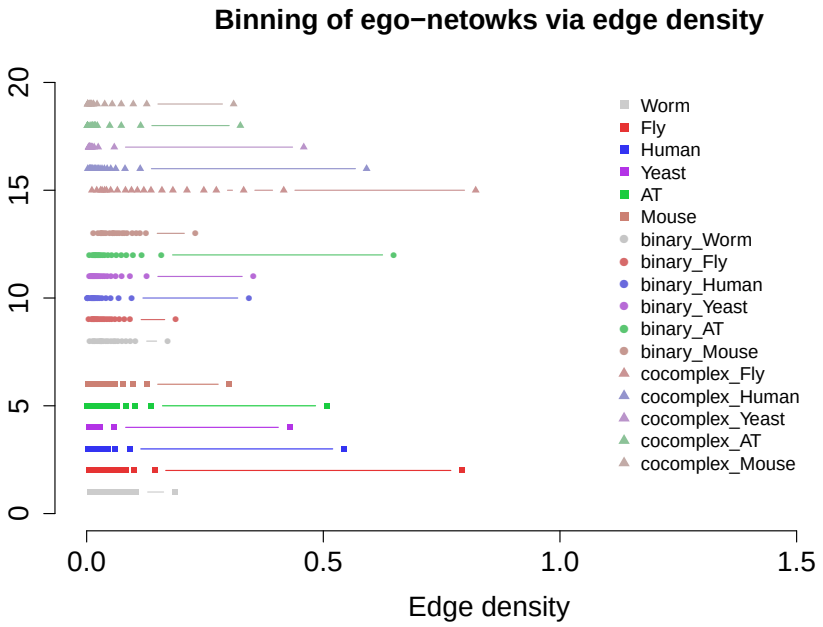


Figure 4.7: Edge density binning for the 2-step ego networks of the PPI networks of of Worm, Fly, Yeast, Human, AT and Mouse and their respective binary and co-complex networks. Most ego-networks across all six organisms have edge densities below 0.1, the smallest density is 0.00158. None of the binnings consist of equally spaced bins as the breaks are based on the quantiles of the ego-networks edge-density. For a more detailed view of the breaks at lower densities, Figure C.3 shows these breaks taking the x axis in log scale.

Figures 4.8 (binary), 4.9 (co-complex) and 4.10 (binary-&-cocomplex) show the p-values of the individual Monte Carlo test performed to assess the fit of each of the eight random graph models to ego-networks in the different edge-density bins, with regards to the nine individual subgraphs on 2 to 4 nodes shown in Figure 4.5. Each figure is arranged as a grid in order to display the p-value that corresponds to a specific PPI network, edge-density bin and subgraph. The x axis displays an array of models $\times$ subgraphs, while the y axis displays an array of different edge-density bins and PPI networks. The edge-density breaks in the y axis are replaced for the

mid edge-density value of each bin to aid readability of the figures. This mid edge-density is represented by a colour scale, as this provides a better comparison across the different PPI networks. White cells in all figures represent cases where the Monte Carlo test could not be performed due to a lack of sample size (fewer than 10 ego-networks). This happened as some random graph models did not generate enough ego-networks with edge-densities that fell in the edge-density bins obtained from the PPI ego-networks.

Results common to binary and co-complex networks. Although the type of Monte Carlo test performed in this section is different to the test performed in Section 4.2 (global fit), there are similarities in the results obtained. Most notably the ERMG model and the DD model, again, were the models which seemed to describe the structure of various PPI networks better than the other models. However, in contrast to Section 4.2 the Configuration model and Chung-Lu model also described the occurrence of some subgraph counts across some edge-density bins.

Contrary to the DD model, the ERMG model, the Chung-Lu model and the Configuration model; the ER model, Goh's power law model and the Geometric 3D model were all rejected by the Monte Carlo test for almost all edge-density bins across all PPI networks. This result was even more evident for the Geometric 3D model, as this model only generated ego-networks in the edge-density bins with the largest edge-densities. Thus leaving the remainder lower density bins with few or no ego-networks with the respective edge-densities, which prevented the execution of the Monte Carlo test in those bins. These bins are shown in white in Figures 4.8, 4.9 and 4.10.

Differential results across binary and co-complex networks. Some differences between the results for the binary, co-complex and binary-&-cocomplex networks could be observed. Firstly, the DD model showed a different performance between the binary networks and the co-complex networks. For the co-complex networks the DD model was either rejected or the test could not be performed due to

a lack of ego-networks coming from the network generated by the DD model, in the majority of edge-density bins. The opposite was the case for the binary networks, where the DD model generated ego-networks whose distribution of subgraph counts was not differentiated from the distribution of subgraph counts observed in the PPI ego-networks in several edge-density bins. However, the DD model still showed a problem across the binary, co-complex and binary-&-cocomplex networks, as it was not able to generate ego-networks with edge-densities that fell within the lower edge-density bins. In contrast, the ERMG model was able to produce ego-networks in most of the lower density bins, and was rejected more frequently for the binary networks rather than the co-complex networks. Finally, the results across Figures 4.8, 4.9 and 4.10 suggested that the DD model and the ERMG model achieved a fit to the distribution of subgraph counts in complementary edge-density regions. The ERMG seemed to achieve a fit in lower edge-density bins whilst the DD model achieved a fit in higher edge-density bins. Take for example the Yeast PPI network in Figure 4.8, or the Worm PPI network in Figures 4.8 and 4.10, or the Mouse PPI network in all three figures. These complementary cases suggested that different local regions in the PPI networks could follow different generation mechanisms.

Furthermore, this complementarity still holds even when considering the average p and q values described in Table 4.3 for the binary networks, co-complex networks and binary-&-cocomplex networks. Figure 4.11 shows the p-values of the Monte Carlo test when considering the DD model with parameters given by the average of the individual estimates of p and q of the networks with the largest consensus value in Table 4.2 within the binary networks, the co-complex networks and binary-&-cocomplex networks. Only two major differences are observed between the results of the DD model and the Avg. DD model. Firstly, the Avg. DD model was able to fit several density bins of the binary-&-cocomplex Fly network that were previously rejected for the DD model. Secondly, for the binary-&-cocomplex Yeast network, the exact opposite was obtained, as the Avg. DD model was rejected in all density bins considered. Apart from these major differences, the results for the Avg. DD model are similar to the ones obtained for the DD model which again suggested the possibility of a common set of parameter values across the organisms considered.

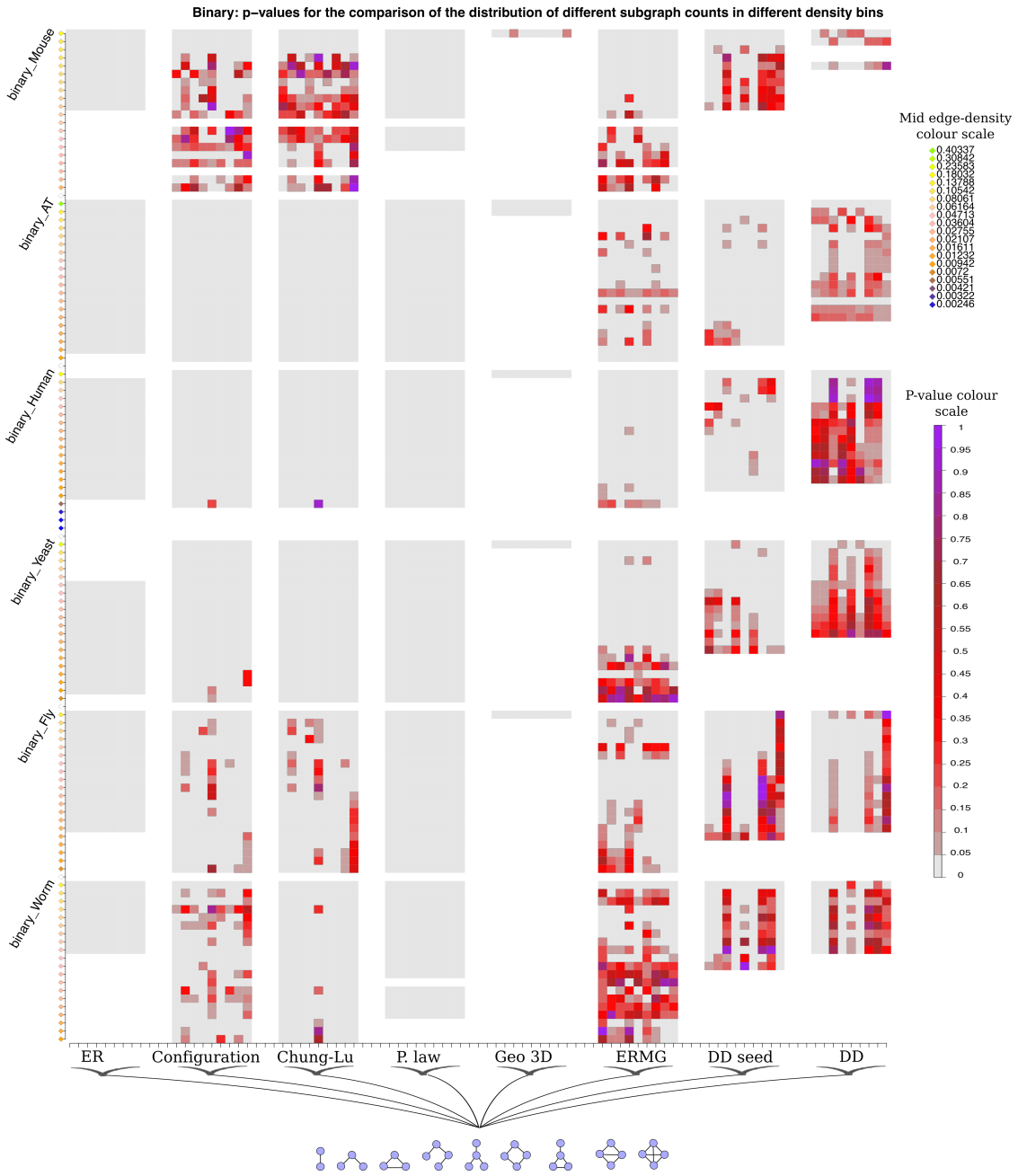


Figure 4.8: Colour coded p-values of a Monte Carlo test, (binary networks), applied to the distributions of different subgraph counts obtained from ego-networks placed in the same edge-density bins. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of edge-densities of all the ego-networks that can be extracted from the corresponding PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10. Note that the binning does not have equally spaced breaks (see Figures 4.7 and C.3).

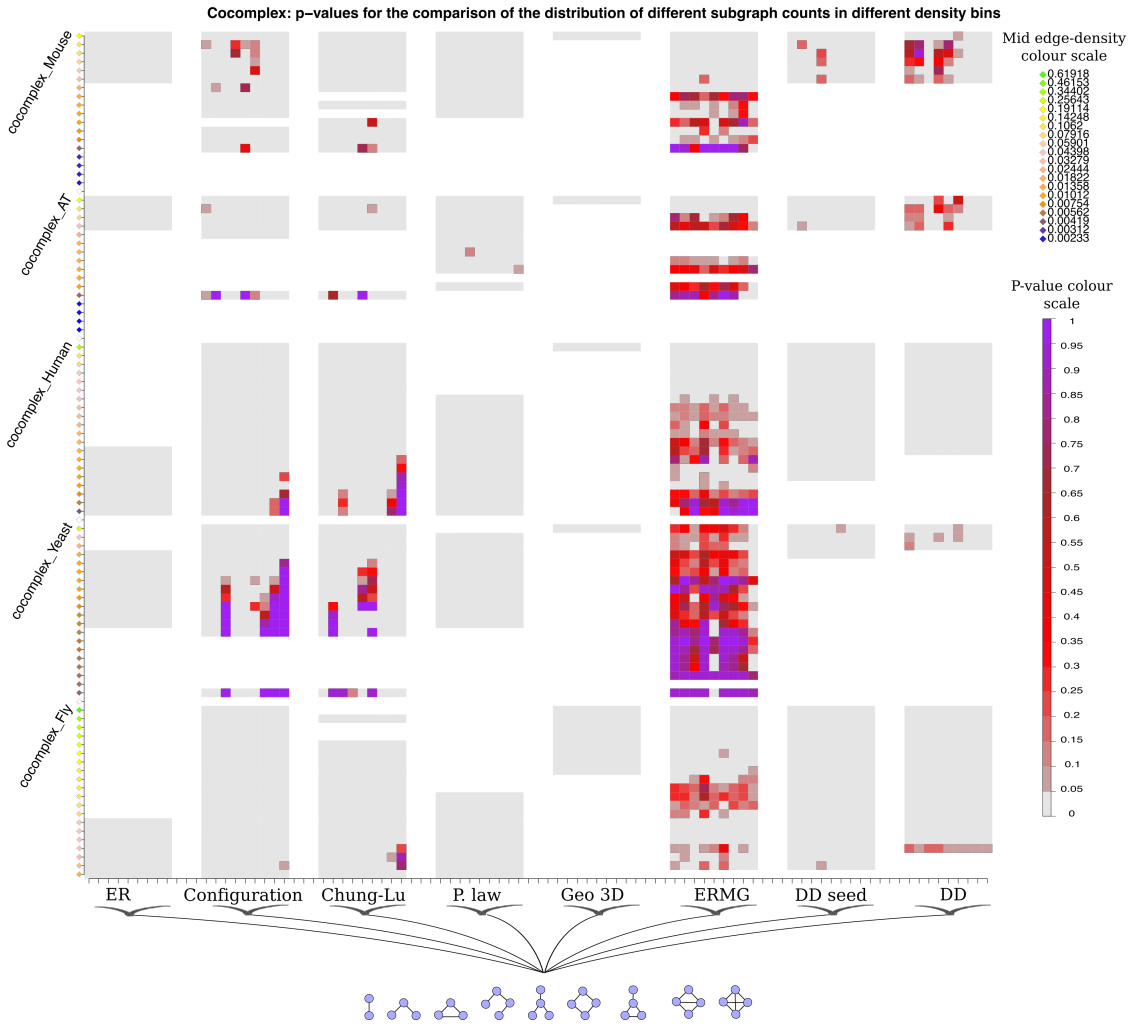


Figure 4.9: Colour coded p-values of a Monte Carlo test, (cocomplex networks), applied to the distributions of different subgraph counts obtained from ego-networks placed in the same edge-density bins. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of edge-densities of all the ego-networks that can be extracted from the corresponding PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10. Note that the binning does not have equally spaced breaks (see Figures 4.7 and C.3).

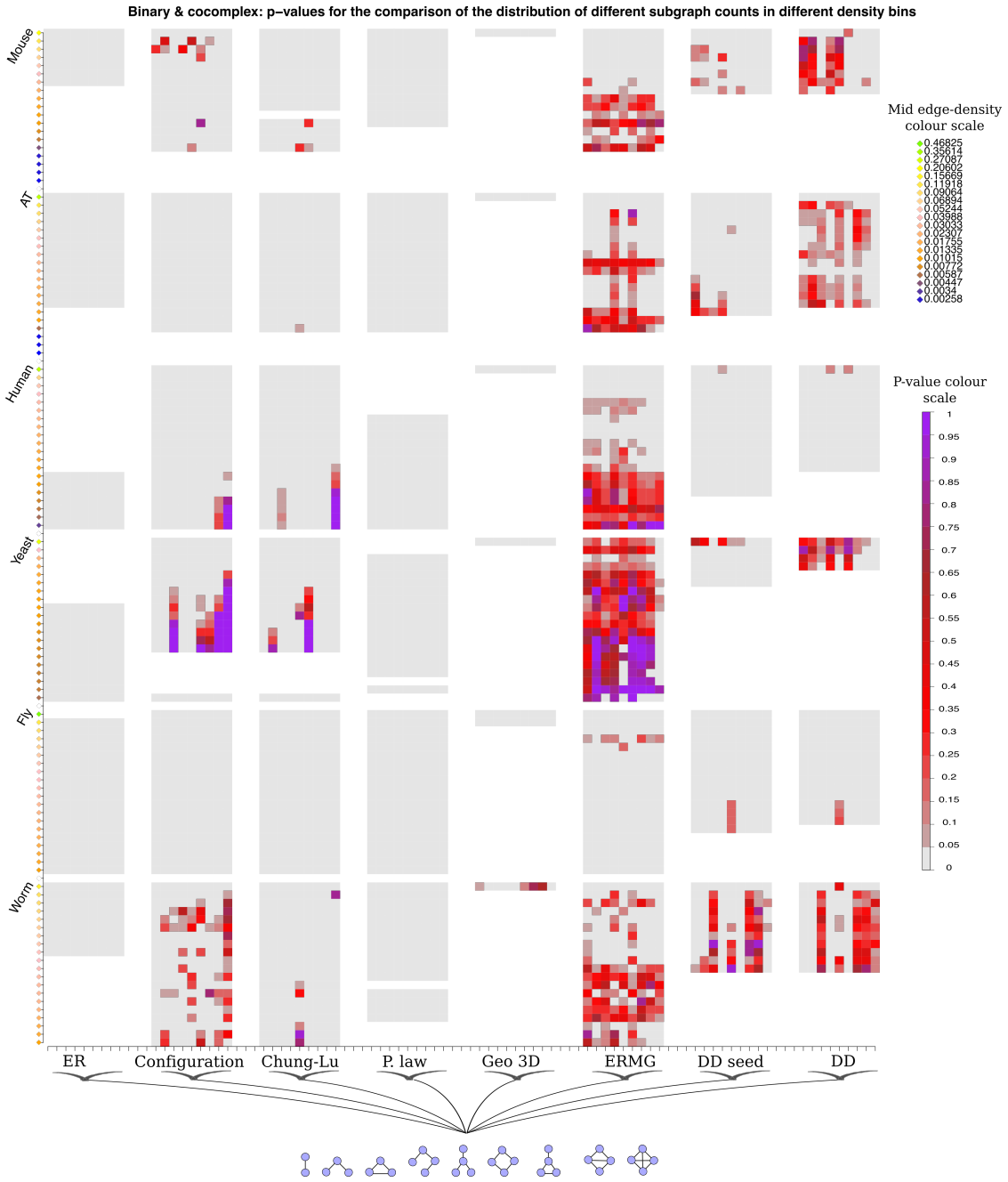


Figure 4.10: Colour coded p-values of a Monte Carlo test, (binary-&-cocomplex networks), applied to the distributions of different subgraph counts obtained from ego-networks placed in the same edge-density bins. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of edge-densities of all the ego-networks that can be extracted from the corresponding PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10. Note that the binning does not have equally spaced breaks (see Figures 4.7 and C.3).

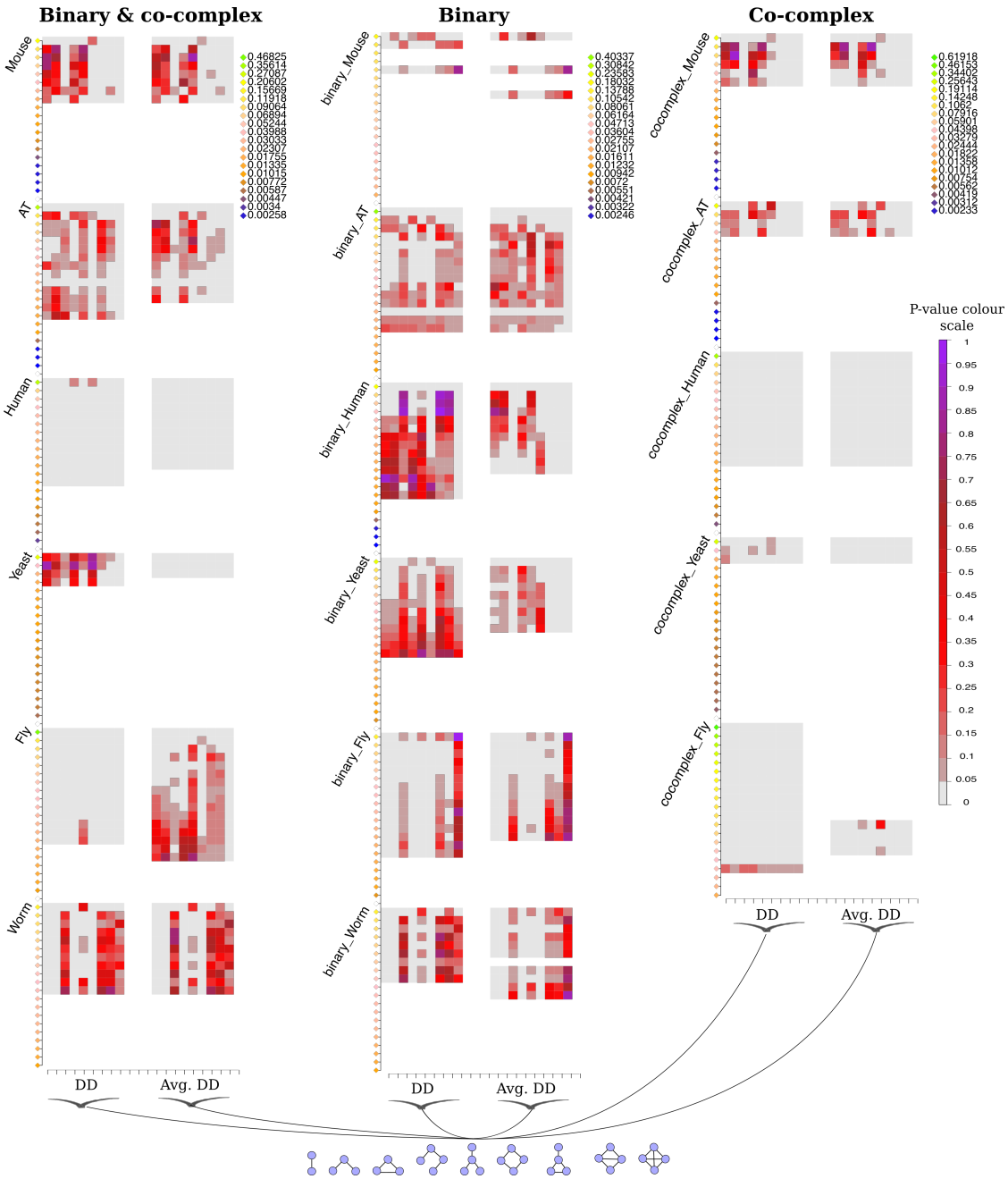


Figure 4.11: Colour coded p-values of the Monte Carlo tests used to test the fit of the DD model (Avg. DD) with parameters given by the mean of the individual estimates of p and q of the networks with the largest consensus value in Table 4.2 within the binary networks, the co-complex networks and binary-&-cocomplex networks. The results for the DD model using the individual parameter estimates of each network (Figure 4.2) is also shown for an easy comparison. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of edge-densities of all the ego-networks that can be extracted from the corresponding PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10. Note that the binning does not have equally spaced breaks (see Figures 4.7 and C.3).

4.4 Robustness of model fit to updates in protein interaction data

The protein-protein interaction networks analysed in this chapter, (obtained in October 2015), are not claimed to be complete. Indeed it is not clear how many interactions there should be in PPI networks. Some estimates can be found in (Stumpf et al., 2008; Hart et al., 2006; Dreze et al., 2011), see Table 4.5.

Organism	Estimated n_e	Reference
Worm	240544	(Stumpf et al., 2008)
Fly	74336	(Stumpf et al., 2008)
Yeast	25229	(Stumpf et al., 2008)
Yeast	75500	(Hart et al., 2006)
Human	672918	(Stumpf et al., 2008)
Human	369000	(Hart et al., 2006)
AT	299000	(Dreze et al., 2011)

Table 4.5: Estimated number of edges of the PPI networks Worm, Fly, Yeast, Human and AT from studies conducted by Dreze et al. (2011); Hart et al. (2006) and Stumpf et al. (2008). The estimates reported here from Stumpf et al. (2008) correspond to the estimates based on DIP PPI networks. Stumpf et al. (2008) provides more estimates using other datasets.

In order to test whether our previous results of global and local fit only applied to the data obtained in October 2015, in this section we repeated the analysis of the previous Sections 4.2 (global fit) and 4.3 (local fit) using PPI networks formed by protein interaction data downloaded from BioGRID in January 2017. In this analysis we used the parameter estimates already found for the PPI networks of 2015, and address the question whether the models which provided a good fit for the October 2015 datasets also provide a good fit for the January 2017 datasets, using the same parameters. Such a result would give an indication of a generally applicable model.

Table 4.6 shows the number of nodes and edges for the PPI networks of 2015 and 2017, as well as the number of new nodes and new interactions. The Yeast and Human PPI networks achieved the largest update in terms of absolute new interactions, while other networks obtained a significant relative increase ($\geq 5\%$) in their number of interactions, particularly the co-complex PPI networks (column n_e_growth). Only the PPI networks of Worm, binary Worm and binary Fly had a

very small change in their number of new nodes or new interactions (less than 20). We note that according to the estimates of Table 4.5, for most organisms, the 2017 data are still incomplete whereas for Yeast the number of reported interactions exceeds both estimates of 25229 (Stumpf et al., 2008) and 75500 (Hart et al., 2006).

	n_v_2015	n_v_2017	n_e_2015	n_e_2017	new nodes	new edges	n_e_growth
Worm	2964	2968	5426	5437	4	11	0%
Fly	7862	8015	36267	37026	153	761	2%
Yeast	5862	5931	79537	85586	69	6061	**8%**
Human	15552	15971	182681	216783	436	35604	**19%**
AT	8864	9177	33172	34461	316	1292	4%
Mouse	5156	5430	11848	13227	281	1432	**12%**
binary_Worm	2902	2904	5247	5255	2	8	0%
binary_Fly	7161	7162	23642	23646	1	5	0%
binary_Yeast	4690	4717	25815	26430	27	615	2%
binary_Human	12152	12860	61138	71099	727	10366	**16%**
binary_AT	6988	7287	27412	28349	302	940	3%
binary_Mouse	1541	1653	2459	2621	119	178	**7%**
cocomplex_Fly	2504	2942	12593	13466	438	874	**7%**
cocomplex_Human	13280	13731	133516	158819	469	26379	**19%**
cocomplex_Yeast	5616	5724	57211	62971	108	5772	**10%**
cocomplex_AT	3659	3892	6444	6969	234	526	**8%**
cocomplex_Mouse	4531	4798	9690	10955	273	1300	**13%**

Table 4.6: Number of nodes (n_v) and edges (n_e) of BioGRID PPI networks downloaded in October 2015 and January 2017. Columns "new nodes" and "new edges" show the increase in the number of new nodes and edges, respectively. Column n_e_growth shows the increase in the number of interactions relative to the previous number of interactions (increases larger than 5% are shown in bold). Network summary statistics of the 2017 PPI networks are shown in Section C.6.

4.4.1 Global Monte Carlo test of previously fitted models on updated PPI networks

In this section we tested the global fit of the ERMG models and DD models to the PPI networks obtained in January 2017. However we used the estimated parameters for the PPI networks from October 2015, and in the case of the DD model, we also used the Avg. DD model, a DD model that uses the average parameters from the 2015 PPI networks that achieved the largest consensus in Table 4.2.

The only change made to the models was to the number of nodes, as this was adjusted according to the observed number of nodes in the PPI networks of 2017. In the case of the DD model, an increase in the number of nodes means the network generation process has more node duplication steps, as more nodes need to

be added. For the ERMG model, an increase in the number of nodes implies that the initial number of nodes that has to be assigned to the different groups increases, and this assignment is done according to the block membership probabilities $\alpha_1, \alpha_2, ..., \alpha_Q$ (see Section 1.4).

