

Università degli Studi di Trento
Facoltà di Scienze Matematiche, Fisiche e Naturali



Dottorato in Matematica XXVI ciclo

Tesi

THEORETICAL AND ALGORITHMIC SOLUTIONS
FOR NULL MODELS IN NETWORK THEORY

Relatore interno:

Prof. Andrea Caranti

Relatore esterno:

Dott. Giuseppe Jurman

Dottorando:

Andrea Gobbi

16 Dicembre 2013

Contents

Introduction	5
1 Basics of graphs	9
1.1 Basic definitions	10
1.2 Representation of graphs	11
1.3 Features in graph	13
1.3.1 Degree distribution and centrality scores	13
2 Generation of complex networks with given characteristics	17
2.1 Initial considerations	18
2.2 Algorithm and results: fixed indegree and power-law outdegree . . .	20
2.3 Scale free directed networks (indegree and outdegree): same distribution	23
2.4 Scale free directed networks (indegree and outdegree): two different distributions	26
2.4.1 Saturation model	29
2.5 New definitions for edge-centrality and a preferential attachment generative algorithm	34
3 Randomisation of graphs preserving the degree distribution	37
3.1 The Switching Algorithm	38
3.2 Motivation: NGS data analysis	40
3.3 Bound for the number of SS in the SA	44
3.4 Pairwise-similarity	52
3.5 Markov chain empirical convergence	59
3.5.1 Simulations	59
3.5.2 Autocorrelation time	62
3.6 Package short description	65
3.6.1 Function description	65

4	Null model for co-expression network based on Pearson correlation	69
4.1	Distribution of Pearson correlation	71
4.2	Coexpression network and threshold selection	75
4.3	Applications in functional genomics	80
4.3.1	Large sample size	80
4.3.2	Small sample size	83
4.4	Conclusion	88
5	Null model for random markets	89
5.1	Introduction	89
5.2	Random Market	90
5.2.1	Notation	90
5.2.2	Uniform Random Market	90
5.2.3	Random market with exponential and negative exponential distribution	93
5.2.4	Heterogeneous Agents	95
5.3	Likelihood Function	96
5.3.1	Uniform with constant preferences case	96
5.3.2	Different preferences and matching probabilities	97
5.4	Conclusion	98
6	Graph and game modelization: TTT solitaire	99
6.1	Description	100
6.2	Graph game construction	100
6.3	Abstractions and strategies	104
6.4	Extension to covered cards	108
6.5	Conclusion	110

Introduction

The graph-theoretical based formulation for the representation of the data-driven structure and the dynamics of complex systems is rapidly imposing as the paramount paradigm [1] across a variety of disciplines, from economics to neuroscience, with biological -omics as a major example. In this framework, the concept of Null Model borrowed from the statistical sciences identifies the elective strategy to obtain a baseline points of modelling comparison [2]. Hereafter, a null model is a graph which matches one specific graph in terms of some structural features, but which is otherwise taken to be generated as an instance of a random network.

In this view, the network model introduced by Erdos & Renyi [3], where random edges are generated as independently and identically distributed Bernoulli trials, can be considered the simplest possible null model. In the following years, other null models have been developed in the framework of graph theory, with the detection of the community structure as one of the most important target[4]. In particular, the model described in [5] introduces the concept of a randomized version of the original graph: edges are rewired at random, with each expected vertex degree matching the degree of the vertex in the original graph. Although aimed at building a reference for the community detection, this approach will play a key role in one of the model considered in this thesis. Note that, although being the first problem to be considered, designing null models for the community structures detection is still an open problem [6, 7].

Real world applications of null model in graph theory have also gained popularity in many different scientific areas, with ecology as the first example: see [8] for a comprehensive overview. More recently, interest for network null models arose also in computational biology [9, 10], geosciences [11] and economics [12, 13], just to name a few.

In the present work the theoretical design and the practical implementation of a series of algorithms for the construction of null models will be introduced, with applications ranging from functional genomics to game theory for social studies. The four chapters devoted to the presentation of the examples of null model are preceded by an introductory chapter including a quick overview of graph theory, together with all the required notations.

The first null model is the topic of the second chapter, where a suite of novel algorithms is shown, aimed at the efficient generation of complex networks under different constraints on the node degrees. Although not the most important example in the thesis, the prominent position dedicated to this topic is due to its strict familiarity with the aforementioned classical null models for random graph construction. Together with the algorithms definition and examples, a thorough theoretical analysis of the proposed solutions is shown, highlighting the improvements with respect to the state-of-the-art and the occurring limitations. Apart from its intrinsic mathematical value, the interest for these algorithms by the community of systems biology lies in the need for benchmark graphs resembling the real biological networks. They are in fact of uttermost importance when testing novel inference methods, and as testbeds for the network reconstruction challenges such as the DREAM series [14, 15, 16].

The following Chapter three includes the most complex application of null models presented in this thesis. The scientific workfield is again functional genomics, namely the combinatorial approach to the modelling of patterns of mutations in cancer as detected by Next Generation Sequencing exome Data. This problem has a natural mathematical representation in terms of rewiring of bipartite networks and mutual-exclusively mutated modules [17, 18], to which Markov chain updates (switching-steps) are applied through a Switching Algorithm SA. Here we show some crucial improvements to the SA, we analytically derive an approximate lower bound for the number of steps required, we introduce BiRewire, an R package implementing the improved SA and we demonstrate the effectiveness of the novel solution on a breast cancer dataset.

A novel threshold-selection method for the construction of co-expression networks based on the Pearson coefficient is the third and last biological example of null model, and it is outlined in Chapter four. Gene co-expression networks inferred by correlation from high-throughput profiling such as microarray data represent a simple but effective technique for discovering and interpreting linear gene relationships. In the last years several approaches have been proposed to tackle the problem of deciding when the resulting correlation values are statistically significant. This is mostly crucial when the number of samples is small, yielding a non negligible chance that even high correlation values are due to random effects. Here we introduce a novel hard thresholding solution based on the assumption that a coexpression network inferred by randomly generated data is expected to be empty. The theoretical derivation of the new bound by geometrical methods is shown together with two applications in oncogenomics.

The last two chapters of the thesis are devoted to the presentation of null models in non-biological contexts.

In Chapter 5 a novel dynamic simulation model is introduced mimicking a

random market in which sellers and buyers follow different price distributions and matching functions. The random market is mathematically formulated by a dynamic bipartite graph, and the analytical formula for the evolution along time of the mean price exchange is derived, together with global likelihood function for retrieving the initial parameters under different assumptions.

Finally in Chapter 6 we describe how graph tools can be used to model abstraction and strategy (see [19, 20, 21]) for a class of games in particular the TTT solitaire. We show that in this solitaire it is not possible to build an optimal (in the sense of minimum number of moves) strategy dividing the big problems into smaller subproblems. Nevertheless, we find some subproblems and strategies for solving the TTT solitaire with a negligible increment in the number of moves. Although quite simple and far from simulating highly complex real-world situations of decision making, the TTT solitaire is an important tool for starting the exploration of the social analysis of the trajectories of the implementation of winning strategies through different learning procedures [22].

Chapter 1

Basics of graphs

The word **graph** was first introduced by James Joseph Sylvester, an English mathematician, in 1878. In his paper, published in *Nature*, he highlighted the analogy between "quantic invariants" and "co-variants" of algebra and molecular diagrams. Euler laid the foundations of graph theory in 1735 sixty years before (without using the word graph!), solving the *Seven Bridges of Königsberg* problem. A century after the Euler's paper, Arthur Cayley started to study *trees* a particular class of graphs, in order to solve an analytical forms arising from differential calculus. In 1937 George Plya gave its contributions in theoretical chemistry studying some techniques related to the enumeration of graphs. All these cases prove that, since from the origins, graph theory has been connoted as an across-the-board interdisciplinary subject. Parallel to graph theory, scientist from different fields had started to use graph elements developing an applicative theory (**complex network theory**) based on real data. These two paths are not independent since usually complex-network problems are solved theoretically (strengthen empirical results), and new theoretical elements can be used to explore and categorise real graphs. In the last years (complex) graph theory has become very popular thanks to the many applications in all scientific field starting from mathematics, physics, chemistry, informatics and bioinformatics, neuroscience, social sciences, economy and also for the exponential growth of collectable data coming from such fields. The necessity to store, manage and analyse these data had driven a huge community of scientist to focus their studies on complex networks theory and, as a consequence, in graph theory.

In this chapter we will shortly present some useful definitions and notations about graph theory. We will also introduce some mathematical tools for the manipulation of graphs. Finally, we will give some important graph's features.

1.1 Basic definitions

Definition 1. A **graph** \mathcal{G} , called also network, is a ordered couple $\{V, E\}$ such that $V = \{x_i\}_{i=1}^n$ is a finite set whose element are called **nodes** and E is a set of e **edges**, which are 2-element subsets of V (an edge is related with two nodes and this relation is represented with unordered pair of nodes).

This kind of graph is called also **simple** (no multiple edges) and **undirected**.

Definition 2. If $E \subset V \times V$, i.e. the edges are ordered pairs of nodes, the the graph is called **digraph** or **directed graph**.

Moreover, if multiple edges are allowed, we are dealing with **multigraphs** in the case of undirected graphs or **quiver** in the case of digraphs. Finally, if at each edge is associated a weight the resulting graph is called **weighted graph** (digraph).

Definition 3. If V can be partitioned into two disjoint sets V_r and V_c such that there are not edges within these two sets, the graph is called **bipartite graph** or **bigraph**.

In a bipartite graph there are two **natural projections**: $P(V_r)$ and $P(V_c)$ defined as $P(V_r) = \{V_r, \{(i, j), i, j \in V_r, \exists k \in V_c \text{ s.t. } (i, k), (j, k) \in E\}\}$ and $P(V_c) = \{V_c, \{(i, j), i, j \in V_c, \exists k \in V_r \text{ s.t. } (i, k), (j, k) \in E\}\}$, in other words, in $P(V_r)$ there is an edge between two nodes if and only if they share a vertex in the graph \mathcal{G} .

Definition 4. A **path** of length n between two nodes u and v is a sequence of n edges connecting the two nodes and is denoted by $p(u, v)$. If n is the minimum number of required edges, than a path of length n is called **shortest paths** and indicated with $sp(u, v)$.

Definition 5. A graph is called **connected** if between each couple of nodes there is a path.

Definition 6. The **distance** or **geodesic distance** d between $u, v, \in \mathcal{G}$ is defined as $d(u, v) = \#sp(u, v)$. The distance of a node u from $A \subset V(\mathcal{G})$ is defined as $d(u, A) = \min_{c \in A} d(u, c)$. The number of shortest paths between two nodes u, v will be denoted with $\sigma_{u, v}$.

It is easy to see that if \mathcal{G} is undirected and connected, then $(V(\mathcal{G}), d)$ is a metric space.

Definition 7. The **diameter** of a connected graph \mathcal{G} is defined as: $d(\mathcal{G}) = \max_{u, v \in V} \#sp(u, v)$. We will call diameter also a set of edges realising this maximum.

Definition 8. The **degree** of a node u in a graph \mathcal{G} is the number of edges containing u . If \mathcal{G} is a digraph, it is possible to distinguish between *indegree*, the number of the $(*, u)$ edges and *outdegree* the number of $(u, *)$ edges.

Definition 9. Let \mathcal{G} be a graph, with \mathcal{G}^* we will denote the **representative graph** or **dual** of \mathcal{G} i.e. the graph whose vertexes are representing the edges of \mathcal{G} (see [23] for more details).

1.2 Representation of graphs

The nodes in a graph are visually represented as circle and an edge between them as an arrow. This representation can be useful to visualise small network and see the interactions between the nodes. There are a lot of layouts for visualise a graph: we can dispose the nodes in a circle or in a grid or use some physical ideas (like force/energy) for drawing graphs in an aesthetically pleasing way. The relations between the nodes in a graph \mathcal{G} can be also represented as a matrix. The most important matrix associated to a graph is the so called adjacency matrix.

Definition 10. The **adjacency matrix** A of a graph is a square matrix $n \times n$ such that every entry $w_{i,j}$ represents the relation between the two nodes i and j according to the type of network.

Some of these adjacency matrices are showed in Tab. 1.1.

From the adjacency matrix we can extrapolate some useful information about the graph, for example in a digraph the i -th row-sum represent the idegree of the node i .

If \mathcal{G} is a bipartite graph with $|V_r| = n_r$ and $|V_c| = n_c$ then its the adjacency matrix A has the following structure:

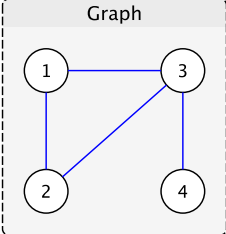
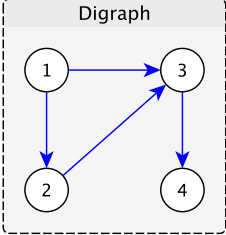
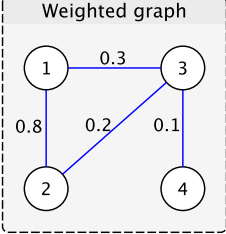
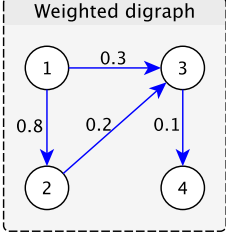
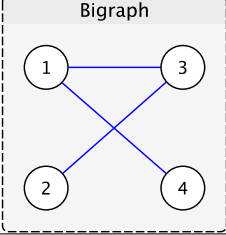
$$A = \left[\begin{array}{c|c} 0 & B \\ \hline B^t & 0 \end{array} \right]$$

where B is called **biadjacency** matrix or **incidence** matrix, i.e. B is a $n_r \times n_c(0-1)$ matrix and $w_{i,j} = 1$ if and only if $(x_{r_i}, x_{c_j}) \in E$.

The representation of \mathcal{G} as a matrix (adjacency matrix, degree matrix, Laplacian matrix, ...) is also a way to store or visualise graphs but naturally leads to the **spectral graph theory** studying, for example, the relations between eigenvectors and random walk stationary distribution or clusters.

The matricial expression is useful for theoretical purpose but it is memory consuming in real applications. An other important structure to store a graph is its **edge-list** L : an $e \times 2$ matrix such that $(l_{i,1}, l_{i,2}) \in E, \forall i = 1, \dots, e$; in the case of weighted graphs an extra column is required. Finally, the **adjacency-list**

A of a graph is a vector of list such that each list $A_i = \{j : (i, j) \in E\}$ contains the **neighbours** of the node i .

Type	Adjacency matrix	Edge-list	Visual representation
Graph	$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 2 & 3 \\ 3 & 4 \end{pmatrix}$	
Digraph	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 2 & 3 \\ 3 & 4 \end{pmatrix}$	
Weighted graph	$\begin{pmatrix} 0 & 0.8 & 0.3 & 0 \\ 0.8 & 0 & 0.2 & 0 \\ 0.3 & 0.2 & 0 & 0.1 \\ 0 & 0 & 0.1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 & 0.8 \\ 1 & 3 & 0.3 \\ 2 & 3 & 0.2 \\ 3 & 4 & 0.1 \end{pmatrix}$	
Weighted digraph	$\begin{pmatrix} 0 & -0.8 & -0.3 & 0 \\ 0.8 & 0 & -0.2 & 0 \\ 0.3 & 0.2 & 0 & -0.1 \\ 0 & 0 & 0.1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 & 0.8 \\ 1 & 3 & 0.3 \\ 2 & 3 & 0.2 \\ 3 & 4 & 0.1 \end{pmatrix}$	
Bigraph	$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 3 \\ 1 & 4 \\ 2 & 3 \end{pmatrix}$	

Tab. 1.1: Adjacency matrices and edge-lists of some introduced graphs.

1.3 Features in graph

The number of nodes and edges, the type of such edges are not the unique features in a graph. The degree distribution is an important feature characterising a graph, in fact it is related to the concept of random graphs or scale-free networks (see the following subsection). An other important group of features are related to centrality scores, *i.e.* the importance of nodes (or edges). This measure, combine with the concept of robustness, plays a fundamental role in the study of percolation and transport in a graph (for example power supply) or in the study of the spreading of an epidemic in a population [24].

Also the mesoscopic structure of a graph, characterised by homogeneous groups of nodes called modules or communities, is crucial to formulate a mechanisms for the genesis of the graph and to uncover (possible) relationships between the nodes not revealed inspecting the whole graph [25].

1.3.1 Degree distribution and centrality scores

A useful feature in graph theory is the **degree distribution**, *i.e.* the probability distribution of the nodes' degrees over the whole graph.

There are two important degree distributions: the binomial (n, p) distribution (related to **random graph**) and the power-law α distribution (for the so called **scale-free** network). The first kind of networks was intensively studied by Erdős in [26] in which the authors showd some important structures (like the size of connected component, largest/giant components, connectivity) based on the probability p and the number of nodes n . Erdős was an important mathematician so that his friends create the so called *Erdős number*, *i.e.* the length of the shortest path (in the coauthor network: a scale-free network) between scientist and him. Approximately 200000 mathematicians have an Erdős number, and some have estimated that 90% of the active mathematicians have an Erdős number smaller than 8 (the so called **small world phenomenon**)¹.

Generate a random $(n - p)$ network is trivial since the whole mechanism can be controlled by a binomial process.

From a complex network point of view, real networks tend to not to be random, for example they are highly right-skewed (a large majority of nodes have low degree but a small number, **hubs**, have an high degree).

The Internet network, some social networks and biological networks approximately follows the power-law distribution $p(k) \sim k^{-\alpha}$ with $2 < \gamma < 3$. This phenomenon has been deeply studied in the last years stating from Price [28] in

¹Actually, my Erdős number is equal to 4 because I wrote [27] with Iorio who has Erdős number is equal to 3. There is a similar number between go players, the so called **Shusaku number** (in honour to Honinbo Shusaku).

1965 and Barabasi [29] in 1999. This last physicist is one of the most influential personality and cited physicists in network science and its aforementioned paper is the one of the ten most cited paper in physical sciences.

This degree distribution implies some properties in the graph like the small world phenomenon (**six-degree of separation**) and the **clustering** coefficient (see [30] for details).

Price and, recently Barabasi, gave also a generative model for the construction of scale-free networks (only for the indegree distribution).

These generative methods (random vs. scale free) are very important whenever a reconstruction method has been to tested (for example looking at its stability) and in the following chapter we will see the importance of construction a suitable benchmark for testing purpose. Moreover we will see the importance of new generative algorithms in which some features are constrained and where the Erdős (and Barabasi) model can not be used as benchmark-builder.

The importance of a node in a graph \mathcal{G} can be measure looking at its degree or using a more general **centrality** measure. A centrality measure can be used to identify influential person in a social network, the spreaders in an epidemics and even a hot spot in an urban network. In literature, in addition to the degree centrality, three more measure has been developed first in sociology and then in social network analysis. The **eigenvector** centrality of the i -th node correspond to the i -th entry of the dominant eigenvector of the adjacency matrix of \mathcal{G} . Google's *PageRank* is a variant of this centrality measure. The **closeness** centrality of a node u is define as $C_c(u) = \sum_{v \neq u} 2^{-d(u,v)}$. Finally, the **betweenness** centrality of a node u is defined as $C_b(u) = \sum_{v \neq u \neq s} \sigma_{v,s}(u) / \sigma_{v,s}$ where $\sigma_{v,s}(u)$ is the number of shortest paths between v and s containing u . In Fig. 1.1 we can see how different measure of centrality weigh differently the importance of the nodes.

With the same arguments we can give also a centrality measure to the edges; unfortunately, in this specific field, the literature is poor and only an extension for the betweenness centrality (counting the edges in the shortest paths) had been studied.

In section 2.5 we will present a natural solution for extending node centrality measures to the edges.

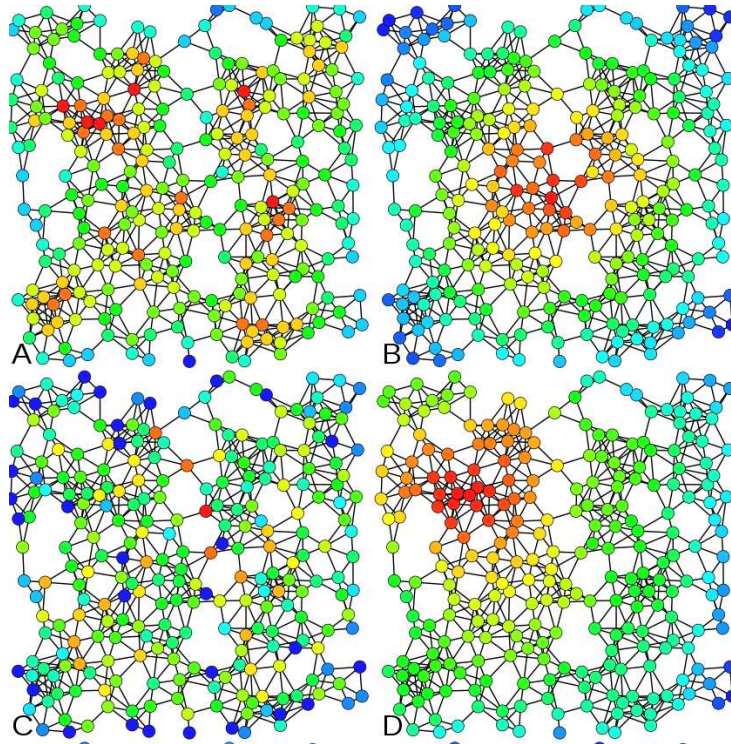


Fig. 1.1: The same graph in which nodes are coloured based on their centrality scores: (A) degree (B) closeness (B) betweenness (C) eigenvector (from <http://en.wikipedia.org/wiki/Centrality>).

Chapter 2

Generation of complex networks with given characteristics

The Erdős-Rényi model with n nodes and probability p is the most important algorithm for the generation of a random graph with n nodes and in which two nodes are connected with probability p . This algorithm was widely studied and is currently used in graph theory (we will find it in Chapter 4). But real networks are far from being random. Let us think for example of the social network or coauthor networks in which there are few hubs and a lot of nodes with low degree. For this reason scale-free networks are more interesting to study.

There are several models for the generation of scale-free networks (for example [31]). Some of them are based on the Price's idea [28] of preferential attachment, later rephrased by Barabasi [29]. These methods are very useful for mimicking some well-known real cases like the Internet and citations' networks. All these methods work on the topology of the network and the edges and/or the nodes are usually added under certain conditions.

As we introduced above, generative models can be used for the creation of a statistical baseline for network inference, network reconstruction, algorithm validation and stability. All these features are resumed in the DREAM (Dialogue for Reverse Engineering Assessments and Methods) challenges started in 2006 [32] and now at the eighth edition. As we can read from their website *the main objective is to catalyze the interaction between theory and experiment, specifically in the area of cellular network inference and quantitative model building. DREAM challenges address how we can assess the quality of our descriptions of networks that underlie biological systems, and of our predictions of the outcomes of novel experiments. These are not simple questions. Researchers have used a variety of algorithms to deduce the structure of biological networks and/or to predict the outcome of perturbations to their systems. They have also evaluated the success of their methodologies using a diverse set of non-standardised metrics. What is still*

needed, and what DREAM aims to achieve, is a fair comparison of the strengths and weaknesses of these methods and a clear sense of the reliability of the models that researchers create.

Also sbv IMPROVER [33] (Systems Biology Verification) is an important challenge that *it is different from other approaches such as DREAM as it focuses on the verification of processes in an industrial context, and not on basic questions in science. sbv IMPROVER could allow an organization to benchmark its methods and verify that these are state of the art performance for their industrial processes.*

In this section we shortly present some simple ideas and methods for the construction of scale-free networks starting from its adjacency matrix.

The adjacency matrix A of a graph \mathcal{G} is a $n \times n$ matrix where n is the number of nodes. The i, j -th entry of the matrix gives us information about the connectivity between the node i and the node j . For example in the case of directed and unweighted network if $A_{i,j} = 1$ then $(i, j) \in E$. Generally, in the row of the adjacency matrix we can read information about indegree, on the columns we can read information about the outdegree.

We have implemented some methods for the generation of unweighted digraphs with some prefixed parameters:

- fixed indegree K and power-law distribution for the outdegree proving that the parameter of the power-law depends on K (see Prop. 1),
- methods in which both indegree and outdegree have a power-law distribution:
 - in Section 2.3 indegree and outdegree have the same distribution,
 - in Section 2.4 the two distributions are different,
 - and in subsection 2.4.1 we revisited the preferential attachment in the case of two power-law distributions.
- In Section 2.5 we extend the concept of centrality to the edges and we propose an innovative growing method based on edge centrality.

2.1 Initial considerations

Let $N > 1$ and $K > 1$ be respectively the number of nodes we want to have in our network and the fixed indegree for each node. Let $D \geq 0$ be the minimum desired outdegree. According to [29], the outdegree distribution has the following form:

$$p(k) = ck^{-\alpha} \quad \text{with } \alpha > 1 .$$

The parameter c is a constant such that:

$$\sum_{k=D}^N ck^{-\alpha} = 1, \quad (2.1)$$

i.e. $p(k)$ is a probability distribution (for $D = 0$ we will consider $p(k) = c(k+1)^{-\alpha}$). Note that for different choices of α we obtain different distributions (and thus different normalisation factors c). The value $p(k)$ represents the probability of a certain node to have outdegree k . This can be viewed as a density and therefore (the integer part of) $Np(k)$ can be interpreted as the number of nodes with outdegree k .

In terms of the adjacency matrix, the outdegree and the indegree of the n -th node can be respectively computed as the sum of the elements on the n -th column and the sum of the n -th row (our graphs are unweighted and direct and thus the elements of the adjacency matrix are either 0 or 1).

Because of the considered constraints, in each row the total number of ones (indegree) is equal to K , while the number of ones for each column (outdegree) follows a power-law distribution. The above considerations allow us to say that the number of columns with k ones are $\lfloor Np(k) \rfloor$.

Therefore, the following identity holds

$$\begin{aligned} & \#\{\text{col. with } k \text{ ones}\} \\ & \sum_{k=D}^N \overbrace{Nck^{-\alpha} k} = \underbrace{NK}, \\ & \#\{\text{ones in the matrix}\} \end{aligned}$$

and, consequently, we have

$$\sum_{k=D}^N ck^{1-\alpha} = K. \quad (2.2)$$

Because of Eq. 2.1 and Eq. 2.2, we can state the following proposition:

Proposition 1

Fixed $N, K, D \in \mathbf{N}$ such that $N > K > D$ and $K < \frac{N-D+1}{\ln N - \ln D}$ there exist only one $\alpha \in \mathbf{R}^+$ such that Eq. 2.1 and Eq. 2.2 hold simultaneously.

Proof. Let:

$$f(x, y) = \sum_{k=D}^N yk^{-x} - 1 \quad \text{and} \quad g(x, y) = \sum_{k=D}^N yk^{1-x} - K.$$

Dividing g by K and using f we obtain:

$$K = \frac{\sum_{k=D}^N k^{1-x}}{\sum_{k=D}^N k^{-x}} = \frac{H_{N,D,x-1}}{H_{N,D,x}} = \frac{H_{N,x-1} - H_{D-1,x-1}}{H_{N,x} - H_{D-1,x}},$$

where $H_{N,x}$ is the N -th generalised harmonic number.

We define $h(x) = \frac{H_{N,D,x-1}}{H_{N,D,x}}$: $h(x)$ is a decreasing function, with initial term

$$\begin{aligned} h(1) &= \frac{H_{N,0} - H_{D,0}}{H_{N,1} - H_{D,1}} \\ &= \frac{N - D + 1}{H_N - H_{D-1}} \quad \text{and using the asymptotic expansion of harmonic numbers,} \\ &> \frac{N - D + 1}{\ln N - \ln(D - 1)}. \end{aligned} \tag{2.3}$$

Computational considerations lead to state that

$$\lim_{x \rightarrow +\infty} h(x) = D.$$

If $D = 1$ it is easy to prove also analytically that the limit is 1.

Using the above limit estimation, the Eq. 2.3 and the hypotheses on K we can find the unique solution to $h(x) = K$. \square

2.2 Algorithm and results: fixed indegree and power-law outdegree

Using Prop. 1, we can design an algorithm to generate a network with the desired properties.

The workflow of the proposed algorithm reads as follows:

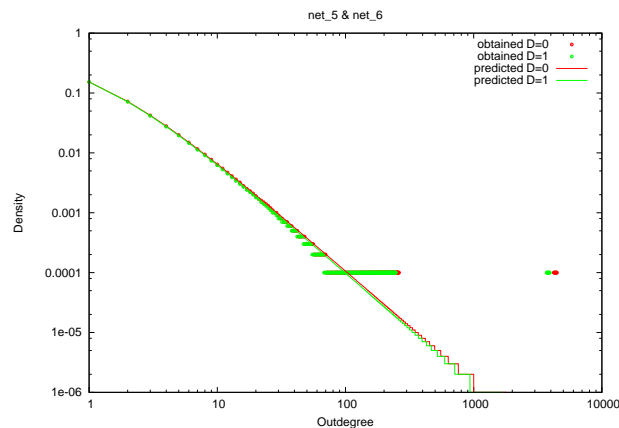
- Given N, K, D compute α (using for example bisection algorithm).
- For each outdegree $D \leq d \leq N$ calculate the number of nodes n_d with outdegree d , i.e. $n_d = Ncd^{-\alpha}$.
- Initialize the adjacency matrix with zeroes.
- Fill the adjacency matrix such that each row has K ones and for each d there are n_d columns with d ones.
- Fill the remaining columns (we are considering the largest natural value that is not greater than n_d and so there are approximation errors i.e. empty columns) with ones in a way that the rows have K ones.

	N	K	D	α
net_1	10000	2	0	2.23028
net_2	10000	2	1	2.470964
net_3	10000	5	0	1.997803
net_4	10000	5	1	2.048527
net_5	10000	10	0	1.855389
net_6	10000	10	1	1.875867

Tab. 2.1: Parameters of generated networks.

We applied the algorithm described above for the generation of 6 networks whose properties are summarised in the table 2.1:

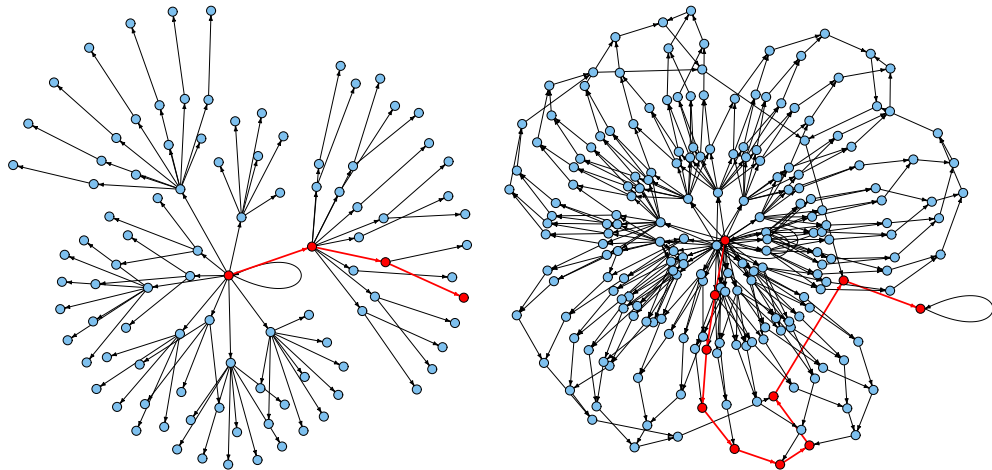
The density of some of the 6 networks net_i as a function of the outdegree is shown in Fig. 2.1.

Fig. 2.1: Log-Log plots distribution for net_5 and net_6 .

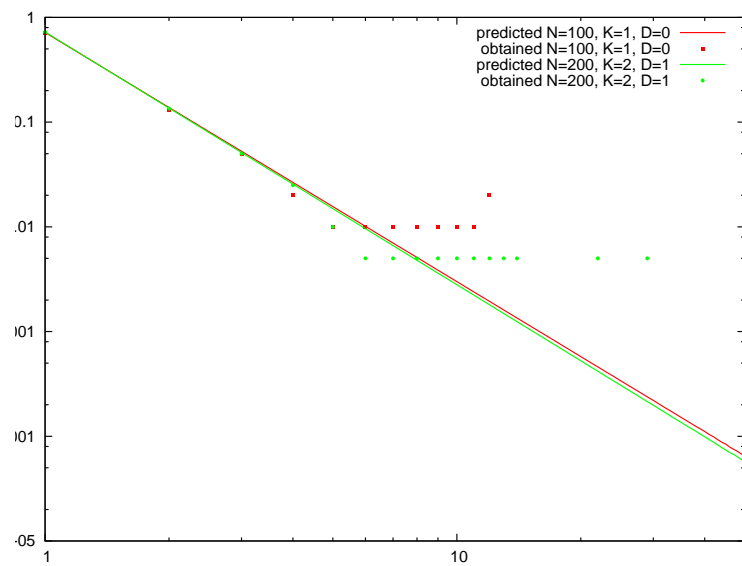
In Fig. 2.2(a), and in Fig. 2.2(b) are showed two networks generated using the method discussed above, and in Fig. 2.2(c) the relative distributions.

The relation between the expected and the obtained characteristics of the constructed networks is promising. In particular, the nodes with low outdegree follow perfectly the power-law distribution.

Some outliers occur, caused by approximation of the distribution (near the predicted line) and some others (dots in the right side of the plots) caused by the filling of the remaining columns (last step of the algorithm).



(a) Generated network with $N = 100$, $K = 1$ and $D = 0$. In red the diameter. (b) Generated network with $N = 100$, $K = 2$ and $D = 1$. In red the diameter.



(c) Distributions relative to the networks in Figs. 2.2(a), 2.2(b).

Fig. 2.2: Topology and Log-Log plot of the distributions for two examples.

2.3 Scale free directed networks (indegree and outdegree): same distribution

In this section we propose an alternative method for the generation of a scale-free directed networks where both indegree and outdegree distributions follow the power-law distribution. We will prove this important fact:

Proposition 2

Let $N > 1$ and $D \geq 1$ be the number of nodes and the minimum degree desired (the same for the indegree and outdegree), and let $p(k) = ck^{-\alpha}$ and $q(k) = dk^{-\beta}$ describe the distributions related to the indegree and outdegree. Then $c = d$ and $\alpha = \beta$

Proof. According to [29] the mean value for out and indegree is the same, and so:

$$\sum_{k=D}^N ck^{1-\alpha} = \sum_{k=D}^N dk^{1-\beta}.$$

We can see it using also the adjacency matrix, indeed, as before, if we count the number of ones in the matrix we obtain:

$$\sum_{k=D}^N \overbrace{Nck^{-\alpha}k}^{\#\{\text{col. with } k \text{ ones}\}} = \sum_{k=D}^N \underbrace{Ndk^{-\beta}k}_{\#\{\text{row with } k \text{ ones}\}}, \quad (2.4)$$

Moreover, we have that:

$$c = \frac{1}{\sum_{k=D}^N k^{-\alpha}} \quad \text{and} \quad d = \frac{1}{\sum_{k=D}^N k^{-\beta}}. \quad (2.5)$$

Using eq. 2.4 and eq. 2.5 we can write:

$$\frac{\sum_{k=D}^N k^{-\alpha}}{\sum_{k=D}^N k^{-\alpha+1}} = \frac{\sum_{k=D}^N k^{-\beta}}{\sum_{k=D}^N k^{-\beta+1}},$$

or, using the notation seen in the proof of the theorem 1,

$$h(\alpha) = h(\beta).$$

The function $h(x)$ is strictly decreasing, and so injective and this is sufficient to conclude that $\alpha = \beta$ and so $c = d$. \square

The algorithm used for this purpose is quite similar to the algorithm showed in the previous sections:

- Given $N > D \geq 1$ and α calculate, for each degree $D \leq d \leq N$, the number of nodes n_d with degree d , i.e. $n_d = Ncd^{-\alpha}$.
- Initialize the adjacency matrix with zeroes.
- Fill the adjacency matrix with blocks of ones following the distribution calculated above. This leads to a symmetric matrix which diagonal is filled with ones. In order to avoid non-connected components and selfloops we need a further step.
- Shift the rows by m positions where m is the minimum value such that $\lfloor Np(m) \rfloor = 0$.

We applied the algorithm described above for the generation of 3 networks whose properties are summarised in Tab. 2.2:

	N	α
net_1	10000	1.5
net_2	10000	2.0
net_3	10000	2.5

Tab. 2.2: Parameters of generated networks.

The density of the 3 networks net_i as a function of the outdegree is shown in Fig. 2.3. A smaller network ($N = 200$) was generated so that we can explore it (see Fig.2.4).

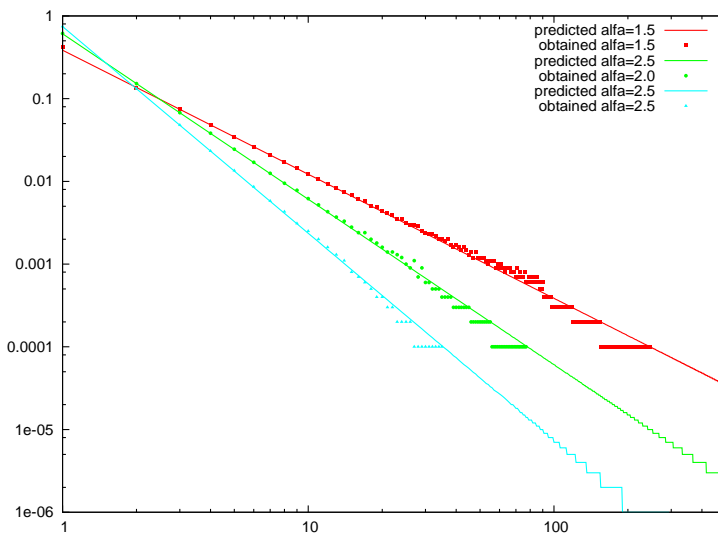
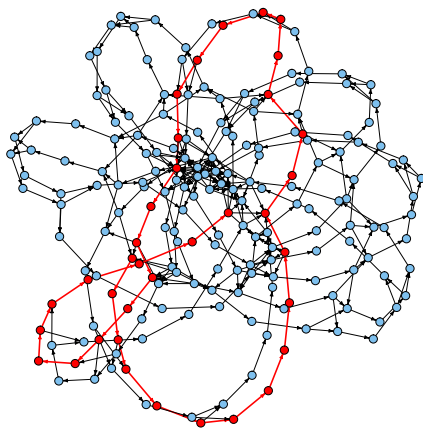
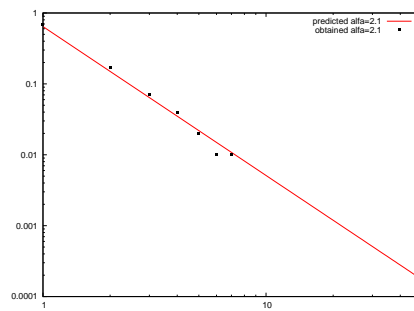


Fig. 2.3: Log-Log plot for net_1 , net_2 and net_3 .



(a) Generated network with $N = 200$.
In red the diameter.



(b) Degree distribution.

Fig. 2.4: Topology and Log-Log plot distribution for the generated network.

2.4 Scale free directed networks (indegree and outdegree): two different distributions

In this section we want to generalise the previous algorithm in order to generate a network with two different power-law distributions. We have just seen that the minimum degree can not be the same (this implies that the two distributions have the same exponent). Fix N, D, E, α and β so that:

$$\begin{aligned} p(k) &= ck^{-\alpha} & k \in [D, N] \text{ represents the indegree distribution and} \\ q(k) &= dk^{-\beta} & k \in [E, N] \text{ represents the outdegree distribution.} \end{aligned}$$

Beyond the mathematical constraints¹, we want to generate a network for all possible coherent choice of the parameters. In order to do that we split into three parts the algorithm:

1. Indegree:

- Compute the number of nodes having k as indegree.
- If $D > 0$ start from the first row of the adjacency matrix, else start from the i -th row where i is the number of nodes with indegree 0. In the same way, start from the first column if $E > 0$ else from the j -th where j is the number of nodes with outdegree equal to 0.
- Compute M such that $\sum_{i=D}^M Nip(i) \leq N$ and $\sum_{i=D}^{M+1} Nip(i) > N$. Starting from M fill the adjacency matrix with $Np(i)$ blocks of i ones with $D \leq i \leq M$ so that in each column only 1 one is present if $E < 2$ or E ones if $E \geq 2$.

2. Outdegree

- Compute the number of nodes having k as outdegree.
- From $l = M + 1$ fill the matrix with $\max\{Np(l), 1\}$ blocks of l starting from the last row filled in the first step and from the i -th column chosen so that the first i columns have sum coherent with the outdegree distribution. (For example, if $E = 1$ then $i = Nq(1)$.)
- Repeat the last step increasing l and completing the outdegree distribution.
- Stop this second part when the last row of the matrix is reached.

¹Approximation errors and small value for N weakens the power of Props. 1 and 2.

3. Balancing

- This step is necessary if you want to produce a connected graph. Moreover at this step we can swap rows and columns for the manipulation of the network.

We used this algorithm to generate 6 networks which properties are summarised in Tab. 2.3. We can see that some generated networks have $E = D$ but different exponents; this choice is made because the approximation errors and the algorithm leave a certain degree of freedom.

	N	α	D	β	E
net_1	100	2.1	0	1.8	1
net_2	1000	2.1	0	1.8	1
net_3	10000	2.1	0	1.8	1
net_4	100	2.1	1	1.8	1
net_5	1000	2.1	1	1.8	1
net_6	10000	2.1	1	1.8	1

Tab. 2.3: Parameters of the generated networks.

In Fig. 2.5 we summarise the obtained and predicted outdegree and indegree distribution for the networks net_4 and net_5 .

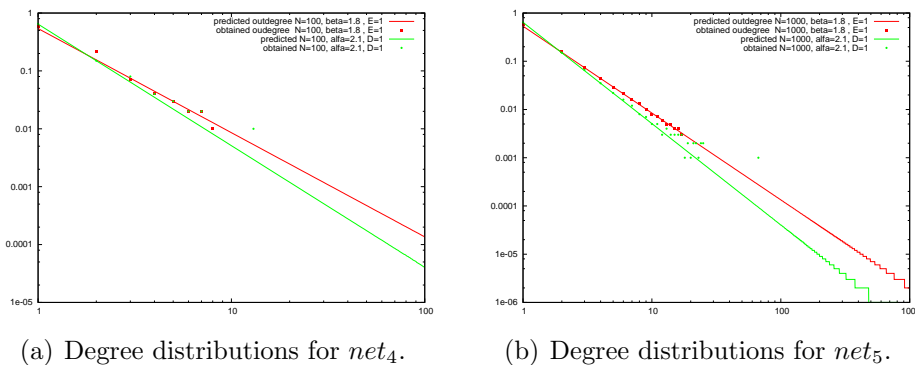
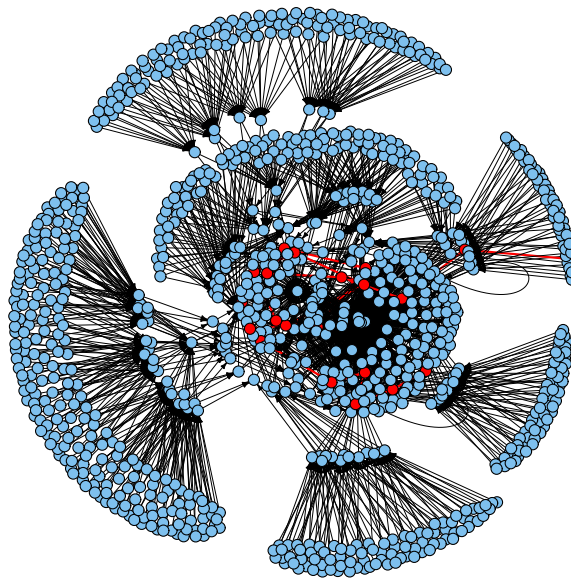
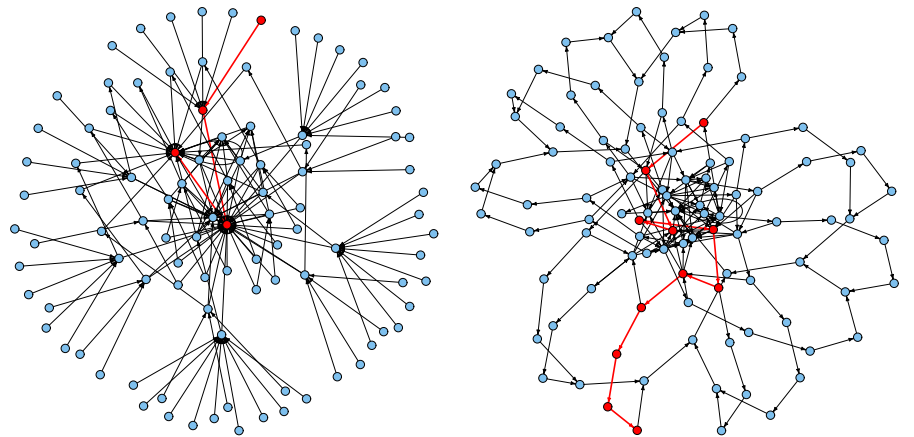


Fig. 2.5: Log-Log plot of the distributions of generated networks net_4 and net_5 .

In Fig. 2.6 we plot the resulting topology of the generated networks net_i $i = 1, 2, 4$.

Fig. 2.6: Topology of (a) net_1 (b) net_2 and (c) net_4 .

2.4.1 Saturation model

In this section we will describe a new method for the generation of directed networks based on the idea of preferential attachment proposed in [29]. Let N, D, E, α and β be numbers so that:

$$\begin{aligned} p(k) &= c(k+1)^{-\alpha} \quad k \in [D, N] \text{ represents the indegree distribution and} \\ q(k) &= d(k+1)^{-\beta} \quad k \in [E, N] \text{ represents the outdegree distribution.} \end{aligned}$$

where c and d are such that $p(k)$ and $q(k)$ are two distribution of probability. Note that we use $k+1$ instead of k in order to allow C and D to be 0. We need to prove the following proposition:

Proposition 3

Let now consider $N > 1, D, E \geq 0$ and $p(k) = c(k+1)^{-\alpha}$ with $k \in [D, N]$ the indegree distribution for a directed network. Then there exists a unique value for β and d such that $q(k) = d(k+1)^{-\beta}$ represents the outdegree distribution of the considered network.

Proof. If $D = E$ the hypotheses of Prop. 2 are satisfied. Suppose than that $D \neq E$. Note that once fixed β also d is determined:

$$\sum_{k=D}^N c(k+1)^{-\alpha} = \sum_{k=E}^N d(k+1)^{-\beta} = 1.$$

Moreover, using the same idea used in Prop. 2, we know that:

$$\sum_{k=D}^N ck(k+1)^{-\alpha} = \sum_{k=E}^N dk(k+1)^{-\beta}.$$

Dividing the second equation by the first we obtain:

$$\frac{\sum_{k=D}^N k(k+1)^{-\alpha}}{\sum_{k=D}^N (k+1)^{-\alpha}} = \frac{\sum_{k=E}^N k(k+1)^{-\beta}}{\sum_{k=E}^N (k+1)^{-\beta}}$$

Using the harmonic numbers and rescaling the indices we can write:

$$\frac{H_{N+1,\alpha-1} - H_{D,\alpha-1} - H_{N+1,\alpha} + H_{D,\alpha}}{H_{N+1,\alpha} - H_{D,\alpha}} = \frac{H_{N+1,\beta-1} - H_{E,\beta-1} - H_{N+1,\beta} + H_{E,\beta}}{H_{N+1,\beta} - H_{E,\beta}},$$

and so:

$$\frac{H_{N+1,\alpha-1} - H_{D,\alpha-1}}{H_{N+1,\alpha} - H_{D,\alpha}} = \frac{H_{N+1,\beta-1} - H_{E,\beta-1}}{H_{N+1,\beta} - H_{E,\beta}}.$$

Notice that the left part of the equation is a number greater than 1 and so we can use Prop. 1 to conclude the proof. \square

This proposition states that we can not choose independently α and β .

The proposed method follows these steps:

- Generate N values M_1, \dots, M_N representing the indegree distribution, i.e. the i -th node (N_i) has M_i as indegree.
- Using a bisection method and the indegree distribution compute the outdegree distribution. Or better, let $W = \sum_{k=1}^N kM_k$ be the number of edges of the desired network, find the value of β (unique using Prop. 3) such that the resulting outdegree distribution has W edges.
- Choose a method to associate indegree with outdegree. For example nodes with low indegree has low outdegree (or high outdegree).
- Connect the nodes.

The last step is crucial and we will explain it in details. First we have to connect the nodes in order to obtain a connected graph. This step is similar to the method described in [29]. We connect N_2 to N_1 , then N_3 is connected to N_1 or N_2 , N_4 to N_1, N_2 or N_3 and so one. We connect N_i to N_j if N_j is the node having the maximum indegree *remaining* among the fist j (i.e. M_j is the maximum among the previous). Once N_i is connected to N_j we decrease the indegree of N_j (i.e. decrease M_j) and the outdegree of N_i . This procedure guarantees us to have a connected graph. Finally, for each (remaining) edge find the two nodes N_i and N_j such that N_i has the maximum indegree remaining and N_j the maximum outdegree remaining and they are not already connected. Connect N_j to N_i and decrease the indegree of N_i and the outdegree of N_j .

The benefit to use this method respect to the previous one is that we can simulate the distribution for the indegree and outdegree (not using the integer part of $Np(k)$). Moreover we can choose the method to associate the indegree to the outdegree, and finally it is easy to modify this procedure in order to have a network with outdegree fixed (see section 1), or an undirected network. Indeed, if we fix the outdegree to K , we have NK ones in the adjacency matrix, and so we can compute α in order to generate a such network. For the generation of undirected graphs is sufficient to consider only the indegree distribution as the degree distribution. Using this algorithm we generated 5 networks which properties are summarised in Tab. 2.4. The networks net_4 and net_5 are generated in order to

simulate two subcase: the graph net_4 has the property to have outdegree constant to 1, and net_5 is undirected with N edges (see [29]).

	N	α	D	E
net_1	100	2.1	0	1
net_2	1000	2.1	0	1
net_3	100	2.1	1	1
net_4	1000	2.245	0	1
net_5	1000	3.062	1	Undirected

Tab. 2.4: Parameters of the generated networks.

In Fig. 2.7 and Fig. 2.8 we summarise some results (degree distributions and topology) for some of the networks net_i . In Fig. 2.7(c) and Fig. 2.7(d) , we compare net_5 with a network generated using the method described in [29] using the R -command $barabasi.game(N=1000)$ in the library *igraph*.

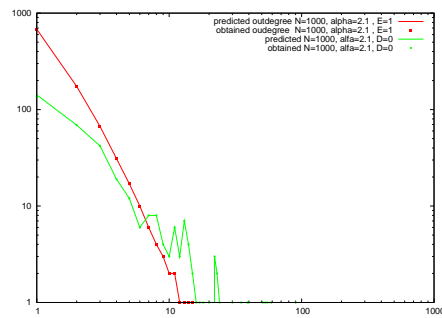
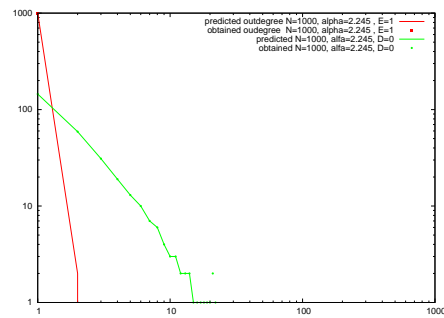
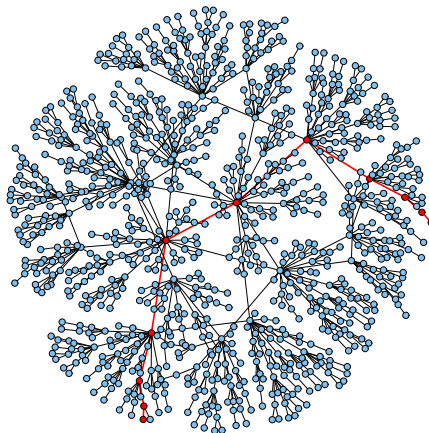
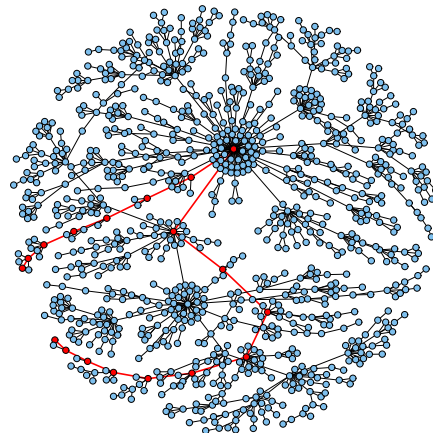
(a) Degree distributions for net_2 .(b) Degree distributions for net_4 .(c) Generated network net_5 .(d) Barabasi game $N = 1000$.

Fig. 2.7: Log-Log plot of the distributions of generated networks.

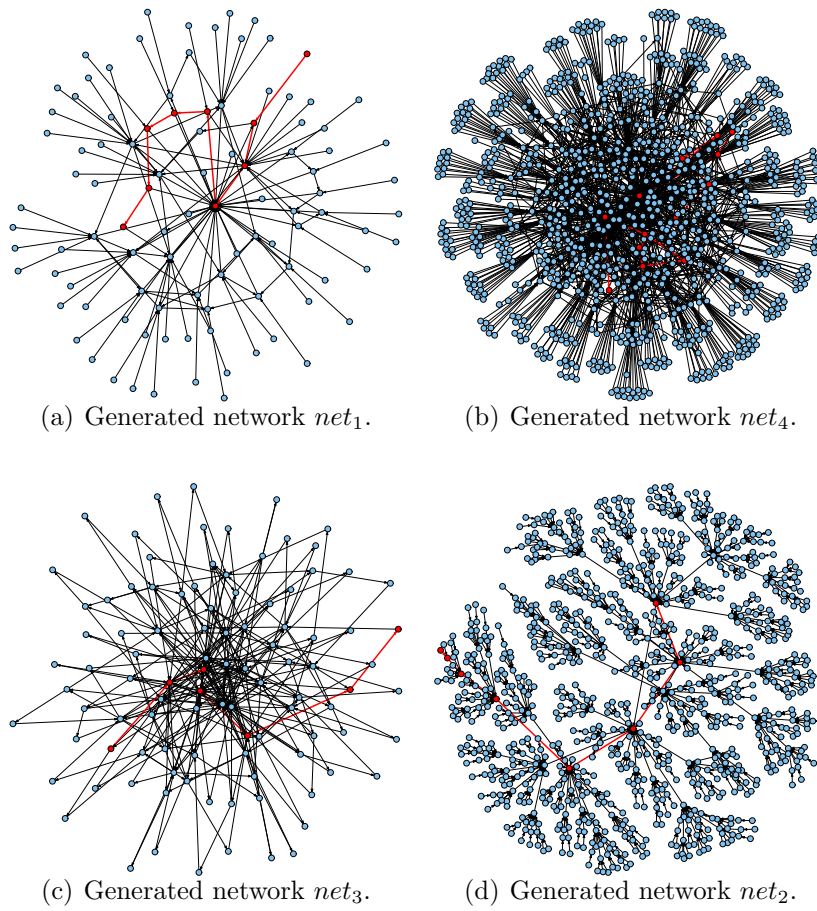


Fig. 2.8: Topology of net_i $i = 1, 2, 3, 4$. In red the diameter's path.

2.5 New definitions for edge-centrality and a preferential attachment generative algorithm

In this section we will describe a growing generation model based on the edge centrality.