The consensus results of the Monte Carlo test from the four network comparison statistics GDDA, GCD, NetEmd and Netdis are shown in Table 4.7, along with the results obtained for the 2015 PPI networks (Table 4.2) in order to facilitate their comparison. It is shown that, overall, the results remain mostly similar. The most striking discrepancy in the results shown in Table 4.7 are the results for the Fly PPI network using the Avg. DD model. The results for the Fly 2015 PPI network retrieve a consensus of 2 whereas the results for the Fly 2017 network retrieved a consensus of 0. This observation may have occurred due to the stochastic nature of the Monte Carlo test, as the results for the binary 2017 Fly and co-complex 2017 Fly networks are still similar to the ones observed for the corresponding 2015 Fly networks. In addition, a couple of p-values from the test are not conclusive, given a significance level α of 10%: the p-values obtained were 0.04, 0.02, 0.10 and 0.08 respectively for the network comparison statistics GDDA, GCD, NetEmd and Netdis see Table C.10.

We highlight the fact that the DD model, which only uses two parameters, was able to be considered as a possible generation mechanism for several updated PPI networks, and furthermore, that the DD model which only used a single pair of parameters for all organisms in the 2017 PPI networks (Avg. DD), obtained even better results than the DD model that used individual parameter values for each of the organisms. This was particularly true for the 2017 binary PPI networks.

An additional result that should be noted is that for the 2017 binary Human and 2017 co-complex Yeast networks, which increased their number of interactions by 16% and 10%, respectively, the ERMG model still achieves a large consensus, 3 for the 2017 binary Human network and 4 for the 2017 co-complex Yeast network.

Models fitted to 2015 data:	**ERMGs**		**DDs**		**Avg DD**	
	2015 PPIs	2017 PPIs	2015 PPIs	2017 PPIs	2015 PPIs	2017 PPIs
Worm	1	0	2	2	2	2
Fly	2	0	0	0	2	0
Yeast	3	3	1	0	0	0
Human	2	2	0	0	2	1
AT	2	2	0	0	2	1
Mouse	2	2	2	1	1	1
binary_Worm	2	0	2	2	2	2
binary_Fly	0	0	3	2	3	2
binary_Yeast	1	2	1	0	1	2
binary_Human	3	3	1	1	1	0
binary_AT	1	1	0	0	1	3
binary_Mouse	0	0	3	2	2	2
cocomplex_Fly	0	0	0	0	1	2
cocomplex_Yeast	4	4	0	0	0	0
cocomplex_Human	1	1	0	0	2	1
cocomplex_AT	1	2	1	1	1	1
cocomplex_Mouse	2	2	1	1	1	1

Table 4.7: Number of Monte Carlo test, for the PPI networks updated up to January 2017, with a p-value larger than 0.05 for the ERMG models and larger than 0.10 for the DD models, across the four network comparison statistics, and for the random graph models ERMG, DD and Avg. DD. The Avg. DD model uses the average parameters from the 2015 PPI networks that achieved the largest consensus in Table 4.2. Overall the results obtained for the updated networks (January 2017) are similar to the results obtained for the networks downloaded in 2015. In addition, the results of the Human PPI networks, which had the largest increased in the number of edges, remained mostly the same for both the ERMG and DD models. In addition the ERMG model still achieved full consensus for the 2017 co-complex Yeast network, which also had one of the largest increases in the number of edges. The p-values for the 2017 PPI networks can be found in Section C.6.

4.4.2 Local Monte Carlo test of previously fitted models on updated PPI networks

In addition to testing the global fit of the ERMG models and DD models fitted to the PPI networks obtained in October 2015, (Avg. DD model included), we tested the local fit of these models to the updated PPI networks of January 2017. Similarly to the previous section, the model parameters used in this section are the parameters obtained for the PPI networks downloaded in October 2015; but changing the number of nodes accordingly to the observed number of nodes in the 2017 PPI networks. In the case of the DD model, an increase in the number of nodes means the network generation process has more duplication steps, as more nodes need to be added. For the ERMG model, an increase in the number of nodes implies that the initial number of nodes that has to be assigned to the different groups increases, and this assignment is done according to the block membership probabilities $\alpha_1, \alpha_2, ..., \alpha_Q$ (see Section 1.4). Figures 4.12 and 4.13 show the results

of the Monte Carlo tests in the different edge-density bins.

Similarly to the results observed for the 2015 PPI networks, the ERMG models, the DD models and the Avg. DD models, achieved a fit to several density bins. This result could be clearly seen for the PPI networks with the largest increase in their number of interactions too, such as the Human network (19% increase), the binary Human network (16% increase) and the co-complex Mouse network (13% increase) (Table 4.6). The results shown in Figures 4.12 and 4.13, again in agreement with the results found for the 2015 PPI networks (Figures 4.9, 4.10 and 4.11), suggest a complementarity between the ERMG model and the DD model as these models seemed to fit ego-networks in complementary edge-density ranges. This finding is most notable in the binary-&-cocomplex networks and in the binary networks, although it is also noted in the co-complex Mouse network. It can also be seen that the DD model achieves in general a better fit for the binary networks rather than the co-complex networks.

4.4.3 Parameters for the DD model in the literature

Across this chapter we have shown the ability of the DD model to generate networks with similar subgraph counts to those observed in PPI networks. The fact that the results we found for the PPI networks obtained in 2015 also hold for the updated 2017 PPI networks, without changing or updating the parameters used, particularly for the DD model using the average consensus parameters (Avg. DD), is exciting . These parameters led to, an overall, larger consensus than the one observed for the individual parameter estimates of p and q for each PPI network (see Table 4.7). To our knowledge, no previous studies made a similar observation, where the PPI networks of several organisms have been modelled well by the DD model of Vázquez et al. (2003) using a single parameter estimate of p and q for all organisms. Table 4.8 shows some parameter estimates used in the literature to model PPI networks via the DD model of Vázquez et al. (2003). We also showed the average consensus parameters we used for the binary-&-cocomplex networks, binary networks and co-complex networks. It can be seen that most of the estimates of p

found in other studies largely differ from our estimates of p, which are the smallest values shown in Table 4.8. In contrast our estimates of q are within the range of previously used values of q in studies that used binary-&-cocomplex networks. The closest parameters used among the other studies shown in Table 4.8 to our own, are the estimates used by Peterson et al. (2012) for a binary Yeast network ($\hat{p} = 0.083$ and $\hat{q} = 0.582$).

$\hat{p}$	$\hat{q}$	PPI	Type of Interaction	Reference
0.0360	0.4232	-	binary-&-cocomplex	**This study**
0.26	0.33	Yeast	binary-&-cocomplex	(Shao et al., 2013)
0.22	0.54	Yeast	binary-&-cocomplex	(Shao et al., 2013)
≈0.2	≈0.4	H. pylori	binary-&-cocomplex	(Thorne and Stumpf, 2012)
≈0.25	≈0.1	Fly	binary-&-cocomplex	(Thorne and Stumpf, 2012)
0.0544	0.4231	-	binary	**This study**
0.083	0.582	Yeast	binary	(Peterson et al., 2012)
0.24	0.88	Yeast	binary	(Gibson and Goldberg, 2011)
0.1	0.7	Yeast	binary	(Vázquez et al., 2003)
0.0183	0.4168	-	co-complex	**This study**

Table 4.8: Parameter estimates used for the DD model in the literature and the average consensus parameters we obtained from the PPI networks that achieved the largest consensus in the Monte Carlo test across the four network comparison statistics (Table 4.2).

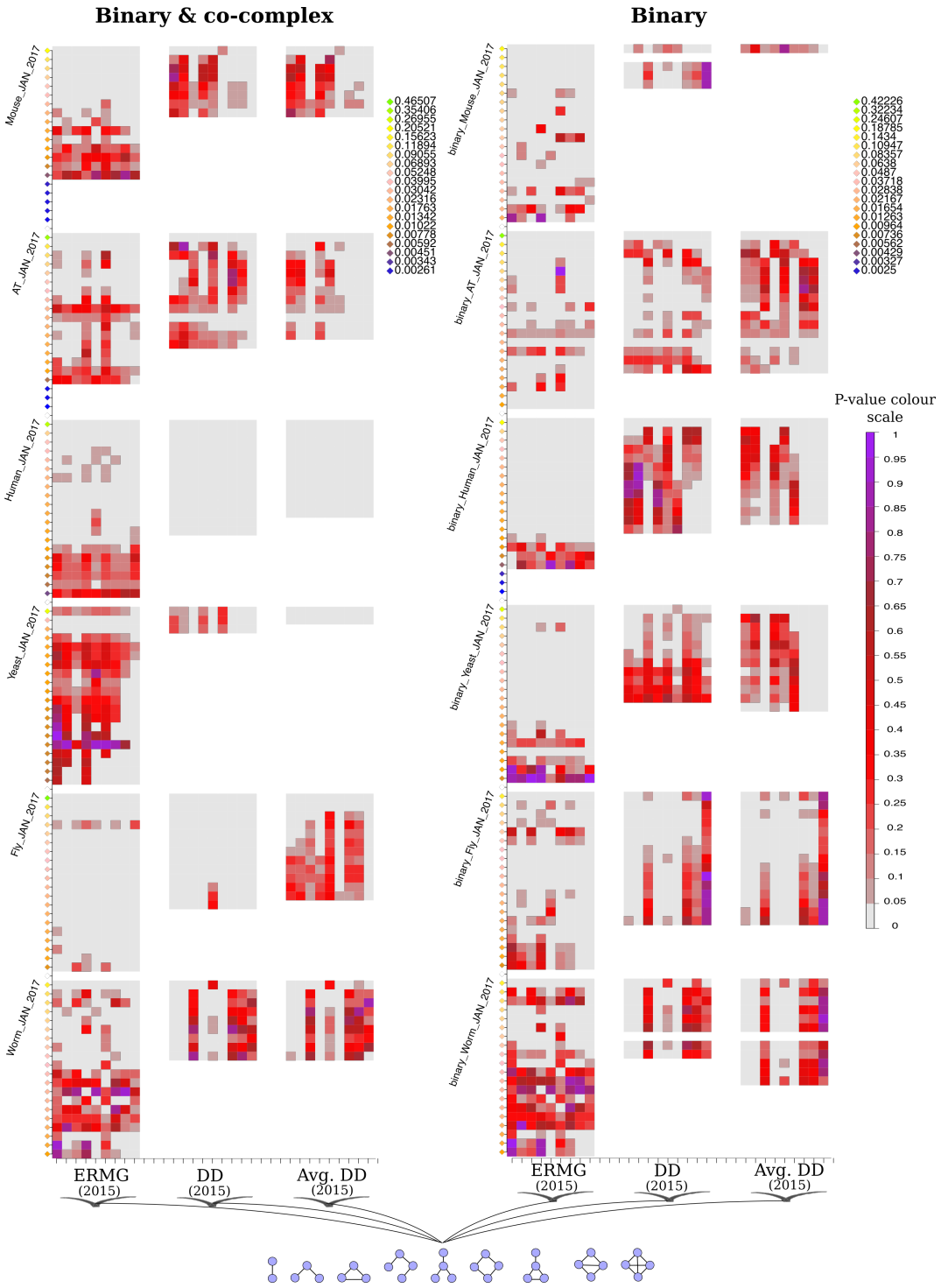


Figure 4.12: Local fit results for 2017 Binary-&-cocomplex networks and 2017 Binary networks. The figure shows colour coded p-values of the Monte Carlo tests used to test the fit of the ERMG model, the DD model and Avg. DD model using the parameter estimates obtained for the PPI networks obtained in October 2015. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of all the ego-networks extracted from the corresponding (2017) PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10.

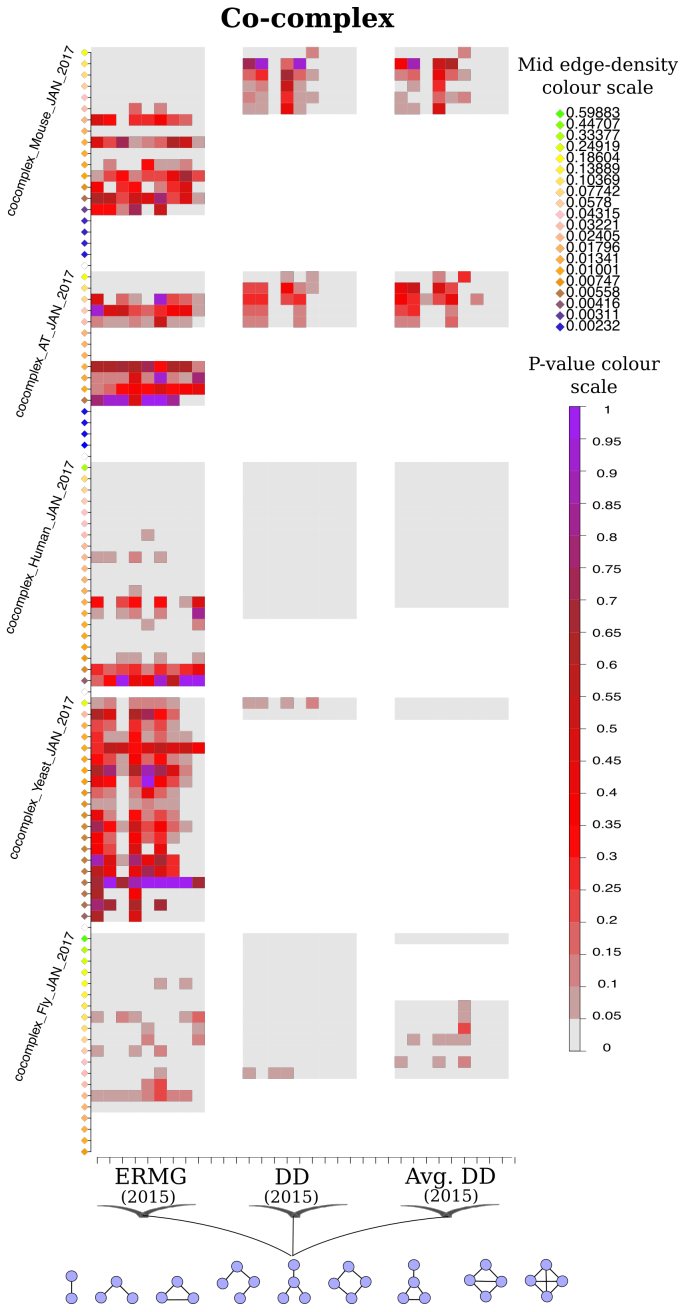


Figure 4.13: Local fit results for 2017 Co-complex networks. The figure shows colour coded p-values of the Monte Carlo tests used to test the fit of the ERMG model, the DD model and Avg. DD model using the parameter estimates obtained for the PPI networks obtained in October 2015. The subgraphs used are shown at the bottom of the figure and the results corresponding to each subgraph are arranged in the x axis for each of the models considered. The edge-density bins are created by taking as breaks the edge-density quantiles $d_0, d_5, d_{10}, d_{15}, ..., d_{95}, d_{100}$ of all the ego-networks extracted from the corresponding (2017) PPI network. For ease of readability the y axis of the plot shows on a sliding colour scale the mid edge density of the respective edge-density bin. White cells are placed for the cases in which the number of ego-networks coming from the model was less than 10.

4.5 Discussion

In this chapter we performed a systematic analysis of the capability of different random graph models to describe the global and local occurrence of small connected subgraphs in the PPI networks of six model organisms. In contrast to several previous studies (Peterson et al., 2012; Thorne and Stumpf, 2012; Shao et al., 2013; Vázquez et al., 2003; Gibson and Goldberg, 2011; Hayes et al., 2013; Przulj et al., 2010), we separately considered PPI networks formed by interactions detected by two major classes of experimental methods, (binary and co-complex methods), as well as the networks formed by all physical interactions regardless of the experimental method used to detect them. We found that, contrary to Hayes et al. (2013), there was no evidence across any of the PPI networks considered or any of the four network comparison statistics, that the Chung-Lu model globally describes PPI networks. However, we found that several PPI networks exhibit a block structure that can be captured by the ERMG model. This result is particularly true for the co-complex yeast network, as none of the four network comparison statistics was able to differentiate this PPI network from the networks generated from the ERMG model (Table 4.2). The ERMG models ranged in their number of blocks and parameters used, from 6 blocks and 28 parameters for the Worm PPI network, to 54 blocks and 1540 parameters for the Human PPI network (Table C.6).

We also found evidence to support that the biologically inspired duplication and divergence mechanic is relevant to the occurrence of small connected subgraphs in PPI networks. This result was primarily evident for some binary networks, which based on the Monte Carlo test, were considered as a possible realisation of the DD model (Table 4.2). In addition, based on the parameter estimates of the DD model for these binary networks, we found that a single parameter set for the DD model sufficed to describe the occurrence of small connected subgraphs in the PPI networks of several organisms for both 2015 and 2017 datasets (Table 4.7).

Based on the analysis of the local neighbourhoods of PPI networks, which were accounted for by all possible 2-step ego-networks of the PPI networks, we found evidence suggesting that different neighbourhoods or proteins in the PPI networks

could be formed via complementary generation mechanisms. We found that regardless of the experimental method considered, the edge-density regions where the DD model fits the data are often the same regions where the ERMG model is rejected as a good descriptor of the subgraph counts observed in the PPI ego-networks (Figures 4.8, 4.9 and 4.10).

Thanks to the consideration of PPI networks formed by interactions detected by binary and co-complex methods alone we found that the local occurrence of small connected subgraphs between the co-complex networks and binary networks is different. While the DD model is able to describe the occurrences of small subgraphs in local regions of different edge-densities in the binary networks, it is not able to achieve a similar performance for the co-complex networks (Figure 4.11).

Lastly, by considering updated PPI networks formed by protein interactions obtained up to January 2017 and testing the global and local fit of the ERMG models and the DD models to these updated networks using the parameter estimates already obtained for the PPI networks of October 2015, we found that the results and conclusion observed for the 2015 PPI networks, were still true for the 2017 PPI networks.

The results obtained for the DD model suggest that a general model that describes the structure of PPI networks with a small number of parameters is possible, as the DD model using only two parameters was able to describe the occurrence of small connected subgraphs for some of the PPI networks.

Chapter 5

Fit of a duplication divergence model to cellular compartment networks

In Chapter 4 we tested the ability of different random graph models to describe the occurrence of small connected subgraphs in the PPI networks of six organisms both globally and locally by considering 2-step ego-networks. We chose 2-step ego-networks, as these subgraphs are large enough to comprise a complex structure, and small enough to not cover the entire PPI network. By creating groups of ego-networks separated by their edge-density we found that the ERMG model and the DD model appear to describe the occurrence of different subgraphs in groups of ego-networks in complementary segments of the edge-density spectrum. Groups of ego-networks with higher edge-densities were often described better by the DD model than by the ERMG model, while ego-networks with lower edge-densities were often described better by the ERMG model than by the DD model.

We are interested in local regions of the PPI networks, as protein interactions do not occur uniformly across the eukaryotic cell. For example, proteins at some particular cellular location can be more prone to interact with each other than with proteins in a different cellular location (Rives and Galitski, 2003; Tarassov et al., 2008; Harrington et al., 2013) (see Figure D.1). In this chapter we considered the protein-protein interaction networks specific to different cellular compartments, as these can offer additional insight regarding the modular structure of PPI networks (Rives and Galitski, 2003; Pereira-Leal et al., 2007).

Based on the results found in Chapter 4, we chose to use the DD model because of its simplicity (2 parameters) and its potential to describe subgraph occurrences. We also restricted our analysis to the binary and co-complex protein interactions of Yeast and Human. These are the two organisms with the most protein interaction data available, and they provide currently the best coverage of interactions present in different cellular compartments.

Similarly to Chapter 4, we were interested in the ability of the DD random graph model to describe the occurrence of small subgraphs in location-specific PPI networks. We investigated whether there are structural similarities or differences in the cellular compartment networks that could be captured by the DD model.

The results found in Chapter 4 pointed to a possible complementarity between the DD model and the ERMG model. Here we explored a model that combined the DD model with features of the ERMG model and information regarding cellular compartmentalisation.

5.1 Methods

5.1.1 Gene ontology annotations

The Gene Ontology (GO) consortium (Ashburner et al., 2000) is an online resource that gathers information that describes the functional roles of genes, their gene products, and their locations. There are three main categories of information in the GO database, called ontologies: "Molecular Function", "Biological Process" and "Cellular Component". Information within each ontology is arranged in a directed acyclic graph (DAG), i.e. a directed graph for which there is no sequence of consecutive edges that can start and end in the same node.

Each node in the DAG is indexed by a key, called a GO term, which is associated with a broad or specific information about the gene product, which is relevant to the respective ontology. Within the molecular function ontology, which provides information regarding activities that occur at the molecular level, a GO term could, for example, refer to "enzyme regulator activity". Similarly, in the biological pro-

cess ontology, which provides information regarding a joint ensemble of molecular functions, the term could refer to "DNA replication" and in the cellular component ontology, which describes locations within the cell, the term could refer to the "nucleus". Figure 5.1 shows an example of the structure of the GO DAGs, taken from (Ashburner et al., 2000). It is shown that the structure of the ontologies is such that for any directed edge (i, j), the description of node j only adds more detail to the description associated with node i. This means that if there is a path of edges $\{(i, j), (j, z), (z, m), ..., (k, l)\}$ then the description of l provides more detail to the compound description given by the nodes $i, j, z, m, ..., k$. For example, consider the DAG in Figure 5.1; the description of node 9 is a more detailed description than the one obtained by considering only nodes 5, 3 and 1 ($\{(1, 3), (3, 5), (5, 9)\}$) or only nodes 6, 2 and 1 ($\{(1, 2), (2, 6), (6, 9)\}$). As it can be seen in Figure 5.1, each gene ontology DAG has a root node, which provides the most basic description in the entire DAG, the ontology itself.

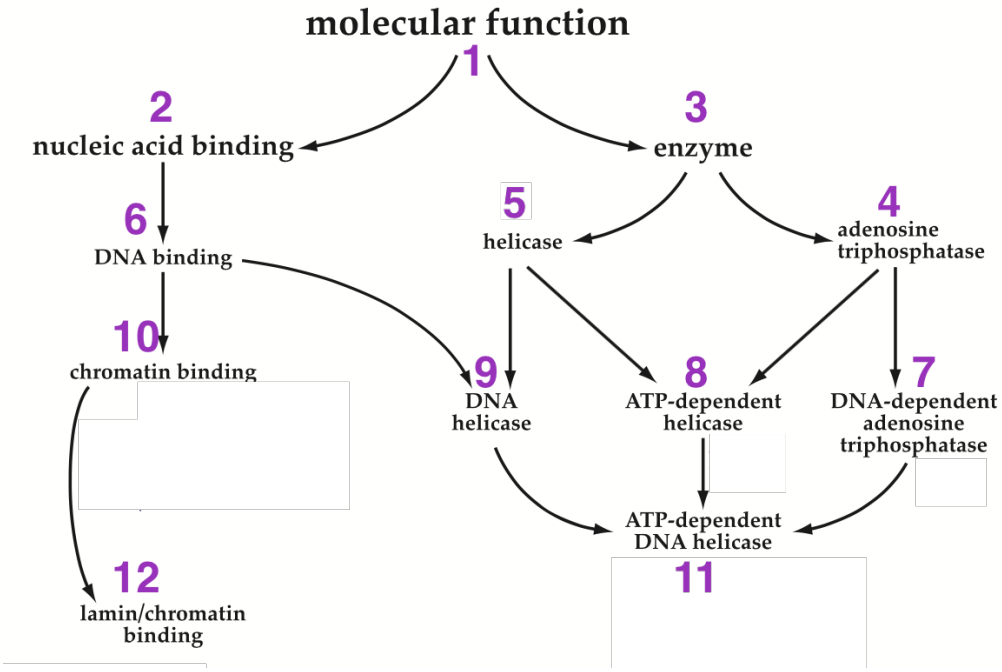


Figure 5.1: DAG example of the molecular function ontology. Figure reproduced from (Ashburner et al., 2000).

Information about specific genes cannot be directly obtained from the gene ontology DAGs, but from "gene association" files which provide the annotation of genes to

GO terms in each of the three ontologies. Each gene annotation to a GO term in the gene association files provides information regarding the evidence from which the annotation is inferred. To obtain all GO terms that are associated with a specific gene or gene product, it suffices to take all GO terms that are part of a path between the root node in the gene ontology DAG and any of the nodes with labels given by any of the GO terms associated with such gene, or gene product, in the gene association file.

Gene association files also contain information regarding the source used to infer an association between a gene and a GO term. The association could, for example, been inferred via experimental evidence, computational methods or literature reviews. However, some associations could have been inferred by using protein interaction data. Hence, the use of associations from these studies may lead us to a circular argument in our subsequent analysis of the PPI networks. For this reason, in this dissertation we refrain from using gene associations via those studies. Those studies can be identified in the gene association file by the type of evidence reported to infer the association: reviewed computational analysis (RCA) and inferred from physical interaction (IPI). We also did not consider studies whose evidence was obtained with no biological data available (ND). The gene association files and the different gene ontology DAGs can be freely obtained from the GO consortium database at `http://www.geneontology.org/` (files downloaded in May 2016). Table D.1 provides summary statistics of the gene association files we used.

5.1.2 Set up of cellular compartment networks

Cellular compartments

One broad division between cell types of organisms is given by a classification between prokaryotes such as the bacterium *E. coli*, and eukaryotes such as Yeast. The main difference between these two cell types is the presence or absence of membrane-bound regions within the cell, specifically the presence or absence of a membrane that encloses the region where the DNA is located (Alberts et al.,

2002, ch. 12). Prokaryotes do not have a membrane enclosing their DNA, while eukaryotes do have a membrane enclosing their DNA. The enclosed DNA forms a cell compartment called *nucleus*. In eukaryotic organisms, the nucleus is not the only membrane-bound space or compartment. In fact, different organisms can have different membrane-bound compartments. For example, algae and plants have chloroplasts, which are in charge of photosynthesis, while animal cells do not have such a compartment.

In this chapter we used two sets of cellular compartments, one for each of the organisms considered here: Yeast and Human. Table 5.1 shows the group of cellular compartments we used for Yeast, and which are the standard cellular compartments used by the *Saccharomyces* genome database (SGD) (Cherry et al., 1998).

Cellular compartments from SGD	
Golgi apparatus	Endoplasmic reticulum
Cell cortex	Extracellular region
Cell wall	Microtubule organizing centre
Cellular bud	Mitochondrial envelope
Chromosome	Mitochondrion
Cytoplasm	Peroxisome
Nucleolus	Ribosome
Nucleus	Cytoplasmic, membrane-bounded vesicle
Site of polarized growth	Membrane
Cytoskeleton	Plasma membrane
Endomembrane system	Vacuole

Table 5.1: Cellular compartments used for Yeast, obtained from the *Saccharomyces* genome database (Cherry et al., 1998) in July 2016.

Table 5.2 shows the cellular compartments considered for Human, which we obtained from the GO database and which cover a generic set of diverse cellular compartments in eukaryotic organisms.

All cellular compartments in Tables 5.1 and 5.2 are represented by a node in the cellular component ontology in the GO database.

Cellular compartment networks

In this chapter we focus on the Yeast and Human organisms, as these are the organism with the largest amount of interaction data available. In Chapter 4 we found that the DD model achieved a better performance for binary networks than

GO Generic cellular compartments	
Cell wall*	Microtubule organizing centre
Chromosome	Mitochondrion
Cilium	Nuclear chromosome
Cytoplasm	Nuclear envelope
Cytoplasmic chromosome	Nucleolus
Cytoplasmic, membrane-bounded vesicle	Nucleoplasm
Cytoskeleton	Nucleus
Cytosol	Organelle
Endoplasmic reticulum	Peroxisome
Endosome	Plasmamembrane
External encapsulating structure*	plastid*
Extracellular region	Protein complex
Extracellular space	Proteinaceous extracellular matrix
Golgi apparatus	Ribosome
Intracellular	Thylakoid*
Lipid particle	Vacuole
Lysosome	Cell

Table 5.2: Generic set of cellular compartments for eukaryotic cells obtained from the gene ontology database (Ashburner et al., 2000) in July 2016. * compartments not present in animal cells.

for co-complex or binary-&-cocomplex networks. Hence, in this chapter we mainly focused on binary networks; although we also considered co-complex networks in order to allow for comparison.

We used protein interactions reported via binary and co-complex methods (Section 1.1.1) from the BioGRID database (downloaded in October 2015). We also used the gene ontology database (cellular compartment ontology and gene associations files, obtained in May 2016), to allocate each protein in the Yeast and Human organisms, to the cellular compartments shown in Tables 5.1 and 5.2. A protein can be allocated to more than one cellular compartment, as the same protein can perform its function in different cellular locations, and some cellular compartments can be part of other cellular compartments, e.g. nucleolus and nucleus.

Many proteins in the PPI networks were not allocated to any of the cellular compartments considered in Tables 5.1 and 5.2, because there was no information available regarding their cellular location. Table 5.3 shows the number of proteins involved in binary interactions and co-complex interactions that were annotated to at least one cellular compartment. As a mean of comparison we show the number of proteins in the binary and co-complex PPI networks as well.

Figure 5.2 shows the distribution of the number of cellular compartments from

	Allocated proteins	Number of proteins in network
binary_Yeast	4468	4690
cocomplex_Yeast	5253	5616
binary_Human	10955	12152
cocomplex_Human	11866	13280

Table 5.3: Number of proteins allocated to at least one cellular compartment from Tables 5.1 and 5.2, for proteins involved in binary interactions alone and proteins involved in co-complex interactions alone. The number of proteins in the binary and co-complex Yeast and Human networks is also shown to aid comparison.

Tables 5.1 and 5.2 associated to the proteins in the binary network of Yeast and Human (similar plots were found for the co-complex networks, see Appendix D).

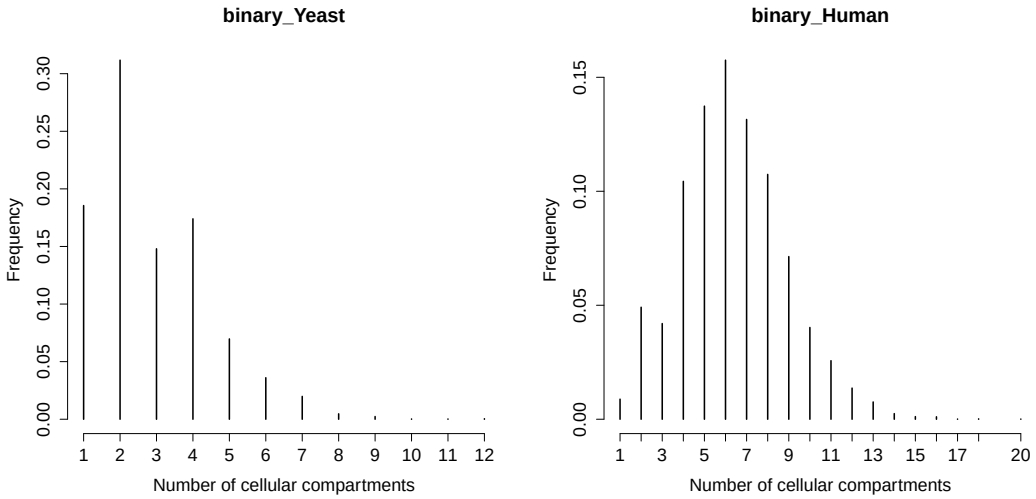


Figure 5.2: Distribution of the number of cellular compartments associated to all proteins in the whole binary networks of Yeast and Human. Similar plots were observed for co-complex Yeast and co-complex human.

The discrepancy in the distributions in Figure 5.2 can be attributed to the total number of different cellular compartments considered (22 for Yeast and 30 Human), the higher structural complexity of the Human, or the better annotation of Human genes compared to the annotation of genes from Yeast. For example, the total number of annotations in the GO database for Human is 413010, while for Yeast it is 111354 (values obtained from `http://www.geneontology.org` on March 2017). Tables 5.4 and 5.5 show the network summary statistics of the binary PPI networks formed by all proteins annotated with each single cellular compartment for Yeast and Human (Tables D.2 and D.3 show the summary statistics for the co-complex

PPI networks for Yeast and Human). The global clustering coefficient observed in the cellular compartment networks was larger than the one observed in the respective binary/co-complex Yeast or Human PPI network; for most of the cellular compartments across all four tables. This characteristic suggested that some cellular compartments contained groups of highly interacting nodes, which is a signal of the possible occurrence of functional modules, as suggested by Barabási and Oltvai (2004). Another characteristic that can be observed across all four tables is the diversity of network sizes, which range from a few tens of nodes to more than a thousand nodes.

In our subsequent analysis we filtered out networks with a small number of edges (< 100). These networks are marked with * across all four Tables 5.4, 5.5, D.2 and D.3.

Compartment	n_v	n_e	C	ρ	L	$Diam$	d
Extracellular region*	0	0	-	-	-	-	-
Cell wall*	5	5	0.4286	0.500000	1.60	3	2.00
Ribosome*	21	25	0.1210	0.119048	2.36	4	2.38
Peroxisome	45	104	0.3268	0.105051	2.66	5	4.62
Microtubule organizing centre	74	219	0.2258	0.081081	2.58	5	5.92
Cytoplasmic, membrane-bounded vesicle	84	172	0.3340	0.049340	4.45	10	4.10
Mitochondrial envelope	120	171	0.1008	0.023950	4.63	13	2.85
Cell cortex	138	518	0.2606	0.054797	2.97	9	7.51
Cellular bud	181	567	0.1468	0.034807	3.07	8	6.27
Golgi apparatus	185	500	0.2581	0.029377	3.55	8	5.41
Nucleolus	190	354	0.2397	0.019716	4.61	11	3.73
Cytoskeleton	223	998	0.2158	0.040318	2.86	6	8.95
Site of polarized growth	224	876	0.1493	0.035074	2.88	7	7.82
Vacuole	278	629	0.2773	0.016336	4.67	13	4.53
Plasma membrane	335	856	0.2722	0.015301	3.90	8	5.11
Endoplasmic reticulum	337	1272	0.2465	0.022467	3.72	10	7.55
Chromosome	373	1302	0.2298	0.018767	3.71	9	6.98
Mitochondrion	393	566	0.0659	0.007348	5.19	13	2.88
Endomembrane system	734	2891	0.1948	0.010747	3.82	10	7.88
Membrane	1433	6317	0.1535	0.006157	3.75	9	8.82
Nucleus	1894	8168	0.0751	0.004556	3.60	9	8.63
Cytoplasm	3271	15312	0.0661	0.002863	3.69	10	9.36
Binary Yeast	4690	25815	0.0617	0.002348	3.63	9	11.01

Table 5.4: Summary statistics of the Yeast cellular compartment PPI networks, for interactions detected via binary methods, in order of number of proteins. The network summary statistics of the binary Yeast network are given for comparison. *Networks not considered in further analysis due to their small number of edges (< 100). Protein interactions obtained in October 2015.

5.1.3 Random graph models and evaluation of fit

Following the results found in Chapter 4, here we chose to use the Duplication Divergence (DD) model of Vázquez et al. (2003) and the Chung-Lu model (Chung

Compartment	n_v	n_e	C	ρ	L	$Diam$	d
Lipid particle*	8	7	0.0000	0.250000	2.68	6	1.75
Peroxisome*	23	31	0.2061	0.122530	2.70	6	2.70
Ribosome*	25	24	0.0000	0.080000	2.35	5	1.92
Lysosome*	76	91	0.1081	0.031930	5.25	11	2.39
Proteinaceous extracellular matrix	91	125	0.0576	0.030525	4.16	10	2.75
Cilium	120	129	0.0039	0.018067	4.31	12	2.15
Nuclear envelope	185	352	0.0945	0.020682	3.43	8	3.81
Microtubule organizing centre	322	622	0.0853	0.012035	4.19	11	3.86
Nucleolus	377	679	0.0792	0.009580	4.38	13	3.60
Nuclear chromosome	400	1534	0.1538	0.019223	3.21	7	7.67
Endosome	426	843	0.0629	0.009312	3.89	12	3.96
Extracellular space	478	689	0.0178	0.006044	4.52	13	2.88
Mitochondrion	502	894	0.0618	0.007109	4.48	11	3.56
Cytoplasmic, membrane-bounded vesicle	594	1152	0.0370	0.006541	3.97	9	3.88
Endoplasmic reticulum	600	963	0.0290	0.005359	4.29	11	3.21
Golgi apparatus	606	1065	0.0396	0.005810	4.13	11	3.51
Vacuole	628	1339	0.0535	0.006801	3.94	11	4.26
Chromosome	662	2663	0.1193	0.012171	3.35	8	8.05
Cytoskeleton	1258	3700	0.0450	0.004680	3.69	9	5.88
Extracellular region	2189	4902	0.0171	0.002047	4.12	11	4.48
Plasma membrane	2322	6219	0.0299	0.002308	3.99	13	5.36
Nucleoplasm	2396	12652	0.0720	0.004410	3.41	8	10.56
Cytosol	2637	10848	0.0379	0.003121	3.52	10	8.23
Protein complex	2826	11883	0.0515	0.002977	3.60	10	8.41
Nucleus	5169	28530	0.0357	0.002136	3.42	8	11.04
Cytoplasm	7402	34416	0.0228	0.001256	3.62	10	9.30
Organelle	9024	47253	0.0254	0.001161	3.60	9	10.47
Intracellular	9778	52419	0.0244	0.001097	3.59	9	10.72
Cell	10520	54559	0.0234	0.000986	3.64	10	10.37
Binary Human	12152	61138	0.0207	0.000828	3.68	10	10.06

Table 5.5: Summary statistics of the Human cellular compartment PPI networks for interactions detected via binary methods in order of number of proteins. * Networks not considered in further analysis due to their small number of edges (< 100). Protein interactions obtained in October 2015.

and Lu, 2002) in order to provide contrast to the results obtained with the DD model (see Section 1.4 for a detailed description of these models).

In this dissertation we were interested in the ability of random graph models to describe the occurrence of small connected subgraphs, as some of these subgraphs have been thought to: (1) be building blocks of networks (Milo et al., 2002) and (2) be more frequently found in densely connected groups of nodes, where functional modules could occur (Barabási and Oltvai, 2004). Thus, to test whether a certain random graph model can describe the occurrence of small connected subgraphs we considered testing the null hypothesis 'H_0: Network G_0 is a realisation of model B" against the general alternative, via a Monte Carlo test based on a network comparison statistic S. This Monte Carlo test is described in Section 4.2.1.

We performed the Monte Carlo tests with four network comparison statistics, namely: GCD (Yaveroglu et al., 2014), GDDA (Pržulj, 2007), NetEmd (Wegner

et al., 2017) and Netdis (Ali et al., 2014). These network comparison methods are described in detail in Chapters 2 and 3. For the network comparison statistic Netdis we used the version with no background expectations as comparisons were made across graphs with the same number of nodes.

5.2 Results

In this chapter we addressed two questions: (1) Can the DD model describe the occurrence of small connected subgraphs in individual cellular compartments? and (2) Can we model whole PPI networks by combining the information of the cellular compartments, the DD model and the possible complementarity between the DD model and the ERMG model (mentioned in Chapter 4)?
We discuss our findings for question (1) in Section 5.2.1. and in Section 5.2.2 we discuss our findings regarding question (2).

5.2.1 Duplication divergence model as a null model for cellular compartment sub-networks

In this section we tested the ability of the DD model to describe the occurrence of small connected subgraphs in each of the specific cellular compartment PPI networks of Yeast and Human. Given that in Chapter 4 the DD model was found to perform better for binary networks, here we focused on cellular compartment networks composed of interactions detected via binary methods. However, we also considered interactions detected via co-complex methods in order to compare the results. Each DD model and Chung-Lu model was fit individually to each of the binary and co-complex cellular compartment PPI networks according to Section 1.4.
Tables 5.6 and 5.7 show the individual DD p-values from the Monte Carlo test using the four network comparison statistics, for each of the binary and co-complex cellular compartment networks of Yeast and Human. The consensus column in the tables shows the number of network comparison statistics with a p-value larger than

0.10 for the DD model. As a point of reference, we also showed in the consensus column, the consensus values for the Chung-Lu model but taking all p-values larger than 0.05. In a similar way as in Chapter 4, we chose a larger significance value for the DD model due to the large variation reported for it (Gibson and Goldberg, 2011), which can lead to a larger probability of not-rejecting the null hypothesis when it is false.

Binary Yeast

Cellular compartment PPI networks			**p-values DD model**				**Consensus**	
Cellular compartment	n_v	n_e	GCD	NetEmd	GDDA	Netdis	DD	Ch-L
Peroxisome	45	104	0.47	0.25	0.46	0.41	4	4
Microtubule organizing centre	74	219	0.46	0.06	0.05	0.26	2	1
Cytoplasmic, membrane-bounded vesicle	84	172	0.19	0.07	0.07	0.35	2	0
Mitochondrial envelope	120	171	0.10	0.47	0.34	0.02	2	1
Cell cortex	138	518	0.45	0.31	0.27	0.35	4	0
Cellular bud	181	567	0.32	0.18	0.07	0.12	3	0
Golgi apparatus	185	500	0.41	0.20	0.06	0.08	2	0
Nucleolus	190	354	0.08	0.06	0.04	0.04	0	0
Cytoskeleton	223	998	0.10	0.32	0.27	0.14	3	0
Site of polarized growth	224	876	0.37	0.48	0.25	0.16	4	1
Vacuole	278	629	0.14	0.01	0.01	0.03	1	0
Plasma membrane	335	856	0.06	0.01	0.01	0.01	0	0
Endoplasmic reticulum	337	1272	0.21	0.01	0.02	0.08	1	0
Chromosome	373	1302	0.43	0.06	0.04	0.04	1	0
Mitochondrion	393	566	0.02	0.42	0.14	0.03	2	0
Endomembrane system	734	2891	0.15	0.01	0.04	0.08	1	0
Membrane	1433	6317	0.05	0.01	0.02	0.13	1	0
Nucleus	1894	8168	0.02	0.16	0.12	0.03	2	0
Cytoplasm	3271	15312	0.01	0.08	0.04	0.01	0	0

Co-complex Yeast

Cellular compartment network			**p-values DD model**				**Consensus**	
Cellular compartment	n_v	n_e	GCD	NetEmd	GDDA	Netdis	DD	Ch-L
Microtubule organizing center	64	148	0.29	0.39	0.16	0.09	3	2
Cytoplasmic, membrane-bounded vesicle	87	348	0.17	0.01	0.02	0.01	1	0
Cell cortex	120	323	0.15	0.01	0.01	0.01	1	0
Cellular bud	164	405	0.32	0.14	0.10	0.15	3	0
Golgi apparatus	194	745	0.27	0.01	0.01	0.01	1	0
Cytoskeleton	206	673	0.03	0.03	0.34	0.02	1	0
Site of polarized growth	211	593	0.26	0.31	0.21	0.25	4	0
Endoplasmic reticulum	261	729	0.04	0.01	0.01	0.01	0	0
Vacuole	270	930	0.01	0.03	0.07	0.02	0	0
Ribosome	286	1340	0.03	0.10	0.41	0.01	1	0
Nucleolus	291	3055	0.14	0.01	0.01	0.01	1	0
Plasma membrane	294	780	0.02	0.16	0.29	0.01	2	0
Mitochondrial envelope	319	1080	0.02	0.06	0.01	0.01	0	0
Chromosome	402	2542	0.07	0.01	0.01	0.01	0	0
Endomembrane system	656	2740	0.07	0.01	0.01	0.01	0	0
Mitochondrion	856	3027	0.01	0.02	0.09	0.01	0	0
Membrane	1473	6903	0.01	0.04	0.01	0.01	0	0
Nucleus	2217	22117	0.01	0.01	0.06	0.01	0	0
Cytoplasm	4007	33445	0.01	0.03	0.01	0.01	0	0

Table 5.6: Yeast: Monte Carlo p-values obtained for the DD model, using network comparison statistics GDDA, GCD, NetEmd and Netdis. The number of network comparison statistics with a p-value larger than 0.10 in shown in the consensus column. The consensus achieved by the Chung-Lu model is also shown as reference point. Consensus values for the Chung-Lu model were obtained by p-values larger than 0.05.

It can be seen in Tables 5.6 and 5.7, that at least two network comparison statistics

Binary Human

Cellular compartment network			***p*-values DD model**				**Consensus**	
Cellular compartment	n_v	n_e	GCD	NetEmd	GDDA	Netdis	DD	Ch-L
Proteinaceous extracellular matrix	91	125	0.17	0.11	0.31	0.41	4	3
Cilium	120	129	0.03	0.14	0.15	0.01	2	1
Nuclear envelope	185	352	0.09	0.48	0.13	0.01	2	1
Microtubule organizing centre	322	622	0.08	0.21	0.04	0.26	2	2
Nucleolus	377	679	0.10	0.29	0.27	0.02	2	3
Nuclear chromosome	400	1534	0.42	0.50	0.40	0.44	4	0
Endosome	426	843	0.04	0.30	0.12	0.01	2	0
Extracellular space	478	689	0.03	0.37	0.44	0.01	2	0
Mitochondrion	502	894	0.08	0.26	0.08	0.48	2	2
Cytoplasmic, membrane-bounded vesicle	594	1152	0.03	0.13	0.14	0.01	2	0
Endoplasmic reticulum	600	963	0.02	0.10	0.26	0.01	1	0
Golgi apparatus	606	1065	0.01	0.13	0.03	0.01	1	0
Vacuole	628	1339	0.01	0.28	0.18	0.01	2	0
Chromosome	662	2663	0.24	0.10	0.14	0.50	3	0
Cytoskeleton	1258	3700	0.02	0.19	0.20	0.01	2	0
Extracellular region	2189	4902	0.02	0.16	0.07	0.01	1	0
Plasma membrane	2322	6219	0.02	0.26	0.32	0.01	2	0
Nucleoplasm	2396	12652	0.23	0.06	0.02	0.16	2	0
Cytosol	2637	10848	0.01	0.19	0.15	0.01	2	0
Protein complex	2826	11883	0.02	0.05	0.11	0.01	1	0
Nucleus	5169	28530	0.01	0.04	0.02	0.01	0	0
Cytoplasm	7402	34416	0.02	0.09	0.05	0.01	0	0
Organelle	9024	47253	0.01	0.08	0.01	0.01	0	0
Intracellular	9778	52419	0.01	0.10	0.03	0.01	0	0
Cell	10520	54559	0.01	0.10	0.03	0.01	0	0

Co-complex Human

Cellular compartment network			***p*-values DD model**				**Consensus**	
Cellular compartment	n_v	n_e	GCD	NetEmd	GDDA	Netdis	DD	Ch-L
Peroxisome	66	102	0.03	0.36	0.45	0.04	2	0
Cilium	128	198	0.03	0.12	0.23	0.05	2	0
Ribosome	179	3461	0.02	0.01	0.03	0.01	0	0
Nuclear envelope	203	453	0.35	0.46	0.43	0.09	3	0
Lysosome	235	496	0.19	0.01	0.01	0.01	1	0
Microtubule organizing centre	392	1026	0.50	0.19	0.30	0.15	4	0
Nuclear chromosome	427	2920	0.39	0.05	0.10	0.15	2	0
Extracellular space	476	895	0.03	0.27	0.42	0.03	2	0
Endosome	495	1536	0.17	0.11	0.10	0.01	2	0
Nucleolus	599	3820	0.22	0.01	0.01	0.01	1	0
Cytoplasmic, membrane-bounded vesicle	707	2252	0.32	0.17	0.03	0.06	2	0
Vacuole	743	2585	0.09	0.03	0.03	0.01	0	0
Chromosome	746	5260	0.39	0.05	0.27	0.13	3	0
Golgi apparatus	757	1938	0.12	0.24	0.11	0.03	3	0
Endoplasmic reticulum	875	2587	0.10	0.08	0.10	0.05	0	0
Mitochondrion	1092	4922	0.08	0.01	0.05	0.01	0	0
Cytoskeleton	1358	5757	0.15	0.01	0.02	0.09	1	0
Nucleoplasm	2655	30354	0.17	0.01	0.04	0.02	1	0
Extracellular region	2676	17307	0.01	0.01	0.01	0.02	0	0
Plasma membrane	2677	11522	0.04	0.05	0.03	0.02	0	0
Cytosol	2927	29161	0.01	0.01	0.01	0.01	0	0
Protein complex	3235	32766	0.12	0.01	0.11	0.03	2	0
Nucleus	5438	61184	0.01	0.01	0.01	0.02	0	0
Cytoplasm	8260	85187	0.01	0.01	0.01	0.01	0	0
Organelle	9815	109687	0.01	0.01	0.02	0.01	0	0
Intracellular	10594	116904	0.02	0.01	0.01	0.01	0	0
Cell	11436	122420	0.01	0.01	0.01	0.01	0	0

Table 5.7: Human: Monte Carlo p-values obtained for the DD model, using network comparison statistics GDDA, GCD, NetEmd and Netdis. The number of network comparison statistics with a p-value larger than 0.10 in shown in the consensus column. The consensus achieved by the Chung-Lu model is also shown as reference point. Consensus values for the Chung-Lu model were obtained by p-values larger than 0.05.

did not reject the DD model as a model from which several binary cellular compartment networks of Yeast and Human could have come from. Similar results can be seen for the co-complex cellular compartment networks, although for a smaller number of cellular compartments, particularly cellular compartment networks with a smaller number of nodes. In contrast to the DD model, the Chung-Lu model was rejected in more than 80% of the cases by all network comparison statistics and for both binary and co-complex networks across both organisms studied.

In Chapter 2 we tested the ability of the Chung-Lu model to describe the subgraph counts in five virus networks which have a number of nodes (Table 2.3) similar to several of the cellular compartment used in this chapter. Contrary to the results we obtained for the Chung-Lu model in Chapter 2, where the Chung-Lu model was able to fit some of the virus networks (see Table 2.5), here the Chung-Lu model was rejected for most cellular compartment networks. This contrasting result obtained for the Chung-Lu model supports the idea that the results obtained for the DD model are meaningful, and not an artefact produced by using smaller PPI networks.

Similar to the results presented in Chapter 4, where we showed the parameter values used in the DD model for the whole binary and co-complex networks, in Figures 5.3 and 5.4 we show the parameter values used for the DD model for each of the cellular compartments. Each pair of parameter values is coloured according to the consensus from the Monte Carlo test shown in Tables 5.6 and 5.7.

One particular characteristic that can be seen in Figures 5.3 and 5.4, across the binary and co-complex networks, is the concentration of the values for the divergence parameter q around values close to 0.4, see Figure D.3 for a histogram of these values where this concentration can be clearly visualised. This characteristic suggests that rates by which duplicated interactions are lost are similar across the entire network regardless of where in the cell the duplicated interaction takes place. This result is complemented by the results found in Chapter 4 for the whole binary PPI networks of Worm, Fly, Yeast, Human, AT and Mouse (Figure 4.2, Table 4.8), where the values of q gathered around values close to 0.4 across all PPI networks. These findings give a potential suggestion about the existence of a global rate by which duplicated interactions are lost across different organisms and

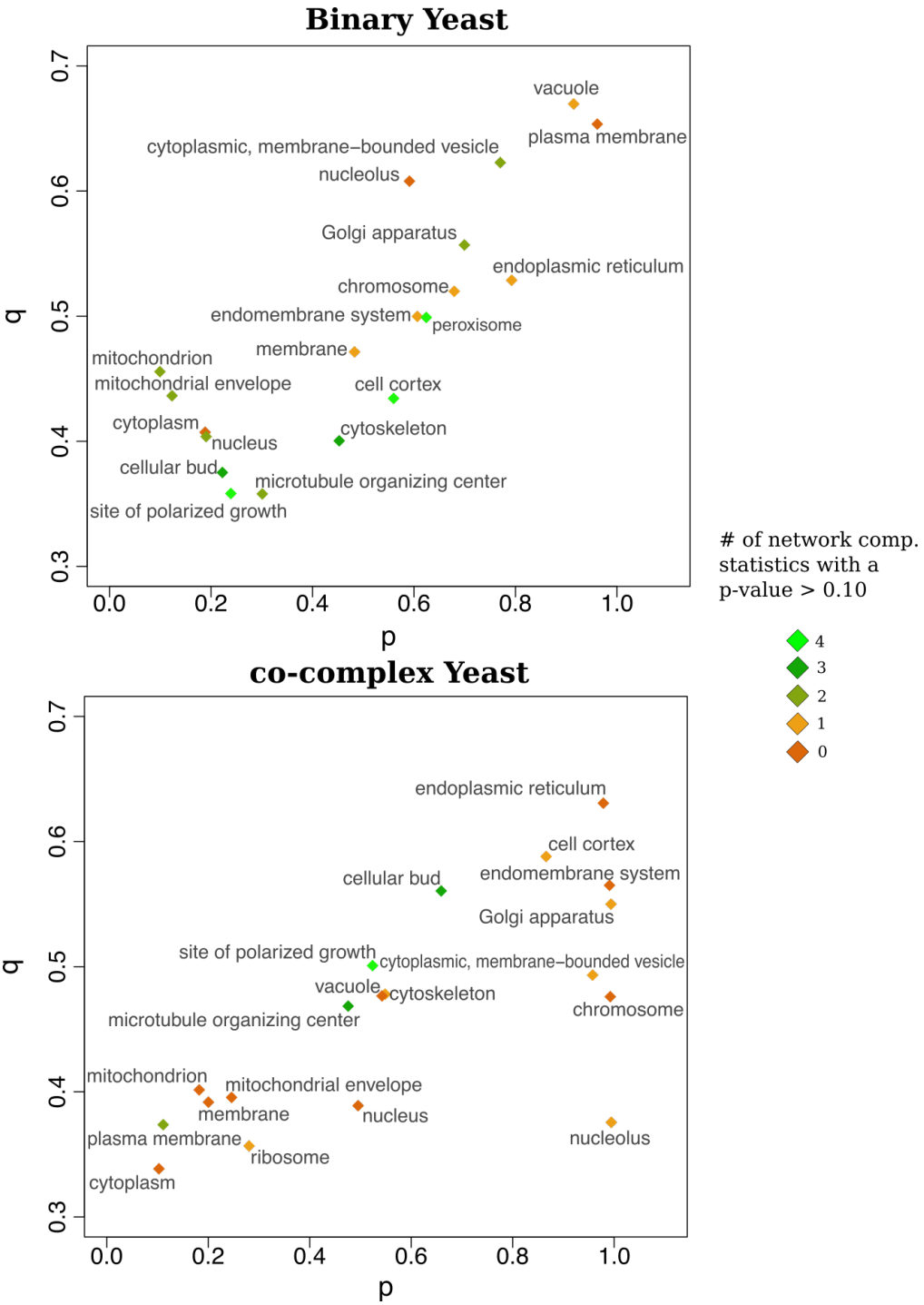


Figure 5.3: Parameter values used in the DD model for the binary and co-complex cellular compartment networks of Yeast. Each point is coloured according to the consensus obtained among the four network comparison statistics in the Monte Carlo test (see Table 5.6).

cellular compartments.