The main idea is to use some measures of vertex centrality on \mathcal{G}^* , the dual graph of \mathcal{G} in order to implement a growing generation method based on edge centrality. In this model we add a fixed number of nodes in each step.

The proposed method follows these steps:

- Fix N the number of desired nodes, choose a centrality measure \mathcal{C} and define two selection rules S_e and S_v . Initialise \mathcal{G} as the connected graph with 3 nodes and 2 edges and let \mathcal{G}^* be its dual.
- At each step we add the i -th vertex and the $i - 1$ -th edge as follows:
 1. Compute the centrality of \mathcal{G}^* using \mathcal{C} and the selected edge e according to S_e .
 2. Select v , one of the two vertexes of e , according to S_v .
 3. Add to \mathcal{G} the vertex i and the edge (i, v) and update \mathcal{G}^* .

We apply this method using some simple selection functions. An edge e is chosen by S_e directly proportional to the score given by the centrality measure (in the same way as the preferential attachment described in [29]). Alternatively we can use non-linear attachment or inverse proportionality.

We perform some simulations using different choice for \mathcal{C} , S_e and S_v . We consider five kind of centrality measures: *PageRank*, *betweenness*, *closeness*, *eigen-vector* and *degree* two different selection functions S_e (**L** and **I**) and three for S_v (**R**, **noR** and **noR2**). For the definition of these last five functions we introduce some notation. Let $c = (x_1, \dots, x_n)$ be a vector of edge-centrality measures, i.e. x_i is the score for the i -th edge with $x_i \geq 0$ and define $W_i = \sum_{j \in A_i} x_j$ where $A_i = \{j \text{ s.t. } (j, i) \in \mathcal{G}\}$, in other words, W_i is the sum of all centrality-scores of the edge having i as vertex.

The function **L** returns the edge i with probability $x_i / \sum_{j=1}^N x_j$, conversely, **I** returns i with probability $\frac{1}{x_i+1} / \sum_{j=1}^N \frac{1}{x_j+1}$. Let now $e = (i, j)$ be an edge, the function **R** returns the vertex i and j with the same probability. A more complex function could be **noR** that selects i with probability equal to $p = W_i / (W_i + W_j)$ or **noR2** that selects i with probability $1 - p$. This method allows also to add more than one edge for each new vertex and the generalisation is quite simple.

In Fig. 2.9 we show some resulting networks and dual network. Notice that the acronyms has to be read as follows:

- $InoR$ $S_e = \mathbf{I}, S_v = \mathbf{noR}$, one edge for each vertex
- $InoR2$ $S_e = \mathbf{I}, S_v = \mathbf{noR2}$, one edge for each vertex
- $LnoR$ $S_e = \mathbf{L}, S_v = \mathbf{noR}$, one edge for each vertex
- $LnoR2$ $S_e = \mathbf{L}, S_v = \mathbf{noR2}$, one edge for each vertex
- IR $S_e = \mathbf{I}, S_v = \mathbf{R}$, one edge for each vertex
- LR $S_e = \mathbf{L}, S_v = \mathbf{R}$, one edge for each vertex
- $2InoR, 2InoR2, 2LnoR, 2LnoR2, 2IR, 2LR$ are the same of before but we add at most two edges for vertex
- $DInoR, DInoR2, DLnoR, DLnoR2, DIR, DLR, 2DInoR, 2DInoR2, 2DLnoR, 2DLnoR2, 2DIR, 2DLR$ are the dual graphs of the respective graphs.

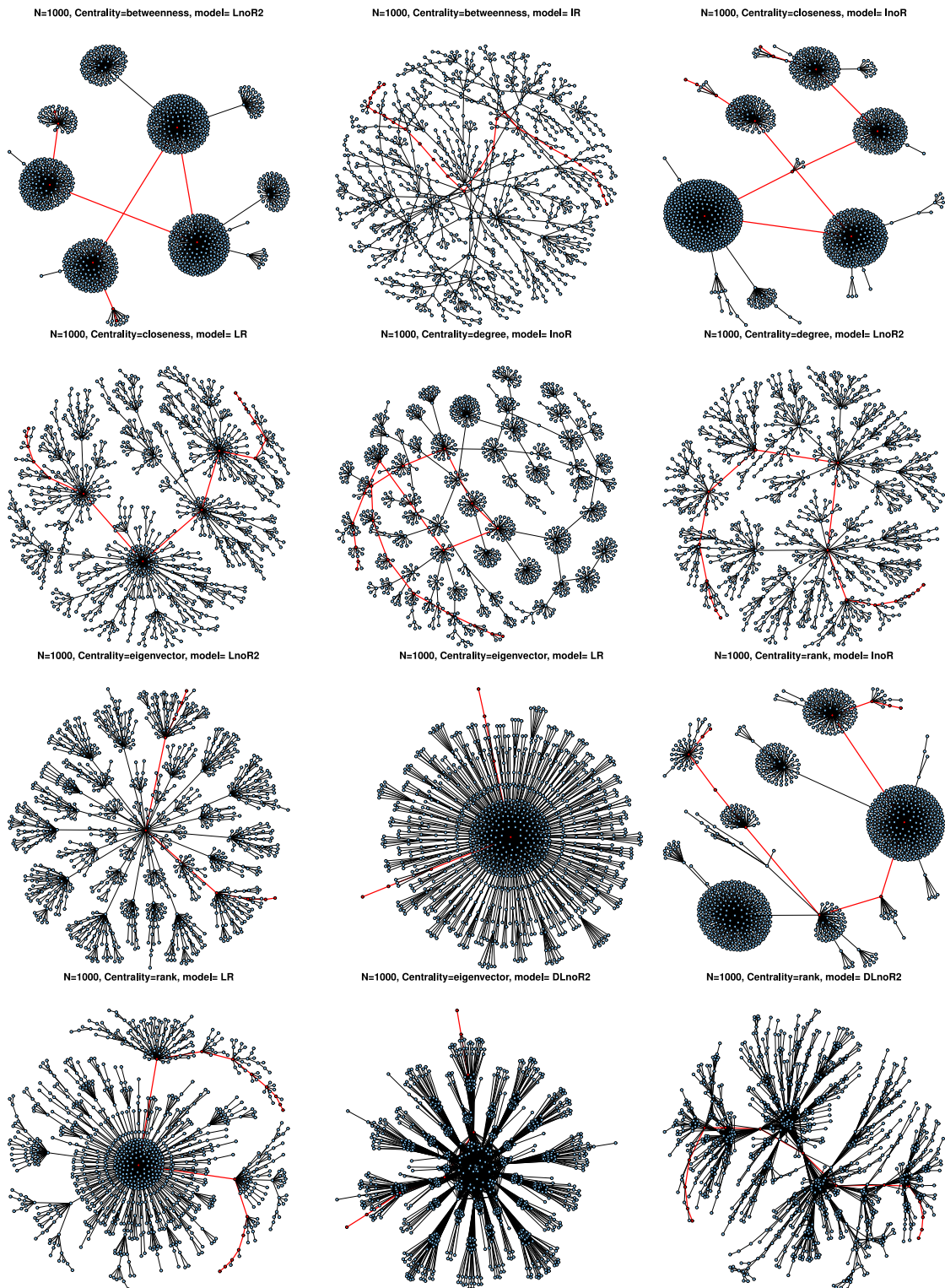


Fig. 2.9: Topology for the resulting graphs using betweenness edge centrality.

Chapter 3

Randomisation of graphs preserving the degree distribution

As we report in the previous chapter, generate a null model for a given graph is a fundamental task for the creation a baseline for modelling comparison. The algorithms showed above belong, in the majority, to the class of so-called **filling** algorithms (first the degree of each node is computed and finally the edge set is created). If the characteristic to be preserved is the degree distribution, the filling methods can drive into biased sample [34]. In order to avoid this, some **switching** methods has been introduced [35, 36, 37]. The importance to preserve the degree distribution, especially in the case of bipartite graph, is well discuss in a large number of recent papers [18, 38, 17, 39, 40, 41, 42, 43]. For example, in [18] the null model, build as a collection of M random version of the initial bigraph, is used for measuring the mutual exclusivity among genes to recognise groups of genes involved in the formation and proliferation of cancers. Other biological applications can be found in Mendelian diseases [44]; drugs and their targets [45, 46]; drugs and diseases [47]. Also in ecological research (see [34, 48, 49] for presence-absence of a certain species in a certain habitat and [50, 51] for pollinators-plant relations) these null models are widely used.

As we outline before, there are two main approaches for generating a random graph with a prescribed degree distribution: the switching algorithm and the filling algorithm. This last algorithm, for which an efficient implementation was developed by Patefield in [52], is based on the joint distribution probability read in the relative adjacency matrices. This method was been criticised [34] for the introduction of systematic biased sample procedure due to the not precise allocation of 1 in the matrices. For this reason several authors [35, 36, 37] prefer to use the switching algorithm (described in Section 3.1) composed by N simple switching steps. This number N is related to the burn-in time (mixing time), *i.e.* the time needed to *forget* the initial graph and typically empiric values are chosen ($100e$ in

[53] where e is the number of edges, $5e$ in [36], $10e$ up to $30e$ in [54]) and little has been rigorously shown in this direction [55]. The absence of trends in the time series of network metrics along the path of a Markov chain sampler has been proposed as a criterion for mixing [36], or at least to give some confidence that the final sampled networks are fairly random [54].

Due to the importance of this kind of null model in the case of bipartite graph (see 3.2 for an extended dissertation), in [27] we derive a theoretical bound for the number of switching steps N and in what follows we extend the result to undirected graphs (section 3.3) and then we will show (Prop. 7) that it is possible to obtain the results for bipartite graph as a special case.

We first introduce the **Jaccard Index (JI)** [56] between two graphs in order to measure the similarity between them. Then we will present the theoretical derivation for the fixed point \bar{x} (Lemma 3) of the JI through the SA and the number of required **Switching Steps (SS)** N in order to reach, on average, this fixed point (Prop. 5). Finally we will empirically show (in section 3.5) that our bound N can be chosen as time convergence (mixing time) for the underlying Markov chain.

This work is the result of a collaboration with the Julio-Saez Rodriguez group at the EBI of Cambridge, in which we derive such results for bipartite graphs.

We can measure how good is a mixing procedure using a **index of similarity** between two graphs with the same number of nodes n . A natural choice of similarity index could be the normalised Hamming distance between two incidence matrices. In biology is preferable to use the JI computed on the adjacency matrices $\mathcal{A} = \{y_{i,j}\}_{i,j=1}^n, \mathcal{B} = \{w_{i,j}\}_{i,j=1}^n$ of the two graphs using the so called **Tanimoto index** defined, in this specific case, as:

$$JI(\mathcal{A}, \mathcal{B}) = \frac{\sum_{i=1}^n \sum_{j=1}^i y_{i,j} \wedge w_{i,j}}{\sum_{i=1}^n \sum_{j=1}^i y_{i,j} \vee w_{i,j}} = \frac{\sum_{i=1}^n \sum_{j=1}^i y_{i,j} w_{i,j}}{\sum_{i=1}^n \sum_{j=1}^i (y_{i,j} + w_{i,j} - y_{i,j} w_{i,j})} \quad (3.1)$$

We will consider N sufficient for generating a random version of the initial graph \mathcal{G} if the mean value of $JI(\mathcal{G}, \mathcal{G}^{(N)})$ does not change, in mean, if we increase N . The condition is often checked [57] to solve the problem of monitoring convergence of the sampler quantifying the forgetting of the initial state. Moreover, we want also that also the pairwise similarity between two instances of the SA to be less similar than each of the two respect to the original. We will see that this first condition is satisfied whenever the first is satisfied.

3.1 The Switching Algorithm

Let \mathcal{G} be a graph. The SA is composed by N basic SS in which:

1. two edge (a, b) and (c, d) are uniformly and independently randomly selected;
2. If $a \neq c, a \neq d, b \neq c, b \neq d$:
 - If $(a, d), (c, b), (a, c)$ and (d, b) are not already in E :
 - with probability $p = 0.5$ the edges (a, d) and (c, b) are added to \mathcal{G} while (a, b) and (c, d) are removed.
 - with probability $q = 1 - p = 0.5$ the edges (a, c) and (d, b) are added to \mathcal{G} while (a, b) and (c, d) are removed.
 - else:
 - If (a, d) and (c, b) are not already in E the edges (a, d) and (c, b) are added to \mathcal{G} while (a, b) and (c, d) are removed.
 - else
 - * If (a, d) and (c, b) are not already in E the edges (a, d) and (c, b) are added to \mathcal{G} while (a, b) and (c, d) are removed.

It is clear that this algorithm preserve the degree distribution. In Fig. 3.1 we can see a schematic representation of the SA.

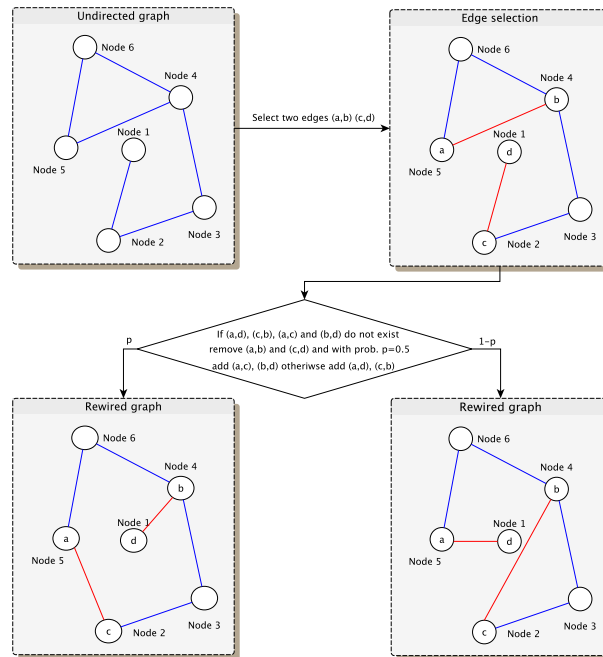


Fig. 3.1: Scheme of a SS in the SA.

3.2 Motivation: NGS data analysis

In the past few years, next generation sequencing (NGS) technologies have progressed very swiftly and currently hundreds of genomes can be simultaneously sequenced in a matter of weeks, at more affordable costs. This opens a wide range of new avenues in biological and biomedical research. In particular, due to the established impact of the genomic background on disease progression and response to drug treatment, cancer research has significantly benefited from these advances. Comprehensive catalogues of mutations in multiple cancer types have been assembled and fruitfully used to identify new diagnostic, prognostic and therapeutic targets [58, 59, 60, 61]. Existing large scale projects, such as the Cancer Genome Atlas (TCGA)[60], the International Cancer Genome Consortium (ICGC) data portal [58] and, more recently, the Genomics of Drug Sensitivity in Cancer (GDSC) [59] and the Cancer Cell Line Encyclopedia (CCLE) [61], provide invaluable opportunities to explore molecular alterations that could potentially play a crucial role in a plethora of different cancer types and their response to therapy [62]. A key task in these projects is to distinguish between driver mutations (i.e., those conferring selective clonal growth advantage) and functionally neutral passenger mutations (which do not contribute to tumour development) [63, 64]. Once key driver mutated genes are identified, a fruitful analysis is to consider them in the context of the pathways where they operate. This allows the identification of cancer driver biological networks, whose altered functionality results in the acquisition of a cancer hallmark [65, 66]. One of the ideas exploited to identify these networks is based on the assumption that sets of mutations exhibiting statistically significant levels of mutual exclusivity are likely to alter genes involved in a common biological process that drives cancer development. Hence, driver mutations in cancer occur in a limited number of pathways and lesions in the same pathway do not tend to occur in the same patient [38]. A possible biological explanation is that if a crucial node is altered in an oncogenic pathway, a secondary mutation on the same pathway is unlikely to provide further selective advantages to the cancer cell, thus it does not tend to be evolutionary selected. On the other hand, mutations of genes participating in different biological pathways may exert a synergistic effect in conferring growth advantages to tumour cells. As a consequence, the combinatorial effects of gene mutations may play key roles in cancer initiation and progression. Therefore investigations have been devoted to searching for groups of genes that are simultaneously mutated more often than expected by random chance [17, 39]. Based on these premises, the emergence of combinatorial properties among patterns of genomic events has been investigated in a number of recent studies, through the application of novel statistical measures quantifying, for example, the mutual exclusivity or the co-occurrence of different genomic lesions [38, 40, 43, 41, 18]. Among these studies, those aimed at identifying groups

of genes whose mutation patterns tend to mutual exclusivity are based on the same principle and are conceptually similar [14-16], although they differ in two crucial methodological aspects: (i) the way sets of genes to be tested for mutual exclusivity are selected and (ii) the way mutual exclusivity (ME) of a gene set is assessed and its statistical significance is quantified. To select candidate sets of genes for ME testing, Vandin et al [43] and Miller et al [41] adopted a data-driven approach, making use of genomic data only. In [43], authors search for sets of candidate sets of genes by solving a modified version of the Maximum Coverage Exclusive Submatrix (MCES) problem, where the objective is to maximise a weight function that specifies a trade-off between mutual exclusivity and coverage. In [15], authors used an online machine-learning algorithm to identify signal of exclusivity against the noisy background of passenger mutations in many irrelevant genes. Differently from the previous two methods, Ciriello et al [18] designed MEMo, a computational framework in which gene sets to be tested for ME are derived from cliques (i.e. groups of genes pair-wisely connected) identified in functional networks, assembled from publicly available signaling- and pathway-maps. For the statistical assessment of ME authors of these works follow heterogeneous strategies. After solving the MCES problem on relatively small datasets (containing few hundreds of samples and genes), authors of [43] perform a significance test simulating a null model by independently permuting the mutations of each gene across patients, thus preserving the mutation frequency gene-wisely (but not sample-wisely). In [41] authors make use of tools from coding theory and the ME significance for a set of genes is computed algorithmically, based on the minimal length of the code needed to compress the corresponding genomic data. This is based on the consideration that the genomic event sub-matrix composed by the patterns of mutations of a set of mutually exclusively mutated genes is less entropic than that of genes that tend to be co-mutated. As a consequence it should be easily compressible (i.e. by using a code of short length). In contrast to these two methods, MEMo [18] quantifies the sample coverage (SC) of a set of genes in terms of the number of samples in which at least one of them is mutated. Then the ME of the gene set under consideration is computed as the divergence of its SC from expectation. To this aim a null model is generated by randomly permuting the analysed dataset, while preserving the overall distribution of observed alterations across both genes and samples. This is crucial to preserve tumour specific alterations, heterogeneity in mutation/copy-number-alteration rates across patients, and to let the SC significance be proportional to the gene set ME. Compared to the methods in [43, 41], the functional relations occurring among a set of mutual-exclusive genes outputted by MEMo are more easily interpretable and the considered null model reflects more comprehensively the statistical properties of the analysed genomic dataset. In order to generate this null model, the authors make use of a per-

mutation strategy based on a random network generation model referred as the switching-algorithm [18]. Empirical p -values are then generated to estimate the significance of the deviation of the observed SC of each gene set from this null model.

NGS data are naturally represented as a bipartite graph in which the two classes of nodes are respectively the genes and the samples and an edge between these two classes indicates a mutation of the gene in the patient. Alternatively it is also used the incidence matrix of the bipartite graph a particular 0 – 1 table described in the first chapter.

In ecological research 0-1 tables, called presence-absence matrices (PAMs) [34], in which rows correspond to different species and columns to different habitats, are randomised to evaluate the deviation of observed phenomena, such as the co-occurrence of different species in the same habitat, from random expectations [67, 48, 68]. Several algorithms exist to generate constrained and non-constrained null models depending on which basic features of the PAM are retained in the computations [48]. In particular, a class of stochastic algorithms (i.e. swap and fill algorithms) generate null models in which the row-wise and column-wise sums of the PAM are preserved [49]. Nevertheless the randomisation of moderately large matrices in a short space of time is still challenging. To the aim of identifying novel cancer driver networks, Ciriello et al [18] took advantage of tools from graph theory by considering a BEM as the incidence matrix of a bipartite graph [69] (Figure 1 (B)). They adapted an algorithm for network randomisation with node degree preservation to the problem of randomising a BEM while preserving its row- and column-wise sums [53]. A bipartite graph (or network) is the natural abstraction of a set of objects and the relationships occurring among them. Bipartite graphs are a subset of networks in which the set of objects (i.e. vertices) are partitioned into two independent sets, so that within each set there are no connected nodes. Bipartite networks occur frequently in biology and in many other fields, and are widely used in bioinformatics and computational biology. Through bipartite networks it is possible to model ontologies with concepts and instances, simulations as Petri nets, biochemical reactions, and anchored maps for genomic mappings [70]. They are very useful in describing complex ecosystems as networks of interacting components and mutual interactions such as plants and their pollinators, plants and seed dispersers, prey and predators. In such situations, the study of the distribution of the number of links per species, or degree distribution, provides insights into the modeled system [50, 51]. In immunology, the immune reactions of a sample of patients to a panel of antigens can be represented as a bipartite graph. Data of unprecedented detail can now be obtained by applying serum sampled from patients to microarrays of purified antigens. For example, patients suffering from allergies can be screened against a large panel of putative allergens. The

resulting bipartite graph can be used as a starting point for the construction of a co-sensitisation graph on the set of antigens [71]. Further examples occur in molecular biology, involving high specificity recognition and signalling between various classes of macromolecules. In a recent work [32], bipartite graphs have been used to represent data on the regulation of protein expression by miRNAs (microRNAs). One set of nodes represents the miRNAs, and the other set of nodes represent the proteins. The presence of an arc between a miRNA node and a protein node indicates that the protein is regulated by the miRNA. The aim is then to use this data (in the form of a bipartite graph) to construct a co-regulation graph on the set of proteins. The criteria for including an arc between two nodes (proteins) in this new graph is based on a comparison of the number of shared miRNAs between the proteins in the observed bipartite graph, with the distribution of these numbers in randomly generated bipartite graphs. Additionally, many kinds of semantic and functional interactions can be easily represented through bipartite networks such those between genes and diseases to uncovers important properties of the nature of Mendelian diseases [44]; drugs and their targets [45, 46]; drugs and diseases [47].

3.3 Bound for the number of SS in the SA

Let $\mathcal{G} = (V, E)$ be an undirected network with e edges without loops and \mathcal{B} its $n \times n$ symmetric binary adjacency matrix; we can denote the node-set as $V = \{1, \dots, n\}$. In what follows, we indicate with $\mathbf{1}$ (respectively $\mathbf{0}$) the entries of a matrix (or vector) assuming value 1 (resp. 0). The number of edges in a complete undirected graph $\frac{n(n-1)}{2}$ will be indicated with t .

Let $\mathcal{B}^{(k)}$ be the adjacency matrix of $\mathcal{G}^{(k)}$ after k SS and $s^{(k)}$ the JI between \mathcal{B} and $\mathcal{B}^{(k)}$. Since each switching step does not alter the node degrees of \mathcal{G} , the total number of $\mathbf{1}$ in \mathcal{B} does not change, as well as its row- and column-wise sums, the JI in Eq. 3 reads:

$$s^{(k)} = JI(\mathcal{B}, \mathcal{B}^{(k)}) = \frac{x^{(k)}}{2e - x^{(k)}} \quad (3.2)$$

where $x^{(k)} \in \{n : n = 0, \dots, e\}$ is equal to the total the number of common edges in the two corresponding networks.

In Tab. 3.1 a scheme of the proof is provided.

1. Computation of the mean-field equation for $x^{(k+1)}$ and consequently for Eq. 3.2 (see Prop. 4 below).
2. Derivation of the fixed point \bar{x} and the convergence time N for the mean-field equation found in Prop. 4 (see Prop. 5 below).
3. Proof that the SA can be used to create null models for \mathcal{G} through N switching steps (see Prop. 6 and section 3.5 below).

Tab. 3.1: Proof Scheme

Let $p_r = P(PR)$ be the probability to perform a rewiring step (PR) and $d = \frac{e}{t}$ the edge density in the network \mathcal{G} .

Proposition 4

The mean-field equation for $x^{(k+1)}$ is equal to

$$x^{(k+1)} = mx^{(k)} + q = \frac{e - 2p_r - ed}{(d - 1)e} x^{(k)} + \frac{2edp_r}{(1 - d)e}. \quad (3.3)$$

Proof. After a switching step, turning $\mathcal{B}^{(k)}$ into $\mathcal{B}^{(k+1)}$, 5 possible values can be assumed by $x^{(k+1)}$:

1. $x^{(k+1)} = f_1(x^{(k)}) = x^{(k)} + 1$: unitary increment. The switching step is successfully performed (for exaple we rewire $(a, b), (c, d)$ with $(a, d), (c, b)$) and one of the following conditions is verified:
 - $(a, b), (c, d) \notin E$ and only one between (a, d) and (c, b) is in E ;
 - only one between (a, b) and (c, d) is in E and $(a, d), (c, b) \in E$.
2. $x^{(k+1)} = f_2(x^{(k)}) = x^{(k)} - 1$: unitary decrement. The rewiring step is successfully performed (for exaple we rewire $(a, b), (c, d)$ with $(a, d), (c, b)$) and one of the following conditions is verified:
 - $(a, b), (c, d) \in E$ and only one between (a, d) and (c, b) is in E ;
 - only one between (a, b) and (c, d) is in E and $(a, d), (c, b) \notin E$.
3. $x^{(k+1)} = f_3(x^{(k)}) = x^{(k)} + 2$: maximal increment. The rewiring step is successfully performed (for example we rewire $(a, b), (c, d)$ with $(a, d), (c, b)$) and $(a, b), (c, d) \notin E$ while $(a, d), (c, b) \in E$.
4. $x^{(k+1)} = f_4(x^{(k)}) = x^{(k)} - 2$: maximal decrement. The rewiring step is successfully performed (for example we rewire $(a, b), (c, d)$ with $(a, d), (c, b)$) and $(a, b), (c, d) \in E$ while $(a, d), (c, b) \notin E$.
5. $x^{(k+1)} = f_5(x^{(k)}) = x^{(k)}$: null variation. Otherwise.

Tab.3.2 contains a summary of the five possible values assumable by $x^{(k+1)}$.

f_1	f_2	f_3	f_4	f_5
+1	-1	+2	-2	+0

Tab. 3.2: Possible values of $x^{(k+1)}$.

If we indicate with $p_i^{(k)} = P(x^{(k+1)} = f_i(x^{(k)}))$ (i.e. probability of each case, for $i = 1, \dots, 5$), then $x^{(k+1)}$ is equal, on average, to:

$$x^{(k+1)} = \sum_{i=1}^5 p_i^{(k)} f_i(x^{(k)}). \quad (3.4)$$

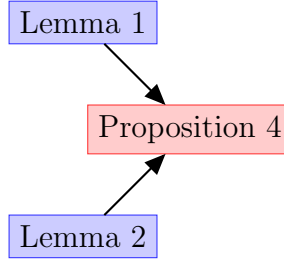


Fig. 3.2: Scheme for the proof of Prop.4.

Explicating the $p_i^{(k)}$ for $i = 1, \dots, 5$ (see Lemma 1 and Lemma 2 below) reduces Eq. 3.4 to Eq. 3.3.

The scheme of this proof is summarized in Fig. 3.2

□

In order to prove Lemma 1 and Lemma 2, we will make use of the following additional notation.

Let us consider now a, b, c, d defined in step 2 of the SA and $w_{i,j}^{(k)}$ the i, j -th element of the adjacency matrix of the graph $\mathcal{B}^{(k)}$.