In contrast, Figures 5.3, 5.4 and D.3 show that the values used for the parameter

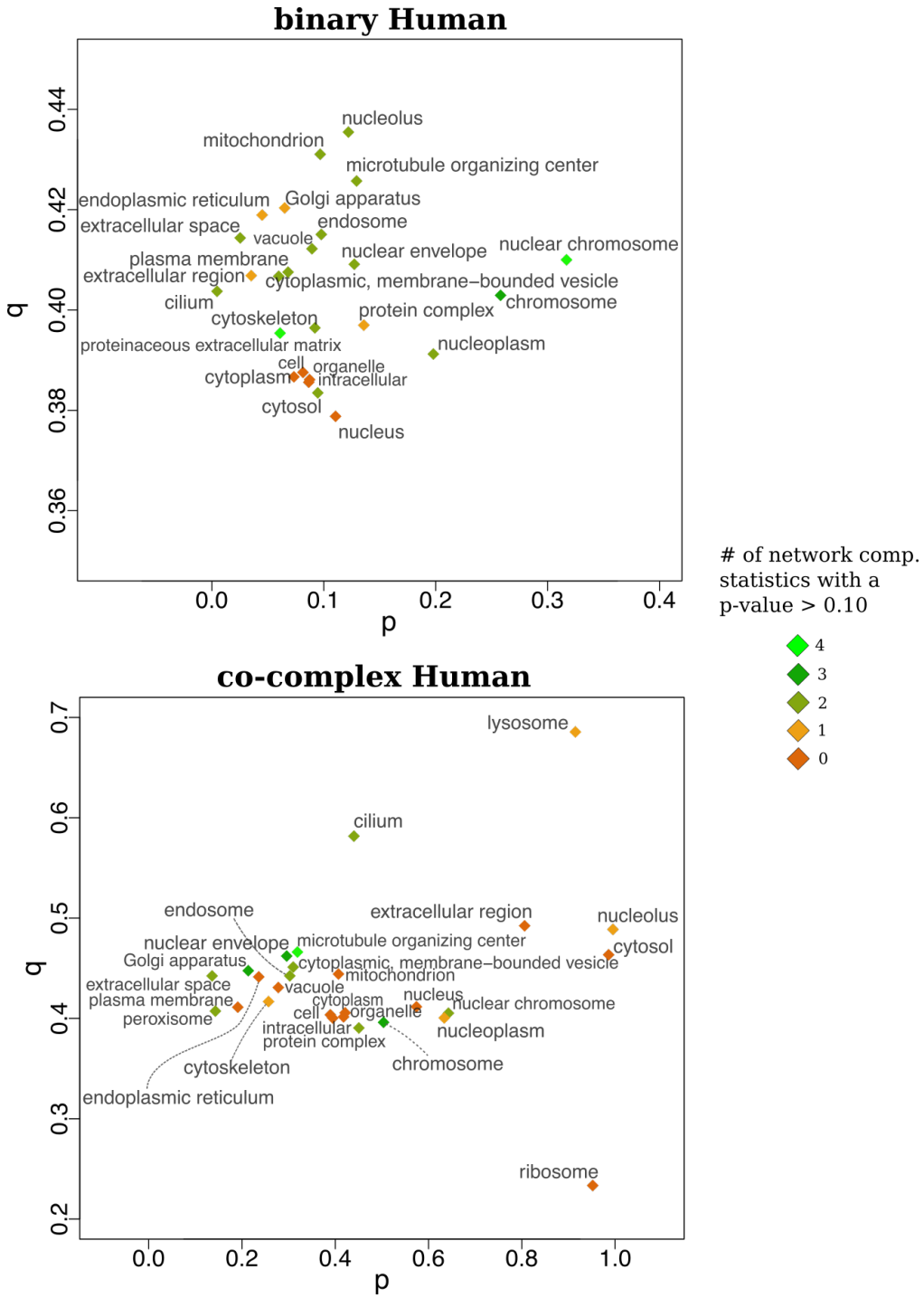


Figure 5.4: Parameter values used in the DD model for the binary and co-complex cellular compartment networks of Human. Each point is coloured according to the consensus obtained among the four network comparison statistics in the Monte Carlo test (see Table 5.7).

p associated to the probability of interaction between node duplicates, were widely spread along the interval $[0, 1]$ for all cases apart from the Human binary cellular

compartments. This result suggests that the differences in the network structure across the different cellular compartments can be mostly explained by the rate at which duplicated proteins are connected to the nodes they were duplicated from, or relatedly by the concentration of triangles present at the different cellular compartments.

5.2.2 Exploring a duplication-divergence model based on cellular compartments

We found in the previous section that the DD model is able to describe the occurrence of small subgraphs in several cellular compartment networks, particularly cellular compartment networks composed of interactions detected via binary methods. In Chapter 4 we found that the ERMG model and the DD model can potentially describe the occurrence of small subgraphs in groups of ego-networks at complementary regions of the edge-density.

In this section we follow these previous analyses to explore a random graph model proposal that assumes a block structure, and where each block is formed by a network generated from a DD model. The model consists of two steps: (1) DD networks are generated with their own individual duplication-divergence parameters. (2) Edges are created between the nodes of the different DD networks.

DD-block model

Given K blocks, let α_k be the probability of a node to be assigned to block C_k and $\eta_k = (p^k, q^k)$ be the duplication and divergence parameters associated to block C_k, $k = 1, 2.., K$. Assume that $\sum_k \alpha_k = 1$, i.e. each node is assigned to exactly one block. Then blocks $C_1, C_2, ...C_K$ have size $c_1, c_2, ..., c_K$ where $(c_1, c_2, ..., c_K) \sim M(n_v, \alpha_1, \alpha_2, ..., \alpha_K)$.

For each block C_k, a DD network of size c_k and parameters η_k is generated, this DD network gives the edges within block C_k. For edges between blocks, let g_i be a variable indicating the block membership of node i, $i = 1, 2, ..., n_v$, where $n_v = \sum_k c_k$. Then, for each pair of nodes i in block C_{g_i} and j in block C_{g_j}, with

$g_i \neq g_j$, an edge is created between these two nodes with probability equal to $d_i d_j \tau_{g_i g_j} \rho$, where d_i and d_j are the degrees of nodes i and j in their corresponding DD networks, $\tau_{g_i g_j}$ is a parameter controlling the interactions between blocks and ρ is a parameter controlling the edge-density of the overall network. Figure 5.5 shows a sketch of this model. The probability $d_i d_j \tau_{g_i g_j} \rho$, is based on the probability of interaction between two nodes in the degree corrected stochastic block models (Karrer and Newman, 2011), which are a generalisation of the ERMG model.
This construction of the model leads to $2K$ DD parameters plus $\binom{K}{2}+1$ parameters for edges between blocks.

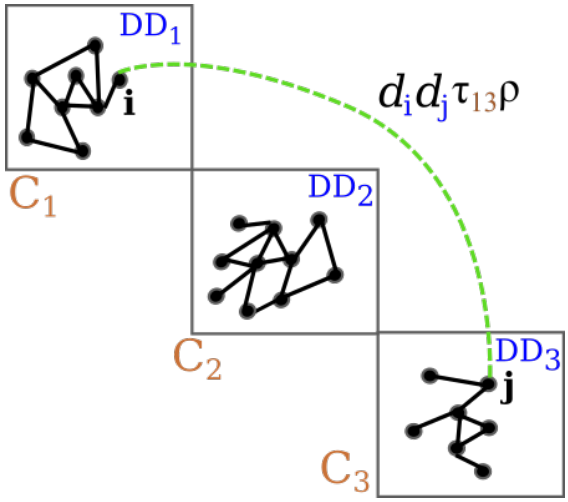


Figure 5.5: Sketch of the DD block model described above. In this example the model starts by generating 3 DD networks, (each associated to a block C_k), with different number of nodes and parameters. Then edges are created between nodes of different blocks with a probability equal to $d_i d_j \tau_{g_i g_j} \rho$, where d_i and d_j are the degrees of nodes i and j in their respective DD networks, τ_{13} is a parameter controlling the interactions between nodes in blocks C_1 and C_3, and ρ is a parameter that controlls the overall edge density of the resulting network.

Application of DD-block model

In this section we explored the fit of the DD-block model to the whole PPI network rather than to smaller regions of the network. We proposed a DD-block model motivated by the large consensus achieved by the DD model across several binary cellular compartments networks (Table 5.6) and by the results obtained for the ERMG model in Chapter 4.

Due to time limitations we were only able to explore this model for the binary Yeast

network. We used Yeast as we believed this was the network which was the closest to being completed. Furthermore, we studied a binary Yeast network, which we called binary-Yeast-C2, and which was formed by nodes in cellular compartments for which the DD model achieved a consensus of at least 2 in Table 5.6.

Figure 5.6 shows the proportion of proteins in each cellular compartment as well as the proportion of 'shared' proteins between each pair of cellular compartments used to form the binary-Yeast-C2 network (compartments that 'shared' less than 1% of the nodes are shown in white).

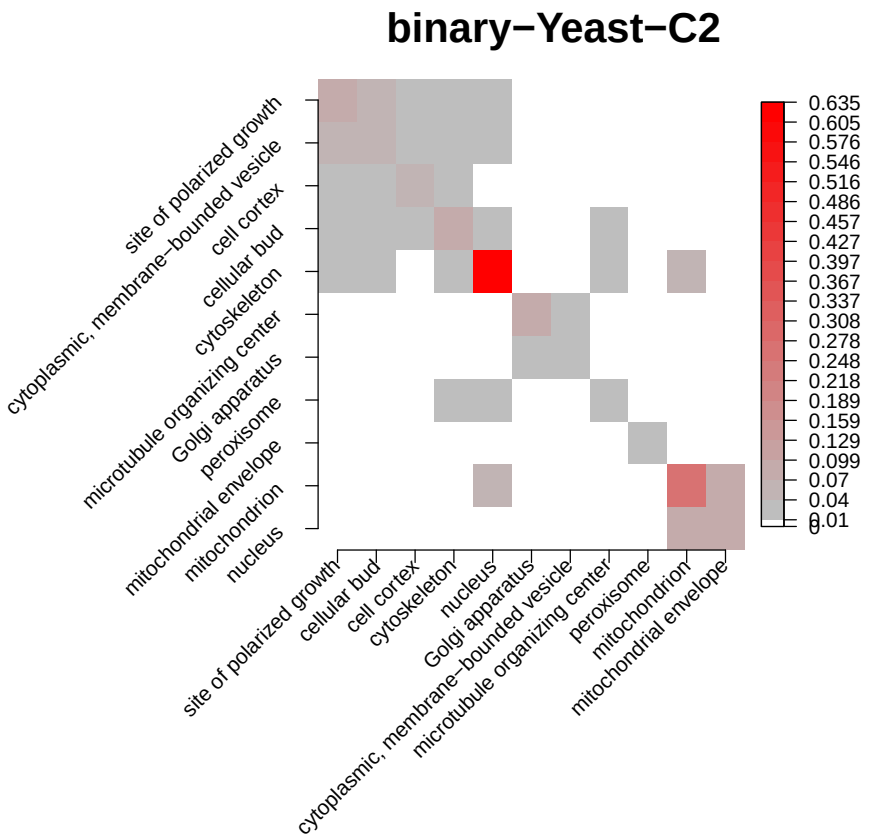


Figure 5.6: Proportion of proteins shared between each pair of cellular compartments used to form the binary-Yeast-C2 network. Compartments that 'shared' less than 1% of the total number of nodes in the binary-Yeast-C2 network are shown in white.

Our primary motivation regarding the application of this model is to test whether, in a favourable scenario where the node memberships are known, this type of model can perform well. Hence, if the node membership to $K = 11$ blocks (cellular compartment) is considered known, then the probabilities of observing a node in block k, α_k, can be estimated by scaling the diagonal proportions, shown in

Figure 5.6, such that their sum is equal to 1. Then as the node memberships are known, the DD parameters, $\eta_k = (p^k, q^k)$, can be obtained by extracting each cellular compartment network individually from the binary-Yeast-C2. However, as the binary-Yeast-C2 was formed by the individual cellular compartment networks used in Section 5.2.1, this procedure would lead to the same parameter estimates of Section 5.2.1. Figure 5.7, shows these parameters (which are extracted from Figure 5.3).

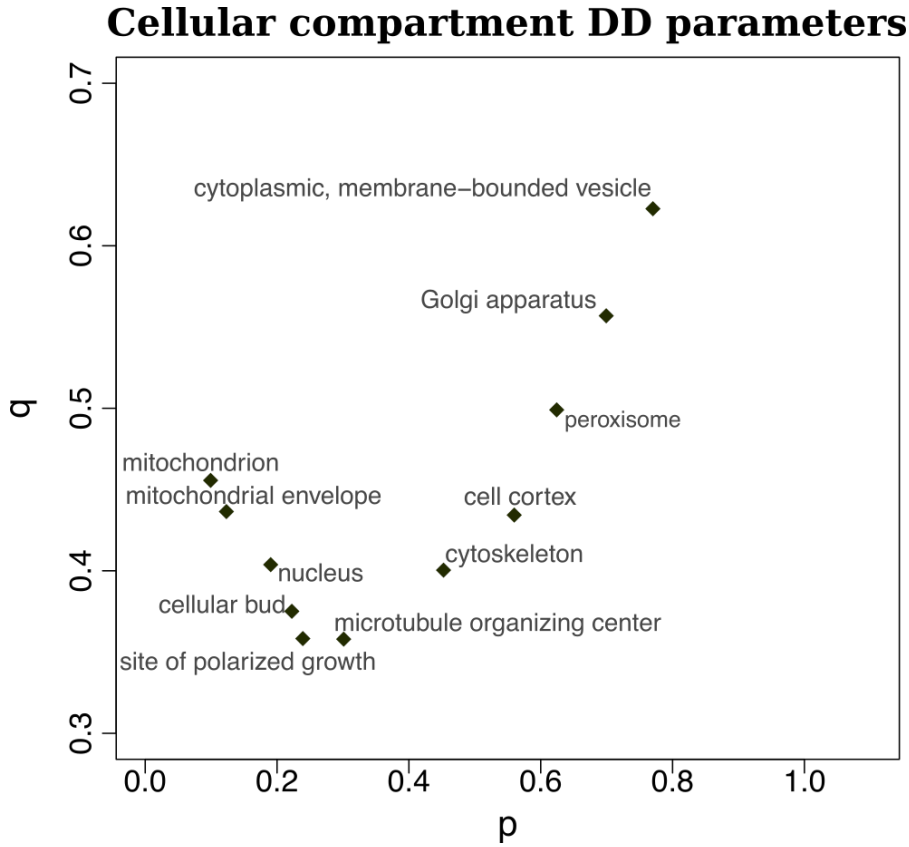


Figure 5.7: The duplication and divergence values $\hat{\eta}_k = (p^k, q^k)$, $k = 1, 2, ..., 11$ used in the DD-block model. Values obtained from Figure 5.7.

Regarding the parameters τ_{kl}, $k, l = 1, ..., 11$, $k \neq l$, we explored three empirical proposals to estimate τ_{kl}, which are based on the probabilities used by the Chung-Lu model and the degree corrected block model. These proposals were:

$$\text{Chung-Lu proposal:} \qquad \hat{\tau}_{kl} = \frac{1}{\sqrt{m_{kk} m_{ll}}},$$

where m_{kl} is the number of edges between nodes in block C_k and nodes in block C_l

observed in the binary-Yeast-C2 network. This proposal is based on the probability $d_i^* d_j^* / 2\sqrt{n_e n_e}$, which is the probability to connect two nodes in the Chung-Lu model. n_e represents the number of edges in the whole network and, d_i^*, d_j^* the degrees of nodes i and j across the whole binary-Yeast-C2 network.

$$\text{Degree corrected ERMG proposal (1):} \qquad \hat{\tau}_{kl} = \frac{m_{kl}}{\sum_{i \in C_k} d_i^* \sum_{j \in C_l} d_j^*},$$

where d_i^* is the number of edges going out of node i to all other nodes in the whole binary-Yeast-C2 network, and m_{kl} is the number of edges between nodes in block C_k and nodes in block C_l in the network. This variation is based on the probability $\frac{d_i^* d_j^* m_{kl}}{\sum_{i \in C_k} d_i^* \sum_{j \in C_l} d_j^*}$ given in (Karrer and Newman, 2011) for the degree corrected stochastic block model. However, as $\sum_{i \in C_k} d_i^*$ considers edges to nodes outside block C_k, we also used the following variation:

$$\text{Degree corrected ERMG proposal (2):} \qquad \hat{\tau}_{kl} = \frac{m_{kl}}{m_{kk} m_{ll}},$$

where m_{ll} provides the number of edges from nodes in block C_l to other nodes in block C_l in the binary-Yeast-C2 network.

Figure 5.8 shows the estimated values from the three proposals for the parameter τ_{kl}, $k, l = 1, 2, ..., 11$.

Lastly, the parameter ρ is taken such that the expected number of edges of the resulting DD-block networks is equal to the observed number of edges of the binary-Yeast-C2 network.

Monte Carlo test

The DD-block model explored in this chapter was a first attempt to explain the overall occurrence of small subgraphs across a whole PPI network based on our previous analysis of the global and local structure of PPI networks. Similar to what was done in Section 5.2.1, here we used the Monte Carlo test and the four network comparison statistics GDDA, GCD, Netdis and NetEmd to test whether the DD-block model is able to describe the occurrence of small subgraphs across the

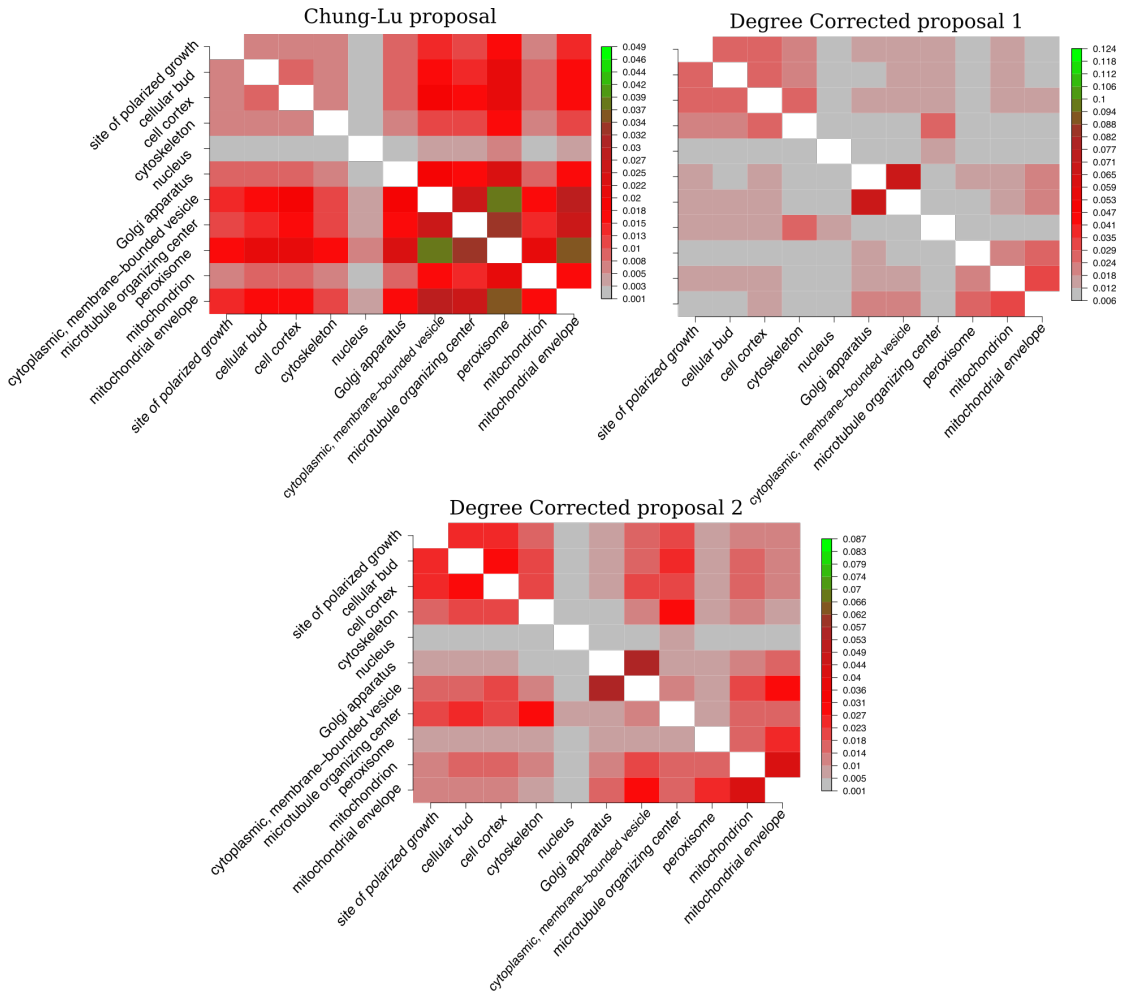


Figure 5.8: Representation of each of the three proposals considered of τ_{kl} $k, l = 1, 2, ..., 11$, estimated from the binary-Yeast-C2 network. We leave the diagonal blank as the edges between nodes of the same cellular compartment (block), are obtained via the DD model.

binary-Yeast-C2 network. Table 5.8 shows the results of the Monte Carlo test for each of the network comparison statistics and for each of the proposals considered in the DD-block model.

The results shown in Table 5.8 rejected the exploratory DD-block model as a model that can describe the occurrence of small connected subgraphs for at least 3 out of the 4 network comparison statistics. Although for the GDDA we obtained p-values larger than 0.05, we believed this result was not sufficient to provide any possible biological significance, as the DD-block model was tested under favourable conditions such as the knowledge of the node memberships to the different cellular compartments and the fact that the binary-Yeast-C2 was composed of nodes in the

	GDDA	GCD	NetEmd	Netdis
DD-block-Chung-Lu	0.22	0.01	0.01	0.01
DD-block-dc-ERMG-1	0.25	0.01	0.03	0.01
DD-block-dc-ERMG-2	0.29	0.01	0.02	0.01

Table 5.8: Monte Carlo p-values using the four network comparison statistics to test whether the proposed DD-block model is able to describe the occurrence of small subgraph in the binary-Yeast-C2 network.

cellular compartments for which the DD model achieved a consensus of at least 2 in Table 5.6.

5.3 Discussion

In this chapter we modelled the network structure of different cellular compartment networks. Following the results of Chapter 4 where we found that the DD model could potentially describe the occurrence of small subgraphs both globally and locally, here we expanded on those results and tested the ability of the DD model to describe the occurrence of small subgraphs in the binary and co-complex cellular compartment networks of Yeast and Human.

We found that the DD model was able to describe the subgraph occurrence of several binary cellular compartment networks. However, we observed some unexpected results from the DD parameter values used across all cellular compartment networks. Firstly, we found that, across all binary and co-complex cellular compartments of Yeast and Human, the divergence values (q) concentrated around a value close to 0.4 (Figure D.3). This result was also in agreement with previous parameter values estimated in Chapter 4 for the binary and co-complex networks of different organisms (Table 4.8). Secondly, the parameter values used for the duplication parameter (p) were widely spread across the interval [0,1] for most of the cases considered (Figure D.3). This, in combination with the previous results, suggests that the value of p alone might be able to capture possible structural differences between the cellular compartment networks.

In Section 5.2.2 we explored a random graph model that incorporated different features we observed in the ERMG and DD models from previous analyses of the

PPI networks. In this model we considered the whole PPI network as being a block structured network where each block was composed of an individual DD network. We considered each of these blocks as representing different cellular compartments that were loosely connected to each other by common rules observed in ERMG models (Karrer and Newman, 2011) and the Chung-Lu model (Chung and Lu, 2002). We tested this model via a Monte Carlo test under a favourable setting where the node memberships were known and where the parameters controlling the edges between blocks were directly obtained from the network considered. At least 3 out of four network comparison statistics rejected the DD-block model.

Therefore, we reject our proposed model. This model used a larger set of parameters than the standard DD model, which using only two parameters also obtained similar results from the Monte Carlo test. However, despite the inability of the DD-block model to describe the whole binary Yeast PPI network, we believe from our prior results that a model that incorporates a block structure, guided by the cellular compartments could provide relevant biological insight about the network structure and formation of functional modules.

In our application of the DD-block model, we used blocks associated only to individual cellular compartments. However, as proteins can be located in more than one cellular compartment, a further application of the DD-block model could be pursued. Instead of only using blocks associated to each individual cellular compartments, additional blocks could be added and associated to combinations of multiple cellular compartments. This new type of block structure would assign nodes with a single location to the individual cellular compartments blocks, and nodes with multiple locations to the corresponding block that jointly considers those locations.

The DD-block model explored in this section was a first attempt to describe the whole structure of PPI networks based on our previous findings for the cellular compartments. This model could be further improved by using different strategies to account for proteins with multiple location assignments, and simplified by using a unique divergence parameter q across all cellular compartments as suggested in Section 5.2.1.

Chapter 6

Conclusions

A common model that can explain and reliably reproduce the underlying general structure of all PPI networks would represent a valuable tool to address further biological questions about protein interactions, protein function and the evolutionary history of organisms (Ali et al., 2010, 2014). This model could be used as a quality control mechanism to detect erroneously reported interactions, to aid in phylogenetic reconstruction and to provide a baseline to detect under-represented or over-represented subgraphs. However, to this date such a model has not yet been obtained, despite the great effort of the scientific community in that direction (Ali et al., 2014; Rito et al., 2012). In this dissertation we addressed the problem of modelling PPI networks and assessing model fit. We approached the network modelling problem from the perspective of the networks local structure captured by the occurrence of small connected subgraphs.

Throughout this dissertation we focused on two topics. (1) Network comparison methods based on subgraph counts, and (2) the assessment of the ability of random graph models to describe the subgraph counts observed in PPI networks. We used and developed network comparison methods based on subgraph counts as different subgraphs have been proposed as possible building blocks of networks (Milo et al., 2002), because there is evidence that suggests that they may be biologically relevant, e.g. they appear in protein complexes (Pereira-Leal et al., 2007) and they might be evolutionary conserved across different organisms (Wuchty et al., 2003).

We studied three state-of-the-art network comparison methods based on subgraph

counts, GDDA, GCD and Netdis (Chapter 2). We found that although all three network comparison methods were based on the same inputs (subgraph counts), sometimes they reached different conclusions. In addition, although all methods aimed to detect networks with similar structure, they struggled at detecting similarities between some networks of different size or edge-density, even when these networks were generated from the same random graph model.