With $\begin{pmatrix} 0 & \alpha & \beta & \gamma \\ \alpha & 0 & \delta & \sigma \\ \beta & \delta & 0 & \tau \\ \gamma & \sigma & \tau & 0 \end{pmatrix}^{(k)}$ we will indicate the submatrix of $\mathcal{B}^{(k)}$ collecting the sixteen positions $\alpha = w_{a,b}^{(k)}$, $\beta = w_{a,c}^{(k)}$, $\gamma = w_{a,d}^{(k)}$, $\delta = w_{b,c}^{(k)}$, $\sigma = w_{b,d}^{(k)}$, $\tau = w_{c,d}^{(k)}$. In what follow, when an entry of the $\begin{pmatrix} 0 & \alpha & \beta & \gamma \\ \alpha & 0 & \delta & \sigma \\ \beta & \delta & 0 & \tau \\ \gamma & \sigma & \tau & 0 \end{pmatrix}^{(k)}$ can be neglected then it will be indicated

with the \cdot symbol. When $k = 0$ we denote $\begin{pmatrix} 0 & \alpha & \beta & \gamma \\ \alpha & 0 & \delta & \sigma \\ \beta & \delta & 0 & \tau \\ \gamma & \sigma & \tau & 0 \end{pmatrix}^{(0)}$ with $\begin{pmatrix} 0 & \alpha & \beta & \gamma \\ \alpha & 0 & \delta & \sigma \\ \beta & \delta & 0 & \tau \\ \gamma & \sigma & \tau & 0 \end{pmatrix}$. Since, for practical and notation purpose, all these values are not required, we will use a more compact notation. Let suppose, w.l.o.g, that $\begin{pmatrix} 0 & 1 & \cdot & 0 \\ 1 & 0 & 0 & \cdot \\ \cdot & 0 & 0 & 1 \\ 0 & \cdot & 1 & 0 \end{pmatrix}^{(k)}$, i.e. PR can be

perform and $\begin{pmatrix} 0 & 0 & \cdot & 1 \\ 0 & 0 & 1 & \cdot \\ \cdot & 1 & 0 & 0 \\ 1 & \cdot & 0 & 0 \end{pmatrix}^{(k+1)}$, i.e. the SA rewires $(a, b), (c, d)$ with $(a, d), (c, b)$. This situation can be easily described using a 4×4 matrix: $\begin{pmatrix} \alpha & \gamma \\ \delta & \tau \end{pmatrix}^{(k)}$ and so in this case $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}$ and $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^{(k+1)}$. Also in this case, when $k = 0$, we denote $\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}^{(0)}$ with $\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$.

Let we introduce four more probabilities and eighth new events:

$$\begin{aligned}
q_s^{(k)} &= P(QS_k^+) = P\left(\begin{pmatrix} 1 \\ \vdots \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right) = P(QS_k^-) = P\left(\begin{pmatrix} \vdots \\ 1 \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right), \\
p_s^{(k)} &= P(PS_k^+) = P\left(\begin{pmatrix} 0 \\ \vdots \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right) = P(PS_k^-) = P\left(\begin{pmatrix} \vdots \\ 0 \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right), \\
q_f^{(k)} &= P(QF_k^+) = P\left(\begin{pmatrix} \vdots \\ 1 \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right) = P(QF_k^-) = P\left(\begin{pmatrix} 1 \\ \vdots \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right), \\
p_f^{(k)} &= P(PF_k^+) = P\left(\begin{pmatrix} \vdots \\ 0 \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right) = P(PF_k^-) = P\left(\begin{pmatrix} 0 \\ \vdots \end{pmatrix} \middle| \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)}\right).
\end{aligned}$$

For example, the value $q_s^{(k)}$ is the probability of having $w_{a,b} = 1 = w_{c,d}$ in the initial graph knowing that the rewiring step is performed. The other events and probabilities have similar interpretations.

Lemma 1

In the above notation:

$$q_s^{(k)} \simeq \frac{x^{(k)}}{e}, \quad p_s^{(k)} \simeq \frac{e - x^{(k)}}{e}, \quad q_f^{(k)} \simeq \frac{e - x^{(k)}}{t - e}, \quad p_f^{(k)} \simeq \frac{t - 2e + x^{(k)}}{t - e}.$$

Proof. Let us suppose that at the step k there are $x^{(k)}$ ones in common between $\mathcal{B}^{(k)}$ and \mathcal{B} , and that $w_{a,b}^{(k)} = 1$, then the probability that in the initial graph $w_{a,b} = 1$ is $\frac{x^{(k)}}{e}$ (positive cases divided by possible cases). Similarly, for $q_f^{(k)}$ the possible cases are $t - e$, i.e. the number of available position in which the new non null entry can be placed, and the positive cases are $e - x^{(k)}$; then

$$q_f^{(k)} \simeq \frac{e - x^{(k)}}{t - e}. \quad (3.5)$$

All the approximations above follow from the simplifications $x^{(k)} - 1 \sim x^{(k)}$ and $e - 1 \sim e$. The rest of the proof can be deduced observing that $p_s^{(k)} = 1 - q_s^{(k)}$ and $p_f^{(k)} = 1 - q_f^{(k)}$.

□

Lemma 2

The probabilities $p_i^{(k)}$, $i = 1, \dots, 5$ are equal to:

$$p_1^{(k)} \simeq \frac{2(x^{(k)} - e)^3(2e - 2x^{(k)} - t)}{(e^2 - et)^2} p_r, \quad p_4^{(k)} \simeq \frac{x^{(k)}(x^{(k)} + t - 2e)^2 x^{(k)}}{(te - e^2)^2} p_r,$$

$$p_2^{(k)} \simeq \frac{2(e - x^{(k)})(x^{(k)} + t - 2e)(2x^{(k)} + t - 2e)x^{(k)}}{(te - e^2)^2} p_r, \quad p_3^{(k)} \simeq \frac{(x^{(k)} - e)^4}{(te - e^2)^2} p_r,$$

$$p_5^{(k)} = 1 - p_4^{(k)} - p_3^{(k)} - p_2^{(k)} - p_1^{(k)}.$$

Proof. Using the definition of $f_1(x^{(k)})$ in Prop.4 and the four probabilities in Fact.1, it follows that:

$$p_1^{(k)} = P(PR \wedge ((PS_k^+ \wedge PS_k^-) \wedge ((QF_k^+ \wedge PF_k^-) \vee (QF_k^- \wedge PF_k^+))) \vee (((QS_k^+ \wedge PS_k^-) \vee (QS_k^- \wedge PS_k^+)) \wedge (QF_k^+ \wedge QF_k^-))).$$

This can be rewritten (omitting the probabilities of the prior events, for sake of simplicity) as:

$$\begin{aligned} p_1^{(k)} &= P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left\{ \left[\begin{pmatrix} 0 & \cdot \\ \cdot & \cdot \end{pmatrix} \wedge \left[\begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 0 \end{pmatrix} \right] \vee \left[\left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 1 \end{pmatrix} \right] \wedge \begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \right\} \right] \\ &= P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left[\begin{pmatrix} 0 & \cdot \\ \cdot & \cdot \end{pmatrix} \wedge \left[\begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 1 \end{pmatrix} \right] \right] + P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left[\left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 1 \end{pmatrix} \right] \wedge \begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \right] \right] \\ &\simeq p_r \left[p_s^2(1 - p_f^2 - q_f^2) + (1 - p_s^2 - q_s^2)q_f^2 \right]. \end{aligned}$$

Similarly:

$$\begin{aligned} p_2^{(k)} &= P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left\{ \left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \wedge \left[\begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 0 \end{pmatrix} \right] \vee \left[\left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 1 \end{pmatrix} \right] \wedge \begin{pmatrix} \cdot & \cdot \\ \cdot & 0 \end{pmatrix} \right\} \right] \\ &= P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \wedge \left[\begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 0 \end{pmatrix} \right] \right] + P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \left[\left[\begin{pmatrix} 1 & \cdot \\ \cdot & \cdot \end{pmatrix} \vee \begin{pmatrix} \cdot & \cdot \\ \cdot & 1 \end{pmatrix} \right] \wedge \begin{pmatrix} \cdot & \cdot \\ \cdot & 0 \end{pmatrix} \right] \right] \\ &\simeq p_r \left[q_s^2(1 - p_f^2 - q_f^2) + (1 - p_s^2 - q_s^2)p_f^2 \right]. \end{aligned}$$

$$p_3^{(k)} = P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \begin{pmatrix} 0 & \cdot \\ \cdot & \cdot \end{pmatrix} \wedge \begin{pmatrix} \cdot & 1 \\ \cdot & \cdot \end{pmatrix} \right] \simeq p_r p_s^2 q_f^2.$$

$$p_4^{(k)} = P \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}^{(k)} \wedge \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} \wedge \begin{pmatrix} & 0 \\ 0 & \end{pmatrix} \right] \simeq p_r q_s^2 p_f^2.$$

□

The mean-field equation for the Tanimoto index $s^{(k+1)}$ resulting combining Eq.3.2 and Eq.3.3 reads:

$$s^{(k+1)} = \frac{e^2 x^{(k)} + (2x^{(k)}t - 2e^2)p_r - ex^{(k)}t}{2e^3 + (2e^2 - 2x^{(k)}t)p_r - e^2x^{(k)} - 2e^2t + ex^{(k)}t - 2p_r x^{(k)}t}. \quad (3.6)$$

The mean-field equation Eq. 3.6 is an approximation because Eq. 3.5 does not consider the preservation of the degree distributions. To take this constrain into account, we slightly modify Eq. 3.5 as follows:

$$q_f^{(k)} \simeq \frac{e - x^{(k)}}{t - e - z}, \quad (3.7)$$

where $t - e - z$ represents the number of *available positions* where the new non null entry can be placed. The value z depends on the initial graph \mathcal{G} that can be neglected (as explained in the demonstration of Prop. 5).

If reformulating Lemma 1, Lemma 2 and Prop. 4 accordingly to this modification the mean-field equation for $x^{(k+1)}$ is equal to:

$$\begin{aligned} x^{(k+1)} &= m(z)x^{(k)} + q(z) = \\ &= \frac{et - 2p_r(t - z) - e^2 - ez}{(t - e - z)e} x^{(k)} + \frac{2e^2 p_r}{(t - e - z)e}. \end{aligned} \quad (3.8)$$

The demonstration of Prop. 5 follows from Prop. 4, Lemma 3 and Lemma 4 (as summarized in Fig.3.3).

Lemma 3

The unique fixed point \bar{x} of Eq.3.8 is:

$$\bar{x} = \frac{e^2}{t - z}. \quad (3.9)$$

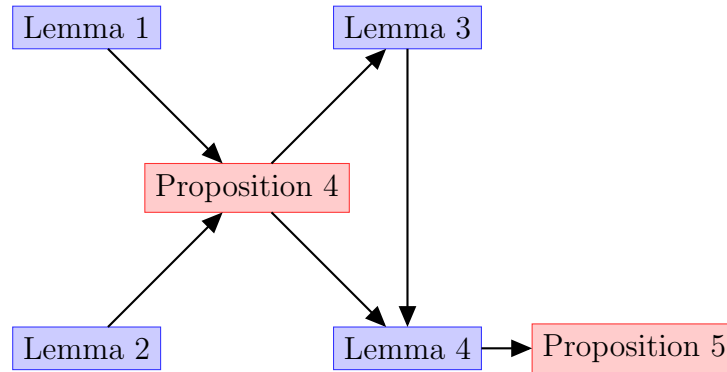


Fig. 3.3: Scheme for the proof of Prop.5.

Proof. Let us solve $x^{(k+1)} = x^{(k)} = \bar{x}$:

$$\begin{aligned}
 0 &= p_1(\bar{x})(\bar{x} + 1) + p_2(\bar{x})(\bar{x} - 1) + p_3(\bar{x})(\bar{x} + 2) + p_4(\bar{x})(\bar{x} - 2) + p_5(\bar{x})\bar{x} - \bar{x} \\
 &= p_1(\bar{x}) - p_2(\bar{x}) + 2p_3(\bar{x}) - 2p_4(\bar{x}) \\
 &= \frac{2(e^2 + \bar{x}z - \bar{x}t)p_r}{(t - e - z)e} \\
 &= \bar{x}z - \bar{x}t + e^2 \\
 \bar{x} &= \frac{e^2}{t - z}.
 \end{aligned}$$

□

Lemma 4

Fixed a positive real number $\epsilon \leq 1$, then $|x^{(k)} - \bar{x}| < \epsilon$ for all $k > N$ with

$$N = \log_{m(z)} g(z, \epsilon) \quad \text{with} \quad g(z, \epsilon) = \frac{\epsilon(t - z)}{(t - e - z)e}.$$

Proof. From Eq.3.8 it follows that:

$$\begin{aligned}
x^{(k+1)} &= m(z)x^{(k)} + q \\
&= (m(z) + 1)x^{(k)} - m(z)x^{(k-1)}, \quad \text{that is a second-order linear recursive sequence admitting} \\
&F(x) = x^2 - (m(z) + 1)x + m(z) \quad \text{as characteristic polynomial. As shown in [72] we can write} \\
x^{(k+1)} &= ar^{k+1} + bs^{k+1}, \quad \text{where } r \text{ and } s \text{ are the two roots of } F \text{ and } a \text{ and } b \text{ are constants} \\
&= am(z)^{k+1} + b, \quad \text{in our case } r = m(z), s = 1, \\
&= \left(e - \frac{q(z)}{1 - m(z)} \right) m(z)^{k+1} + \frac{q(z)}{1 - m(z)} \tag{3.10}
\end{aligned}$$

given that $x^{(0)} = e$ and $x^{(1)} = m(z)e + q(z)$.

Fixed $\epsilon \leq 1$,

$$\begin{aligned}
|x^{(N)} - \bar{x}| < \epsilon &\iff \left| \left(e - \frac{q(z)}{1 - m(z)} \right) m(z)^k \right| < \epsilon \iff \\
N > \log_{m(z)} g(z, \epsilon) &\quad \text{with} \quad g(z, \epsilon) = \frac{\epsilon(t - z)}{(t - e - z)e}. \tag{3.11}
\end{aligned}$$

Since $0 < m(z) \leq 1$ the previous inequality holds. \square

Proposition 5

Let d denotes the edge density of \mathcal{G} , namely $d = \frac{e}{t} \in [0, 1]$ and $\epsilon = 1$, then N is equal to:

$$\frac{e(1 - d)}{2p_r} \ln(e - de). \tag{3.12}$$

Proof. Since $m'(z) = \frac{-2p_r}{(e+z-t)^2} < 0$ and $\frac{\partial}{\partial z} g(z, \epsilon) = -\frac{(t-z)^2}{e^2} < 0$, the maximum value for N of Eq. 3.11 is reached for $z = 0$ and its value is:

$$\begin{aligned}
N &= \log_{\frac{et - 2p_r(t-z) - e^2 - ez}{(t-e-z)e}} \frac{t^2}{t^2e - e^2t} \\
&= \log_{1 + \frac{2p_r t}{(e-t)e}} \frac{1}{e - de} \\
&= \frac{\ln \frac{1}{e - de}}{\ln 1 + \frac{2p_r t}{(e-t)e}} \\
&\sim \frac{(t - e)e}{2p_r t} \ln(e - de) \quad \text{using } \ln[1 + x] \sim x \text{ for } |x| < 1 \\
&= \frac{(1 - d)e}{2p_r} \ln(e - de).
\end{aligned}$$

□

Remark 1. For an implementative point of view, the value p_r is not important because we can count only the number of SSs correctly perform i.e. consider $p_r = 1$. For some sparse and regular graphs (see [27]) it is possible to compute p_r but for a general graph this probability depends strongly by the degree distribution.

3.4 Pairwise-similarity

Let $r^{(k)} = s(\mathcal{B}^{(k)}, \mathcal{C}^{(k)})$ where $\mathcal{B}^{(k)}$ and $\mathcal{C}^{(k)}$ are the adjacency matrices of two rewired version of \mathcal{G} , obtained through the SA with at the k -th SSs. In this section we will show that the similarity between any pair of rewired versions of \mathcal{G} obtained through different instances of the SA, with k SSs, is lower than their individual similarity to \mathcal{G} .

Lemma 5

Using the same notation and Prop. 4 and Prop. 5, with $z = 0$ it follows that:

$$r^{(k+1)} = \bar{m}r^{(k)} + \bar{q} = \frac{et - e^2 - 4p_r t}{te - e^2} r^{(k)} + \frac{4e^2 p_r}{te - e^2}.$$

Proof. Similarly to Prop. 4 the value $r^{(k+1)}$ can be estimated as:

$$r^{(k+1)} = \sum_{i=1}^9 q_i^{(k)} g_i(r^{(k)}),$$

where the values of g_i are listed below and summarise in Tab.3.3.

g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9
+4	-4	+3	-3	+2	-2	+1	-1	+0

Tab. 3.3: Possibilities for $r^{(k+1)}$.

Using the letters a, b, c, d for $\mathcal{B}^{(k)}$ and $\alpha, \beta, \gamma, \delta$ for $\mathcal{C}^{(k)}$ and introducing $F^{(k)}$ as the set of the common edges between $\mathcal{B}^{(k)}$ and $\mathcal{C}^{(k)}$ and using $(\cdot)^{(k)}$ instead of $\begin{pmatrix} 0 & \dots \\ \dots & 0 \end{pmatrix}^{(k)}$ we have:

1. $g_1(r^{(k)}) = r^{(k)} + 4$: we gain four ones. The two rewiring steps are performed (one for $\mathcal{B}^{(k)}$ and one for $\mathcal{C}^{(k)}$) and $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \notin F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.

2. $g_2(r^{(k)}) = r^{(k)} - 4$: we lose four ones. The two rewiring steps are performed and $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \in F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.
3. $g_3(r^{(k)}) = r^{(k)} + 3$: we gain three ones. The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \notin F^{(k)}$ and only three among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ are elements of $F^{(k)}$ or
 - One among $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta)$ is in $F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.
4. $g_4(r^{(k)}) = r^{(k)} - 3$: we lose three ones. The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \in F^{(k)}$ and only one among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ is a element of $F^{(k)}$ or
 - Three among $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta)$ are in $F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.
5. $g_5(r^{(k)}) = r^{(k)} + 2$: we gain two ones.
 - The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \notin F^{(k)}$ and only two among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ are elements of $F^{(k)}$.
 - $(a, b) \in F^{(k)}$ (or one of the other) and
 - * if $(a, d) \in F^{(k)}$ two among $(c, b), (\alpha, \delta), (\gamma, \beta)$ are in $F^{(k)}$,
 - * if $(a, d) \notin F^{(k)}$ $(c, b), (\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.
 - $(a, b)(c, d) \in F^{(k)}$ (or any other couple) and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.
 - Only one of the two rewiring steps are performed (let say $\mathcal{B}^{(k)}$) and:
 - $(a, b), (c, d) \notin F^{(k)}$ and $(a, d), (c, b) \in F^{(k)}$.
6. $g_6(r^{(k)}) = r^{(k)} - 2$: we lose two ones.
 - The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \in F^{(k)}$ and only two among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ are elements of $F^{(k)}$.
 - $(a, b) \notin F^{(k)}$ (or one of the other) and
 - * if $(a, d) \notin F^{(k)}$ one among $(c, b), (\alpha, \delta), (\gamma, \beta)$ is in $F^{(k)}$,
 - * if $(a, d) \in F^{(k)}$ $(c, b), (\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.

– $(a, b)(c, d) \notin F^{(k)}$ (or any other couple) and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.

- Only one of the two rewiring steps are performed (let say $\mathcal{B}^{(k)}$) and:
 - $(a, b), (c, d) \in F^{(k)}$ and $(a, d), (c, b) \notin F^{(k)}$.

7. $g_7(r^{(k)}) = r^{(k)} + 1$: we gain a one.

- The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \notin F^{(k)}$ and only one among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ is an element of $F^{(k)}$.
 - $(a, b) \in F^{(k)}$ (or one of the other) and
 - * if $(a, d) \in F^{(k)}$ one among $(c, b), (\alpha, \delta), (\gamma, \beta)$ is in $F^{(k)}$,
 - * if $(a, d) \notin F^{(k)}$ two among $(c, b), (\alpha, \delta), (\gamma, \beta)$ are in $F^{(k)}$.
 - $(a, b)(c, d) \in F^{(k)}$ (or any other couple) and
 - * if $(a, d), (c, b) \in F^{(k)}$ one among $(\alpha, \delta), (\gamma, \beta)$ is in $F^{(k)}$,
 - * if $(a, d) \in F^{(k)}$ and $(c, b) \notin F^{(k)}$ (or viceversa) $(\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.
 - Three among $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta)$ are in $F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \in F^{(k)}$.
- Only one of the two rewiring steps are performed (let say $\mathcal{B}^{(k)}$) and:
 - $(a, b), (c, d) \notin F^{(k)}$ and only one among (a, d) and (c, b) is an element of $F^{(k)}$ or
 - $(a, b) \in F^{(k)}, (c, d) \notin F^{(k)}$ and $(a, d), (c, b) \in F^{(k)}$.

8. $g_8(r^{(k)}) = r^{(k)} - 1$: we lose a one.

- The two rewiring steps are performed and:
 - $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta) \in F^{(k)}$ and three among $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta)$ are elements of $F^{(k)}$.
 - One among $(a, b), (c, d), (\alpha, \beta), (\gamma, \delta)$ is in $F^{(k)}$ and $(a, d), (c, b), (\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.
 - $(a, b)(c, d) \in F^{(k)}$ (or any other couple) and
 - * if $(a, d), (c, b) \notin F^{(k)}$ one among $(\alpha, \delta), (\gamma, \beta)$ is in $F^{(k)}$,
 - * if $(a, d) \in F^{(k)}$ and $(c, b) \notin F^{(k)}$ (or viceversa) $(\alpha, \delta), (\gamma, \beta) \notin F^{(k)}$.
 - $(a, b) \notin F^{(k)}$ (or one of the other) and
 - * if $(a, d) \notin F^{(k)}$ two among $(c, b), (\alpha, \delta), (\gamma, \beta)$ are in $F^{(k)}$,
 - * if $(a, d) \notin F^{(k)}$ one among $(c, b), (\alpha, \delta), (\gamma, \beta)$ is in $F^{(k)}$.

- Only one of the two rewiring steps are performed (let say $\mathcal{B}^{(k)}$) and:
 - $(a, b), (c, d) \in F^{(k)}$ and only one among (a, d) and (c, b) is an element of $F^{(k)}$ or
 - $(a, b) \in F^{(k)}, (c, d) \notin F^{(k)}$ and $(a, d), (c, b) \notin F^{(k)}$.

9. $g_9(r^{(k)}) = r^{(k)}$: no variation.

The rest of proof follows from the explication of the probabilities $q_i^{(k)}$ $i = 1, \dots, 9$ given in Lemma 6, as summarised in Fig. 3.4).

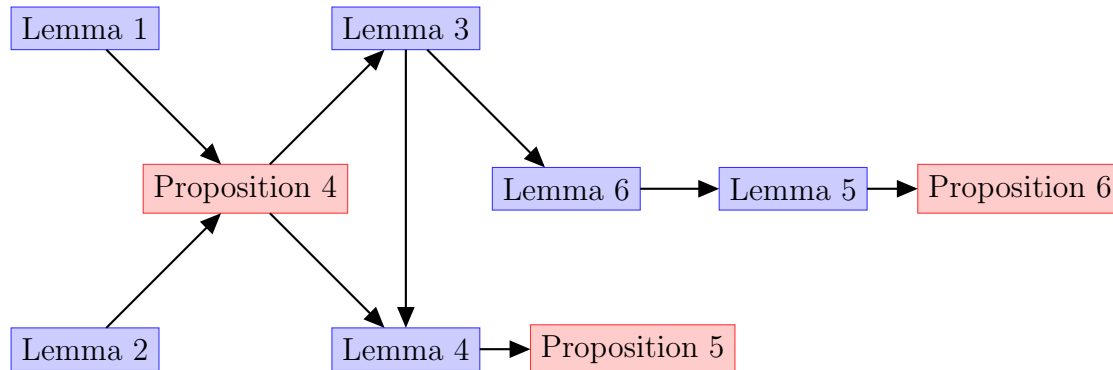


Fig. 3.4: Scheme for the proof of Prop.6.

□

Lemma 6

From the definition of the probabilities in Lemma 3 we can compute

$q_1^{(k)}, i = 1, \dots, 9$:

$$q_1^{(k)} \sim \frac{(e - r^{(k)})^8 p_r^2}{(e^2 - te)^4}.$$

$$q_2^{(k)} \sim \frac{r^{(k)^4} (2e - r^{(k)} - t)^4 p_r^2}{(e^2 - te)^4}.$$

$$q_3^{(k)} \sim \frac{4 * (r^{(k)} - e)^7 (2e - 2r^{(k)} - t) p_r^2}{(e^2 - te)^4}.$$

$$q_4^{(k)} \sim \frac{4(e - r^{(k)})(2e - 2r^{(k)} - t) (r^{(k)}(2e - r^{(k)} - t))^3 p_r^2}{(e^2 - te)^4}.$$

$$q_5^{(k)} \sim \frac{2(e - r^{(k)})^4 (11e^4 p_r - 52e^3 p_r r^{(k)} - 10e^3 p_r t + 82e^2 p_r r^{(k)^2} + 38e^2 p_r r^{(k)} t + 2e^2 p_r t^2 - 56e p_r r^{(k)^3} - 40e p_r r^{(k)^2} t - 6e p_r r^{(k)} t^2 + 14p_r r^{(k)^4} + 14p_r r^{(k)^3} t + 3p_r r^{(k)^2} t^2 + e^4 - 2e^3 t + e^2 t^2) p_r}{(e^2 - te)^4}.$$

$$q_6^{(k)} \sim \frac{2(2e - r^{(k)} - t)^2 (11e^4 p_r - 52e^3 p_r r^{(k)} - 10e^3 p_r t + 82e^2 p_r r^{(k)^2} + 38e^2 p_r r^{(k)} t + 2e^2 p_r t^2 - 56e p_r r^{(k)^3} - 40e p_r r^{(k)^2} t - 6e p_r r^{(k)} t^2 + 14p_r r^{(k)^4} + 14p_r r^{(k)^3} t + 3p_r r^{(k)^2} t^2 + e^4 - 2e^3 t + e^2 t^2) p_r r^{(k)^2}}{(e^2 - te)^4} +$$

$$q_7^{(k)} \sim \frac{-4(e - r^{(k)})^3 (2e - 2r^{(k)} - t) (3e^4 p_r - 22e^3 p_r r^{(k)} - 2e^3 p_r t + 39e^2 p_r r^{(k)^2} + 15e^2 p_r r^{(k)} t - 28e p_r r^{(k)^3} - 18e p_r r^{(k)^2} t - 2e p_r r^{(k)} t^2 + 7p_r r^{(k)^4} + 7p_r r^{(k)^3} t + p_r r^{(k)^2} t^2 + e^4 - 2e^3 t + e^2 t^2) p_r}{(e^2 - te)^4} +$$

$$q_8^{(k)} \sim \frac{4(e - r^{(k)})(2e - 2r^{(k)} - t)(2e - r^{(k)} - t) (3e^4 p_r - 22e^3 p_r r^{(k)} - 2e^3 p_r t + 39e^2 p_r r^{(k)^2} + 15e^2 p_r r^{(k)} t - 28e p_r r^{(k)^3} - 18e p_r r^{(k)^2} t - 2e p_r r^{(k)} t^2 + 7p_r r^{(k)^4} + 7p_r r^{(k)^3} t + p_r r^{(k)^2} t^2 + e^4 - 2e^3 t + e^2 t^2) p_r r^{(k)}}{(e^2 - te)^4}.$$

$$q_9^{(k)} \sim 1 - \sum_{i=1}^8 q_i^{(k)}.$$

Proof. A key point for the calculation of these probabilities is that the number of admissible configuration should be correctly enumerated. As an example, to compute $q_7^{(k)}$, i.e. probability of unitary increment, factors 4 and 24 are defined considering that if originally all the four selected edges are not in $F^{(k)}$ we gain a one if and only if one of the rewired edge is a element of the set, and there are

exactly 4 configurations possible since the rewired edge in the set could be one of the four possible edges. If we are in the second case the possible configurations are summarised in Fig.3.5

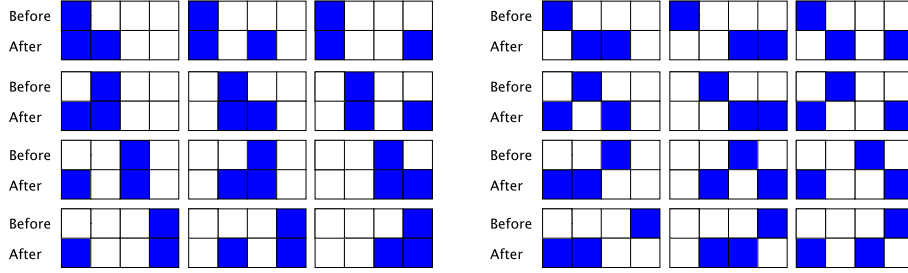


Fig. 3.5: All the 24 configurations producing a unitary increment in the second case. The four columns represent the edges and those in blue are elements of $F^{(k)}$. The first row of each bloc represents the configuration before the rewiring step, while the second one represents the situation after the step.

Similarly to the proof of Prop.4:

$$q_1^{(k)} = P(PR_b \wedge PR_c \wedge (PS_b^+ \wedge PS_b^- \wedge PS_b^+ \wedge PS_b^-) \wedge (QF_c^+ \wedge QF_c^- \wedge QF_c^+ \wedge QF_c^-)) \sim p_r^2 p_s^4 q_f^4.$$

$$q_2^{(k)} = P(PR_b \wedge PR_c \wedge (QS_b^+ \wedge QS_b^- \wedge QS_b^+ \wedge QS_b^-) \wedge (PF_c^+ \wedge PF_c^- \wedge PF_c^+ \wedge PF_c^-)) \sim p_r^2 q_s^4 p_f^4.$$

$$q_3^{(k)} = P(PR_b \wedge PR_c \wedge ((PS_b^+ \wedge PS_b^- \wedge PS_b^+ \wedge PS_b^-) \wedge ((QF_c^+ \wedge QF_c^- \wedge QF_c^+ \wedge PF_c^-) \vee (QF_c^+ \wedge QF_c^- \wedge PF_c^+ \wedge QF_c^-) \vee (QF_c^+ \wedge PF_c^- \wedge QF_c^+ \wedge QF_c^-) \vee (PF_c^+ \wedge QF_c^- \wedge QF_c^+ \wedge QF_c^-)) \vee ((PS_c^+ \wedge PS_c^- \wedge PS_c^+ \wedge QS_c^-) \vee (PS_c^+ \wedge PS_c^- \wedge QS_c^+ \wedge PS_c^-) \vee (PS_c^+ \wedge QS_c^- \wedge PS_c^+ \wedge PS_c^-) \vee (QS_c^+ \wedge PS_c^- \wedge PS_c^+ \wedge PS_c^-)) \wedge (QF_b^+ \wedge QF_b^- \wedge QF_b^+ \wedge QF_b^-))) \sim p_r^2 (4p_s^4 q_f^3 p_f + 4p_s^3 q_s q_f^4).$$

Similarly:

$$\begin{aligned}
q_4^{(k)} &\sim p_r^2(4q_s^4q_f p_f^3 + 4q_s^3p_s p_f^4). \\
q_5^{(k)} &\sim p_r^2(6q_f^2p_f^2p_s^4 + 16q_f^3p_f p_s^3q_s + 6q_f^4p_s^2q_s^2) + 2pr(1-pr)(q_f^2p_s^2). \\
q_6^{(k)} &\sim p_r^2(6q_f^2p_f^2q_s^4 + 16p_f^3q_f q_s^3p_s + 6p_f^4p_s^2q_s^2) + 2p_r(1-p_r)(p_f^2q_s^2). \\
q_7^{(k)} &\sim p_r^2(4q_f p_f^3p_s^4 + 24p_f^2q_f^2p_s^3q_s + 24p_f q_f^3p_s^2q_s^2 + 4q_f^4p_s q_s^3) + 2p_r(1-p_r)(2p_s^2q_f p_f + 2p_s q_s q_f^2). \\
q_8^{(k)} &\sim p_r^2(4q_f^3p_f q_s^4 + 24q_f^2p_f^2q_s^3p_s + 24q_f p_f^3p_s^2q_s^2 + 4p_f^4q_s p_s^3) + 2pr(1-pr)(2q_s^2p_f q_f + 2p_s q_s p_f^2).
\end{aligned}$$

□

Proposition 6

Let $x^{(k)}$ defined as in Lemma 4 and $z = 0$ then the fixed point \bar{r} of Eq.3.12 is

$$\bar{r} = \frac{e^2}{t},$$

and for all $k = 1, \dots, N$, follows that:

$$r^{(k)} \leq x^{(k)}.$$

Proof. From Eq.3.12, \bar{r} is a fixed point if and only if:

$$0 = r^{(k+1)} - r^{(k)} = \frac{-4(t-e)^2(e^2 - r^{(k)}t)}{e(e-t)t^2}.$$

for which the unique admissible root is $\frac{e^2}{t}$. The sequence in Eq.3.12 is again a second order linear sequence for which a closed form can be computed as shown in [72]:

$$\begin{aligned}
r^{(k)} &= \frac{te - e^2}{t} \left(\frac{te - e^2 - 4p_r t}{te - e^2} \right)^k + \frac{e^2}{t} \quad \text{and} \\
x^{(k)} &= \frac{te - e^2}{t} \left(\frac{te - e^2 - 2p_r t}{te - e^2} \right)^k + \frac{e^2}{t} \quad \text{so} \\
r^{(k)} \leq x^{(k)} &\iff \frac{te - e^2 - 4p_r t}{te - e^2} \leq \frac{te - e^2 - 2p_r t}{te - e^2} \iff -2p_r \leq 0.
\end{aligned}$$

In conclusion $r^{(k)} \leq x^{(k)}$. □

Finally, in Prop. 4, we prove how to derive the bound showed in [27] for bipartite graphs.

Proposition 7

Let $\mathcal{G} = (\{V_r, V_c\}, E)$ represents a bipartite graph such that $|E| = e$, then the bound for N in Prop. 5 reads as:

$$N = \frac{e}{2(1-d)} \ln(e - de).$$

Proof. Let t be number of edges in the complete bipartite graph, i.e. $t = |V_r||V_c|$, and let $d = \frac{e}{t}$ be the graph's density. It is easy to see that $p_r = (1-d)^2$ and so the thesis. \square

3.5 Markov chain empirical convergence

3.5.1 Simulations

Let X_n be the Markov chain underlying the SA. It is easy to see that this Markov chain is irreducible, aperiodic with a finite space state, and so there exists a unique stationary distribution π . In Prop. 5 we derive a convergence bound for the mean value of X_n , i.e. we prove that $\mathbb{E}(X_n)$ converges after N SSs, and clearly $\mathbb{E}(X_n) \rightarrow \mathbb{E}(\pi)$. In this section we will show that this bound can be chosen as convergence time for all the distribution. Also if the transition matrix associate to the SA could be easily written (using the probabilities computed in Lemma 2), it is not trivial to compute a closed form for the probability density function F_n of X_n (this closed form could be used to write π and so check the distance from the stationary distribution).

For these reason, we generate a random graph with $n = 2000$ and $d = 10\%$ and we perform 2000 independent runs of the SA. We assume that π is reached after $100e \sim 35N$ SSs we compute the first five moments of X_n each 1000 SSs, and the Kolmogorov distance and the total variation distance between F_n and π .

In Fig. 3.6 we plot the trend of the first five normalised moments and in Fig. 3.7 we plot the trend of the two distances. Moreover, in the small box in Fig. 3.7, we plot also the evolution of the F_n during the SA.

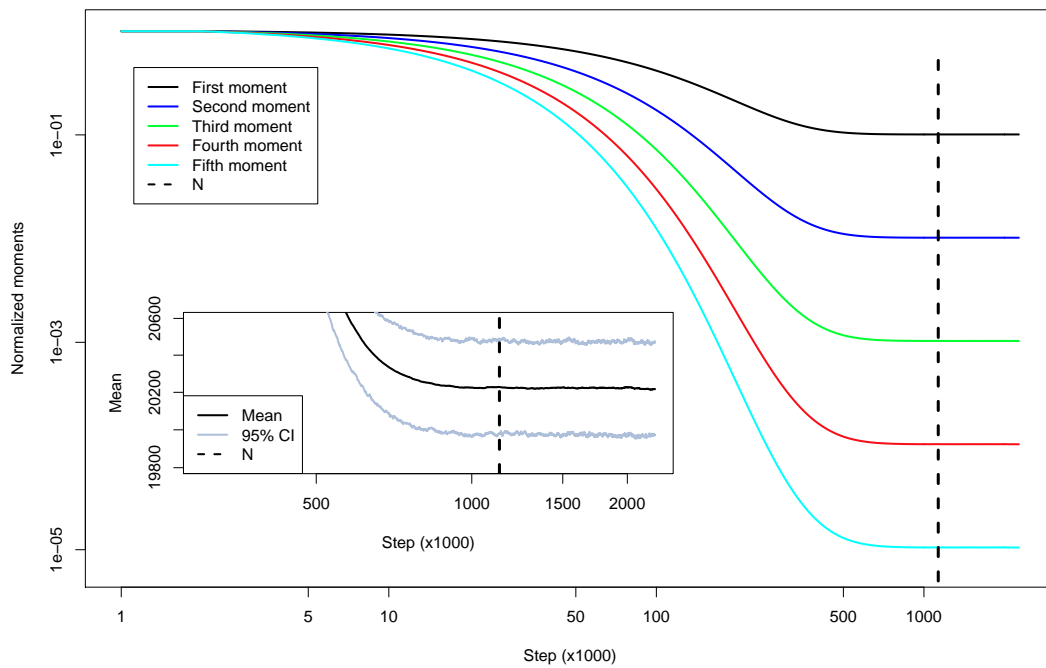


Fig. 3.6: The first five normalized moments of F_n computed every 1000 successful SSs (log-log plot). After N SSs all these moments tend to converge. In the small figure: trend of the mean value in black with the 95% CI in gray.

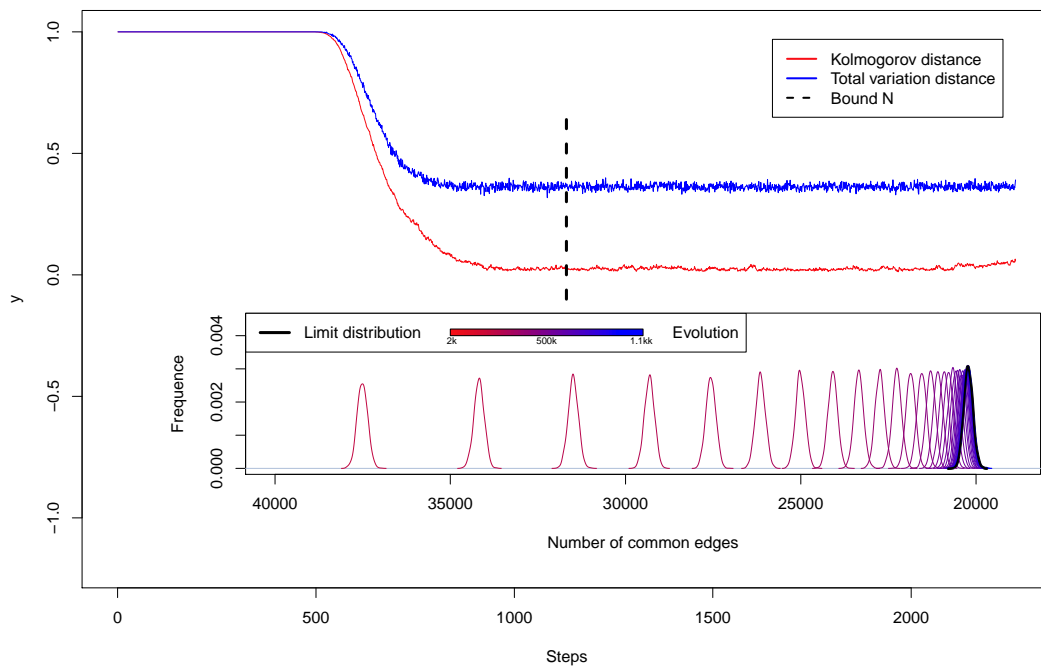


Fig. 3.7: The trend of two measure (Kolmogorov in red and total variation in blue) between F_n and π plotted between 1 and $2N$ successful SSs. The distances tend to converge after N SSs (dotted black line). In the small figure: the evolution of F_n form 2000 up to N SSs. The distributions are plotted every 20000 SSs. The distribution F_N (blue line) can be consider as π (black line).

3.5.2 Autocorrelation time

Following the same strategy described in [36], we can determinate the mixing time of the underlying Markov chain looking at the **autocorrelation time**.

Definition 11. The **autocorrelation** $A_t(X)$ of a signal $X = (x_1, \dots, x_n)$ is the covariance of itself and the same signal given a lag time t :

$$A_t(X) = \frac{\mathbb{E}[(x_i - \mu)(x_{i+t} - \mu)]}{\sigma^2},$$

where μ and σ are respectively, the mean and the variance of X .

If the samples X are sampled from the stationary distribution, the relative value of autocorrelation would be near to 0 as n , the number of samples, increases. Varying the lag time t and monitoring $A_t(X)$ we can control the mixing time of the Markov chain. We can compute $A_t(X)$ as:

$$A_t(X) = C(t)/C(0) \quad \text{where} \quad C(t) = \frac{1}{n-t} \sum_{i=1}^{n-t} (x_i - \mu)(x_{i+t} - \mu) \quad (3.13)$$

The signals considered here, following [36], are the presence/absence of the whole set of possible edges. Looking at the values of $A_t(X)$ for increasing values of t it is possible to determinate when the autocorrelation assume a *random* behaviour (and so the signal at lag time t can be consider uncorrelated).

Since the presence/absence of an edge can be read in the adjacency matrix, since we prove that after N SS the number of common edge reach its minimum (further SSs does not decrease it), and that the common edges tend not to be the same for multiple rewired graphs, and since the rewired edges are selected uniformly, we can state that a lag time $t = N$ guarantees an acceptable level of autocorrelation between the samples.

In Ex.1 we will show how the Jaccard index (or more generally, the number of common edges $x^{(k)}$) is correlated with the autocorrelation (Fig.3.8) and that our bound N can be effectively considered a *good* autocorrelation time (Fig.3.9). Here we obtain compatible results respect to which obtained in [27] for bipartite graphs.

Example 1. We generate a random undirected network with $n = 400$, $e = 10632$ and density $d = 6.6\%$. Using Eq.3.12 with $p_r = 1$, counting only successful SS, we obtain a bound for the number of SS N equal to 44836. We run the SA for $20N$ SS and extract a sample each 250 SS. For each lag time $t \in \{250, 500, \dots, 179500\}$ (it is unfeasible to extract samples for unitary increment of t) we compute $A_t(X)$ using Eq.3.13 for a tenth of the possible edges. We plot the trend of the autocorrelation for nine of these 7980 edges and the mean value. In Fig.3.8 we can see that there is a strong correlation between the (mean) autocorrelation and the mean Jaccard index and in Fig.3.9 we plot the lag time t versus the autocorrelation.

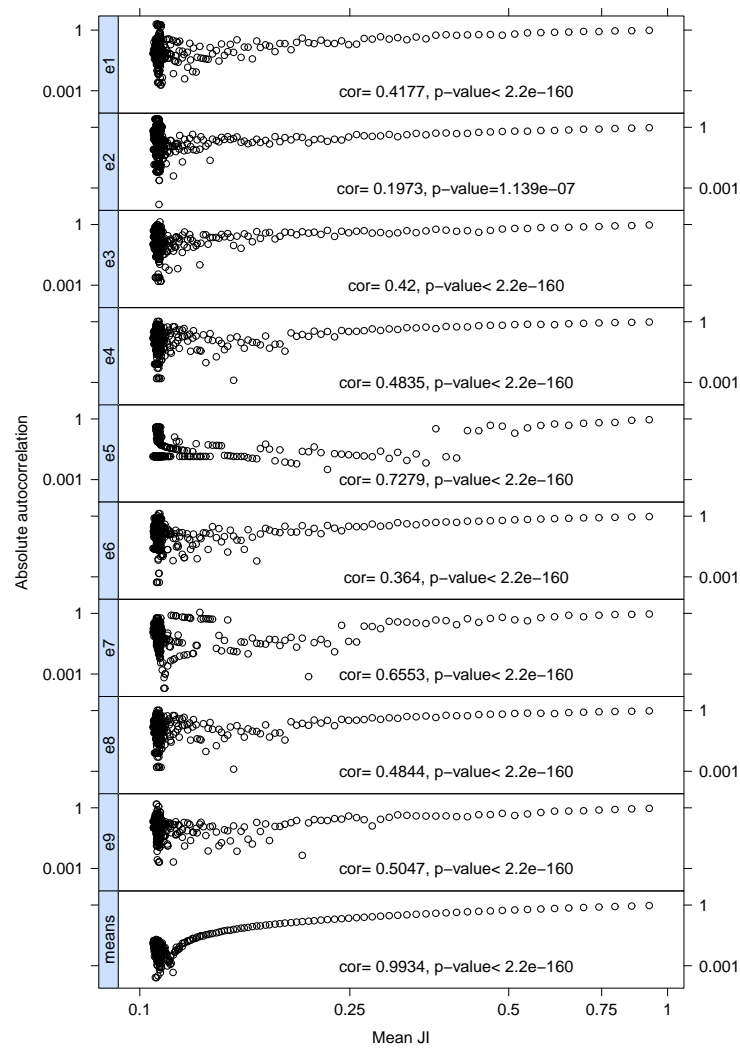


Fig. 3.8: The (log-log) scatterplots of the mean Jaccard Index VS (the absolute value of) the autocorrelation. In the lowest box we clearly see a strong autocorrelation between the mean Jaccard Index and the mean value of the autocorrelation.

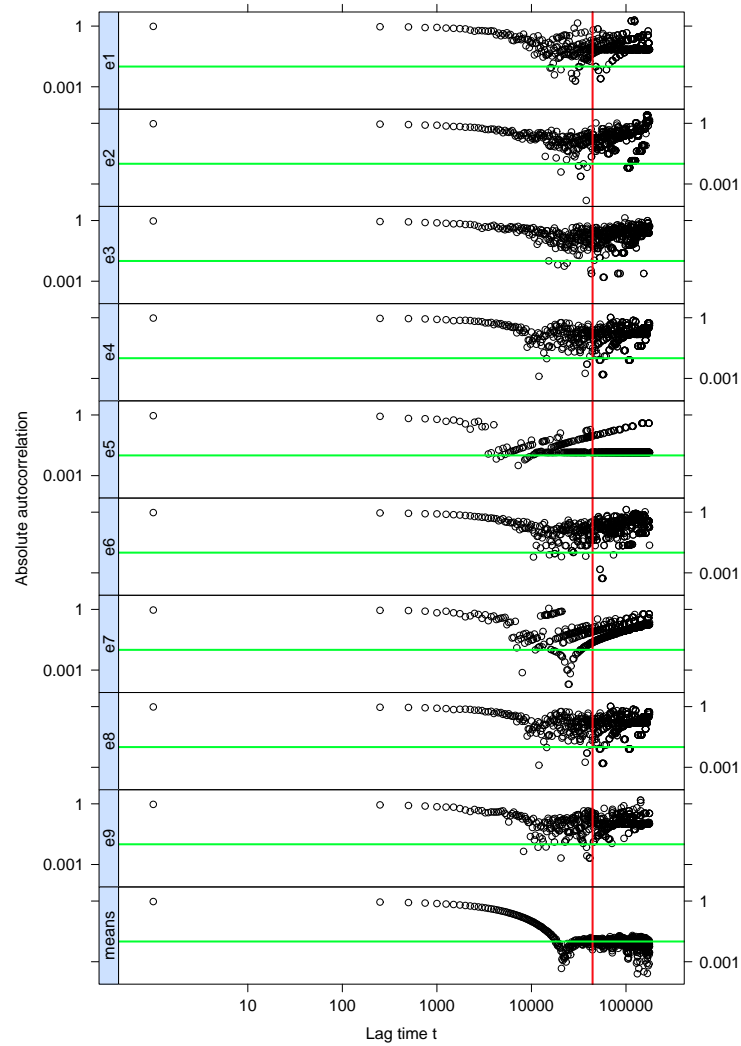


Fig. 3.9: The (log-log) plots relative to the trend of (the absolute value of) the autocorrelation for nine random edges. In red we draw our bound N and in green a value of (the absolute value of) autocorrelation equal to 0.01.

3.6 Package short description

We collect some useful functions related to the SA into a *R-package* called *BiRewire*. It is possible to download the package from <http://www.ebi.ac.uk/~iorio/BiRewire> and install it with the shell-command:

```
R CMD INSTALL BiRewire_xx.yy.zz.tar.gz
```

or with `biocLite()` directly in R:

```
source("http://bioconductor.org/biocLite.R")
biocLite("BiRewire")
```

To load *BiRewire* use the following commands:

```
library(BiRewire)
```

BiRewire requires the R package **igraph** (see [73]) available at the CRAN repository, or downloadable at <http://cran.r-project.org/web/packages/igraph/index.html>.

Most of the functions in the package are written in C and then wrapped in R. It is possible to work (at low dimensions) directly using the incidence matrix (for bipartite) or adjacency matrix (for undirected) or using edge-lists (for bipartite) or adjacency-lists (for undirected). These choices make the implemented functions very fast. For instance, for a random undirected graph with 1000 nodes and 50000 edges the *igraph* routine *rewire* needs ~ 52 s in order to perform 10000 SS, while the *BiRewire* routine *birewire.rewire* (working on adjacency lists) needs ~ 0.035 s.

In the package is also included a real dataset of breast cancer samples and their respective mutations downloaded from the Cancer Cancer Genome Atlas [60] at the address <http://tcga.cancer.gov/dataportal/> and a vignette in which we show all the functionalities of this package.

3.6.1 Function description

In this subsection are described all the functions implemented in *BiRewire* with a simple practical example in which a real breast cancer dataset is modeled as a bipartite network, and randomised preserving the mutation-rate both across samples and genes (i.e. the corresponding bipartite network is rewired). In each of the following functions it is possible to perform N **successful** switching steps, as discussed before, using the flag `exact=TRUE`. To prevent a possible infinite loop, the program performs at maximum `MAXITER_MUL*max.iter` iterations.

First of all, we create a bipartite network modeling a genomic breast cancer dataset downloaded from the Cancer Genome Atlas (TCGA) projects data portal

<http://tcga.cancer.gov/dataportal/>. From this dataset germline mutations were filtered out with state-of-the-art softwares; synonymous mutations and mutations identified as benign and tolerated were also removed. The resulting bipartite graph has $n_r = 757$ nodes (corresponding to samples), $n_c = 9,757$ nodes (corresponding to genes), and $e = 19,758$ edges connecting a node in n_r to a node in n_c if the gene corresponding to the node in n_r is mutated to the samples corresponding to the node in n_c . The edge density of this network is 0.27%.

The genomic dataset (in the form of a binary matrix in which rows correspond to samples, columns correspond to genes and the (i, j) entry is non null if the i -th sample harbours a mutation in the j -th gene) can be loaded and modeled as a bipartite graph, with the following R commands:

```
data(BRCA\_binary\_matrix)##loads an binary genomic event matrix
                        ##for the breast cancer dataset
g=birewire.bipartite.from.incidence(BRCA_binary_matrix)##models
                        ##the dataset as igraph bipartite graph
```

Once the bipartite graph is created it is possible to conduct the analysis by calling the **birewire.analysis** function, using the following commands:

```
step=5000
max=100*sum(BRCA_binary_matrix)
scores<-birewire.analysis(BRCA_binary_matrix,step,verbose=FALSE,
max.iter=max)
plot(x=step*seq(1:length(scores$similarity_scores)),
     y= scores$similarity_scores,
     type='l', xlab="Number of switching steps",
     ylab="Jaccard Similarity Score",ylim=c(0,1))
legend(max*0.8,1, c("Jaccard Similarity","N"),
       cex=0.9, col=c("black","red"), lty=1:1,lwd=3)
abline(v=scores$N,col='red')
plot(x=step*seq(1:length(scores$similarity_scores)),
     y= scores$similarity_scores,
     type='l',xlab="Number of switching steps",
     ylab="Jaccard Similarity Score",log="xy",main="Log-Log plot")
legend("topright", c("Jaccard Similarity","N"),
       cex=0.9, col=c("black","red"), lty=1:1,lwd=3)
abline(v=scores$N,col='red')
```

The function **birewire.analysis** returns the Jaccard similarity sampled every 5000 SSs. In the resulting plots the value of the analytically derived lower bound

to the number of switching steps is also visualised $\$N$. For more details see the the documentation.

The same analysis can be performed on general undirected networks (not bipartite).

```
g.und<-erdos.renyi.game(directed=F,loops=F,n=1000,p.or.m=0.01)
m.und<-get.adjacency(g.und,sparse=FALSE)
step=100
max=100*length(E(g.und))
scores.und<-birewire.analysis.undirected(m.und,step=step,
  verbose=FALSE,max.iter=max)
plot(x=step*seq(1:length(scores.und$similarity_scores)),
  y= scores.und$similarity_scores,
  type='l', xlab="Number of switching steps",
  ylab="Jaccard Similarity Score",ylim=c(0,1))
legend(max*0.8,1, c("Jaccard Similarity","N"),
  cex=0.9, col=c("black","red"), lty=1:1,lwd=3)
abline(v=scores.und$N,col='red')
plot(x=step*seq(1:length(scores.und$similarity_scores)),
  y= scores.und$similarity_scores,
  type='l',xlab="Number of switching steps",
  ylab="Jaccard Similarity Score",log="xy",main="Log-Log plot")
legend("topright", c("Jaccard Similarity","N"),
  cex=0.9, col=c("black","red"), lty=1:1,lwd=3)
abline(v=scores.und$N,col='red')
```

To rewire a bipartite graph two modalities are available. Both of them can be used with the analytical bound N as number of switching steps or with a user defined value. The function takes in input an incidence matrix \mathcal{B} or the an *igraph* bipartite graph.

```
m2<-birewire.rewire.bipartite(BRCA_binary_matrix,verbose=FALSE)
g2<-birewire.rewire.bipartite(g,verbose=FALSE)
```

The first function gives in output the incidence matrix of the rewired graph while the second one a rewired *igraph* graph. See documentation for further details.

To rewire a general undirected graph the following functions can be used:

```
m2.und<-birewire.rewire(m.und,verbose=FALSE)
g2.und<-birewire.rewire(g.und,verbose=FALSE)
```

This function computes the Jaccard index between two incidence matrices with same dimensions and node degrees.

```
sc=birewire.similarity(BRCA_binary_matrix,m2)
sc=birewire.similarity(BRCA_binary_matrix,t(m2))#also works
```

The following functions execute the Switching Algorithm and computes similarity trends across its switching steps for the two natural projections of the starting bipartite graph

```
#use a smaller graph!
gg <- simplify(graph.bipartite( rep(0:1,length=100),
c(c(1:100),seq(1,100,3),seq(1,100,7),100,seq(1,100,13),
seq(1,100,17),seq(1,100,19),seq(1,100,23),100)))
result=birewire.rewire.bipartite.and.projections(gg,step=10,
max.iter="n",accuracy=1,verbose=FALSE)
plot(result$similarity_scores.proj2,type='l',col='red',ylim=c(0,1))
lines(result$similarity_scores.proj1,type='l',col='blue')
legend("top",1,c("Proj2","Proj1"),cex=0.9,col=c("blue","red"),lwd=3)
```


Chapter 4

Null model for co-expression network based on Pearson correlation

Universally acknowledged by the scientific community as the basic task of the systems biology, the network inference is the prototypical procedure for moving from the classical reductionist approach to the novel paradigm of data-driven complex systems in the interpretation of biological processes [1]. The core of all the network inference (or network reconstruction) procedures is the detection of the topology of a graphy, *i.e.*, its wiring diagram, whose nodes are a given set of biological entities, starting from measurements of the entities themselves. In the last 15 years, the reconstruction of the regulation mechanism of a gene network and of the interactions among proteins from high-throughput data such as expression microarray of, more recently, from Next Generation Sequencing data has become a major line of research for laboratories worldwide. The proposed solutions rely on techniques ranging from deterministic to stochastic, and their number is constantly growing in the literature. Nonetheless, network inference is still considered an open, unsolved problem [74]. In fact, in many practical cases, the performances of the reconstruction algorithms are poor, due to several factors limiting the inference accuracy [75, 76] to the point of making it no better than coin tossing in some situations [15]. The major problem is the underdeterminacy of the task [77], due to the overwhelming number of interactions to predict starting from a usually small number of available measurements. In general, size and quality of available data are critical factors for all inference algorithms.