We observed that the network comparison methods GDDA, GCD and Netdis had difficulties at detecting that networks with different number of nodes or edges share the same network generation mechanism. Hence, we proposed a network comparison method, NetEmd (Chapter 3), that tackled this problem indirectly by proposing a method that is invariant to translations and rescalings of subgraph count distributions, and which was better able to detect similarities across networks with different number of nodes and edge-densities than the other state-of-the-art network comparison methods based on subgraph counts.

By means of the four network comparison statistics GDDA, GCD, Netdis and NetEmd we next tested whether the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse could be considered as a realisation of any of seven random graph models studied, some of which were previously suggested for PPI networks. The studied models were the ER model, the Configuration model, the Chung-Lu model, Goh's et al. power law model, the Geometric random graph model, the ERMG model and the DD model. Due to the different types of errors introduced by the different experimental methods used to detect protein interactions, we also analysed separately PPI networks which were formed by one of the two main types of interaction detection methods (De Las Rivas and Fontanillo, 2010; Keskin et al., 2016), namely binary methods and co-complex methods.

Our analysis of all binary and co-complex networks was split into two parts, a global fit (Section 4.2) where our interest lay in an overall fit to the whole PPI network, and a local fit (Section 4.3) where we assessed whether the global fit model is suited to model some local neighbourhoods of proteins in the PPI networks.

In the global fit analysis we found that among the seven random graph models considered, the only models that were 'not rejected' as possible null models for

several PPI networks were the ERMG model and the DD model. Specifically, some binary networks were considered as realisations of the DD model, while some co-complex networks, particularly the co-complex Yeast network, were considered as realisations of the ERMG model.

In the local fit analysis, where we examined wether groups of 2-step ego-networks, extracted from networks globally fitted to the PPI networks, could also fit groups of 2-step ego-networks of PPI networks. We found, similarly to the analysis of the global fit, that the DD and ERMG models were the only models that were not rejected as possible descriptors for several groups of PPI ego-networks across multiple organisms, and that these two models could fit ego-networks at complementarity regions of the edge-density, which suggested that whole PPI networks could be generated by two complementary network generation mechanisms (see Section 4.3).

From the local fit analysis we also found that other models that were previously linked to the structure of PPI networks, such as the geometric random graph model (Higham et al., 2008; Pržulj et al., 2004), were rejected as models for the occurrence of subgraph counts locally or globally. In particular we provided compelling evidence that suggested that the claim made by Hayes et al. (2013) about the Chung-Lu model being able to describe the PPI networks of several organisms does not hold true for any of the PPI networks of the six organism studied.

It should be noted that although neither ERMG nor DD models were rejected as possible descriptors of the local and global structure of some PPI networks, the number of parameters used for these two models is largely different. The number of parameters used in the ERMG model ranged from 28 parameters (Worm) to 1540 parameters (Human) (see Table C.6), while the DD model only required 2 parameters. The fact that the DD model with only two parameters was able to achieve a better or similar performance than models with a larger number of parameters such as the Chung-Lu model (for which the degree sequence is required), or the ERMG model, suggests that the DD model could potentially offer biologically relevant information. Hence, we further inspected the parameters of the DD model used for all PPI networks studied in the global fit analysis (Section 4.2), and found

that the parameters of the binary networks were concentrated in a specific region of the parameter space (see Figure 4.2). Based on this observation and the previous results of the global fit for the DD model, we used a combined DD parameter for all binary networks, a combined parameter for all co-complex networks and a combined parameter for all binary-&-cocomplex networks.

We found that for both global and local fits, the DD model that used the combined parameters obtained, overall, similar results as the DD model that used specific parameters for each of the different organisms. To test the robustness of our prior conclusions for the PPI networks to updates in the PPI data, we constructed PPI networks with protein interactions updated up to January 2017. We found that our previous conclusions for the global and local fit to the (October 2015) PPI networks held true for the updated networks (Section 4.4). This result pointed to a possible common global parameter that could explain the occurrence of subgraphs across the different organisms. We believe this result is encouraging as it supports the idea that there is a common network structure and network generation mechanism that relates the different organisms. It also suggests that there is still underlying biologically meaningful information that could be extracted from PPI networks (Hakes et al., 2008), as despite the different errors that we know are present in the data, we observed similar behaviours across the different organisms and across the different time stamps for PPI networks which accounted for the type of experiment (binary or co-complex) interactions were reported from.

As proteins are not present uniformly across the cell, and they tend to interact more frequently in different cellular locations (Tarassov et al., 2008), we studied the PPI networks of specific cellular compartments for the Yeast and Human organisms. We tested whether the different cellular compartment networks could be considered as realisations of the DD model and found that, for several cellular compartment networks, the DD model was not rejected as the possible network generation mechanism.

Contrary to the DD parameters we obtained for the whole binary PPI networks of the different organisms used in Chapter 4, the DD parameters for most cellular compartment networks, other than the Human binary networks, did not cluster

around any particular value (see Figures 5.3 and 5.4). This result suggested that although the overall structure of the binary PPI networks could be generated by a DD model (Section 4.2), the different cellular compartment networks could have slight variations in their structure which made them differ from the global structure and from the structure of other cellular compartments.

We found that the discrepancy between the global structure and the structure of the different cellular compartments can be captured alone by the DD parameter that accounts for the propensity of interaction between node duplicates p, as we found that the divergence parameter q appeared to concentrate around a value close to 0.4 for all binary and co-complex cellular compartment networks of Yeast and Human (see Figure D.3). In contrast, for all cases but the binary human cellular compartment networks, the values of the parameter p appeared to be widely spread across the interval $[0, 1]$.

The observed concentration of the DD parameter q for the cellular compartments, which also coincides with the combined parameter values of q used for the whole PPI networks (see Table 4.8), again suggested that a common constant proportion of duplicated interactions is lost (divergence) from duplicated proteins, regardless of the organism or the cellular location.

The results we obtained for the binary cellular compartment networks of Yeast, and the possible complementarity between the ERMG model and the DD model previously observed in Chapter 4, suggested to combine these models in order to obtain a model for the whole binary Yeast PPI network. Our proposed model consisted in generating DD networks for each of the cellular compartment networks and then, based on a generalisation of ERMG model, edges between the different cellular compartments were created. We tested this model under favourable conditions and we rejected it due to its poor performance. A key drawback of our model was that nodes were assumed to belong to exactly one cellular compartment, whereas proteins often belong to multiple compartments. Our model, which combines the information about cellular compartments and the generation mechanism of the DD model, could be improved by allowing nodes to be present in multiple cellular compartments. Different mechanisms could then be devised

for finding a consensus for the appearance of the same protein in multiple compartments. The model could be improve further by considering a more rigorous modelling framework. A Bayesian framework could be a particular good method, as it could for example reduce the number of parameters of the model by proposing a distribution over the DD parameters of the different cellular compartments. In addition it could follow a strategy similar to the one used for ERMG models (Mariadassou et al., 2010) or the more general degree corrected stochastic block models (Karrer and Newman, 2011), for the estimation of interaction between cellular compartments. Additionally, the modelling approach could also incorporate a model for the error. However, the selection of such model requires caution as common procedures used to decrease error rates, (in terms of false positives), can lead to changes in the resulting structure of the network that contradict the expected biological behaviour of the network (Hakes et al., 2008). Introduction of new and randomly allocated, deleted or rewired interactions is a commonly used error model, and although it is too simple to model the whole error process in PPI networks, which combines a mixture of scientific interest bias, experimental error and random noise; it can be used to test the robustness of the results obtained (e.g. Ali et al., 2014). Gold-standard datasets could also be used to propose error models, however, as mentioned by Hakes et al. (2008), a gold-standard or high-confidence dataset needs to be selected carefully, as it may portray a network structure that contradicts expected biological features (see Section 1.1). Thus, in order to increase confidence in biologically relevant results, it is important to replicate studies across multiple organisms and across different types of datasets that isolate different types of error or bias (e.g. datasets formed by the type of experiments used to detect interactions).

The results obtained throughout this dissertation have provided a framework where practical characteristics of network comparison statistics can be easily found. By applying this framework we found previously unknown differences between the network comparison statistics GDDA, GCD and Netdis. Due to the size and edge-density dependence of the results provided by GDDA, GCD and Netdis, we proposed NetEmd, which was better able to detect similarities across networks with

different number of nodes and edge-densities.

This dissertation provided an in-depth analysis of PPI networks by studying two major classes of experimental data. We found that the structure present in binary and co-complex networks largely differs from one another. We also provided evidence that suggested that the structure of the PPI networks of different organisms can be globally explained by a common generation mechanism with a common set of parameters. Furthermore, by considering PPI networks specific to different cellular compartments we found evidence for the hypothesis that there is a constant proportion of deletion of duplicated interactions that is common to all cellular compartments across different organisms.

Bibliography

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In Petrov, B. and Csaki, F., editors, *2nd International Symposium in Information theory*, pages 267–81. Akademia Kiado.

Aladağ, A. E. and Erten, C. (2013). Spinal: scalable protein interaction network alignment. *Bioinformatics*, 29(7):917.

Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47.

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P. (2002). *Molecular Biology of the cell*. Garland Science, fourth edition.

Ali, W., Deane, C. M., and Reinert, G. (2010). *Handbook of Statistical Systems Biology*, chapter Protein interaction networks and their statistical analysis. John Wiley & Sons.

Ali, W., Rito, T., Reinert, G., Sun, F., and Deane, C. M. (2014). Alignment-free protein interaction network comparison. *Bioinformatics*, 30:i430–i437.

Ali, W., Wegner, A. E., Gaunt, R. E., Deane, C. M., and Reinert, G. (2016). Comparison of large networks with sub-sampling strategies. *Scientific Reports*, 6.

Aliakbary, S., Motallebi, S., Rashidian, S., Habibi, J., and Movaghar, A. (2015). Distance metric learning for complex networks: Towards size-independent comparison of network structures. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 25(2):023111.

Alon, U. (2007). Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461.

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., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis,

A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29.

Asta, D. and Shalizi, C. R. (2014). Geometric network comparison. *arXiv preprint arXiv:1411.1350*.

Banerjee, A. and Jost, J. (2008). On the spectrum of the normalized graph laplacian. *Linear algebra and its applications*, 428(11-12):3015–3022.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.

Barabási, A.-L. and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113.

Barnett, I., Malik, N., Kuijjer, M. L., Mucha, P. J., and Onnela, J. (2016). Feature-based classification of networks. *CoRR*, abs/1610.05868.

Beal, M. J. (2003). *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London.

Berlingerio, M., Koutra, D., Eliassi-Rad, T., and Faloutsos, C. (2013). Network similarity via multiple social theories. In *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on*, pages 1439–1440.

Borgwardt, K. M., Kriegel, H.-P., Vishwanathan, S. V. N., and Schraudolph, N. N. (2007). Graph kernels for disease outcome prediction from protein-protein interaction networks. In *Pacific symposium on biocomputing*, volume 12, pages 4–15.

Borgwardt, K. M., Ong, C. S., Schönauer, S., Vishwanathan, S. V. N., Smola, A. J., and Kriegel, H.-P. (2005). Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl 1):i47–i56.

Brent, R. P. (1971). An algorithm with guaranteed convergence for finding a zero of a function. *The Computer Journal*, 14(4):422–425.

Chang, C, C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Cherry, J. M., Adler, C., Ball, C., Chervitz, S. A., Dwight, S. S., Hester, E. T., Jia, Y., Juvik, G., Roe, T., Schroeder, M., Weng, S., and Botstein, D. (1998). Sgd: *Saccharomyces* genome database. *Nucleic Acids Research*, 26(1):73.

Chipman, H., George, E. I., McCulloch, R. E., Clyde, M., Foster, D. P., and Stine, R. A. (2001). The practical implementation of bayesian model selection. *Lecture Notes-Monograph Series*, pages 65–134.

Chung, F. and Lu, L. (2002). Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics*, 6(2):125–145.

Chung, F. R. K. (1997). *Spectral graph theory*, volume 92. American Mathematical Society.

Clark, C. and Kalita, J. (2014). A comparison of algorithms for the pairwise alignment of biological networks. *Bioinformatics*, 30(16):2351–2359.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Crawford, J. and Milenković, T. (2015). GREAT: GRaphlet Edge-based network AlignmenT. In *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 220–227.

Das, J. and Yu, H. (2012). Hint: High-quality protein interactomes and their applications in understanding human disease. *BMC Systems Biology*, 6(1):1–12.

Daudin, J.-J., Picard, F., and Robin, S. (2008). A mixture model for random graphs. *Statistics and Computing*, 18(2):173–183.

Davison (2003). *Statistical Models*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, New york.

De Las Rivas, J. and Fontanillo, C. (2010). Protein–protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS Computational Biology*, 6(6):e1000807.

Deane, C. M., Salwiński, Ł., Xenarios, I., and Eisenberg, D. (2002). Protein interactions: Two methods for assessment of the reliability of high throughput observations. *Molecular & Cellular Proteomics*, 1(5):349–356.

Debnath, A. K., Lopez de Compadre, R. L., Debnath, G., Shusterman, A. J., and Hansch, C. (1991). Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34(2):786–797.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Didelot, X., Everitt, R. G., Johansen, A. M., and Lawson, D. J. (2011). Likelihood-free estimation of model evidence. *Bayesian Anal.*, 6(1):49–76.

Division, U. N. S. (2015). United nations commodity trade statistics database (un comtrade). http://comtrade.un.org/.

Dobson, P. D. and Doig, A. J. (2003). Distinguishing enzyme structures from non-enzymes without alignments. *Journal of Molecular Biology*, 330(4):771–783.

Drees, H., Haan, L. d., and Resnick, S. (2000). How to make a Hill plot. *The Annals of Statistics*, 28(1):254–274.

Dreze, M., Carvunis, A.-R., Charloteaux, B., Galli, M., Pevzner, S. J., Tasan, M., and Ahn (2011). Evidence for network evolution in an *Arabidopsis* interactome map. *Science*, 333(6042):601–607.

Dupuy, D., Bertin, N., Cusick, M. E., Han, J.-D. J., and Vidal, M. (2006). Reply to toward the complete interactome. *Nature Biotechnology*, 24(6):615–615.

El-Kebir, M., Heringa, J., and Klau, G. W. (2011). *Lagrangian Relaxation Applied to Sparse Global Network Alignment*, pages 225–236. Springer Berlin Heidelberg, Berlin, Heidelberg.

Erdös, P. and Rényi, A. (1960). On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:17–61.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recogn. Lett.*, 27(8):861–874.

Feenstra, R. C., Lipsey, R. E., Deng, H., Ma, A. C., and Mo, H. (2005). World trade flows: 1962-2000. Technical report, National Bureau of Economic Research.

Fossum, E., Friedel, C. C., Rajagopala, S. V., Titz, B., Baiker, A., Schmidt, T., Kraus, T., Stellberger, T., Rutenberg, C., Suthram, S., et al. (2009). Evolutionarily conserved herpesviral protein interaction networks. *PLoS Pathogens*, 5(9):e1000570.

Freedman, D. and Diaconis, P. (1981). On the histogram as a density estimator: L_2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 57(4):453–476.

Friedel, C. C. and Zimmer, R. (2006). Toward the complete interactome. *Nature Biotechnology*, 24(6):614–615.

Gibson, T. A. and Goldberg, D. S. (2011). Improving evolutionary models of protein interaction networks. *Bioinformatics*, 27(3):376–382.

Gilbert, E. N. (1959). Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144.

Gilbert, E. N. (1961). Random plane networks. *Journal of the Society for Industrial & Applied Mathematics*, 9(4):533–543.

Goh, K.-I., Kahng, B., and Kim, D. (2001). Universal behavior of load distribution in scale-free networks. *Physical Review Letters*, 87(27):278701.

Gu, J., Jost, J., Liu, S., and Stadler, P. F. (2016). Spectral classes of regular, random, and empirical graphs. *Linear algebra and its applications*, 489:30–49.

Guruharsha, K., Rual, J.-F., Zhai, B., Mintseris, J., Vaidya, P., Vaidya, N., Beekman, C., Wong, C., Rhee, D. Y., Cenaj, O., et al. (2011). A protein complex network of *Drosophila melanogaster*. *Cell*, 147(3):690–703.

Haasdonk, B. (2005). Feature space interpretation of svms with indefinite kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):482–492.

Hakes, L., Pinney, J. W., Robertson, D. L., and Lovell, S. C. (2008). Protein-protein interaction networks and biology[mdash]what's the connection? *Nat Biotech*, 26(1):69–72.

Harrington, H. A., Feliu, E., Wiuf, C., and Stumpf, M. P. (2013). Cellular compartments cause multistability and allow cells to process more information. *Biophysical Journal*, 104(8):1824–1831.

Hart, G. T., Ramani, A. K., and Marcotte, E. M. (2006). How complete are current yeast and human protein-interaction networks. *Genome Biology*, 7(11):120.

Hartwell, L. H., Hopfield, J. J., Leibler, S., and Murray, A. W. (1999). From molecular to modular cell biology. *Nature*, 402(c47-c52).

Hashemifar, S. and Xu, J. (2014). HubAlign: an accurate and efficient method for global alignment of protein–protein interaction networks. *Bioinformatics*, 30(17):i438–i444.

Hayes, W., Sun, K., and Pržulj, N. (2013). Graphlet-based measures are suitable for biological network comparison. *Bioinformatics*, 29(4):483–491.

Hermjakob, H., Montecchi-Palazzi, L., Lewington, C., Mudali, S., Kerrien, S., Orchard, S., Vingron, M., Roechert, B., Roepstorff, P., Valencia, A., Margalit, H., Armstrong, J., Bairoch, A., Cesareni, G., Sherman, D., and Apweiler, R. (2004). Intact: an open source molecular interaction database. *Nucleic Acids Research*, 32(Database issue):D452–D455.

Higham, D. J., Rašajski, M., and Pržulj, N. (2008). Fitting a geometric graph to a protein–protein interaction network. *Bioinformatics*, 24(8):1093–1099.

Higueruelo, A. P., Jubb, H., and Blundell, T. L. (2013). Protein–protein interactions as druggable targets: recent technological advances. *Current Opinion in Pharmacology*, 13(5):791–796.

Hill, B. M. (1975). A simple general approach to inference about the tail of a distribution. *The Annals of Statistics*, 3(5):1163–1174.

Hočevar, T. and Demšar, J. (2014). A combinatorial approach to graphlet counting. *Bioinformatics*, pages 559–565.

Holland, P. W., Laskey, K. B., and Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137.

Hooten, M. B. and Hobbs, N. T. (2015). A guide to bayesian model selection for ecologists. *Ecological Monographs*, 85(1):3–28.

Huang, H. and Bader, J. S. (2009). Precision and recall estimates for two-hybrid screens. *Bioinformatics*, 25(3):372–378.

Ispolatov, I., Krapivsky, P., and Yuryev, A. (2005). Duplication-divergence model of protein interaction network. *Physical Review E*, 71(6):061911.

Janjič, V., Sharan, R., and Pržulj, N. (2014). Modelling the yeast interactome. *Scientific Reports*, 4(4273):10.1038/srep04273.

Janowski, S. J., Kaltschmidt, B., and Kaltschmidt, C. (2014). Biological network modeling and analysis: Towards the virtual cell. In *Approaches in Integrative Bioinformatics*, chapter 8, pages 203–244. Springer, Berlin Heidelberg.

Jayasumana, S., Hartley, R., Salzmann, M., Li, H., and Harandi, M. (2015). Kernel methods on riemannian manifolds with gaussian rbf kernels. *IEEE transactions on pattern analysis and machine intelligence*, 37(12):2464–2477.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., and Barabási, A.-L. (2000). The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654.

Jordan, M., Ghahramani, Z., Jaakkola, T., and Saul, L. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Jure, L. and Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`.

Karrer, B. and Newman, M. E. J. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E*, 83:016107.

Keshava Prasad, T. S., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., Telikicherla, D., Raju, R., Shafreen, B., Venugopal, A., Balakrishnan, L., Marimuthu, A., Banerjee, S., Somanathan, D. S., Sebastian, A., Rani, S., Ray, S., Harrys Kishore, C. J., Kanth, S., Ahmed, M., Kashyap, M. K., Mohmood, R., Ramachandra, Y. L., Krishna, V., Rahiman, B. A., Mohan, S., Ranganathan, P., Ramabadran, S., Chaerkady, R., and Pandey, A. (2009). Human protein reference database—2009 update. *Nucleic Acids Research*, 37(suppl 1):D767–D772.

Keskin, O., Tuncbag, N., and Gursoy, A. (2016). Predicting protein-protein interactions from the molecular to the proteome level. *Chemical Reviews*, 116(8):4884–4909. PMID: 27074302.

Kolaczyk, E. (2009). *Statistical Analysis of Network Data: Methods and Models*. Mathematics and Statistics. Springer, New York.

Kuchaiev, O. and Pržulj, N. (2011). Integrative network alignment reveals large regions of global network similarity in yeast and human. *Bioinformatics*, 27(Sep):2539–2561.

Kuchaiev, O., Stevanović, A., Hayes, W., and Pržulj, N. (2011). Graphcrunch 2: Software tool for network modeling, alignment and clustering. *BMC Bioinformatics*, 12(1):24.

Latouche, P., Birmelé, E., and Ambroise, C. (2012). Variational bayesian inference and complexity control for stochastic block models. *Statistical Modelling*, 12(1):93–115.

Leskovec, J., Kleinberg, J., and Faloutsos, C. (2005). Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pages 177–187. ACM.

Lewis, A., Jones, N., Porter, M., and Deane, C. M. (2010). The function of communities in protein interaction networks at multiple scales. *BMC Systems Biology*, 4(1):100.

Lewis, A., Jones, N., Porter, M., and Deane, C. M. (2012). What evidence is there for the homology of protein-protein interactions? *PLoS Computational Biology*, 8(9):e1002645.

Luss, R. and d'Aspremont, A. (2008). Support vector machine classification with indefinite kernels. In *Advances in Neural Information Processing Systems*, pages 953–960.

Malod-Dognin, N. and Pržulj, N. (2015). L-graal: Lagrangian graphlet-based network aligner. *Bioinformatics*.

Mamano, N. and Hayes, W. B. (2017). Sana: Simulated annealing far outperforms many other search algorithms for biological network alignment. *Bioinformatics*.

Mariadassou, M., Robin, S., and Vacher, C. (2010). Uncovering latent structure in valued graphs: A variational approach. *The Annals of Applied Statistics*, 4(2):pp. 715–742.

Maugis, P. G., Priebe, C. E., Olhede, S. C., and Wolfe, P. J. (2017). Statistical inference for network samples using subgraph counts. *ArXiv e-prints*.

Mehla, J., Caufield, J., Sakhawalkar, N., and Uetz, P. (2017). Chapter seventeen - a comparison of two-hybrid approaches for detecting protein-protein interactions. In Shukla, A. K., editor, *Proteomics in Biology, Part B*, volume 586 of *Methods in Enzymology*, pages 333 – 358. Academic Press.

Mengersen, K. L., Pudlo, P., and Robert, C. P. (2013). Bayesian computation via empirical likelihood. *Proceedings of the National Academy of Sciences*, 110(4):1321–1326.

Milo, R., Itzkovitz, S., Kashtan, N., Levitt, R., Shen-Orr, S., Ayzenshtat, I., Sheffer, M., and Alon, U. (2004). Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002). Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827.

Mohar, B., Alavi, Y., Chartrand, G., and Oellermann, O. R. (1991). The laplacian spectrum of graphs. *Graph theory, combinatorics, and applications*, 2(871-898):12.

Molloy, M. and Reed, B. (1995). A critical point for random graphs with a given degree sequence. *Random Structures & Algorithms*, 6(2-3):161–180.

Moreira, D. and Lopez-Garcia, P. (2009). Ten reasons to exclude viruses from the tree of life. *Nature Reviews Microbiology*, 7(4):306–311.

Newman, M. E. (2003). The structure and function of complex networks. *SIAM Review*, 45(2):167–256.

Newman, M. E. (2005). Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, 46(5):323–351.

Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74:036104.

Newman, M. E. J. (2010). *Networks: an introduction*. Oxford University Press, New York, NY, USA.

Neyshabur, B., Khadem, A., Hashemifar, S., and Arab, S. S. (2013). NETAL: a new graph-based method for global alignment of protein–protein interaction networks. *Bioinformatics*, 29(13):1654–1662.

Niepert, M., Ahmed, M., and Kutzkov, K. (2016). Learning convolutional neural networks for graphs. *CoRR*, abs/1605.05273.

Noh, H. J., Ponting, C. P., Boulding, H. C., Meader, S., Betancur, C., Buxbaum, J. D., Pinto, D., Marshall, C. R., Lionel, A. C., Scherer, S. W., and Webber, C. (2013). Network topologies and convergent aetiologies arising from deletions and duplications observed in individuals with autism. *PLOS Genetics*, 9(6):1–12.

Onnela, J.-P., Fenn, D. J., Reid, S., Porter, M. A., Mucha, P. J., Fricker, M. D., and Jones, N. S. (2012). Taxonomies of networks from community structure. *Physical Review E*, 86:036104.

Parrish, J. R., Yu, J., Liu, G., Hines, J. A., Chan, J. E., Mangiola, B. A., Zhang, H., Pacifico, S., Fotouhi, F., DiRita, V. J., et al. (2007). A proteome-wide protein interaction map for campylobacter jejuni. *Genome Biology*, 8(7):R130.

Patro, R. and Kingsford, C. (2012). Global network alignment using multiscale spectral signatures. *Bioinformatics*, 28(23):3105.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Penrose, M. (2003). *Random Geometric Graphs*. Oxford University Press.

Peregrín-Alvarez, J. M., Xiong, X., Su, C., and Parkinson, J. (2009). The modular organization of protein interactions in *Escherichia coli*. *PLoS Computational Biology*, 5(10):e1000523.

Pereira-Leal, J., Levy, E., Kamp, C., and Teichmann, S. (2007). Evolution of protein complexes by duplication of homomeric interactions. *Genome Biology*, 8(4):R51.

Peterson, G. J., Presse, S., Peterson, K. S., and Dill, K. A. (2012). Simulated evolution of protein-protein interaction networks with realistic topology. *PloS One*, 7(6):e39052.

Philip, J. (2007). *The probability distribution of the distance between two random points in a box*. KTH mathematics, Royal Institute of Technology.

Pinkert, S., Schultz, J., and Reichardt, J. (2010). Protein interaction networks?more than mere modules. *PLoS Comput Biol*, 6(1):1–13.

Porter, M. A., Onnela, J.-P., and Mucha, P. J. (2009). Communities in networks. *Notices of the AMS*, 56(9):1082–1097.

Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., and Feldman, M. W. (1999). Population growth of human y chromosomes: a study of y chromosome microsatellites. *Molecular Biology and Evolution*, 16(12):1791–1798.

Pržulj, N., Corneil, D. G., and Jurisica, I. (2004). Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515.

Pržulj, N. (2007). Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183.

Pržulj, N. and Higham, D. J. (2006). Modelling protein–protein interaction networks via a stickiness index. *Journal of the Royal Society Interface*, 3(10):711–716.

Przulj, N., Kuchaiev, O., Stevanovic, A., and Hayes, W. (2010). Geometric evolutionary dynamics of protein interaction networks. In *Pacific Symposium on Biocomputing*, pages 178–189.