In what follows the impact of data size is discussed for one of the simplest inference techniques, *i.e.*, the gene coexpression network, where interaction strength between two genes is a function of the correlation between the corresponding expression levels across the available tissue samples. The biological underlying hy-

pothesis is that functionally related genes have similar expression patterns [78], and thus that coexpression is correlated with functional relationships, although this does not imply causality. In particular, as highlighted in [79], correlation can help unveiling the underlying cellular processes, since coordinated coexpression of genes encode interacting proteins, and Pearson correlation coefficient can be used as the standard measure. However, as noted in [80], correlation between genes may sometimes be due to unobserved factors affecting expression levels. Coexpression analysis has been intensively used as an effective algorithm to explore the system-level functionality of genes, sometimes outperforming much more refined approaches [81, 82]. The observation that simpler approaches such as correlation can be superior even on synthetic data has been explained by some authors [83, 84] with the difficulties of complex algorithm in detecting the subtleties of the combinatorial regulation. Moreover, coexpression network can capture more important features that the conventional differential expression approach [85], and its use has been extended to other tasks, for instance the investigation of complex biological traits [86]) Finally, these network can be crucial for understanding regulatory mechanisms [87], for the development of personalised medicine [88] or, more recently, in metagenomics [89].

Despite its success, a major issue affects coexpression networks: deciding when a given correlation value between two nodes can be deemed statistically significant and thus worthwhile assigning a link connecting them. This translates mathematically into choosing (a function of) a suitable threshold, as in the case of mutual information and relevance networks [90]. As reported in [91], in literature statistical methods for testing the correlations are underdeveloped, and thresholding is often overlooked even in important studies [92]. The two main approaches known in literature can be classified as soft or hard thresholding. The soft thresholding is adopted in a well-known framework called Weighted Gene Coexpression Network Analysis (WGCNA) [93], recently used also for other network types [94, 95]. All genes are mutually connected, and the weight of the link is a positive power of the absolute value of the Pearson correlation, where the exponent is chosen as the best fit of the resulting network according to a scale-free model [28, 29]. This approach, without discarding any correlation, promotes high correlation values and penalises low values. In the hard thresholding approach, instead, only correlation values larger than the threshold are taken into account, and an unweighted link is set for each of these values, so that a binary network is generated (see [96] for one of the earliest references). Clearly, an incorrectly chosen threshold value can jeopardize the discussed results with false negative links (for too strict threshold) or false positive links (for too loose threshold). Many different heuristics have been proposed for setting the threshold values, such as using the False Discovery Rate [97, 98, 99, 100], or the p -value of the correlation test [88], or employing partial cor-

relation [101], or using rank-based techniques [102, 103, 104] or more complex randomization techniques [105]. Alternatively, correlation distribution has been studied, experimentally [106] or at level of single interaction, not as whole network [107]. However, in many studies in literature, the threshold is not chosen accordingly to a soundly bases procedure, but referring to standard choices [108, 109, 110, 111], or to heuristics not directly related to the correlation values, but rather with the resultining network topology [112, 113, 114, 115, 116, 117, 118, 119, 120]. In [121] a comparison of some coexpression thresholds is shown on a few microarray datasets.

Here we propose a new a priori and non-parametric model for the computation of an hard threshold based on the assumption that a random coexpression graph should not have any edge. The threshold is theoretically derived by means of a geometric approach based on the work of Bevington [122], and, as a deterministic independent null model, it depends only on the dimensions of the starting data matrix, with assumptions on the skewness of the data distribution compatible with the structure of gene expression levels data [123, 124]. By definition, this threshold is aimed at minimising the possible false positive links, paying a price in terms of false negative detected edges.

To conclude with, we show four applications, in both the large and the small sample size settings. The first two are examples in a large sample size settings, with a synthetic dataset and with an ovarian epithelial carcinoma dataset on a large cohort of 285 cases [125, 126]. Two more applications in the opposite situations are demonstrated on two publicly available datasets, the former regarding a pancreatic cancer study [127] on a tiny cohort of six patients, and the latter on a Alzheimer dataset with 28 samples on two different phenotypes [128, 129, 130].

4.1 Distribution of Pearson correlation

Let $x, y \in \mathbf{R}^n$ with $n \geq 3$. The **Pearson correlation coefficient** ρ between x and y is defined as:

$$\rho(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}},$$

where \bar{w} denotes the arithmetic mean $\frac{1}{n} \sum_{i=1}^n w_i$ of the n -dimensional vector w .

The first step towards the construction of a null model for random absolute Pearson coexpression network is the estimation, for $0 < p < 1$, of the function $F(n, p) = P(|\rho(x, y)| > p)$, where x and y are two independent normal vectors of

length n . Define two new random variable \tilde{x} and \tilde{y} as follows:

$$\tilde{x} = \frac{x - \bar{x}}{\sigma_x \sqrt{n-1}}, \quad \text{and} \quad \tilde{y} = \frac{y - \bar{y}}{\sigma_y \sqrt{n-1}}, \quad (4.1)$$

where σ_x and σ_y are the standard deviations of x and y . From the definition, the following identities immediately descend:

$$\begin{aligned} \sum_{i=1}^n \tilde{x}_i &= 0 = \sum_{i=1}^n \tilde{y}_i \\ \sum_{i=1}^n \tilde{x}_i^2 &= 1 = \sum_{i=1}^n \tilde{y}_i^2 \\ \rho(x, y) &= \rho(\tilde{x}, \tilde{y}) = \sum_{i=1}^n \tilde{x}_i \tilde{y}_i. \end{aligned} \quad (4.2)$$

We can now state and prove two key results.

Proposition 8

Let $x, y, \tilde{x}, \tilde{y}$ be defined as in Eq. 4.1. Then $\tilde{x}, \tilde{y} \in S_{n-1} \cap \mathcal{H} \sim S_{n-2}$, where \mathcal{H} is the vectorial hyperplane defined as $\sum_{i=1}^n w_i = 0$ and w_i are the coordinates of \mathbf{R}^n .

Proof. Since $\|\tilde{x}\| = 1$, the following identity holds:

$$\begin{aligned} \sum_{i=1}^n \tilde{x}_i &= \sum_{i=1}^n \frac{x_i - \bar{x}}{\sigma_x \sqrt{n-1}} = \frac{1}{\sigma_x \sqrt{n-1}} \sum_{i=1}^n (x_i - \bar{x}) \\ &= \frac{1}{\sigma_x \sqrt{n-1}} \left[\left(\sum_{i=1}^n x_i \right) - n\bar{x} \right] = \frac{1}{\sigma_x \sqrt{n-1}} (n\bar{x} - n\bar{x}) = 0, \end{aligned}$$

and the same holds for \tilde{y} , too. □

An example for $n = 3$ of the situation described in Prop. 8 is plotted in Fig. 4.2.

Proposition 9

Let x, y be as in Prop. 8 and $0 < p < 1$ be a real number. Then the function $F(n, p)$ has the following close form

$$F(n, p) = P(|\rho(x, y)| > p) = \frac{2}{\sqrt{\pi}} \frac{\Gamma\left(\frac{n-1}{2}\right)}{\Gamma\left(\frac{n-2}{2}\right)} \int_0^{\arccos p} \sin^{n-3}(\vartheta) d\vartheta, \quad (4.3)$$

where $\Gamma(x)$ is the gamma function $\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-t} dt$.

Proof. Using Eq. 4.1 and Eq. 4.2, we have that

$$\rho(x, y) = \rho(\tilde{x}, \tilde{y}) = \tilde{x}\tilde{y} = \cos \beta, \quad (4.4)$$

where β is the angle between the two vectors \tilde{x} and \tilde{y} . Eq. 4.4 and Prop. 8 yields that $P(|\rho(x, y)| > p)$ is the proportion between the area of the spherical cap in $n - 2$ dimensions included within an angle β from x and the whole surface of the $n - 2$ -dimensional sphere [131]. A compact formula for the area $A_{n-1}^{\text{cap}}(r)$ of a $n - 2$ -spherical cap is given in [132] as:

$$A_{n-1}^{\text{cap}}(r) = \frac{2\pi^{(n-2)/2}}{\Gamma\left(\frac{n-2}{2}\right)} r^{n-2} \int_0^\beta \sin^{n-3}(\vartheta) d\vartheta,$$

and, since the area of the whole surface is

$$S_{n-2}(r) = \frac{2\pi^{(n-1)/2}}{\Gamma\left(\frac{n-1}{2}\right)} r^{n-2},$$

the thesis follows from the setting $r = 1$. \square

An alternative derivation of the same result can be found in [122].

In Prop. 8 the transformed vectors are assumed to be uniformly distributed on the spherical surface. This assumption holds in the case of a normal distribution, but it does not hold in general. However, in the following paragraph we show that is a good approximation, since x and y are independent. In fact, Prop. 4.3 can be generalised to other distributions [133, 134, 135, 136]), when data skewness can be bounded [131].

Let $G^\delta(p, n)$ be an empirical distribution generated by k couples of two vectors $x, y \in \mathbf{R}^n$ sampled according to a given distribution function δ . Let then

$$E_t(F, G^\delta) = \left(\int_0^1 |F(p, n) - G^\delta(p, n)|^t dp \right)^{\frac{1}{t}}$$

$G^\delta(p, 8)$		x		
		$U(0, 1)$	$N(0, 1)$	$L(2, 3)$
y	$U(0, 1)$	0.001832	0.00137	0.021202
	$N(0, 1)$	0.001195	0.00142	0.001432
	$L(2, 3)$	0.022961	0.00139	0.080803
$G^\delta(p, 20)$		x		
		$U(0, 1)$	$N(0, 1)$	$L(2, 3)$
y	$U(0, 1)$	0.0016851	0.0007752	0.0248819
	$N(0, 1)$	0.0008008	0.0014559	0.0008381
	$L(2, 3)$	0.0238804	0.0011422	0.1038271
$G^\delta(p, 100)$		x		
		$U(0, 1)$	$N(0, 1)$	$L(2, 3)$
y	$U(0, 1)$	0.0006978	0.0008244	0.015630
	$N(0, 1)$	0.0009281	0.0007388	0.001441
	$L(2, 3)$	0.0159969	0.0014090	0.104998

Tab. 4.1: Error function $E_2(F, G^\delta)$, for $n = 8, 20, 100$ and different distributions δ .

be the t -error function evaluating the difference between the theoretical distribution $F(p, n)$ and the empirical distribution $G^\delta(p, n)$. Hereafter we report the results of the simulations for $k = 50000$ and $n = 8, 20, 100$, where δ is one of the following three distribution functions:

- $U(0, 1)$, the uniform distribution in $[0, 1]$;
- $N(m, s)$, the normal distribution with mean m and standard deviation s ;
- $L(ml, sl)$, the lognormal distribution with mean-log ml and standard deviation-log sl .

In particular, in Tab. 4.1 we list the values of $E_2(F, G^\delta)$ and in Fig. 4.1 we display the curves of the cumulative distribution functions (CDF) of $G^\delta(p, n)$ corresponding to the three functions δ , separately for the different values of n .

Regardless of the value of n , the empirical distribution fits the exact formula Eq.4.3 when x and y are uniformly sampled, while it does not fit the same equation when the two vectors come from extremely skewed distributions such as the lognormal. Note that non-Gaussian asymmetric distributions can occasionally be detected in some array studies [124]: however, techniques for reducing the skewness are routinely applied during preprocessing [123], and thus the aforementioned results can be safely used in the microarray framework.

Finally, we conclude this paragraph deriving the mean and the variance of the function $|\rho|$. Starting from Eq. 4.3, the density function $f(p, n)$ can be computed as

$$f(p, n) = \frac{2}{\sqrt{\pi}} \frac{\Gamma\left(\frac{n-1}{2}\right)}{\Gamma\left(\frac{n-2}{2}\right)} (1-p^2)^{\frac{n-4}{2}}.$$

Using the above expression for $f(p, n)$, the two moments follow straightforwardly:

$$\begin{aligned} \mathbb{E}(|\rho|, n) &= \int_0^1 pf(p, n)dp \\ &= \frac{2}{\sqrt{\pi}} \frac{\Gamma\left(\frac{n-1}{2}\right)}{(n-2)\Gamma\left(\frac{n-2}{2}\right)} \\ \text{Var}(|\rho|, n) &= \int_0^1 p^2 f(p, n)dp - \mathbb{E}^2(p, n) \\ &= \frac{1}{n-1} - \frac{4\Gamma^2\left(\frac{n-1}{2}\right)}{\pi(n-2)^2\Gamma^2\left(\frac{n-2}{2}\right)}. \end{aligned}$$

4.2 Coexpression network and threshold selection

The results derived in the previous section are used here to construct a null model for the correlation network, thus yielding a threshold for the inference of a coexpression network from nodes' data.

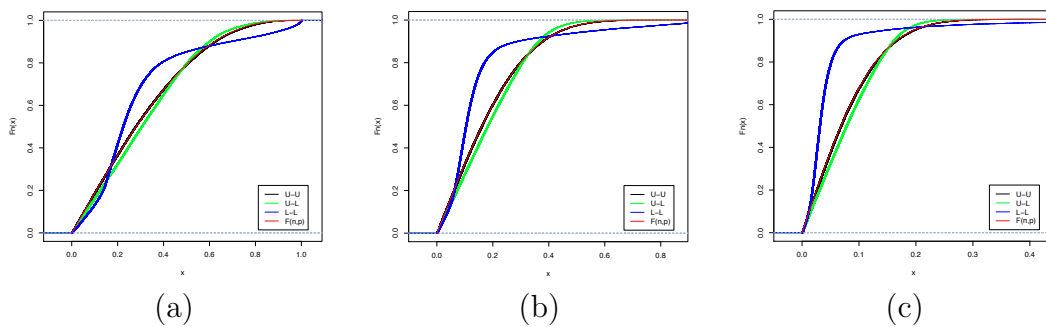


Fig. 4.1: CDFs relative to the different distributions $\delta = U$ and $\delta = L$ compared with the theoretical curve $F(n, p)$, for the three cases $n = 8$ (a), $n = 20$ (b) and $n = 100$ (c). In all cases, the red curve of $F(n, p)$ and the black curve for the double uniform distribution $U - U$ are almost coincident.

Let $\mathcal{X} = \{x_i\}_{i=1}^m$ be a set such that $x_i \in \mathcal{U}[0, 1]^n \forall i = 1, \dots, m$. Then the coexpression p -graph $\mathcal{G}_p = \{V, E_p\}$ is the graph where

$$V = \{v_1, \dots, v_m\} \quad \text{and} \quad (v_i, v_j) \in E_p \iff |\rho(x_i, x_j)| > p .$$

The first result characterises the coexpression graphs in terms of null models:

Proposition 10

The graph \mathcal{G}_p is an Erdős-Rényi model [3] with m nodes and probability p as in Eq. 4.3.

Proof. The proof follows immediately from the definition of \mathcal{G}_p and Eq. 4.3. \square

Example Consider a dataset \mathcal{Y} consisting of $n = 3$ samples described by $m = 100$ genes. Then \mathcal{Y} can be represented by 100 points in $[0, 1]^3 \subset \mathbf{R}^3$ as shown in Fig. 4.2(a). The new variables \tilde{x}_i are built through a two-stages procedure applied to each gene. First the mean is subtracted, so the transformed dataset lies on the hyperplane \mathcal{H} described in Prop. 8 as displayed in Fig. 4.2(b,c). Finally, each gene is normalised to unitary variance, and the resulting dataset lies on $S_{n-1} \cap \mathcal{H}$, which is the circumference in Fig. 4.2(d). Using the results in the previous section, it is now possible to define, for n nodes measured on m samples, the secure threshold \bar{p} as the minimum value of p such that the corresponding random coexpression network $\mathcal{G}_{\bar{p}}$ is on average an empty graph, that is

$$\bar{p} = \min_{p \in (0,1]} \left\{ F(n, p) \frac{m(m-1)}{2} < 1 \right\} . \quad (4.5)$$

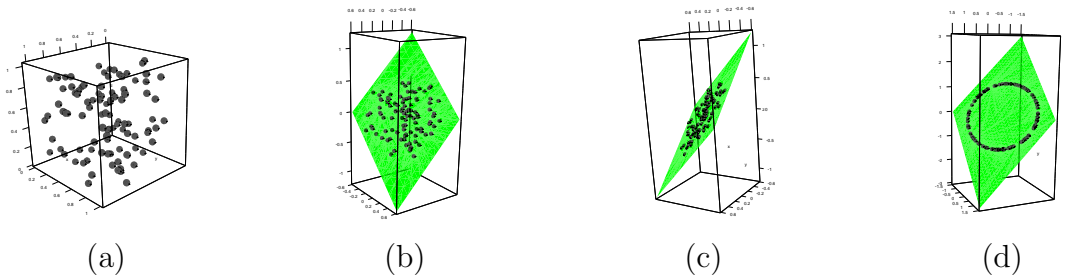


Fig. 4.2: Transformation of the initial dataset preserving the Pearson correlation. (a) Original dataset (b,c) Mean subtraction (d) Variance normalisation. In green the hyperplane \mathcal{H} .

n \ m	100	500	1000	2000	10000	50000	100000
8	0.95629	0.98520	0.99070	0.99415	0.99800	0.99932	0.99957
15	0.81681	0.89170	0.91323	0.93036	0.95800	0.97456	0.97949
20	0.73825	0.82388	0.85077	0.87330	0.91286	0.93973	0.94852
30	0.62814	0.71776	0.74817	0.77485	0.82534	0.86367	0.87729
50	0.50225	0.58534	0.61513	0.64213	0.69607	0.74036	0.75705
75	0.41647	0.49026	0.51740	0.54238	0.59353	0.63709	0.65394
100	0.36343	0.42999	0.45477	0.47774	0.52537	0.56662	0.58279

Tab. 4.2: A subset of values of the secure threshold \bar{p} for different number of samples m and genes n .

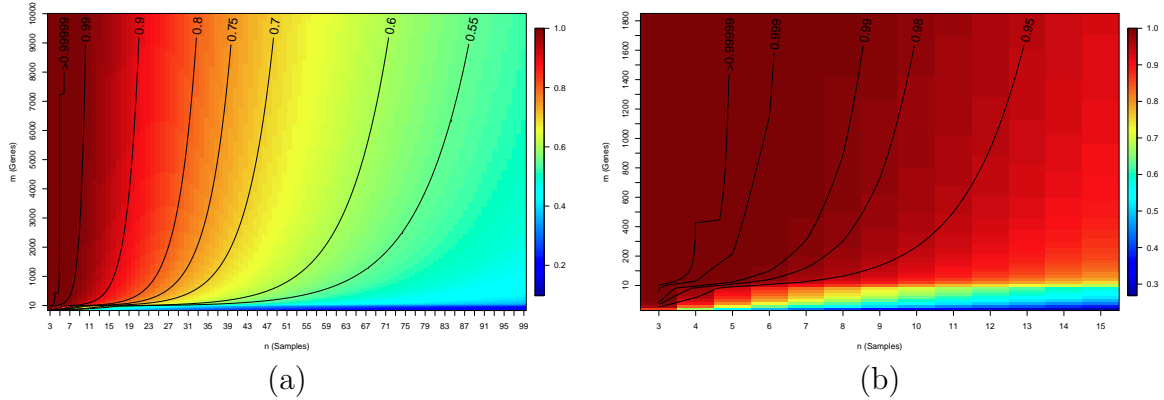


Fig. 4.3: Contour plot of the function $\bar{p}(m, n)$ on (a) a large (m, n) range and (b) zoomed on the small sample size area.

The underlying hypothesis for Eq. 4.5 is the assumption that in a random dataset we do not expect an kind of edge, *i.e.* any kind of relation. Due to its definition, the secure threshold \bar{p} is biased towards avoiding the false positive links, paying a price in terms of false negatives. In fact, all the links passing the filter are induced by correlation only due to the inference data, while all links whose correlation value can be generated either by relation between data or by random noise are discarded. In Tab. 4.2 a collection of values of \bar{p} is listed for different m and n , while in Fig. 4.3 the contourplot of the function $\bar{p}(n, m)$ is shown first on a large range of values and then zooming on the small sample size area. In the Tab. 4.3 we show the comparison on a set of synthetic and array datasets of the secure threshold \bar{p} with another well known hard thresholding methods, the clustering

coefficient-based threshold C^* [117] and with the statistical thresholds based on the adjusted p-values of 0.01, 0.05 or 0.1. In almost all cases, the threshold \bar{p} is the strictest. As shown in the previous section, for not very skewed distribution, the good approximation provided by the exact formula for $F(n, p)$ given in Eq. 4.3 guarantees the effectiveness of the secure threshold \bar{p} in detecting actual links between nodes. Nonetheless, whenever a stricter threshold is needed, it is still possible to follow the construction proposed, with the following refinement. The edge-creation process in the Erdős-Rényi model follows a binomial distribution, where n is the number of trials and p the probability associated to the success of a trial. The mean np of this distribution is one of the contributing term in the definition of secure threshold Eq. 4.5. To further restrict the number of falsely detected links, the variance term ($np(1 - p)$ for the binomial distribution) can be added to the formula through the Chebyshev's inequality

$$P(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2} ,$$

Dataset type	#samples	#nodes	C^*	B0.01	B0.05	B0.1	\bar{p}
Simulated	50	1000	0.57	0.58	0.54	0.52	0.6152
Simulated	25	1000	0.69	0.76	0.72	0.70	0.7956
H-U133P	23	897	0.72	0.78	0.74	0.72	0.8125
H-U133P	10	897	0.78	0.96	0.94	0.93	0.9723
H-U133P	10	675	0.77	0.96	0.93	0.92	0.9681
H-U133P	9	897	0.79	0.97	0.96	0.95	0.9821
H-U133P	8	897	0.81	0.98	0.97	0.96	0.98999
H-U133P	7	897	0.81	0.99	0.99	0.98	0.99558
H-U133P	6	897	0.86	>0.99	>0.99	0.99	0.99872
H-U133P	5	897	0.92	>0.99	>0.99	>0.99	0.99984
H-U133P	4	897	0.99	>0.99	>0.99	>0.99	> 0.9999
H-U133A	4	675	0.99	>0.99	>0.99	>0.99	> 0.9999
H-I6	4	675	0.99	>0.99	>0.99	>0.99	> 0.9999
M-U74	4	401	0.97	>0.99	>0.99	>0.99	0.9999

Tab. 4.3: Comparison of the secure threshold \bar{p} with the clustering coefficient-based threshold C^* [117] and the statistical thresholds based on the adjusted p-values B0.01, B0.05 or B0.1 on a collection of synthetic and array datasets.

n \ m	100	500	1000	2000	10000	50000	100000
8	0.97584	0.99179	0.99484	0.99675	0.99889	0.99962	0.99977
15	0.86282	0.91826	0.93437	0.94723	0.96810	0.98065	0.98439
20	0.78966	0.85726	0.87876	0.89686	0.92883	0.95068	0.95784
30	0.68082	0.75573	0.78151	0.80425	0.84759	0.88074	0.89256
50	0.55034	0.62269	0.64902	0.67302	0.72135	0.76137	0.77651
75	0.45887	0.52436	0.54881	0.57145	0.61820	0.65834	0.67394
100	0.40153	0.46116	0.48369	0.50471	0.54865	0.58703	0.60214

Tab. 4.4: A subset of values of the secure threshold \tilde{p}_2 for different number of samples m and genes n .

Tab. 4.5: A subset of values of the secure threshold \tilde{p}_5 for different number of samples m and genes n .

n \ m	100	500	1000	2000	10000	50000	100000
8	0.98553	0.99508	0.99691	0.99805	0.99934	0.99978	0.99986
15	0.89287	0.93585	0.94842	0.95849	0.97486	0.98474	0.98768
20	0.82530	0.88080	0.89858	0.91361	0.94025	0.95853	0.96454
30	0.71934	0.78401	0.80647	0.82636	0.86445	0.89373	0.90420
50	0.58686	0.65162	0.67541	0.69720	0.74130	0.77803	0.79198
75	0.49164	0.55125	0.57373	0.59463	0.63803	0.67552	0.69015
100	0.43124	0.48595	0.50683	0.52640	0.56752	0.60368	0.61796

where μ and σ are the mean and the standard deviation of X . Thus, the definition of secure threshold can be sharpened to \tilde{p}_k as follows:

$$\tilde{p}_k = \min_{p \in (0,1)} \left\{ F(n, p) \frac{m(m-1)}{2} + k \sqrt{(1 - F(n, p)) F(n, p) \frac{m(m-1)}{2}} < 1 \right\} .$$

For instance, the binomial distribution, for large value of n , can be approximated as a normal distribution for which the 95.45% of the values lie in the interval $(\mu - 2\sigma, \mu + 2\sigma)$. In Tab 4.4 we show, for \tilde{p}_2 , the analogous of Tab. 4.2 for \bar{p} .

Finally, the Chebyshev's inequality implies that at least the 96% of the values lie in the interval $(\mu - 5\sigma, \mu + 5\sigma)$: the corresponding threshold values for $k = 5$ are listed in Tab. 4.5.

4.3 Applications in functional genomics

4.3.1 Large sample size

Synthetic dataset First a correlation matrix M_G on 20 genes G_1, \dots, G_{20} is created, together with a dataset \mathcal{G} of the corresponding expression G_i^{1000} across 1000 synthetic samples, so that $M_G(i, k) = |\text{cor}(G_i^{1000}, G_j^{1000})|$ is the absolute Pearson correlation between the expression of the genes G_i and G_j from \mathcal{G} .

In particular, M_G has two 10×10 blocks highly correlated on the main diagonal, and two 10×10 poorly correlated blocks on the minor diagonal, as shown in Fig. 4.4. These blocks derived from the following generating rule, given uncorrelated starting element G_1^{1000} and G_{11}^{1000} :

$$|\text{cor}(G_k^{1000}, G_j^{1000})| \approx \begin{cases} 1 - 0.03j & \text{for } k = 1, 2 \leq j \leq 10 \\ 0.7 - 0.015j & \text{for } k = 11, 12 \leq j \leq 20 \end{cases}.$$

Outside the two main blocks, all correlation values range between 0.002 and 0.074. In Fig. 4.4 we also show the heatmap of the gene expression dataset \mathcal{G} . Then a subset of n_s samples is selected from the starting 1000, and the corresponding coexpression networks is built, for the 100 hard threshold values $0.01j$, for $1 \leq j \leq 100$. The secure threshold for these cases are respectively 0.799, 0.596 and 0.389. These procedure is repeated 500 times for each value $n_s = 10, 20, 50$. The same experiment is then repeated adding a 20% and a 40% level of Gaussian noise to the original data. Using M_G as the ground truth where all values outside the two main blocks are thresholded to 0, for each hard threshold $0.01j$ we evaluate the

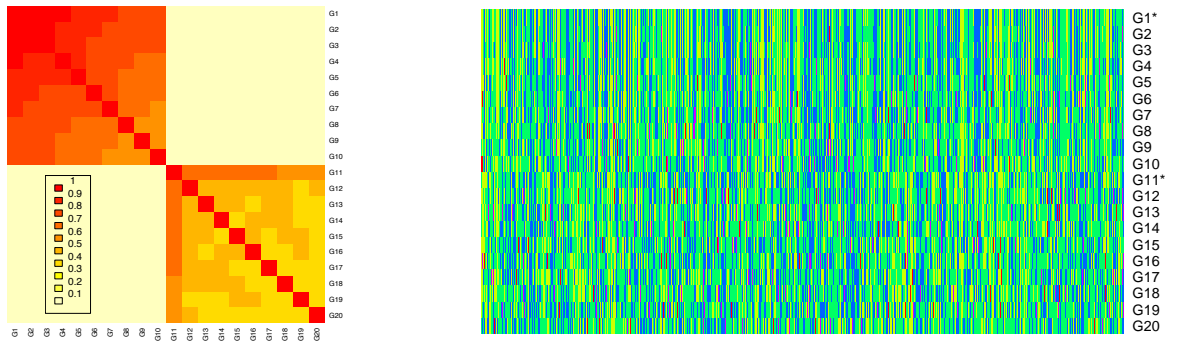


Fig. 4.4: Levelplot of the structure of the correlation matrix M_G (left) and heatmap of the dataset \mathcal{G} . The generating gene expression vectors G_1^{1000} and G_{11}^{1000} are marked with *.

ratio of False Positive links, the ratio of False Negative links and the Hamming-Ipsen-Mikhailov (HIM) distance from the gold standard¹ The graphs summarising the experiments, separately for sample size, are displayed in Fig. 4.5.

In all cases, the secure threshold \bar{p} corresponds to the strictest value yielding a coexpression network with no false positive links included, which its characterising property. Moreover, in almost all displayed situations, thresholding at \bar{p} still guarantees an acceptable HIM distance from the ground truth, and a false negative ratio always smaller than 0.4.

Ovarian cancer The aforementioned results obtained in a synthetic case are then tested here in a large array study on 285 patients of ovarian cancer at different stages [126], recently used in a comparative study on conservation of coexpressed modules across different pathologies [125]. In details, a whole tumor gene expression profiling was conducted on 285 predominately high-grade and advanced stage serous cancers of the ovary, fallopian tube, and peritoneum; the samples were hybridized on the Affymetrix Human Genome HG-U133 Plus 2.0 Array, including 54621 probes. The goal of the original study was to identify novel molecular subtypes of ovarian cancer by gene expression profiling with linkage to clinical and pathological features. As a major result, the authors presented two ranked gene lists supporting their claim that molecular subtypes show distinct survival characteristics. The two gene lists characterise the Progression Free Survival (PSF) and the poor Overall Survival (OS), respectively.

Following the procedure of the previous, synthetic example, first we individuate the sample subset corresponding to the homogeneous cohort of 161 grade three patients and a set T of 20 genes, belonging to the top good OS and PFS genes (EDG7, LOC649242, SCGB1D2, CYP4B1, NQO1, MYCL1, PRSS21, MGC13057, PPP1R1B, KIAA1324, LOC646769) and to the top poor OS/PFS genes (THBS2, SFRP2, DPSG3, COL11A1, COL10A1, COL8A1, FAP, FABP4, POSTN), thus generating a dataset \mathcal{O}_T of dimension 161 samples and 20 features. The corresponding absolute Pearson correlation matrix O_T is then used as the ground truth for the subsampling experiments: the levelplot of O_T and the heatmap of \mathcal{O}_T is shown in Fig. 4.6. In these experiments, a random subdataset of n_s samples is extracted from \mathcal{O}_T , and the corresponding absolute Pearson coexpression network on the nodes T is built, for increasing threshold values. In Fig. 4.7 we report the HIM and the ratio of False Positive and False Negative links for 500 runs of the experiments, separately for $n_s = 5, 10, 20$ and 50.

Again, the secure threshold \bar{p} corresponds to the smallest Pearson value warranting no false positive links included. Moreover, in almost all displayed situa-

¹The HIM distance [137, 138] is a metric between networks having the same nodes, ranging between 0 for identical networks and 1, attained comparing the clique with the empty graph.

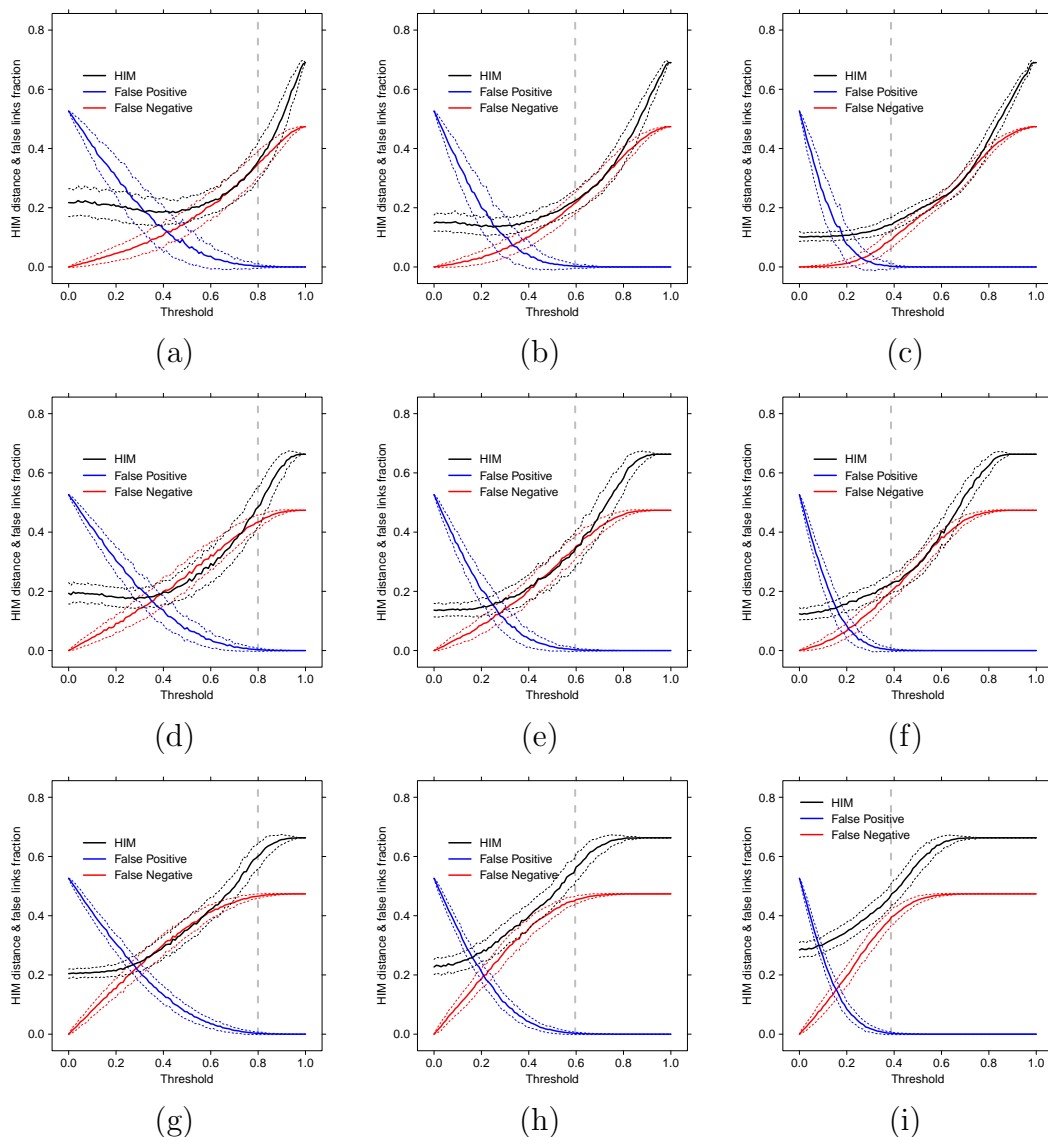


Fig. 4.5: Coexpression inference of the M_G network from random subsampling of the \mathcal{G} dataset, without noise (a,b,c), with 20% Gaussian noise (d,e,f) and with 40% Gaussian noise (g,h,i), on 10 (a,d,g), 20 (b,e,h) and 50 (c,f,i) samples. Solid lines indicate mean over 500 replicates of HIM distance (black), ratio of False Positive (blue) and ratio of False Negative (red); dotted lines of the same color indicate $\pm 1\sigma$, while grey vertical dashed lines correspond to the secure threshold \bar{p} .

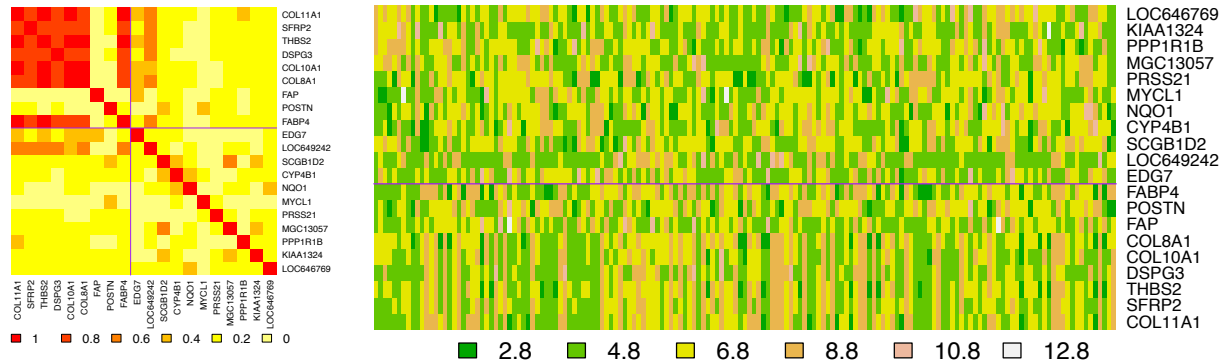


Fig. 4.6: Levelplot of the structure of the correlation matrix O_T (left) and heatmap of the Ovarian dataset O_T restricted to the set of 20 selected genes T . Solid lines separate the group of good and poor PFS/OS top genes.

tions, the threshold \bar{p} is approximately the value where the HIM distance starts growing quicker, while the false positive rate remains under 0.8.

4.3.2 Small sample size

When the sample size is very small, the novel hard thresholding introduced here can severely limit the conclusions than can be drawn without incurring in the risk of discussing false positive links. This problem can be particularly evident in differential network analysis tasks [139, 140, 141, 100, 142], where loosening the threshold may lead to consider unsupported variations between networks in different conditions. In what follows we show two cases of (almost) negative results, where the experimental conditions tightly bound the possible differential coexpression network analysis.

Pancreatic Cancer The first example is based on a pancreatic cancer dataset, publicly available at GEO <http://www.ncbi.nlm.nih.gov/geo/>, at the accession number GDS4329 and originally analysed in [127]. The dataset consists of 24 samples from 6 patients suffering from pancreatic ductal adenocarcinoma, divided in 4 subgroup samples, *i.e.*, circulating tumor cells (C), haematological cells (G), original tumour (T), and non-tumoural pancreatic control tissue (P). The aim of the original study was to develop a circulating tumor cells gene signature and to assess its prognostic relevance after surgery, while here we concentrate on the feasibility of a differential coexpression network analysis. Namely, we explore the Pearson correlation networks build separately on the four classes of samples on a specific set of genes S , defined by the differential expression analysis. In particular, the

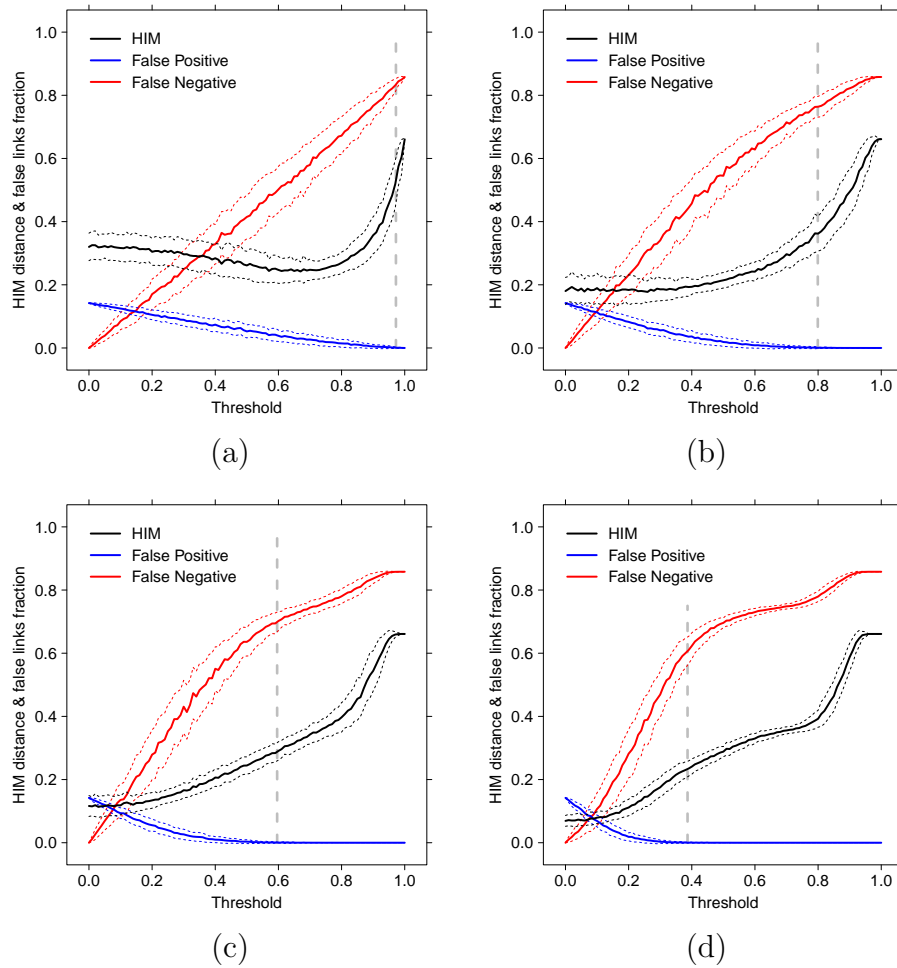


Fig. 4.7: Coexpression inference of the coexpression network from subsampling of the \mathcal{O}_T dataset, on 5 (a), 10 (b), 20 (c) and 50 (d) samples. Solid lines indicate mean over 500 replicates of HIM distance (black), ratio of False Positive (blue) and ratio of False Negative (red); dotted lines of the same colour indicate $\pm 1\sigma$, while grey vertical dashed lines correspond to the secure threshold \bar{p} .

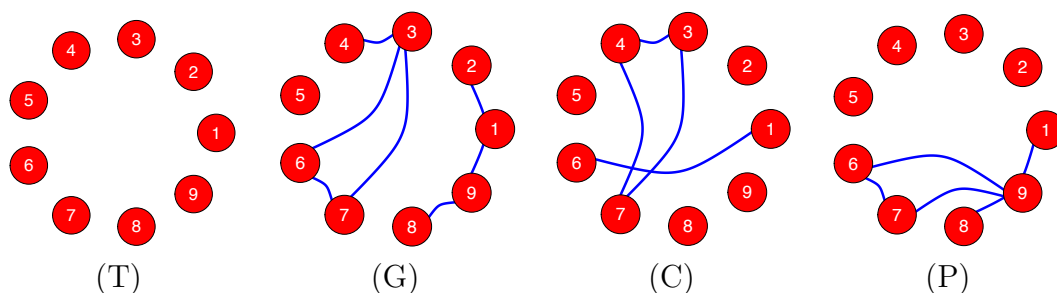


Fig. 4.8: Correlation networks on the set S for the four classes T, G, C and P, thresholded at Pearson correlation coefficient 0.8508.

set S include as nodes the genes resulting upregulated in the C subgroup and associated with both the p38 mitogen-activated protein kinase (MAPK) signaling pathway and the cell motility pathway, which were ranked as the pathways with the highest expression ratio. In details, the nine genes are Talin-1 (TLN1), signal transducer and activator of transcription 3 (STAT3), Vinculin (VCL), CCL5, autocrine motility factor receptor (AMFR), Tropomyosin alpha-4 chain (TPM4), arachidonate 12-lipoxygenase (ALOX12), Rho-guanine nucleotide exchange factor 2 (ARHGEF2), and engulfment and cell motility protein (ELMO1), respectively denoted by $1, \dots, 9$ in the plots.

Following the formula in Definition 4.5, the secure threshold for nine genes and six samples is 0.8508: hard thresholding the four coexpression networks results in the graphs collected in Fig. 4.8. As shown by the plots, the number of edges that result statistically significant over the secure threshold 0.8508 is small: namely 6 for the class G, 4 for the classes C and P and none for the primary tumoral cells T. In particular, the classes C and G share the links VCL–CCL5 and VCL–ALOX12, while P and G share the link TPM4–ALOX12 and P and C have no common links. Clearly, the paucity of statistically significant links prevents any further quantitative comparison: in Fig 4.9 we show, for each networks, the number of links at a given correlation.

Alzheimer data A similar situation occurs with the Alzheimer dataset studied in [128, 129, 130] and available at GEO <http://www.ncbi.nlm.nih.gov/geo/>, at the accession number GSE4226. The dataset collect the expression of peripheral blood mononuclear cells from normal elderly control (NEC) and Alzheimer disease (AD) subjects. The NEC and AD subjects were matched for age and education; the Mini-Mental State Examination (MMSE) [143] was administered to all subjects, and the mean MMSE score of the AD group was significantly lower than that of the NEC subjects. Targets from biological replicates of female (F) and male

(M) NEC and female and male AD were generated and the expression profiles were determined using the NIA Human MGC custom cDNA microarray. Each combinations of the sex and disease phenotypes has a cohort size of seven samples.

The original aim of the studies was the comparison between NEC and AD and the identification of genes with disease and gender expression patterns. In what follows, we show that, given the small sample size, very little can be assessed by a differential coexpression network analysis (see [144] for a recent larger miRNA coexpression study on a cohort of 363 individuals). In particular, from the KEGG Database <http://www.genome.jp/kegg/> [145, 146] we extracted the Alzheimer’s disease pathway in Homo sapiens (KEGG accession hsa05010) and we extracted, from the original 32 genes included in the pathway, the 10 genes spotted on the platform with no missing value across the 28 total samples. The ten resulting genes are apolipoprotein E (APOE), amyloid beta (A4) precursor protein (APP), glycogen synthase kinase 3 beta (GSK3B), cyclin-dependent kinase 5 (CDK5), microtubule-associated protein tau (MAPT), presenilin 2 (Alzheimer disease 4) (PSEN2), amyloid beta A4 precursor protein-binding, family B, member 1 Fe65 (APBB1), lipoprotein lipase (LPL), synuclein alpha non A4 component of amyloid precursor (SNCA) and anterior pharynx defective 1 homolog A (APH1A), numbered from 1 to 10 in the above order in what follows. The resulting heatmap is shown in Fig. 4.10. The coexpression networks for the four combinations of sex (M/F) and disease (NEC/AD) are shown in Fig. 4.11, where the secure threshold is $\bar{p} = 0.8166$. Again, the number of links whose correlation

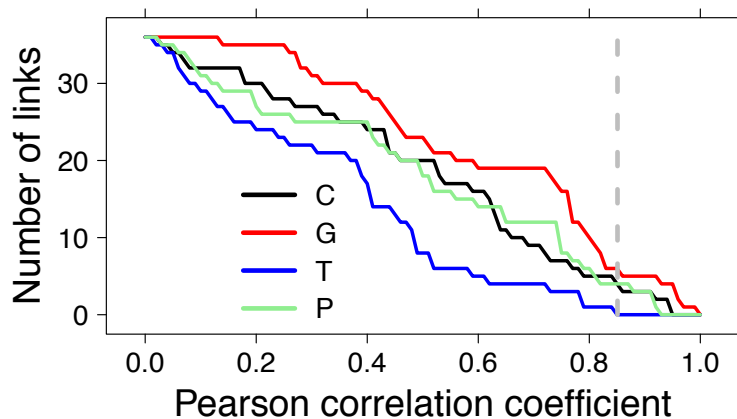


Fig. 4.9: Number of links with correlation values larger than a given threshold for the coexpression networks C, P, T, and G; the vertical gray dashed line corresponds to Pearson correlation 0.8508, the secure threshold for 9 nodes and 6 samples.

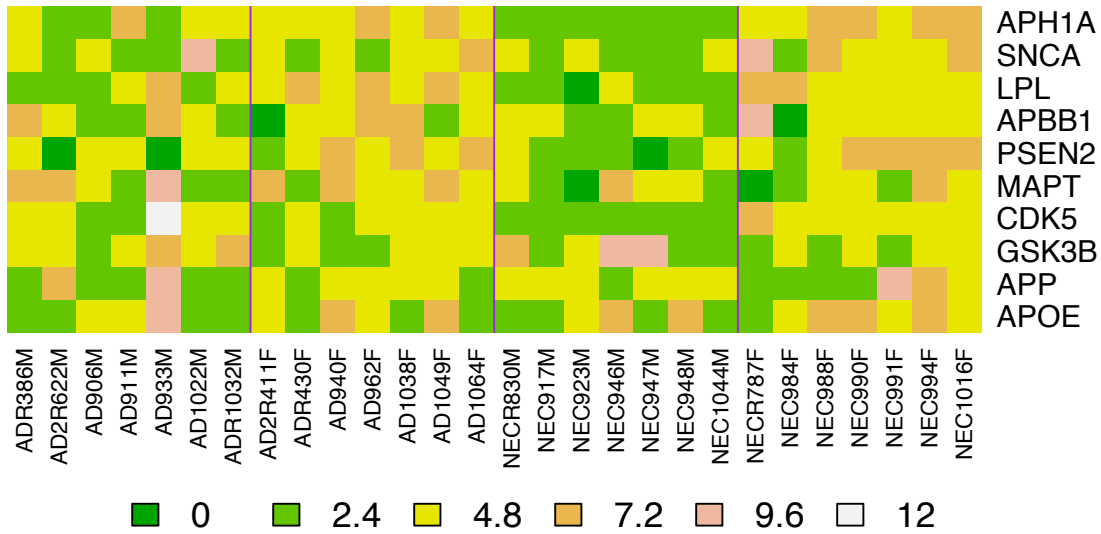


Fig. 4.10: Heatmap of the expression of the ten genes of the Alzheimer pathway on the 28 samples of the Alzheimer dataset. Vertical lines separate samples groups.

value is above the secure threshold is very small: however, all the retrieved links are well known in literature [147] and in dedicated webservers such as GeneMANIA <http://www.genemania.org> [148]. Clearly, if we consider the two main classes AD and NEC, the number of samples grows to 14 for each class, and the threshold \bar{p} can be relaxed down to 0.5943. The two resulting networks are displayed in Fig. 4.12, together with the trend of the HIM distance between AD and NEC as a function of the threshold, both globally and separately for gender, where we can see that the selected threshold, in all cases, falls after the maximal distance between disease and control group. As a major effect emerging when comparing

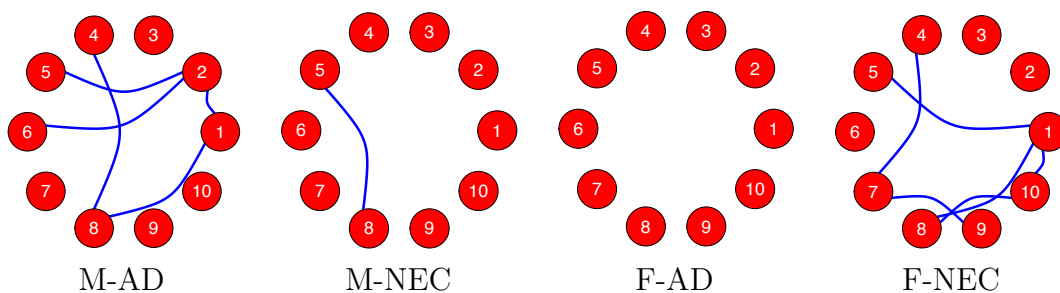


Fig. 4.11: Correlation networks on the Alzheimer dataset S for the four classes M-AD, M-NEC, F-AD, F-NEC, thresholded at Pearson correlation coefficient 0.8166.

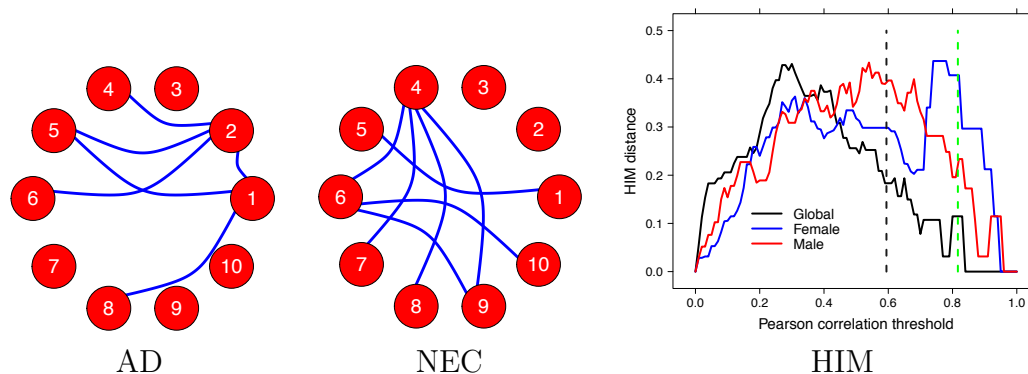


Fig. 4.12: Alzheimer dataset: correlation networks for the two classes AD and NEC, thresholded at Pearson correlation coefficient 0.5943 (AD, NEC) and HIM curve of the distance between AD and NEC network versus the Pearson correlation threshold, globally (black) and separately for Male (red) and Female (blue) patients and controls. Grey dashed vertical line indicates the secure threshold $\bar{p} = 0.5943$ for the global case, while the green line corresponds to the secure threshold $\bar{p} = 0.8166$ for the sex disaggregated case.

the coexpression network of the AD patients versus the NEC individuals we note that the connections between CDK5 and PSEN2, APBB1, LPL, SNCA are lost in the disease networks, while connections appear between APP and APOE, CDK5, MAPT, PSEN2; changing of regulation of CDK5 and APP in AD patients are well known in literature: see for instance [149, 150, 151].

4.4 Conclusion

A simple a priori, theoretical and non-parametric method is proposed for the selection of an hard threshold for the construction of correlation networks. This model is based on the requirements of filtering random data due to noise and reducing the number of false positive, and it is implemented by means of geometric properties of the Pearson correlation coefficient. This new approach can be especially useful in small sample size case, probably the most common situation in profiling studies in functional genomics. Finally, when the number of samples increase, coupling this method with soft thresholding approaches, can help recovering false negative links neglected by too strict thresholds.

Chapter 5

Null model for random markets

5.1 Introduction

Recent years have seen an increasing use of computational techniques for the study of human behaviour and economic systems [152], motivated by these facts in this chapter we extend the ideas about null models of networks to the analysis of institutional rules and behaviour in simple markets.

There is a long tradition in economics on the role that behavioural rules, specifically rationality, have on the dynamics of markets starting from [153] which first noted how individual random behaviour is enough for the emergence of system properties typical of markets. Following works from [154] focused the market dynamics of randomly choosing agents in a double auction [155] [156] institution, i.e. a set of rules typical of financial markets which obliges sellers to propose asks lower than the lowest ask proposed and sellers to bids more than the highest bid. Leading to interesting insights on the ability of the double auction rules to efficiently extract wealth from exchanges in absence of optimal behaviour. More recently the same concept has been applied by [157] to deduce the departure from randomness of trades behaviour in financial markets. Here, inspired from the definition of microeconomic system of [158], we extend this framework by defining a generative network model of stochastic trading that under specific assumptions over the distribution functions generating the behaviour and the matching of agents can be suitably be used as a null-model.

We model markets as a bipartite temporal network of agents linked by the contracts they sign, limiting to the simple case of two agents classes, buyers and sellers, that trades one commodity for a currency,. At each time step, edges are generated by an underlying matching and bargaining process. The first is expressed as a probability distribution over the permutation set of buyer-seller dyads while the latter is modelled as the probability density function of contracts that increase the

wealth of both agents under random bargaining proposals. In our final formulation the probability of links creation is constant during the process, but different among all the possible couples, leading to a simple expression of the likelihood function for the temporal sequence of networks.