Rajagopala, S. V., Sikorski, P., Kumar, A., Mosca, R., Vlasblom, J., Arnold, R., Franca-Koh, J., Pakala, S. B., Phanse, S., Ceol, A., Hauser, R., Siszler, G., Wuchty, S., Emili, A., Babu, M., Aloy, P., Pieper, R., and Uetz, P. (2014). The binary protein-protein interaction landscape of escherichia coli. *Nature Biotechnology*, 32(3):285–290.

Ratmann, O., Andrieu, C., Wiuf, C., and Richardson, S. (2009). Model criticism based on likelihood-free inference, with an application to protein network evolution. *Proceedings of the National Academy of Sciences*, 106(26):10576–10581.

Razick, S., Magklaras, G., and Donaldson, I. M. (2008). irefindex: A consolidated protein interaction database with provenance. *BMC Bioinformatics*, 9(1):405.

Rhodes, D. R., Tomlins, S. A., Varambally, S., Mahavisno, V., Barrette, T., Kalyana-Sundaram, S., Ghosh, D., Pandey, A., and Chinnaiyan, A. M. (2005). Probabilistic model of the human protein-protein interaction network. *Nature Biotechnology*, 23(8):951–959.

Rito, T., Deane, C. M., and Reinert, G. (2012). The importance of age and high degree, in protein-protein interaction networks. *Journal of Computational Biology*, 19(6):785–795.

Rito, T., Wang, Z., Deane, C. M., and Reinert, G. (2010). How threshold behaviour affects the use of subgraphs for network comparison. *Bioinformatics*, 26(18):i611–i617.

Rives, A. W. and Galitski, T. (2003). Modular organization of cellular networks. *Proceedings of the National Academy of Sciences*, 100(3):1128–1133.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer-Verlag New York.

Runber, Y., Tomasi, C., and Guibas, L. J. (1998). A metric for distributions with applications to image databases. In *In IEEE International Conference Computer Vision*, pages 59–66.

Sanfeliu, A. and Fu, K. S. (1983). A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(3):353–362.

Sarajlić, A., Janjić, V., Stojković, N., Radak, D., and Pržulj, N. (2013). Network topology reveals key cardiovascular disease genes. *PloS One*, 8(8):e71537.

Saraph, V. and Milenković, T. (2014). MAGNA: Maximizing Accuracy in Global Network Alignment: Maximizing accuracy in global network alignment. *Bioinformatics*, 30(20):2931–2940.

Sato, S., Shimoda, Y., Muraki, A., Kohara, M., Nakamura, Y., and Tabata, S. (2007). A large-scale protein–protein interaction analysis in *Synechocystis* sp. pcc6803. *DNA Research*, 14(5):207–216.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

Shao, M., Yang, Y., Guan, J., and Zhou, S. (2013). Choosing appropriate models for protein-protein interaction networks: a comparison study. *Briefings in Bioinformatics*, page 10.1093/bib/bbt014.

Shao, M., Zhou, S., and Guan, J. (2015). Revisiting topological properties and models of protein protein interaction networks from the perspective of dataset evolution. *Systems Biology, IET*, 9(4):113–119.

Shen-Orr, S. S., Milo, R., Mangan, S., and Alon, U. (2002). Network motifs in the transcriptional regulation network of escherichia coli. *Nature Genetics*, 31(1):64–68.

Shervashidze, N., Schweitzer, P., Leeuwen, E. J. v., Mehlhorn, K., and Borgwardt, K. M. (2011). Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(Sep):2539–2561.

Shervashidze, N., Vishwanathan, S. V. N., Petri, T., Mehlhorn, K., and Borgwardt, K. M. (2009). Efficient graphlet kernels for large graph comparison. In *AISTATS*, volume 5, pages 488–495.

Shimoda, Y., Shinpo, S., Kohara, M., Nakamura, Y., Tabata, S., and Sato, S. (2008). A large scale analysis of protein-protein interactions in the nitrogen-fixing bacterium mesorhizobium loti. *DNA Research*, 15(1):13–23.

Snel, B., Lehmann, G., Bork, P., and Huynen, M. A. (2000). String: a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene. *Nucleic Acids Research*, 28(18):3442–3444.

Stark, C., Breitkreutz, B.-J., Chatr-aryamontri, A., Boucher, L., Oughtred, R., Livstone, M. S., Nixon, J., Van Auken, K., Wang, X., Shi, X., Reguly, T., Rust, J. M., Winter, A., Dolinski, K., and Tyers, M. (2011). The biogrid interaction database: 2011 update. *Nucleic Acids Research*, 39:D698–D704.

Stark, C., Breitkreutz, B.-J., Reguly, T., Boucher, L., Breitkreutz, A., and Tyers, M. (2006). Biogrid: a general repository for interaction datasets. *Nucleic Acids Research*, 34:D535–D539.

Stumpf, M. P. H., Thorne, T., de Silva, E., Stewart, R., An, H. J., Lappe, M., and Wiuf, C. (2008). Estimating the size of the human interactome. *Proceedings of the National Academy of Sciences*, 105(19):6959–6964.

Sturges, H. A. (1926). The choice of a class interval. *Journal of the American Statistical Association*, 21(153):65–66.

Sugiyama, M. and Borgwardt, K. M. (2015). Halting in random walk kernels. In *Advances in neural information processing systems*, pages 1639–1647.

Sun, Y., Crawford, J., Tang, J., and Milenković, T. (2015). *Simultaneous Optimization of both Node and Edge Conservation in Network Alignment via WAVE*, pages 16–39. Springer Berlin Heidelberg, Berlin, Heidelberg.

Tarassov, K., Messier, V., Landry, C. R., Radinovic, S., Molina, M. M. S., Shames, I., Malitskaya, Y., Vogel, J., Bussey, H., and Michnick, S. W. (2008). An in vivo map of the yeast protein interactome. *Science*, 320(5882):1465–1470.

Thorne, T. and Stumpf, M. P. H. (2012). Graph spectral analysis of protein interaction network evolution. *Journal of The Royal Society Interface*, 9(75):2653–2666.

Thüne, M. (2013). *Eigenvalues of matrices and graphs*. PhD thesis, University of Leipzig.

Topirceanu, A., Udrescu, M., and Vladutiu, M. (2013). Network fidelity: A metric to quantify the similarity and realism of complex networks. In *Cloud and Green Computing (CGC), 2013 Third International Conference on*, pages 289–296.

Traud, A. L., Mucha, P. J., and Porter, M. A. (2012). Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165 – 4180.

Vázquez, A., Flammini, A., Maritan, A., and Vespignani, A. (2003). Modeling of protein interaction networks. *Complexus*, 1(1):38–44.

Venables, W. N. and Ripley, B. D. (2002). *Modern Aapplied Statistics with S*. Statistics and Computing. Springer, New York, fourth edition.

Venkatesan, K., Rual, J.-F., Vazquez, A., Stelzl, U., Lemmens, I., Hirozane-Kishikawa, T., Hao, T., Zenkner, M., Xin, X., Goh, K.-I., Yildirim, M. A., Simonis, N., Heinzmann, K., Gebreab, F., Sahalie, J. M., Cevik, S., Simon, C., de Smet, A.-S., Dann, E., Smolyar, A., Vinayagam, A., Yu, H., Szeto, D., Borick, H., Dricot, A., Klitgord, N., Murray, R. R., Lin, C., Lalowski, M., Timm, J., Rau, K., Boone, C., Braun, P., Cusick, M. E., Roth, F. P., Hill, D. E., Tavernier, J., Wanker, E. E., Barabási, A.-L., and Vidal, M. (2009). An empirical framework for binary interactome mapping. *Nat Meth*, 6(1):83–90.

Wale, N., Watson, I. A., and Karypis, G. (2008). Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowl. Inf. Syst.*, 14(3):347–375.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *Nature*, 393(6684):440–442.

Wegner, A. E. (2014). Subgraph covers: An information-theoretic approach to motif analysis in networks. *Physical Review X*, 4:041026.

Wegner, A. E., Ospina-Forero, L., Gaunt, R. E., Deane, C. M., and Reinert, G. (2017). Identifying networks with common organizational principles. *Journal of Complex networks, preprint arXiv:1704.00387*.

West, J., Widschwendter, M., and Teschendorff, A. E. (2013). Distinctive topology of age-associated epigenetic drift in the human interactome. *Proceedings of the National Academy of Sciences*, 110(35):14138–14143.

Wilson, R. C. and Zhu, P. (2008). A study of graph spectra for comparing graphs and trees. *Pattern Recognition*, 41(9):2833–2841.

Winterbach, W., Van Mieghem, P., Reinders, M., Wang, H., and de Ridder, D. (2013). Topology of molecular interaction networks. *BMC Systems Biology*, 7(1):90.

Wodak, S. J., Vlasblom, J., Turinsky, A. L., and Pu, S. (2013). Protein–protein interaction networks: the puzzling riches. *Current Opinion in Structural Biology*, 23(6):941–953.

Wuchty, S., Oltvai, Z. N., and Barabási, A.-L. (2003). Evolutionary conservation of motif constituents in the yeast protein interaction network. *Nature Genetics*, 35(2):176–179.

Wuchty, S. and Uetz, P. (2014). Protein-protein interaction networks of *E. coli* and *S. cerevisiae* are similar. *Science Reports*, 4.

Xenarios, I., Rice, D. W., Salwinski, L., Baron, M. K., Marcotte, E. M., and Eisenberg, D. (2000). Dip: the database of interacting proteins. *Nucleic Acids Research*, 28(1):289–291.

Xing, E. P., Ng, A. Y., Jordan, M. I., and Russell, S. (2002). Distance metric learning with application to clustering with side-information. In *NIPS*, volume 15, page 12.

Yanardag, P. and Vishwanathan, S. V. N. (2015). Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1365–1374. ACM.

Yaveroglu, O. N., Malod-Dognin, N., Davis, D., Levnajic, Z., Janjic, V., Karapandza, R., Stojmirovic, A., and Przulj, N. (2014). Revealing the hidden language of complex networks. *Scientific Reports*, 4.

Zoraghi, R. and Reiner, N. E. (2013). Protein interaction networks as starting points to identify novel antimicrobial drug targets. *Current Opinion in Microbiology*, 16(5):566 – 572.

Appendix A

Supplementary background information

A.1 Estimation of the power law exponents

To estimate the power law exponent α of the degree distribution two known methods can be used. One is a least squares regression, based on a linear relation between the log-probabilities of the model and the logarithms of the degrees, known as log-log plot. The other method is based on Hill plots and the Hill estimator (Newman, 2005; Hill, 1975; Kolaczyk, 2009, pp. 81-85).

Hill plots and log-log plots are often described in terms of a sample of i.i.d. random variables $X_1, X_2, ..., X_n$ from a Pareto distribution. X is said to follow a Pareto distribution if its p.d.f. is

$$f(x) = s\frac{l^s}{x^{s+1}} = sl^s x^{-(s+1)}, \qquad \text{for } x \in [l, \infty),$$

where the parameters l, s are positive and are known as the *location* and *shape* parameters, respectively. To state the Pareto in terms of power law exponents we take $\alpha = 1 + s$, which is only a reparametrization of the density function.

The tails of power law distributions behave in the same way as the tails of Pareto distributions, hence when we concentrate on the tails of the distribution the Pareto assumption is reasonable.

In this dissertation we initially considered both, a linear regression fit of $\log(P(X > x))$ to estimate α and Hill-plots. The two methods are described next.

A.1.1 Simple linear regression and log-log plots

Consider a random variable X (either continuous or discrete) with density function (or probability function) $f(x) = Cx^{-\alpha}$, $\forall x \geq x_{min}$ and 0 otherwise, where $C, \alpha, x_{min} > 0$, then

$$f(x) = Cx^{-\alpha} \implies \log(f(x)) = \log(C) - \alpha \log(x).$$

Hence, $\log(f(x))$ behaves linearly with respect to $\log(x)$ with change rate α. The exponent α can be estimated by a linear regression fit to the points $\log(f(x))$ and $\log(x)$. If X is continuous an initial estimate of $f(x)$ can be obtained by a histogram; if X is discrete then $f(x)$ can be estimated by the relative frequencies. With this initial estimation of $f(x)$, α can then be estimated by the linear regression fit to the points $\log(\hat{f}(x))$ and $\log(x)$ (Newman, 2005). However, due to the nature of the distribution, random samples could have a small number of the values at the tail of the distribution, which could add noise to the estimation of $f(x)$ through the histogram and therefore to the estimation of α as well.

Newman (2005) reviewed different ways to reduce this noise and concluded that a better way to deal with the noise present is using tail probabilities and the c.d.f. $F(x)$, i.e. $\log(P(X > x)) = \log(1 - F(x))$ instead of $\log(f(x))$. Noting that $P(X > x)$ has the same functional form of $f(x)$, though with exponent $\alpha - 1$:

$$P(X > x) = \int_x^\infty t^{-\alpha} dt = Cx^{-(\alpha-1)}/(\alpha - 1).$$

A similar result holds for discrete random variables, for example variables following a Zipf law (Newman, 2005).

Figure A.1 shows a linear regression fit to a sample of 3740 i.i.d. random variables from a Pareto distribution with parameters: location $l = 1$ and shape $s = 2.8$, ($\alpha = 3.8$). The linear regression fit in plot (b) using tail probabilities gives a closer parameter estimate ($\alpha = 3.8$) than the linear regression fit in plot (a) ($\alpha = 3.2$).

As a rule of thumb, if the power law exponent is estimated for power law degree distributions, the range of the data should at least span three orders of magnitude

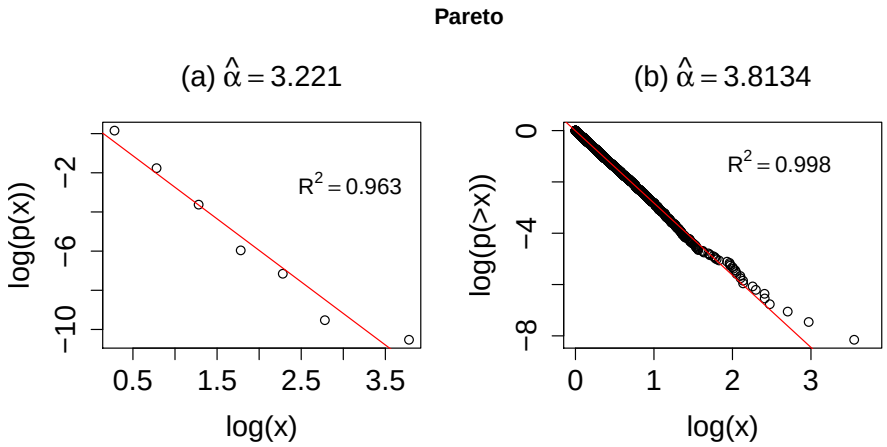


Figure A.1: (a-b) Log-log plots of a realisation of 3740 i.i.d. random variables from a Pareto distribution with parameters: location $l = 1$ and shape $s = 2.8$, ($\alpha = 3.8$). Plot (a) show a linear regression fit to $\log(\hat{f}(x))$ and plot (b) show a linear regression fit to $\log(\hat{P}(X > x))$. Plot (b) shows a clear linear behaviour using the cumulative empirical distribution; a parameter estimate of $\alpha = 3.8144$ was obtained. The coefficient of determination (R^2) of the simple linear regression fits (a-b) are 0.963 and 0.998 respectively.

and have a coefficient of determination of $R^2 \geq 0.99$ to obtain a 'good' estimation of α.

A.1.2 Hill plots

Hill plots (Drees et al., 2000; Kolaczyk, 2009, pp. 82-85) are based on the Hill estimator (Hill, 1975), which is an approximate maximum likelihood estimator of $\gamma = (\alpha - 1)^{-1}$ when X_i, $i = 1, 2, ..., n$ are considered i.i.d. variables from a distribution that satisfies

$$1 - F(x) \sim x^{-\alpha} L(x), \text{ when } x \to \infty, \tag{A.1}$$

with $L(x)$ being a slowly varying function, i.e. $L(x)$ such that $\lim_{t \to \infty} \frac{L(tx)}{L(t)} = 1$, $\forall x > 0$ (Drees et al., 2000; Newman, 2005).

The Hill estimator is

$$\hat{\alpha}_k = 1 + (H_{k,n})^{-1}, \quad \text{where} \quad H_{k,n} := \frac{1}{k} \sum_{i=n-k+1}^{n} \log(X_{(i)}/X_{(n-k)}), \quad k = 1, 2, ..., n-1,$$

and $X_{(1)} \leq X_{(2)} \leq ... \leq X_{(n)}$. If X_i, $i = 1, 2, ..n$ are i.i.d. random variables Pareto

distributed, then $H_{n-1,n}$ is the maximum likelihood estimator of α (Drees et al., 2000).

In practice k is selected as a value for which the plot of all points $(k, \hat{\alpha}_k)$, stabilises as k increases. The former plot is known as **Hill plot**. Figure A.2 shows an example of a Hill plot for a sample of 3740 i.i.d. random variables from a Pareto distribution. The Hill plot clearly stabilises as k increases.

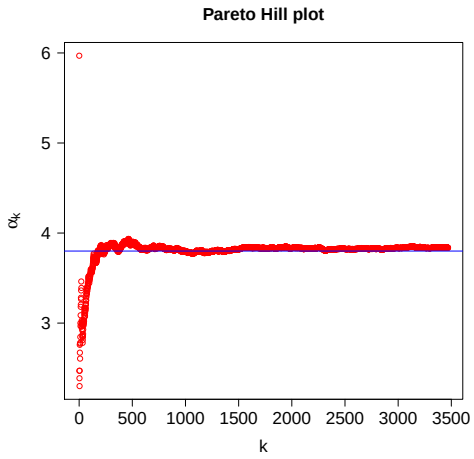


Figure A.2: Hill plot of a sample of 3740 i.i.d. random variables from a Pareto distribution with parameters: location $l = 1$ and shape $s = 2.8$, ($\alpha = 3.8$). The plot clearly stabilises around the true parameter (blue line).

A.1.3 Hill plots of protein-protein interaction networks

The degree distribution of some PPI networks has been observed to follow a power law distribution (Albert and Barabási, 2002). However, in general terms the degree distribution of PPI networks might not follow a power law distribution (Friedel and Zimmer, 2006; Dupuy et al., 2006; Pržulj and Higham, 2006; Kolaczyk, 2009, pp. 84-85).

In our studies we did not observe that the degree distributions of the PPI networks followed a power law distribution, as the Hill plots we obtained for most PPI networks did not show the values of α_k stabilising as the degree k increased (see Figure A.3). Hence, estimation of the power law exponent via Hill plots for these PPI networks could have been subjective, as there is no clear value of k that can be used to obtain the respective $\hat{\alpha}_k$. This type of behaviour suggested that the degree

distribution of these PPI networks might not follow a power law distribution.

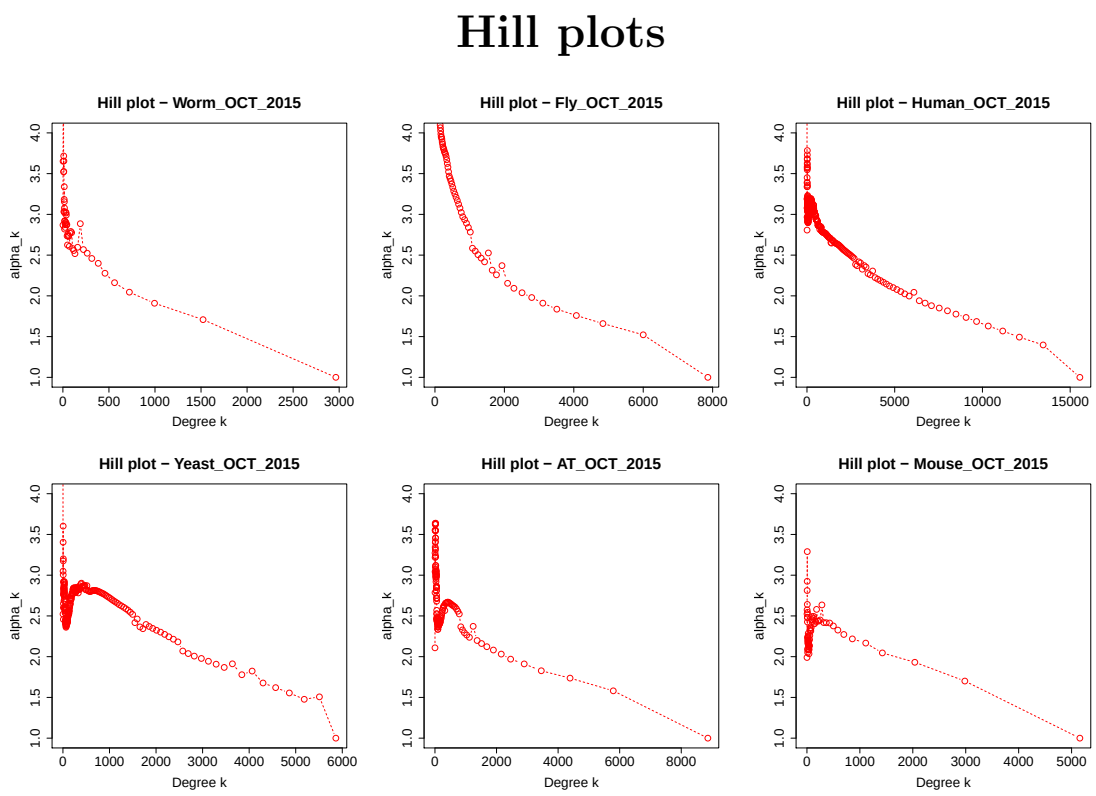


Figure A.3: Hill plots for the degree distributions of the Worm Fly, Human, Yeast, AT and Mouse PPI networks (BioGRID - OCT 2015). It can be seen that there is no clear value for which the Hill plot 'stabilises' as the degree increases, as it is shown in Figure A.2.

Appendix B

Netdis Scores

We showed in Section 2.4 that the variation of Netdis with background expectations led to unexpected results of the Monte Carlo tests shown in Figure 2.10. We showed in Figure 2.12 and Figure 2.13 that the unexpected behaviour was caused by the appearance of large Netdis values (close to 1), which suggested that the networks being compared had different structure, even when the networks were two realisations of the same model.

We conjectured that a pair of sparse networks generated with a larger average degree were more likely to contain ego-networks with subgraph counts at opposite sides of the expected number of subgraphs $E_w(.)$ than a pair of networks with smaller average degrees. This conjecture was based on the results of Figure 2.12 and on the fact that at lower average degrees, certain subgraphs are less likely to occur, but as the average degree increases the likelihood of occurrence of those subgraphs also increases. For example, in ER networks a 4-clique is less likely to occur in networks with small average degrees. Hence, as the average degree increases, a larger variation in the distribution of subgraph counts between networks can appear, thus providing networks with subgraph counts that have a number of subgraphs significantly smaller than expected, while another network has a number of subgraphs significantly larger than expected. For example, most ER networks with 1000 nodes and average degree 10 used in Figure 2.12 contained 0 4-clique subgraphs ($w = 8$, see Figure 2.1). However, ER networks with average degrees of 40 or 80 did contain 4-clique subgraphs. Figure B.1 shows such a scenario by

considering the distribution of the number of ego-networks that contain x 4-clique subgraphs ($w = 8$) in 10 different ER networks.

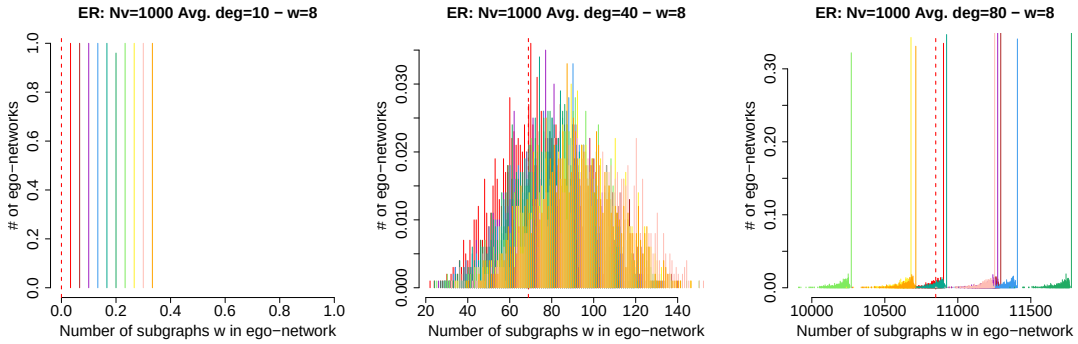


Figure B.1: Distribution of the number of ego-networks that contain x 4-clique subgraphs ($w = 8$) in an ER network with 1000 nodes and average degrees 10, 40 and 80. The distributions of 10 ER networks are plotted in different colours. For visualisation purposes the distributions for average degree 10 are placed over the x axis in intervals of length 0.03, as these distributions are mostly concentrated at $x = 0$.

Figure B.2 shows the respective $S_w(G) = \sum_i \left(N_{w,i}(G) - \binom{n_i}{k} E_w(\rho(i)) \right)$ values for each of the networks in Figure B.1. It can be seen that as the average degree increases the number of pairs of ER networks with S_w values with opposite sign also increases. A similar behaviour can be seen in some of the other subgraphs and networks of different size, see for example Figure B.3.

Note that, as discussed in Section 2.4, S_w values with opposite signs lead to larger Netdis values, as they provide a positive contribution to

$$Netdis(k) = 0.5 - \frac{1}{2 \times \sqrt{M(k)}} \sum_{w \in A(k)} \left(\frac{S_w(G) S_w(H)}{\sqrt{S_w(G)^2 + S_w(H)^2}} \right).$$

This behaviour is shown in Figure 2.12, for ER networks and again in Figure B.4 for Geometric 3D networks.

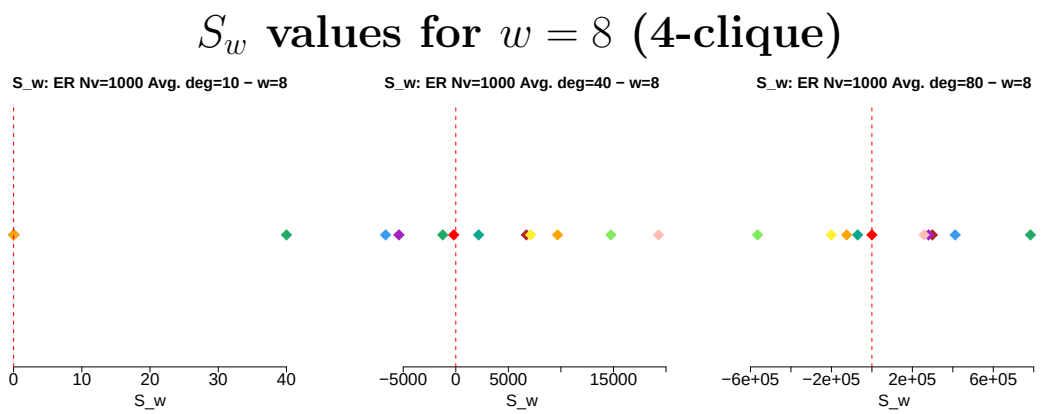


Figure B.2: S_w values for 4-clique subgraphs ($w = 8$) of 10 ER networks with 1000 nodes and average degrees 10, 40 and 80. It can be seen that as the average degree increases, the appearance of S_w values with opposite signs also increases.

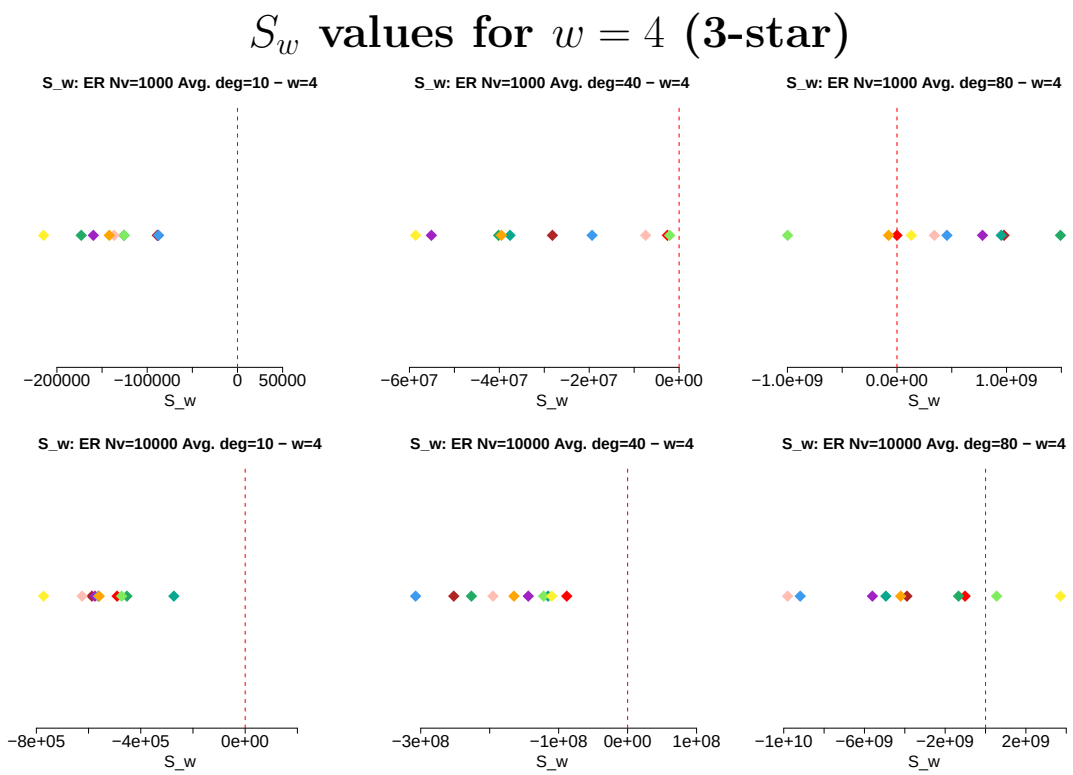


Figure B.3: S_w values for 3-star subgraphs ($w = 4$) of 10 ER networks with 1000 and 10000 nodes and average degrees 10, 40 and 80. It can be seen that as the average degree increases, the appearance of S_w values with opposite signs also increases.

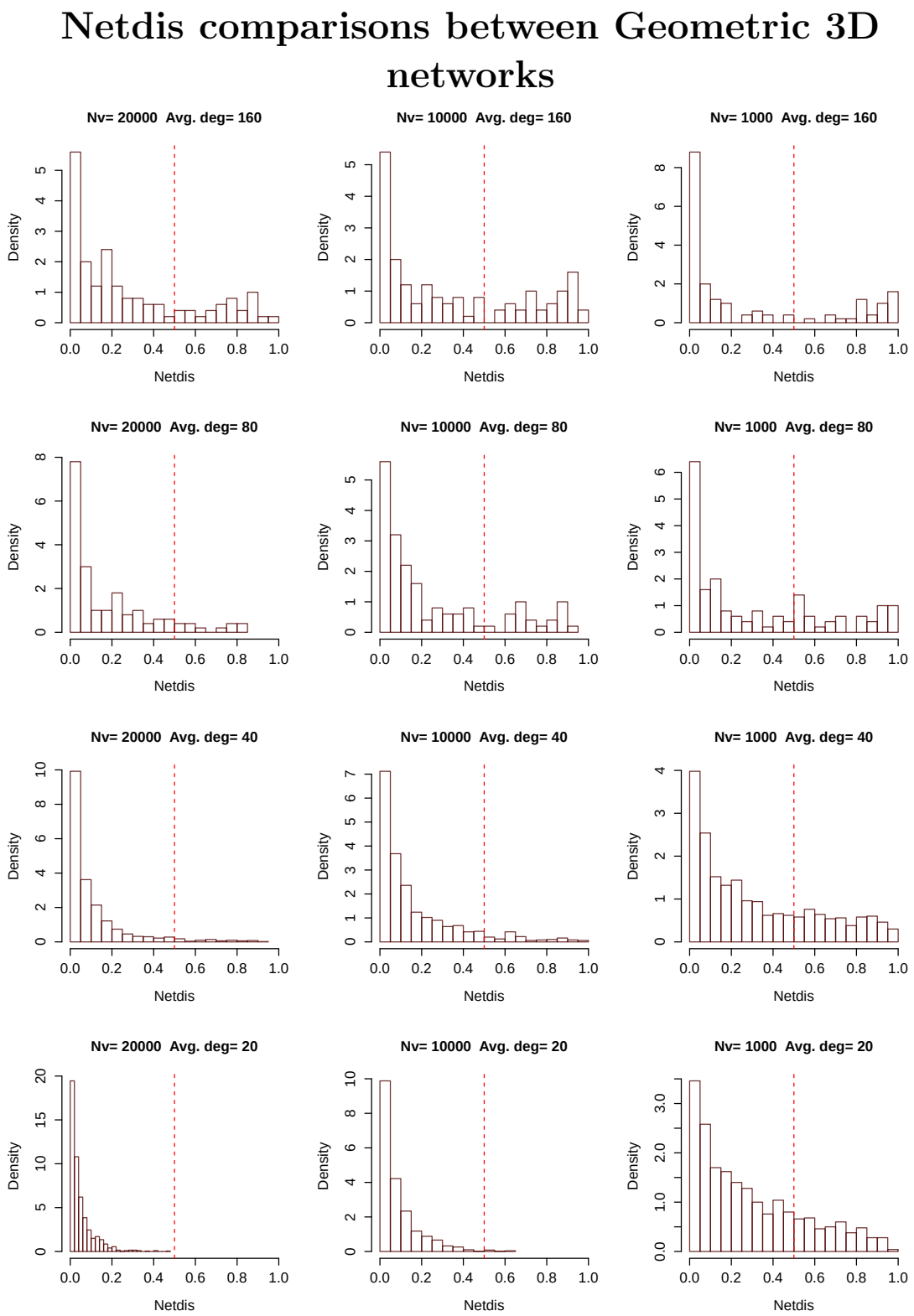


Figure B.4: Histograms of independent one-to-one Netdis comparisons for Geometric 3D networks with 1000, 10000 and 20000 nodes and with average degrees, $\bar{d}$, 20, 40, 80 and 160. It can be seen that as the edge-density increases, larger Netdis values occur more frequently.

Appendix C

Additional information to global and local fit of protein-protein interaction networks

C.1 Binary and co-complex protein-protein interaction data

Table 1.1 shows the classification of the different experimental methods used to discover the physical protein-protein interactions into either binary or co-complex (De Las Rivas and Fontanillo, 2010). Binary methods measure direct interactions between pairs of proteins in a one-two-one fashion, while co-complex methods measure direct and indirect interactions between groups of proteins, often in a one-to-many fashion (De Las Rivas and Fontanillo, 2010). The most used binary method is Yeast-Two-Hybrid and the most used co-complex method is Affinity Capture-MS. Binary and co-complex methods introduce particular types of errors and biases to the interaction data that they generate and these propagate to the resulting network of protein-protein interactions. Because of their nature, binary methods are less likely to report interactions of the type (A-B, B-C, A-C) than co-complex methods, since co-complex methods can report all possible interactions between a group of detected proteins, even when some of these interactions do not exist in real life.

A possible example of the implications of this type of error can be seen in the Fly

PPI network, where the interactions reported by a single study using co-complex methods lead to clear changes in the structure of the Fly PPI network. Table C.1 illustrates this case by showing the large discrepancy between the global clustering coefficient of the Fly PPI network when it considers all physical interactions (whole Fly network) and when it considers all interactions except the ones reported by Guruharsha et al. (2011) (Fly_without_Guruharsha).

	n_v	n_e	C	ρ	L	$Diam$	$\bar{d}$
Whole Fly	7862	36267	**0.1560**	0.001174	4.14	10	9.23
Fly_without_Guruharsha	7413	25784	**0.0168**	0.000939	4.26	11	6.96

Table C.1: Number of nodes (n_v), number of edges (n_e), global clustering coefficient (C), network density (ρ), average shortest path length of the largest connected component (L), diameter ($Diam$) and average degree ($\bar{d}$) of the Fly PPI network, considering all physical interactions (Whole Fly) and the Fly PPI network that consider all interactions except the ones reported by Guruharsha et al. (2011) (Fly_without_Guruharsha).

C.2 Tables of the global fit of random graph models to 2015 protein-protein interaction networks

The following tables show the Monte Carlo p-values obtained in the global fit analysis of Section 4.2, for the Erdös-Rényi model (ER), the Configuration model, the Chung-Lu model, Goh's power law model, the Geometric random graph model (Geo-3D), the Erdös-Rényi Mixture Graph model (ERMG) and the Duplication-Divergence model (DD) by Vázquez et al. (2003), along with a variation of the DD model where, similarly to (Gibson and Goldberg, 2011), an ER network on 100 nodes is given as a starting point for the network generation process (DD seed). Tables C.2, C.3, C.4 and C.5, show the Monte Carlo p-values obtained for the network comparison statistics GDDA, GCD, NetEmd and Netdis, respectively. It can be seen across all tables that only the ERMG model and the DD model obtained p-values larger than 0.05 for several PPI networks.

GDDA

	ER	Conf_mod	Chung-Lu	P_Law	GEO_3D	ERMG	DDseed	DD
Worm	0.01	0.01	0.01	0.01	0.01	0.09	0.18	0.31
Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Yeast	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.06
Human	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
AT	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Mouse	0.01	0.01	0.01	0.01	0.01	0.01	0.04	0.40
binary_Worm	0.01	0.01	0.01	0.01	0.01	0.04	0.18	0.17
binary_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.04
binary_Yeast	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.04
binary_Human	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02
binary_AT	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.03
binary_Mouse	0.01	0.03	0.03	0.01	0.01	0.01	0.12	0.46
cocomplex_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_Yeast	0.01	0.01	0.01	0.01	0.01	0.06	0.01	0.01
cocomplex_Human	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_AT	0.01	0.01	0.01	0.01	0.01	0.01	0.04	0.49
cocomplex_Mouse	0.01	0.01	0.01	0.01	0.01	0.02	0.02	0.24

Table C.2: *P*-values of Monte Carlo test (M=99, N=30) performed to test the global fit of the random graph models to the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse by means of the network comparison statistic **GDDA**. The test considered PPI networks based on interactions reported by binary methods (binary), co-complex methods (cocomplex) and, binary and co-complex methods. The minimum possible p-value in these tests is 0.01.

GCD

	ER	Conf_mod	Chung-Lu	P_Law	GEO_3D	ERMG	DDseed	DD
Worm	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02
Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Yeast	0.01	0.01	0.01	0.01	0.01	0.23	0.01	0.01
Human	0.01	0.01	0.01	0.01	0.01	0.13	0.01	0.01
AT	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.01
Mouse	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
binary_Worm	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.03
binary_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.11
binary_Yeast	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02
binary_Human	0.01	0.01	0.01	0.01	0.01	0.11	0.01	0.02
binary_AT	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02
binary_Mouse	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02
cocomplex_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_Yeast	0.01	0.01	0.01	0.01	0.01	0.49	0.01	0.01
cocomplex_Human	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_AT	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_Mouse	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Table C.3: *P*-values of Monte Carlo test (M=99, N=30) performed to test the global fit of the random graph models to the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse by means of the network comparison statistic **GCD**. The test considered PPI networks based on interactions reported by binary methods (binary), co-complex methods (cocomplex) and, binary and co-complex methods. The minimum possible p-value in these tests is 0.01.

NetEmd

	ER	Conf_mod	Chung-Lu	P_Law	GEO_3D	ERMG	DDseed	DD
Worm	0.01	0.01	0.01	0.01	0.01	0.05	0.01	0.20
Fly	0.01	0.01	0.01	0.01	0.01	0.07	0.01	0.01
Yeast	0.01	0.01	0.01	0.01	0.01	0.35	0.01	0.02
Human	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.01
AT	0.01	0.01	0.01	0.01	0.01	0.08	0.01	0.05
Mouse	0.01	0.01	0.01	0.01	0.01	0.25	0.01	0.12
binary_Worm	0.01	0.04	0.01	0.01	0.01	0.06	0.01	0.36
binary_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.29
binary_Yeast	0.01	0.01	0.01	0.01	0.01	0.09	0.01	0.11
binary_Human	0.01	0.01	0.01	0.01	0.01	0.45	0.01	0.15
binary_AT	0.01	0.01	0.01	0.01	0.01	0.04	0.01	0.04
binary_Mouse	0.01	0.01	0.01	0.01	0.01	0.01	0.03	0.47
cocomplex_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_Yeast	0.01	0.01	0.01	0.01	0.01	0.06	0.01	0.01
cocomplex_Human	0.01	0.01	0.01	0.01	0.01	0.05	0.01	0.01
cocomplex_AT	0.01	0.01	0.01	0.01	0.01	0.49	0.01	0.05
cocomplex_Mouse	0.01	0.01	0.01	0.01	0.01	0.27	0.01	0.09

Table C.4: *P*-values of Monte Carlo test (M=99, N=30) performed to test the global fit of the random graph models to the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse by means of the network comparison statistic **NetEmd**. The test considered PPI networks based on interactions reported by binary methods (binary), co-complex methods (cocomplex) and, binary and co-complex methods. The minimum possible p-value in these tests is 0.01.

Netdis

	ER	Conf_mod	Chung-Lu	P_Law	GEO_3D	ERMG	DDseed	DD
Worm	0.01	0.01	0.01	0.01	0.01	0.03	0.01	0.02
Fly	0.01	0.01	0.01	0.01	0.01	0.19	0.01	0.01
Yeast	0.01	0.01	0.01	0.01	0.01	0.24	0.01	0.01
Human	0.01	0.01	0.01	0.01	0.01	0.41	0.01	0.01
AT	0.01	0.01	0.01	0.01	0.01	0.35	0.01	0.01
Mouse	0.01	0.01	0.01	0.01	0.01	0.33	0.01	0.01
binary_Worm	0.01	0.01	0.01	0.01	0.01	0.10	0.01	0.02
binary_Fly	0.01	0.01	0.01	0.01	0.01	0.05	0.07	0.28
binary_Yeast	0.01	0.01	0.01	0.01	0.01	0.05	0.01	0.02
binary_Human	0.01	0.01	0.01	0.01	0.01	0.20	0.01	0.01
binary_AT	0.01	0.01	0.01	0.01	0.01	0.09	0.01	0.02
binary_Mouse	0.01	0.01	0.01	0.01	0.01	0.05	0.01	0.10
cocomplex_Fly	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
cocomplex_Yeast	0.01	0.01	0.01	0.01	0.01	0.42	0.01	0.01
cocomplex_Human	0.01	0.01	0.01	0.01	0.01	0.23	0.01	0.01
cocomplex_AT	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.01
cocomplex_Mouse	0.01	0.01	0.01	0.01	0.01	0.26	0.01	0.01

Table C.5: *P*-values of Monte Carlo test (M=99, N=30) performed to test the global fit of the random graph models to the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse by means of the network comparison statistic **Netdis**. The test considered PPI networks based on interactions reported by binary methods (binary), co-complex methods (cocomplex) and, binary and co-complex methods. The minimum possible p-value in these tests is 0.01.

C.3 Parameters for the ERMG model

The ERMG model, also known as stochastic block model, proposed by Holland et al. (1983), is based on a classification of nodes into Q classes and where the

interaction probabilities between nodes only depends on the node class membership. The ERMG model has several parameters: $\alpha = (\alpha_1, \alpha_2, ..., \alpha_Q)$ a vector of class membership probabilities, such that $\sum_k \alpha_k = 1$, and a symmetric matrix π of interaction probabilities between classes, or blocks, with entries π_{ql}, $q, l = 1, 2, ..., Q$. The total number of parameters in this model is then $1 + 2Q + \binom{Q}{2}$, as the number of blocks also has to be selected.

Table C.6 shows the number of blocks and number of parameters used in the ERMG models fitted to the different 2015 PPI networks.

	Number of blocks	Number of parameters
Worm_OCT_2015	6	28
Fly_OCT_2015	30	496
Yeast_OCT_2015	46	1128
Human_OCT_2015	54	1540
AT_OCT_2015	33	595
Mouse_OCT_2015	15	136
binary_Worm_OCT_2015	6	28
binary_Fly_OCT_2015	14	120
binary_Yeast_OCT_2015	26	378
binary_Human_OCT_2015	32	561
binary_AT_OCT_2015	31	528
binary_Mouse_OCT_2015	6	28
cocomplex_Fly_OCT_2015	25	351
cocomplex_Yeast_OCT_2015	43	990
cocomplex_Human_OCT_2015	54	1540
cocomplex_AT_OCT_2015	13	105
cocomplex_Mouse_OCT_2015	11	78

Table C.6: Number of blocks and parameters used in the ERMG models fitted to the 2015 PPI networks.

In this dissertation estimation of the ERMG parameters is based on the optimisation of the lower limit of the the likelihood $p(X, Z, \pi, \alpha)$

$$J(R_X, \pi, \alpha) = H(R_X) + \sum_Z R_X(Z) \log p(X, Z, \alpha, \pi),$$

with respect to $R_X(Z)$, π and α (see Section 1.4). The number of blocks or groups in the ERMG model is the number of blocks Q such that the Integrated Classification Likelihood (ICL) criterion is maximised. The ICL criterion for a model, m_Q, with Q blocks is (Mariadassou et al., 2010; Daudin et al., 2008)

$$ICL(Q) = p_{m_Q}(X, \tilde{Z}, \alpha, \pi) - \frac{1}{2}\{Q(Q+1)/2 \times \log[n_v(n_v - 1)] - (Q-1)\log[n_v]\},$$

where n_v is the number of nodes and $p_{m_Q}(X, Z, \alpha, \pi)$ is the complete data likelihood

of model m_Q, and where the unknown block memberships Z are replaced by the predicted node memberships $\tilde{Z}$, which can be obtained by the optimisation of $J(R_X, \pi, \alpha)$. For more details see (Mariadassou et al., 2010; Daudin et al., 2008).

C.4 Illustration of global model fit to the distribution of subgraph counts in 2015 PPI networks

In Section 4.2 (global fit to 2015 PPI networks) we tested whether different random graph models were able to describe the occurrence of subgraph counts in several PPI networks. Figure 4.1 and Table 4.2 showed the results of the Monte Carlo tests, where it can be seen that the ERMG model and the DD model were able to fit different PPI networks.

In this section we show an example of how the fit looks like when comparing the distributions of subgraph counts of networks generated by the ERMG model and the DD model to two PPI networks.

To explain the fit of the ERMG model and DD model to the subgraph counts of the PPI networks, we show in Figures C.1 and C.2 the cumulative distribution function (cdf) of the first 12 subgraph degree distributions, (similar to graphlet degree distributions described in Section 2.1.1), of the co-complex Yeast network and the binary Fly network, respectively. Figure 2.1 shows the form of each of these 12 subgraphs. Each figure shows the cdf of the PPI networks in black, and the cdfs of 10 networks generated by the the DD model (Figure C.1) and the ERMG model (Figure C.2) in red.

Both Figures C.1 and C.2 portray several features. For example, in most of the cdf plots, the 10 realisations from the random graph model do not perfectly overlap the cdf of the PPI network of the corresponding subgraph. This behaviour is not a problem, as our goal, rather than being able to replicate exactly the subgraph counts of a given PPI network, is to obtain a model that is able to produce subgraph occurrences that can be thought as proceeding from PPI networks.

Figures C.1 and C.2 also show that some subgraphs are not well described by the models. This is the case of subgraph-7, a diamond shaped subgraph (see Figure 2.1 for a plot of the different subgraphs). For this subgraph, the cdf of the binary Fly network looks like an upper limit for the cdfs obtained from the 10 realisations of the DD model in the range $[0, 50]$ of subgraph-7 degrees. This means that the DD model generates a higher number of nodes with larger subgraph-7 degrees than what is observed in the binary Fly network. In contrast, the cdf of the co-complex Yeast network seems like a lower limit for the cdfs obtained from the 10 realisations of the ERMG model in the range $[0, 30000]$, meaning that the co-complex Yeast network portrays more nodes with larger subgraph-7 degrees than what is seen in the networks generated from the ERMG model. Subgraph-8 (complete subgraph in four nodes), displays the behaviour of subgraph-7 for the co-complex Yeast network more clearly.

Some properties of the models themselves can also be observed from these plots. In the case of the DD model, the large variability in the networks generated by the model can be seen by the discrepancy among the cdfs of the different realisations of the DD model for some of the subgraphs, for example subgraph-11. On the other hand, the ERMG model does not portray such discrepancy for most subgraphs. However, this is possibly due to the large number of parameters involved, for example 990 parameters (43 blocks) for the co-complex Yeast network (Table C.6). In contrast, the DD model always considers 2 parameters. Despite the number of parameters that ERMG model uses, it seems not to over fit the data as it still portrays larger discrepancies between the cdfs among networks generated from the model and the observed PPI cdf. The cdfs of subgraph-4 in Figure C.2 illustrate this point.

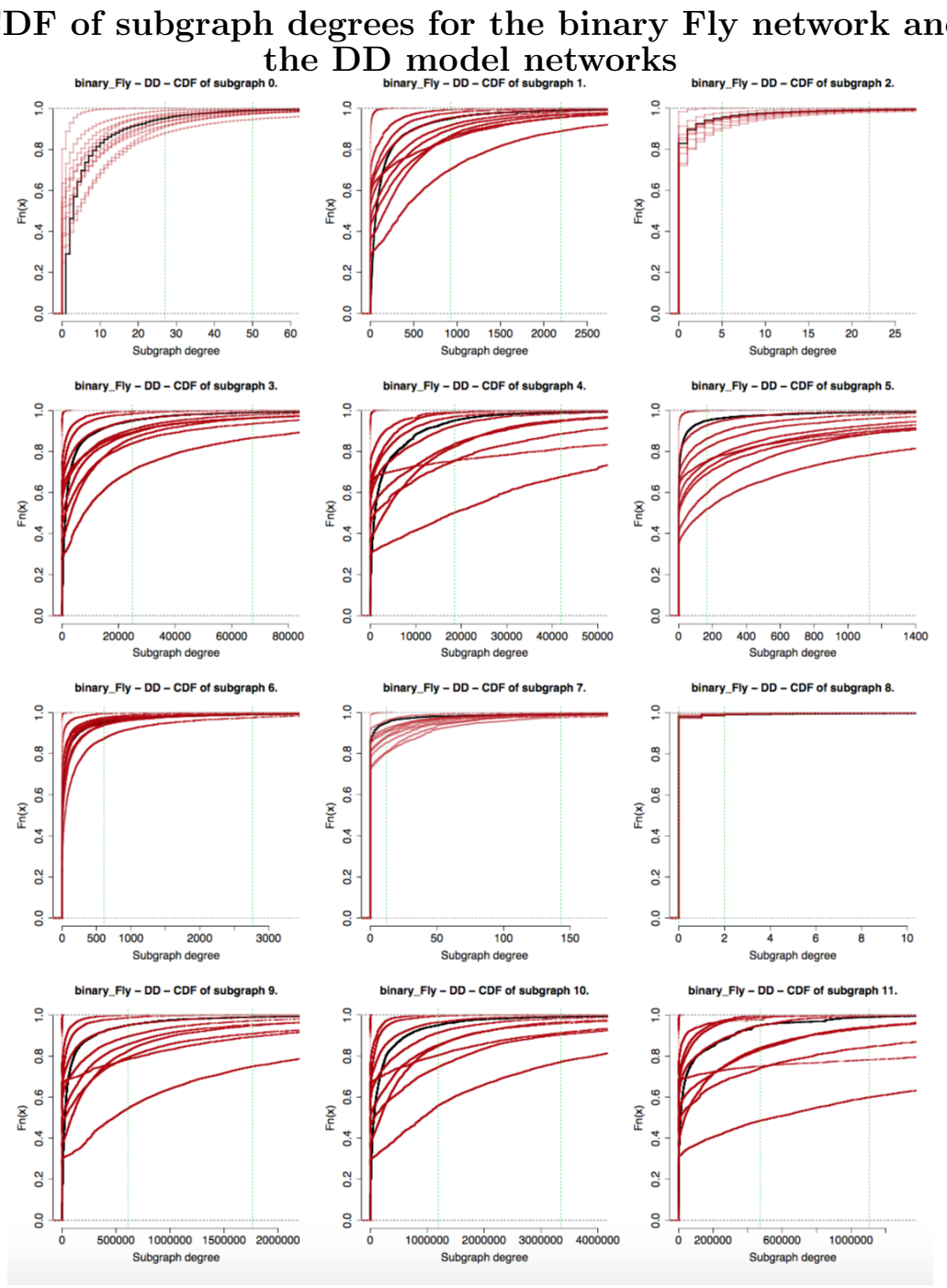


Figure C.1: Cumulative distribution function of subgraph degree distributions of subgraphs 0 to 11 for the binary Fly PPI network (black). Shown in red, the cumulative distribution function of 10 realisations of the DD model. Two green dashed vertical lines are added to each plot to mark the 95% and 99% quantiles of the respective subgraph degree distribution. Figure 2.1 illustrates the form of each subgraph.

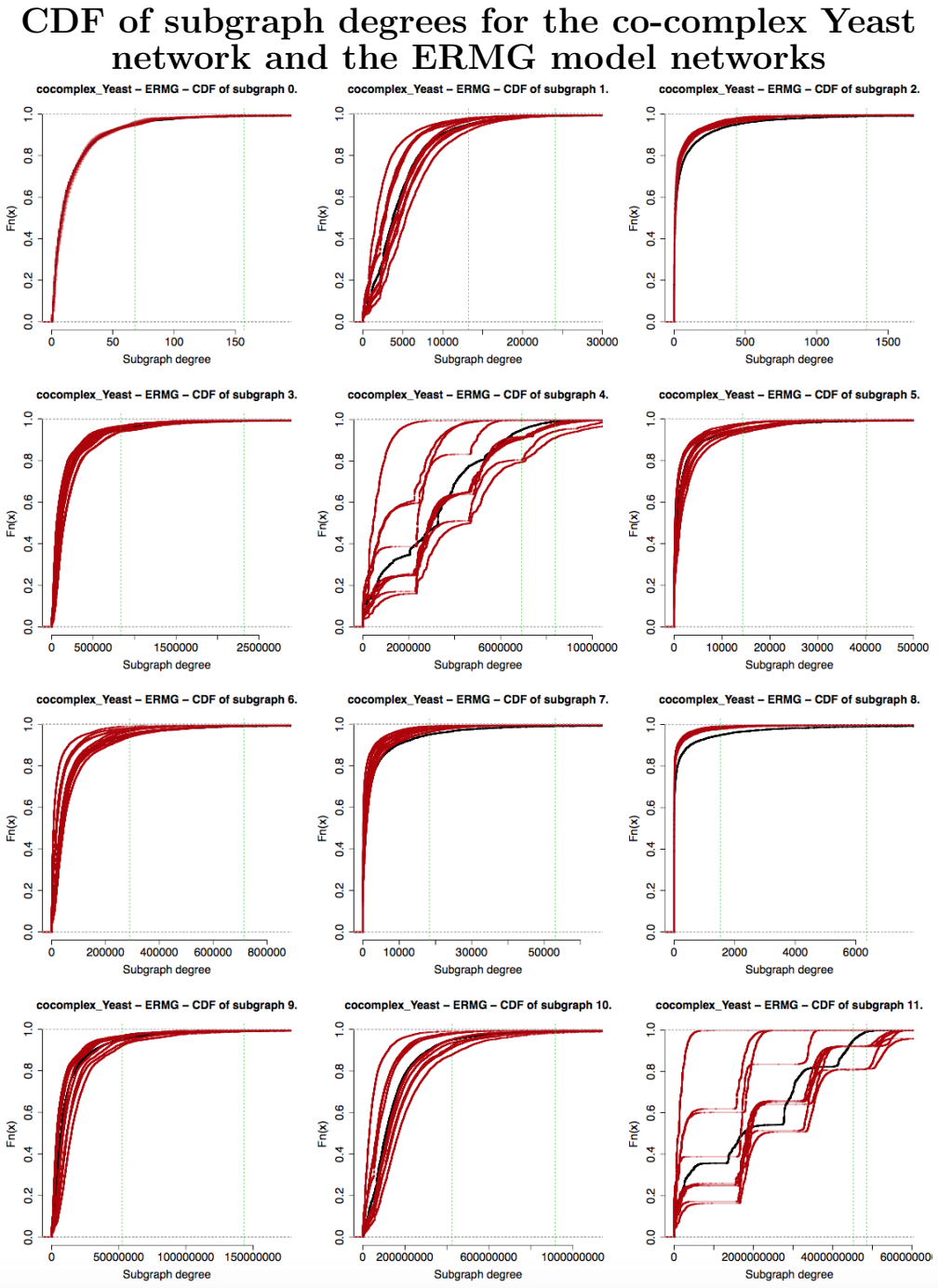


Figure C.2: Cumulative distribution function of subgraph degree distributions of subgraphs 0 to 11 for the the co-complex Yeast PPI network (black). Shown in red, the cumulative distribution function of 10 realisations of the ERMG model. Two green dashed vertical lines are added to each plot to mark the 95% and 99% quantiles of the respective subgraph degree distribution. Figure 2.1 illustrates the form of each subgraph.

C.5 Edge-density binning for PPI ego-networks

In Section 4.3 (local fit analysis for 2015 PPI networks), we extracted and grouped the 2-step ego-networks of Worm, Fly, Yeast, Human, AT and Mouse PPI networks. We grouped the ego-networks based on an edge-density binning formed by the quantiles 5%, 10%, 15%,...,90%, 95% and 100%, of the PPI ego-network edge-densities. We chose this type of binning as it provided a large quantity of groups (20) while still comprising a large enough sample size to perform a Monte Carlo test within each bin. Figure 4.7 showed the binning we obtained for all PPI networks evaluated. However, as there is a larger concentration of bins below 0.1, the plot in Figure 4.7 does not show a clear separation between the bin breaks. To avoid this visualisation problem Figure C.3 shows the binning in a logarithmic scale. It can be seen that most of the bins are concentrated between 0.01 and 0.1 and that the binning obtained for the binary networks tends to have breaks at slightly larger edge-densities than the ones obtained for the co-complex networks.

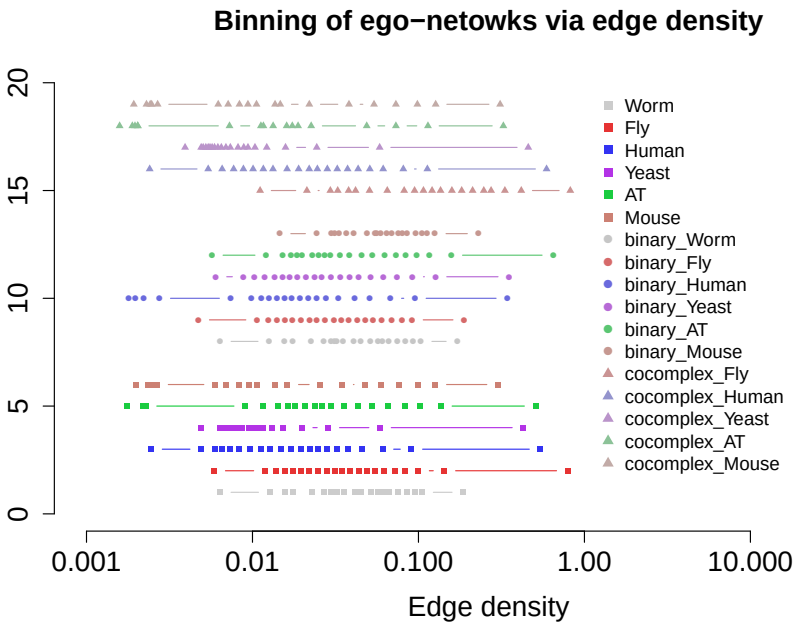


Figure C.3: Edge-density binning (in logarithmic scale) for the 2-step ego-networks of the PPI networks of Worm, Fly, Yeast, Human, AT and Mouse and their respective binary and co-complex networks. Most ego-networks across all six species have edge densities below 0.1 up to a minimum of 0.001583117. None of the binning considers equally spaced bins as the breaks are based on the quantiles of the ego-network edge-densities.

C.6 Protein-protein interactions networks obtained in 2017

In Section 4.4 (robustness to data updates), we tested whether the results of global and local fit shown in Section 4.2 and Section 4.3 only applied to the data obtained in October 2015, or if those results would also apply to the updated PPI networks of January 2017.

In Table C.7 we show the network summary statistics of all 2017 PPI networks used in Section 4.4.

In Tables C.8, C.9 and C.10 we show the Monte Carlo p-values and consensus across the four network comparison statistics obtained in Section 4.4 for the global fit of the ERMG model, the DD model and the Avg. DD model to these 2017 PPI networks. The Avg. DD model is the DD model that uses the average parameters from the 2015 PPI networks that achieved the largest consensus in Table 4.2.

Note that in all Monte Carlo tests performed for the 2017 data in Section 4.4, the model parameters used were the same parameter estimates that were found for the 2015 PPI networks.

Method	Organism	n_v	n_e	C	ρ	L	$Diam$	$\bar{d}$
Binary & Co-complex	Worm_JAN_2017	2968	5437	0.0204	0.001235	4.80	13	3.66
	Fly_JAN_2017	8015	37026	0.1465	0.001153	4.10	10	9.24
	Yeast_JAN_2017	5931	85586	0.0497	0.004867	2.50	6	28.86
	Human_JAN_2017	15971	216783	0.0615	0.001700	3.17	8	27.15
	AT_JAN_2017	9177	34461	0.0234	0.000818	4.28	12	7.51
	Mouse_JAN_2017	5430	13227	0.0120	0.000897	4.01	15	4.87
Binary	Worm_JAN_2017	2904	5255	0.0174	0.001247	4.78	13	3.62
	Fly_JAN_2017	7162	23646	0.0145	0.000922	4.33	12	6.60
	Yeast_JAN_2017	4717	26430	0.0625	0.002376	3.62	9	11.21
	Human_JAN_2017	12860	71099	0.0233	0.000860	3.61	10	11.06
	AT_JAN_2017	7287	28349	0.0331	0.001068	4.81	15	7.78
	Mouse_JAN_2017	1653	2621	0.0316	0.001920	5.99	16	3.17
Co-complex	Worm_JAN_2017	23	24	0.0800	0.094862	3.53	7	2.09
	Fly_JAN_2017	2942	13466	0.3175	0.003113	4.35	13	9.15
	Yeast_JAN_2017	5724	62971	0.0442	0.003845	2.50	6	22.00
	Human_JAN_2017	13731	158819	0.0713	0.001685	3.26	9	23.13
	AT_JAN_2017	3892	6969	0.0058	0.000920	3.89	12	3.58
	Mouse_JAN_2017	4798	10955	0.0108	0.000952	3.94	14	4.57

Table C.7: Number of nodes (n_v), number of edges (n_e), global clustering coefficient (C), network density (ρ), average shortest path length (L), diameter ($Diam$) and average degree ($\bar{d}$) of the largest connected component of the PPI networks of Worm, Fly, Human, Yeast, AT and Mouse downloaded from the BioGRID database on January 28^{th} 2017.

ERMG

	GDDA	GCD	NetEmd	Netdis	Consensus
Worm_JAN_2017	0.03	0.01	0.04	0.04	0
Fly_JAN_2017	0.01	0.01	0.02	0.01	0
Yeast_JAN_2017	0.01	0.15	0.40	0.35	3
Human_JAN_2017	0.01	0.38	0.01	0.39	2
AT_JAN_2017	0.01	0.03	0.22	0.28	2
Mouse_JAN_2017	0.02	0.01	0.38	0.39	2
binary_Worm_JAN_2017	0.02	0.01	0.04	0.03	0
binary_Fly_JAN_2017	0.01	0.01	0.01	0.04	0
binary_Yeast_JAN_2017	0.01	0.01	0.10	0.06	2
binary_Human_JAN_2017	0.01	0.45	0.30	0.28	3
binary_AT_JAN_2017	0.01	0.01	0.03	0.08	1
binary_Mouse_JAN_2017	0.01	0.01	0.01	0.03	0
cocomplex_Fly_JAN_2017	0.01	0.01	0.02	0.01	0
cocomplex_Yeast_JAN_2017	0.11	0.28	0.14	0.13	4
cocomplex_Human_JAN_2017	0.01	0.03	0.03	0.35	1
cocomplex_AT_JAN_2017	0.03	0.01	0.22	0.21	2
cocomplex_Mouse_JAN_2017	0.01	0.01	0.38	0.24	2

Table C.8: Monte Carlo test p-values (M=99, N=30) and consensus ($\alpha = 0.05$) across the four network comparison statistics to test whether the 2017 PPI networks can be considered as realisations of the ERMG model. The **ERMG** models used in the Monte Carlo test considered the same parameters that were obtained for the October 2015 PPI networks, see Section 4.2. The minimum possible p-value in these tests is 0.01.

DD

	GDDA	GCD	NetEmd	Netdis	Consensus
Worm_JAN_2017	0.50	0.01	0.30	0.03	2
Fly_JAN_2017	0.01	0.01	0.01	0.01	0
Yeast_JAN_2017	0.02	0.01	0.01	0.01	0
Human_JAN_2017	0.01	0.01	0.01	0.03	0
AT_JAN_2017	0.03	0.01	0.10	0.01	0
Mouse_JAN_2017	0.48	0.01	0.03	0.01	1
binary_Worm_JAN_2017	0.24	0.02	0.36	0.02	2
binary_Fly_JAN_2017	0.02	0.08	0.30	0.33	2
binary_Yeast_JAN_2017	0.02	0.02	0.03	0.03	0
binary_Human_JAN_2017	0.01	0.01	0.15	0.01	1
binary_AT_JAN_2017	0.03	0.01	0.07	0.06	0
binary_Mouse_JAN_2017	0.39	0.01	0.43	0.04	2
cocomplex_Fly_JAN_2017	0.01	0.01	0.01	0.01	0
cocomplex_Yeast_JAN_2017	0.01	0.01	0.01	0.01	0
cocomplex_Human_JAN_2017	0.01	0.01	0.01	0.01	0
cocomplex_AT_JAN_2017	0.46	0.01	0.03	0.01	1
cocomplex_Mouse_JAN_2017	0.38	0.01	0.06	0.01	1

Table C.9: Monte Carlo test p-values (M=99, N=30) and consensus ($\alpha = 0.10$) across the four network comparison statistics to test whether the 2017 PPI networks can be considered as realisations of the DD model. The **DD** models used in the Monte Carlo test considered the parameters that were obtained for the October 2015 PPI networks, see Section 4.2 and Figure 4.2. The minimum possible p-value in these tests is 0.01.

Avg. DD

	GDDA	GCD	NetEmd	Netdis	Consensus
Worm_JAN_2017	0.43	0.02	0.13	0.01	2
Fly_JAN_2017	0.04	0.02	0.10	0.08	0
Yeast_JAN_2017	0.01	0.01	0.01	0.04	0
Human_JAN_2017	0.01	0.01	0.03	0.26	1
AT_JAN_2017	0.08	0.01	0.13	0.01	1
Mouse_JAN_2017	0.34	0.01	0.06	0.01	1
binary_Worm_JAN_2017	0.32	0.01	0.32	0.02	2
binary_Fly_JAN_2017	0.01	0.08	0.35	0.33	2
binary_Yeast_JAN_2017	0.09	0.01	0.23	0.44	2
binary_Human_JAN_2017	0.04	0.02	0.09	0.01	0
binary_AT_JAN_2017	0.24	0.01	0.14	0.25	3
binary_Mouse_JAN_2017	0.47	0.01	0.46	0.09	2
cocomplex_Fly_JAN_2017	0.41	0.02	0.09	0.11	2
cocomplex_Yeast_JAN_2017	0.08	0.01	0.05	0.02	0
cocomplex_Human_JAN_2017	0.07	0.01	0.08	0.24	1
cocomplex_AT_JAN_2017	0.23	0.01	0.05	0.01	1
cocomplex_Mouse_JAN_2017	0.46	0.01	0.09	0.01	1

Table C.10: Monte Carlo test p-values (M=99, N=30) and consensus ($\alpha = 0.10$) across the four network comparison statistics for the **Avg. DD** model with parameters $p = 0.0360$ and $q = 0.4232$ for the 2017 binary-&-cocomplex networks, $p = 0.0544$ and $q = 0.4231$ for 2017 binary networks and $p = 0.0183$ and $q = 0.4168$ for 2017 co-complex networks. These parameters are the same parameters used for the 2015 networks, see Section 4.2 and Table 4.8. The minimum possible p-value in these tests is 0.01.

Appendix D

Cellular compartments

D.1 Number of interactions between different cellular compartments

Figure D.1, reproduced from (Tarassov et al., 2008), shows the number of interactions between a similar set of cellular compartments to the one used in Chapter 5. By comparing randomised networks Tarassov et al. (2008) highlighted in this figure the cellular compartments with a lower (blue) and higher (red) than expected number of interactions. It can be seen that in general most cellular compartments had a larger than expected number of interactions.

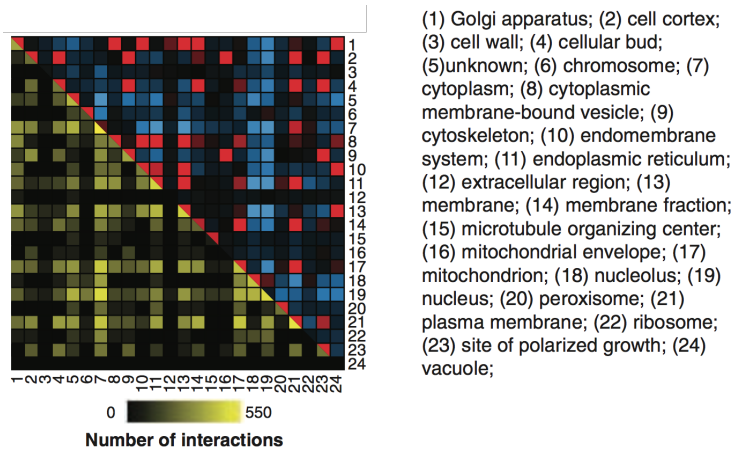


Figure D.1: Number of protein interactions between different cellular compartments (CC) of a Yeast PPI network analysed by Tarassov et al. (2008) (yellow). CC with a larger than expected number of interactions are shown in red, while CC with a lower than expected number of interactions are shown in blue. Figure reproduced from (Tarassov et al., 2008).

D.2 Gene association files

Gene association files provide the associations of genes to GO terms in each of the three ontologies "Molecular Function" (MF), "Biological Process" (BP) and "Cellular Component" (CC) (see Section 5.1.1). In this dissertation we used the gene association files of both Yeast and Human to obtain information about the cellular location of different Yeast and Human proteins. Table D.1 provides some summary statistics of the gene association files we downloaded in May 2016 from the GO consortium database at `http://www.geneontology.org`.

	Number of associations	Number of associations to CC	Number of genes
Yeast	98662	37266	6381
Human	481733	153381	18521

Table D.1: Summary statistics of the 'raw' gene association files of Yeast and Human downloaded in May 2016 from the GO consortium database. Each column represents the total number of associations in each file, the total number of associations in the cellular component ontology (CC), and the total number of different genes with at least one association.

D.3 Cellular location of Yeast and Human proteins

In Chapter 5, we associated Yeast and Human proteins to different cellular compartments (see Tables 5.1 and 5.2). In this section we provide supplementary Figures and Tables of the resulting allocation of proteins to the different cellular compartments.

Figure D.2 shows the distribution of the number of cellular compartments associated to all proteins in the whole co-complex networks of Yeast and Human. Similar distributions were found for the binary networks of Yeast and Human (Figure 5.2).

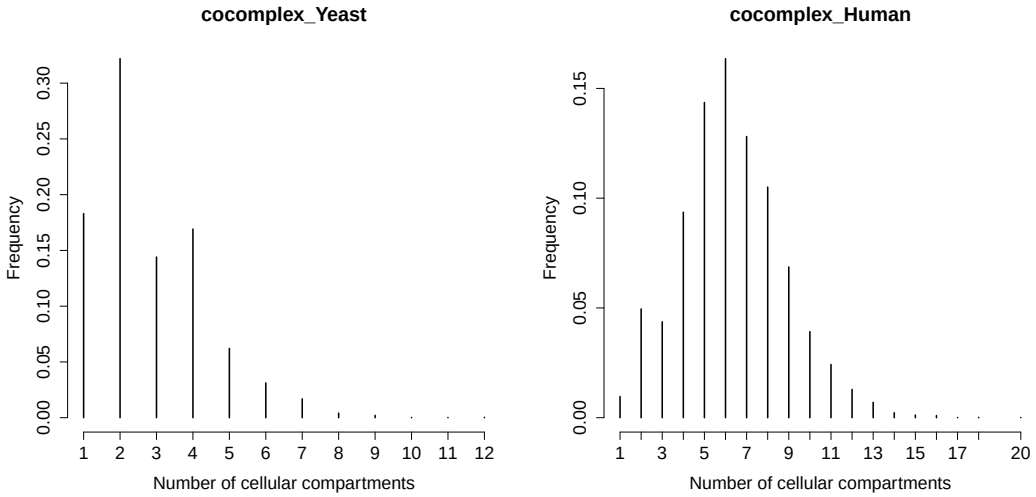


Figure D.2: Distribution of the number of cellular compartments associated to all proteins in the whole co-complex networks of Yeast and Human.

D.3.1 Network summary statistics of cellular compartment networks

Tables D.2 and D.3 show the network summary statistics of the co-complex PPI networks formed by all proteins annotated with each single cellular compartment in the Yeast and Human organisms (see Tables 5.1 and 5.2). Networks marked with * were not used in subsequent analysis.

Compartment	n_v	n_e	C	ρ	L	$Diam$	$\bar{d}$
Cell wall*	6	7	0.4000	0.466667	1.53	2	2.33
Extracellular region*	7	7	0.2308	0.333333	1.86	3	2.00
Peroxisome*	35	87	0.5513	0.146218	2.58	6	4.97
Microtubule organizing center	64	148	0.2763	0.073413	2.91	6	4.62
Cytoplasmic, membrane-bounded vesicle	87	348	0.3825	0.093023	3.09	7	8.00
Cell cortex	120	323	0.2888	0.045238	3.29	8	5.38
Cellular bud	164	405	0.2571	0.030301	3.63	10	4.94
Golgi apparatus	194	745	0.2868	0.039795	3.24	8	7.68
Cytoskeleton	206	673	0.2348	0.031873	3.10	7	6.53
Site of polarized growth	211	593	0.2284	0.026766	3.50	9	5.62
Endoplasmic reticulum	261	729	0.3069	0.021485	3.75	10	5.59
Vacuole	270	930	0.2220	0.025609	3.30	8	6.89
Ribosome	286	1340	0.1564	0.032879	3.04	7	9.37
Nucleolus	291	3055	0.3860	0.072402	2.51	7	21.00
Plasma membrane	294	780	0.0769	0.018110	3.00	7	5.31
Mitochondrial envelope	319	1080	0.1386	0.021293	2.77	7	6.77
Chromosome	402	2542	0.2774	0.031538	2.92	7	12.65
Endomembrane system	656	2740	0.2973	0.012754	3.82	11	8.35
Mitochondrion	856	3027	0.0883	0.008272	3.01	9	7.07
Membrane	1473	6903	0.0821	0.006367	2.97	8	9.37
Nucleus	2217	22117	0.1274	0.009004	2.67	7	19.95
Cytoplasm	4007	33445	0.0343	0.004167	2.52	7	16.69
co-complex Yeast	5616	57211	0.0445	0.003629	2.57	6	20.37

Table D.2: Summary statistics of the Yeast cellular compartment PPI networks, for interactions detected via co-complex methods. The network summary statistics of the co-complex Yeast network are given for ease of comparison. * Networks not considered in further analysis due to their small number of edges (< 100). Protein interactions obtained in October 2015.

Compartment	n_v	n_e	C	ρ	L	$Diam$	$\bar{d}$
Lipid particle*	17	22	0.1558	0.161765	2.32	4	2.59
Proteinaceous extracellular matrix*	47	49	0.0492	0.045328	5.57	14	2.09
Peroxisome	66	102	0.1272	0.047552	3.19	9	3.09
Cilium	128	198	0.2245	0.024360	5.27	13	3.09
Ribosome	179	3461	0.7717	0.217249	2.00	5	38.67
Nuclear envelope	203	453	0.1714	0.022094	3.92	12	4.46
Lysosome	235	496	0.2566	0.018040	4.56	12	4.22
Microtubule organizing center	392	1026	0.1578	0.013388	3.87	12	5.23
Nuclear chromosome	427	2920	0.2161	0.032105	2.79	7	13.68
Extracellular space	476	895	0.0825	0.007917	4.50	11	3.76
Endosome	495	1536	0.1441	0.012563	3.55	8	6.21
Nucleolus	599	3820	0.3163	0.021329	3.05	8	12.75
Cytoplasmic, membrane-bounded vesicle	707	2252	0.1366	0.009023	3.67	8	6.37
Vacuole	743	2585	0.1247	0.009378	3.57	9	6.96
Chromosome	746	5260	0.1699	0.018929	2.94	7	14.10
Golgi apparatus	757	1938	0.1057	0.006773	4.02	10	5.12
Endoplasmic reticulum	875	2587	0.1090	0.006766	3.77	10	5.91
Mitochondrion	1092	4922	0.1444	0.008263	3.51	10	9.01
Cytoskeleton	1358	5757	0.1016	0.006248	3.53	9	8.48
Nucleoplasm	2655	30354	0.1375	0.008615	2.87	8	22.87
Extracellular region	2676	17307	0.1735	0.004836	3.52	10	12.93
Plasma membrane	2677	11522	0.0697	0.003217	3.66	9	8.61
Cytosol	2927	29161	0.1963	0.006810	3.06	8	19.93
Protein complex	3235	32766	0.1102	0.006264	3.08	9	20.26
Nucleus	5438	61184	0.1130	0.004139	3.06	7	22.50
Cytoplasm	8260	85187	0.0867	0.002497	3.19	10	20.63
Organelle	9815	109687	0.0817	0.002277	3.19	9	22.35
Intracellular	10594	116904	0.0776	0.002083	3.20	8	22.07
Cell	11436	122420	0.0761	0.001872	3.26	9	21.41
Co-complex Human	13280	133516	0.0711	0.001514	3.32	9	20.11

Table D.3: Summary statistics of the Human cellular compartment PPI networks, for interactions detected via co-complex methods. * Networks not considered in further analysis due to their small number of edges (< 100). Protein interactions obtained in October 2015.

D.4 Dispersion of DD parameter values used for cellular compartment networks

Histogram of the parameter values obtained for the binary and co-complex cellular compartment networks of Yeast and Human shown in Figures 5.3, 5.4. It can be seen that the while the values of the parameter q for the cellular compartment networks concentrated around similar values across the binary and co-complex Yeast and Human networks, the values of the parameter p are widely spread across the interval $[0, 1]$.

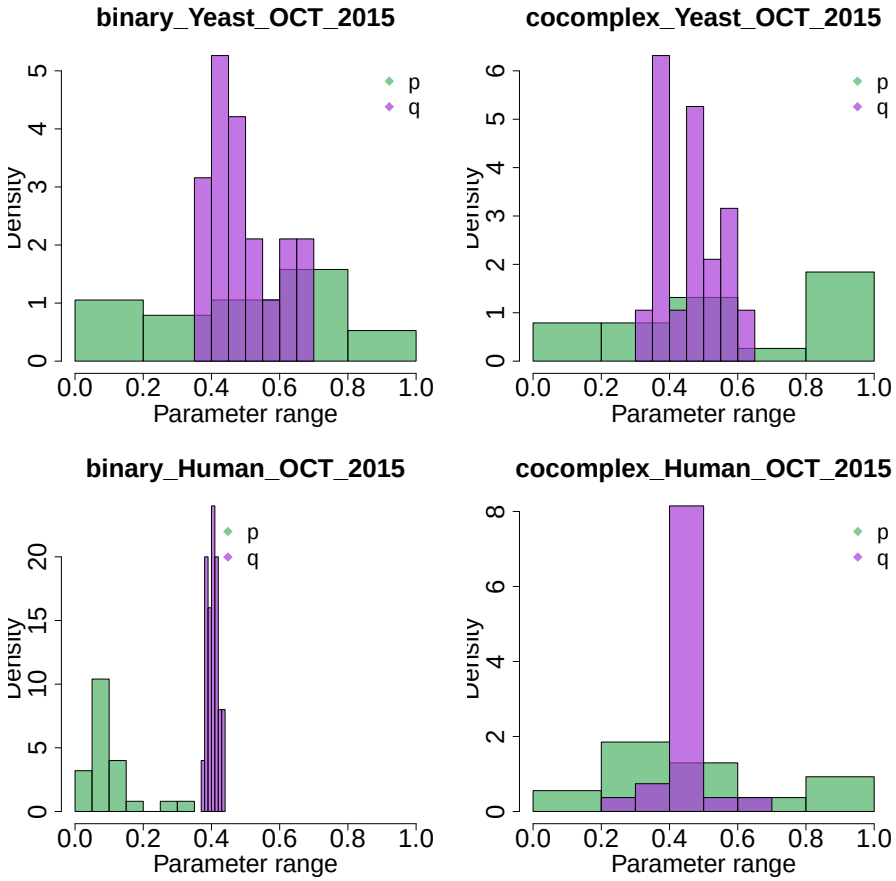


Figure D.3: Histogram of the parameter values obtained for the binary and co-complex cellular compartment networks of Yeast and Human. It can be seen that the values of the parameter q for the cellular compartment networks concentrated around similar values across the binary and co-complex Yeast and Human networks.