5.2 Random Market

5.2.1 Notation

With $\mathcal{S} = (s_1, \dots, s_T)$ we will represent the T sellers while with $\mathcal{B} = (b_1, \dots, b_T)$ the T buyers. Let L be the maximum number of timesteps considered in the system described above. Let $\mathcal{M}^{(k)} = \{(i, \sigma(i)) \mid i \in 1, \dots, T\}$ be the random matching at the time k where σ is an element of \mathcal{S}_T the set of all permutation of T elements. Let m and M respectively be the maximum and the minimum price in the market. Let also s_i and b_i be respectively the minimum price acceptable for the sellers and the maximum for the buyers, which correspond the bound below (above) which they would not gain from the trade. Let $X_i = X$ be the the random variable associated to the price of the i -th seller with support $\Omega_X = [\bar{s}, M]$ and density function $f_X(x)$ and with $Y_i = Y$ we will indicate the random variable associated to the price of the i -th buyer with support $\Omega_Y = [m, \bar{b}]$ and density function $f_Y(x)$. Let finally $Z_i = Z$ be the random variable associate to $X - Y$, i.e. the difference of price between the seller and the buyer. An edge between i and $\sigma(i)$ is created if and only if $Z \leq 0$.

5.2.2 Uniform Random Market

We first start by the definition of a random market where preferences are constant across buyers and seller, agents choose uniformly in their acceptable range and they are matched uniformly. That is $b = \bar{b}$ and $s = \bar{s}$ respectively for each buyer and seller, with $\bar{s} < \bar{b}$, and X and Y are independent and uniforms.

Then we can study the wealth extracted by the interaction of two agents studying Z :

1. if $\bar{s} - m < M - \bar{b}$:

$$f_Z(z) = \begin{cases} \frac{z - (\bar{s} - \bar{b})}{(M - \bar{s})(\bar{b} - m)} & \bar{s} - \bar{b} \leq z < \bar{s} - m \\ \frac{z}{M - \bar{s}} & \bar{s} - m \leq z < M - \bar{b} \\ \frac{-z + (M - m)}{(M - \bar{s})(\bar{b} - m)} & M - \bar{b} \leq z < M - m \\ 0 & \text{otherwise} \end{cases}$$

2. if $\bar{s} - m > M - \bar{b}$:

$$f_Z(z) = \begin{cases} \frac{z - (\bar{s} - \bar{b})}{(M - \bar{s})(\bar{b} - m)} & \bar{s} - \bar{b} \leq z < M - \bar{b} \\ \frac{z}{\bar{b} - m} & M - \bar{b} \leq z < \bar{s} - m \\ \frac{-z + (M - m)}{(M - \bar{s})(\bar{b} - m)} & \bar{s} - m \leq z < M - m \\ 0 & \text{otherwise} \end{cases}$$

3. if $\bar{s} - m = M - \bar{b}$:

$$f_Z(z) = \begin{cases} \frac{z - (\bar{s} - \bar{b})}{(M - \bar{s})(\bar{b} - m)} & \bar{s} - \bar{b} \leq z < M - \bar{b} \\ \frac{-z + (M - m)}{(M - \bar{s})(\bar{b} - m)} & \bar{s} - m \leq z < M - m \\ 0 & \text{otherwise} \end{cases}$$

Where

$$\begin{aligned} c_1 &= \frac{(\bar{s} - \bar{b})^2}{2(M - \bar{s})(\bar{b} - m)} & (5.1) \\ c_3 &= \frac{-(M - m)^2}{2(M - \bar{s})(\bar{b} - m)} + 1 \\ c_2 &= \frac{(\bar{s} - m)^2 - 2(\bar{s} - \bar{b})(\bar{s} - m)}{2(M - \bar{s})(\bar{b} - m)} + \frac{\bar{s} - m}{M - \bar{s}} + c_1 & \text{in case 1} \\ &= \frac{(M - \bar{b})^2 - 2(\bar{s} - \bar{b})(M - \bar{b})}{2(M - \bar{s})(\bar{b} - m)} + \frac{M - \bar{b}}{\bar{b} - m} + c_1 & \text{in case 2.} \end{aligned}$$

In this simple situation, $P(Z \leq 0) = c_1$ in any case because $\bar{s} - m \geq 0$ and $M - \bar{b} \geq 0$.

Moreover, from f_z it is possible to compute the mean value \hat{Z} of Z as:

$$\hat{Z} = \frac{1}{c_1} \int_{\bar{s} - \bar{b}}^0 z f_z(z) dz = \frac{1}{c_1} \int_{\bar{s} - \bar{b}}^0 \frac{z^2 - z(\bar{s} - \bar{b})}{(M - \bar{s})(\bar{b} - m)} dz = \frac{(\bar{s} - \bar{b})^3}{6c_1(M - \bar{s})(\bar{b} - m)} = \frac{\bar{s} - \bar{b}}{3},$$

and define the mean value of the trade $v := -\frac{\hat{Z}}{2}$. We are interested to study also the profit of the sellers and the buyers after L steps. We have to compute the mean value of the sellers (and buyers) when a trade is performed ($Z \leq 0$):

$$\begin{aligned} \hat{s} &:= E(X|Z \leq 0) = E(X|X - Y \leq 0) = E(X|X \leq Y) \\ &= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} x f_{x,y}(x, y) dy dx \\ &= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} \frac{x}{(M - \bar{s})(\bar{b} - m)} dy dx = \frac{2\bar{s} + \bar{b}}{3} \end{aligned}$$

$$\begin{aligned}
\hat{b} &:= E(Y|Z \leq 0) = E(Y|X - Y \leq 0) = E(Y|Y \geq X) \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} y f_{x,y}(x, y) dy dx \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} \frac{y}{(M - \bar{s})(\bar{b} - m)} dy dx = \frac{\bar{s} + 2\bar{b}}{3}.
\end{aligned}$$

Finally we can estimate the mean profit of the sellers, p_s , and buyers, p_b and the mean weight of each edge \bar{v} as:

$$\begin{aligned}
p_s &= (\hat{s} + v - \bar{s})Lc_1 = \left(\frac{2\bar{s} + \bar{b}}{3} + \frac{\bar{b} - \bar{s}}{6} - \bar{s} \right) c_1 L = \frac{\bar{b} - \bar{s}}{2} c_1 L \\
p_b &= (\bar{b} - \hat{b} - v)Lc_1 = \left(\bar{b} - \frac{2\bar{b} + \bar{s}}{3} + \frac{\bar{b} - \bar{s}}{6} \right) c_1 L = \frac{\bar{b} - \bar{s}}{2} c_1 L \\
\bar{v} &= -\hat{Z}Lc_1 = \frac{Lc_1(\bar{b} - \bar{s})}{3}.
\end{aligned}$$

Example 2. We simulate a simple random market with parameters: $T = 6, m = 10, M = 110, \bar{b} = 60, \bar{s} = 30$ and follow its evolution for $L = 10000$ timestep. In Fig.5.1 we can see some snapshots of the entire process.

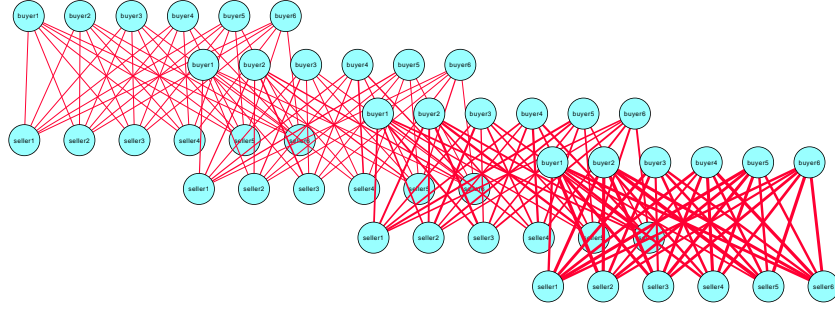


Fig. 5.1: Evolution of the bipartite weighted graph.

5.2.3 Random market with exponential and negative exponential distribution

More sophisticated hypothesis on behaviour can be implemented by modifying the distribution of agents choices. As an example we show how the intuitive idea that buyers tries to buy low and sellers try to sell high by using skewed distributions. Let suppose that the sellers follow a truncate exponential distribution and the buyers a negative exponential distribution, in other words:

$$f_X(x) = k_1 e^{\mu x} = \frac{\mu e^{\mu x}}{e^{M\mu} - e^{s\mu}}, \quad f_Y(y) = k_2 e^{-\lambda y} = \frac{\lambda e^{-\lambda y}}{e^{-m\lambda} - e^{-b\lambda}}.$$

Since we are interested at the market only when $X - Y \leq 0$, we shall compute $F_Z(z)$ only for negative values of z :

$$\begin{aligned} F_Z(z) &= \int \int_D f_{X,Y}(x,y) dx dy \quad \text{where } D \text{ is the domain for which } X < Y \\ &= \int \int_D f_X(x) f_Y(y) dx dy \quad \text{using the independence between } X \text{ and } Y \\ &= k_1 k_2 \int_{\bar{s}}^{\bar{b}+z} \int_{x-z}^{\bar{b}} e^{-\lambda y} e^{\mu x} dy dx \\ &= \frac{k_1 k_2}{\lambda} \int_{\bar{s}}^{\bar{b}+z} e^{-\lambda(x-z)+\mu x} - e^{\mu x - \lambda \bar{b}} dx \\ &= \frac{k_1 k_2}{\lambda} \left[\frac{e^{\bar{b}(\mu-\lambda)+\mu z} - e^{\bar{s}(\mu-\lambda)+\lambda z}}{\mu - \lambda} - \frac{e^{\bar{b}(\mu-\lambda)+\mu z} - e^{\mu \bar{s} - \lambda \bar{b}}}{\mu} \right] \\ &= \frac{k_1 k_2}{\lambda \mu (\mu - \lambda)} \left[\mu e^{\mu \bar{s} - \lambda \bar{b}} - \mu e^{\bar{s}(\mu-\lambda)+\lambda z} + \lambda e^{\bar{b}(\mu-\lambda)+\mu z} - \lambda e^{\mu \bar{s} - \lambda \bar{b}} \right] \quad (5.2) \\ f_Z(z) = F'_Z(z) &= \frac{k_1 k_2}{\mu - \lambda} \left[e^{\bar{b}(\mu-\lambda)+\mu z} - e^{\bar{s}(\mu-\lambda)+\lambda z} \right]. \end{aligned}$$

From Eq.5.2 we get:

$$c_1 = F_Z(0) = \frac{k_1 k_2}{\lambda \mu (\mu - \lambda)} \left[\mu e^{\mu \bar{s} - \lambda \bar{b}} - \mu e^{\bar{s}(\mu-\lambda)} + \lambda e^{\bar{b}(\mu-\lambda)} - \lambda e^{\mu \bar{s} - \lambda \bar{b}} \right].$$

We can compute now the mean value of z as:

$$\begin{aligned}
\hat{Z} &= \frac{1}{c_1} \int_{\bar{s}-\bar{b}}^0 z f_z(z) dz = \frac{k_1 k_2}{(\mu - \lambda) c_1} \int_{\bar{s}-\bar{b}}^0 z \left[e^{\bar{b}(\mu-\lambda)+\mu z} - e^{\bar{s}(\mu-\lambda)+\lambda z} \right] dz \\
&= \frac{k_1 k_2}{(\mu - \lambda) c_1} \left[\frac{1}{\mu} \left(z e^{\bar{b}(\mu-\lambda)+\mu z} - \frac{e^{\bar{b}(\mu-\lambda)+\mu z}}{\mu} \right) - \frac{1}{\lambda} \left(z e^{\bar{s}(\mu-\lambda)+\lambda z} - \frac{e^{\bar{s}(\mu-\lambda)+\lambda z}}{\lambda} \right) \right] \Big|_{\bar{s}-\bar{b}}^0 \\
&= \frac{k_1 k_2}{(\mu - \lambda) c_1} \left[\frac{1}{\mu} \left((\bar{b} - \bar{s}) e^{\mu \bar{s} - \lambda \bar{b}} + \frac{e^{\mu \bar{s} - \lambda \bar{b}} - e^{\bar{b}(\mu-\lambda)}}{\mu} \right) - \frac{1}{\lambda} \left((\bar{b} - \bar{s}) e^{\mu \bar{s} - \lambda \bar{b}} + \frac{e^{\mu \bar{s} - \lambda \bar{b}} - e^{\bar{s}(\mu-\lambda)}}{\lambda} \right) \right] \\
&= \frac{k_1 k_2 e^{\mu \bar{s} - \lambda \bar{b}}}{(\mu - \lambda) c_1} \left[\frac{1}{\mu} \left((\bar{b} - \bar{s}) + \frac{1 - e^{\mu(\bar{b}-\bar{s})}}{\mu} \right) - \frac{1}{\lambda} \left((\bar{b} - \bar{s}) + \frac{1 - e^{\lambda(\bar{b}-\bar{s})}}{\lambda} \right) \right].
\end{aligned}$$

As before we can compute \hat{s} and \hat{b} :

$$\begin{aligned}
\hat{s} &:= E(X|Z \leq 0) = E(X|X - Y \leq 0) = E(X|X \leq Y) \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} x f_{x,y}(x, y) dy dx \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} k_1 k_2 x e^{-\lambda y} e^{\mu x} dy dx \\
&= \frac{k_1 k_2}{c_1 \lambda} \int_{\bar{s}}^{\bar{b}} x e^{(\mu-\lambda)x} - e^{\mu x - \lambda \bar{b}} dx \\
&= \frac{k_1 k_2}{c_1 \lambda} \left[\frac{x e^{(\mu-\lambda)x}}{\mu - \lambda} - \frac{e^{(\mu-\lambda)x}}{(\mu - \lambda)^2} - \frac{x e^{\mu x - \lambda \bar{b}}}{\mu} + \frac{e^{\mu x - \lambda \bar{b}}}{\mu^2} \right] \Big|_{\bar{s}}^{\bar{b}} \\
&= \frac{k_1 k_2}{c_1 \lambda} \left[\frac{\bar{b} e^{(\mu-\lambda)\bar{b}}}{\mu - \lambda} - \frac{e^{(\mu-\lambda)\bar{b}}}{(\mu - \lambda)^2} - \frac{\bar{b} e^{\mu \bar{b} - \lambda \bar{b}}}{\mu} + \frac{e^{\mu \bar{b} - \lambda \bar{b}}}{\mu^2} - \frac{\bar{s} e^{(\mu-\lambda)\bar{s}}}{\mu - \lambda} + \frac{e^{(\mu-\lambda)\bar{s}}}{(\mu - \lambda)^2} + \frac{\bar{s} e^{\mu \bar{s} - \lambda \bar{b}}}{\mu} - \frac{e^{\mu \bar{s} - \lambda \bar{b}}}{\mu^2} \right]
\end{aligned}$$

$$\begin{aligned}
\hat{b} &:= E(Y|Z \leq 0) = E(Y|X - Y \leq 0) = E(Y|Y \geq X) \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} y f_{x,y}(x, y) \, dy \, dx \\
&= \frac{1}{c_1} \int_{\bar{s}}^{\bar{b}} \int_x^{\bar{b}} k_1 k_2 y e^{-\lambda y} e^{\mu x} \, dy \, dx \\
&= \frac{k_1 k_2}{c_1 \lambda^2} \int_{\bar{s}}^{\bar{b}} \lambda x e^{x(\mu-\lambda)} + e^{x(\mu-\lambda)} - \lambda \bar{b} e^{x\mu-\lambda\bar{b}} - e^{x\mu-\lambda\bar{b}} \, dx \\
&= \frac{k_1 k_2}{c_1 \lambda^2} \left[\frac{\lambda x e^{x(\mu-\lambda)}}{\mu-\lambda} - \frac{\lambda e^{x(\mu-\lambda)}}{(\mu-\lambda)^2} + \frac{e^{x(\mu-\lambda)}}{\mu-\lambda} - \frac{\lambda \bar{b} e^{x\mu-\lambda\bar{b}}}{\mu} - \frac{e^{x\mu-\lambda\bar{b}}}{\mu} \right] \Big|_{\bar{s}}^{\bar{b}} \\
&= \frac{k_1 k_2}{c_1 \lambda^2} \left[\frac{\lambda \bar{b} e^{\bar{b}(\mu-\lambda)}}{\mu-\lambda} - \frac{\lambda e^{\bar{b}(\mu-\lambda)}}{(\mu-\lambda)^2} + \frac{e^{\bar{b}(\mu-\lambda)}}{\mu-\lambda} - \frac{\lambda \bar{b} e^{\bar{b}\mu-\lambda\bar{b}}}{\mu} - \frac{e^{\bar{b}\mu-\lambda\bar{b}}}{\mu} - \frac{\lambda \bar{s} e^{\bar{s}(\mu-\lambda)}}{\mu-\lambda} + \frac{\lambda e^{\bar{s}(\mu-\lambda)}}{(\mu-\lambda)^2} - \frac{e^{\bar{s}(\mu-\lambda)}}{\mu-\lambda} + \frac{\lambda \bar{b} e^{\bar{s}\mu-\lambda\bar{b}}}{\mu} + \frac{e^{\bar{s}\mu-\lambda\bar{b}}}{\mu} \right].
\end{aligned}$$

Finally we can estimate the mean profit of the sellers, p_s , and buyers, p_b , as:

$$\begin{aligned}
p_s &= (\hat{s} + v - \bar{s}) L c_1 \\
p_b &= (\bar{b} - \hat{b} - v) L c_1.
\end{aligned}$$

Notice that it is possible to recover \bar{s} and \bar{b} (and so p_s and p_b) also for other probability distributions (and a closed form can be computed based on the kind of integral functions we are dealing with).

5.2.4 Heterogeneous Agents

Assumption on the heterogeneity of preferences are implementable by modeling choices with distributions over different different supports. Here we study the case of uniform distributions over heterogeneous supports:

Let B and S be two random variables with support $\Omega_B = \Omega_S = [m, M]$ and with uniform density functions f_B and f_S . i.e. for $i \in 1, \dots, T$ the random variable X_i has support $\Omega_{X_i} = [\bar{s}_i, M]$ and Y_i has support $\Omega_{Y_i} = [m, \bar{b}_i]$ for some value of \bar{s}_i and b_i choosing according to f_S and f_B . Let $Z_{i,j} = X_i - Y_j$ the difference between the i -th seller and the j -th buyer. Let $W = S - B$ the difference between the two random variables introduced above. The value of $Z_{i,j}$ depends on the distribution of W so it is necessary to study first this random variable. Using the same argumentation showed above we get:

$$f_W(w) = \begin{cases} \frac{w-(m-M)}{(M-m)^2} & m-M \leq w < 0 \\ \frac{-w+(M-m)}{(M-m)^2} & 0 \leq w < M-m \\ 0 & \text{otherwise} \end{cases}$$

$$F_W(w) = \begin{cases} 0 & w < m-M \\ \frac{w^2-2w(m-M)}{2(M-m)^2} + \frac{1}{2} & m-M \leq w < 0 \\ \frac{-w^2+2w(M-m)}{2(M-m)^2} + \frac{1}{2} & 0 \leq w < M-m \\ 1 & M-m \leq z \end{cases}$$

These considerations lead us to compute $P(W \leq 0) = F_W(0) = \frac{1}{2}$ and the mean value of W , $\hat{W} = \frac{m-M}{3}$ and the mean value of \bar{b} and \bar{s} , whenever $W \leq 0$, indicated with $\tilde{b} = \frac{m+2M}{3}$ and $\tilde{s} = \frac{2m+M}{3}$. Let $\delta(a, b)$ a function such that $\delta(a, b) = 1$ if $a > b$ and $\delta(a, b) = 0$ if $a \leq b$. If the values \bar{b}_i, \bar{s}_j are known using the same augmentations showed above we can compute $V = \bar{v}_{i,j}$, $i, j \in 1, \dots, T$ as:

$$V = L \begin{pmatrix} \delta(\bar{b}_1, \bar{s}_1) \frac{c_{1,1}(\bar{s}_1 - \bar{b}_1)}{3T} & \dots & \delta(\bar{b}_T, \bar{s}_1) \frac{c_{1,T}(\bar{s}_1 - \bar{b}_T)}{3T} \\ \vdots & \ddots & \vdots \\ \delta(\bar{b}_1, \bar{s}_T) \frac{c_{T,1}(\bar{s}_T - \bar{b}_1)}{3T} & \dots & \delta(\bar{b}_T, \bar{s}_T) \frac{c_{T,T}(\bar{s}_T - \bar{b}_T)}{3T} \end{pmatrix}$$

where $c_{i,j}$ is the value c_1 of Eq.(1) with parameters \bar{s}_i, \bar{b}_j .

Viceversa, if the values \bar{b}_i, \bar{s}_j are unknown, the mean value of V is

$$\hat{V} = \frac{1}{2} L \bar{c}_1 \begin{pmatrix} \frac{\tilde{b} - \tilde{s}}{3T} & \dots & \frac{\tilde{b} - \tilde{s}}{3T} \\ \vdots & \ddots & \vdots \\ \frac{\tilde{b} - \tilde{s}}{3T} & \dots & \frac{\tilde{b} - \tilde{s}}{3T} \end{pmatrix}$$

where \bar{c}_1 is the value c_1 of Eq. 5.1 with parameters \tilde{b}, \tilde{s} .

5.3 Likelihood Function

5.3.1 Uniform with constant preferences case

In this section we will derive the likelihood function in the case of uniform distributions. In order to do this, we will assume that M and m are given. Moreover we suppose that the buyers and sellers follow a uniform distribution for their prices and that all matching are equiprobable.

The observed data is the effective price of exchange between the seller and the buyer. This amount of money is, in terms of random variables, equal to $W = Y + \frac{Z}{2}$ if $Z < 0$ i.e. the exchange success or equal to $W = 0$ otherwise. These two events have probability respectively c_1 and $1 - c_1$. In order to write the likelihood function we need to compute the probability distribution of W conditioned to $Z < 0$, using that X and Y are independent and so the join probability f_{XY} is the product $f_X f_Y$:

$$\begin{aligned}
F_{W|Z \leq 0}(t) &= P\left(Y + \frac{Z}{2} \leq t \mid Z \leq 0\right) = \frac{P(X + Y \leq 2t \vee X - Y \leq 0)}{c_1} \\
&= \int_{\bar{s}}^t \int_x^{2t-x} f_Y(y) f_X(x) dy dx \\
&= \frac{1}{(M - \bar{s})(\bar{b} - m)c_1} \int_{\bar{s}}^t \int_x^{2t-x} 1 dy dx \\
&= \frac{1}{(M - \bar{s})(\bar{b} - m)c_1} \int_{\bar{s}}^t 2t - 2x dx \\
&= \frac{t^2 - 2t\bar{s} - \bar{s}^2}{(M - \bar{s})(\bar{b} - m)c_1} \\
&= \frac{2(t^2 - 2t\bar{s} - \bar{s}^2)}{(\bar{s} - \bar{b})^2} \\
f_{W|Z \leq 0}(t) &= \frac{4(t - \bar{s})}{(\bar{s} - \bar{b})^2}
\end{aligned}$$

Let now $N_+ = \{x_i\}_i$ set of non-null observed prises and let n_0 the number failed bargain. The log-likelihood function related to W is:

$$\mathcal{L}(\bar{s}, \bar{b} | \{x_i\}_i) = \sum_{x_i \in N_+} \log \frac{4(x_i - \bar{s})}{(\bar{s} - \bar{b})^2} + n_0 \log \frac{2(M - \bar{s})(\bar{b} - m) - (\bar{s} - \bar{b})^2}{2(M - \bar{s})(\bar{b} - m)}. \quad (5.3)$$

So we can recover the initial data \bar{b}, \bar{s} maximising $\mathcal{L}(\bar{s}, \bar{b} | \{x_i\}_i)$ under the following constrains:

- $m < \bar{s} < \bar{b} < M$,
- $\bar{b} > \max(x_i; x_i > 0)$,
- $\bar{s} < \min(x_i; x_i > 0)$.

5.3.2 Different preferences and matching probabilities

Analogously to the previous section we extend the specification of the likelihood function to the more general case where:

- buyers and sellers may have different preferences,
- the matching function $g(s_i, b_j)$ between the i -th seller and j -th buyer is not uniform, but depends on i and j , i.e. $g(s_i, b_j) = g(i, j)$. An other way to see this matching function is like a stochastic matrix (a non-negative matrix s.t. the row-sum and column-sum is equal to one and the (i, j) -th entry represent the probability to match the i -th seller and the j -th buyer.

In this case the likelihood function has the following expression:

$$\begin{aligned} \mathcal{L}(\{\bar{s}_i\}_i, \{\bar{b}_j\}_j | \{x_i\}_i) &= \sum_{i=1}^T \sum_{j=1}^T \left(\sum_{x_i \in N_{i,j}} \log \frac{4(x_i - \bar{s}_i)}{(\bar{s}_i - \bar{b}_j)^2} \right. \\ &\quad + (g(i, j)L - |N_{i,j}|) \log \frac{2(M - \bar{s}_i)(\bar{b}_j - m) - (\bar{s}_i - \bar{b}_j)^2}{2(M - \bar{s}_i)(\bar{b}_j - m)} \\ &\quad \left. + \sum_{k \neq j} g(i, k)(L - |N_{i,j}|) \log \frac{2(M - \bar{s}_i)(\bar{b}_k - m) - (\bar{s}_i - \bar{b}_k)^2}{2(M - \bar{s}_i)(\bar{b}_k - m)} \right). \end{aligned}$$

Where $g(i, j)L - |N_{i,j}|$ is the number of failed bargains between the seller i and the buyer j and $g(i, k)(L - |N_{i,j}|)$ is the number of failed bargains between the seller i and the buyer k .

5.4 Conclusion

We propose a flexible framework to model simple markets in a probabilistic fashion for the case of agents with linear preferences, and showed how different intuitions over behavioural or institutional rules can be introduced modifying the parameters of the model. The successful ability of the model in retrieving its own parameters from simulation suggests the possibility to generate interesting experiments by fitting the model on ad hoc simulated and human data. As an example it might be possible to measure the impact of complicated institutional rules like the double auction by fitting the random model to the data: a good fit would indicate a small effect from the new rules while a bad fit would indicate radical changes generated by the rules. Analogously it might be possible to investigate the distance of human behaviour from randomness, *i.e.* a null model, by fitting the model to human experimental data.

Chapter 6

Graph and game modelization: TTT solitaire

The TTT (Target the Two) solitaire was first introduced by Cohen and Bacadayan in 1994 [19] to show how *individuals store their components of organisational routines in procedural memory* and so that *the procedural memory may be the source of distinctive properties reported by observers of organisational routines*. This solitaire was used two years later by Narduzzo and Egidi in [21] in order to understand how individuals tend to divide into subproblems and to *routinize their behaviours* accordingly to an induced strategy. The authors show that the game admits a large number of configurations and some of them can be more easily solved by adopting one (locally optimal) strategy, while others can be easily solved by a different, locally optimal, strategy. The authors focused their attention to the different choice made by the players after different training sessions, in particular they observe the persistence of some player to use only the strategy learnt before, experimentally showing a certain degree of routinisation in players' behaviour. In 2003 Egidi [20] formalise the concept of strategy, subproblems and categories showing that *decomposition patterns are usually non invariant and therefore the final result is not an optimal strategy*. Egidi showed that when the initial problem is divided into smaller subproblems (using heuristic decomposition patterns), the players (consciously or not) introduces hidden sub-optimalties also if each subproblem is solved in the optimal way. Here we will embed the TTT solitaire in its natural graph representational world and we will try to formalise some concepts introduced in the mentioned bibliography under the light of network theory. In this framework, there are natural formalisations for the ideas aforementioned that allows a practically systematic treatise of pattern decomposition strategies. Under these new elements, it is quite easy to show that there is not a non-trivial decomposition pattern solving this solitaire in the optimal way, but we can show that more elaborate is a strategy, less we move away from the optimum.

After a brief introduction about the solitaire in Section 6.2 we show a simple algorithm for generating the whole graph game. After that, the definition of abstraction and strategies are introduced and finally a new approach solve the solitaire in the case of hidden cards is showed.

6.1 Description

The TTT solitaire is played using six cards $2\clubsuit, 3\clubsuit, 4\clubsuit, 2\heartsuit, 3\heartsuit$ and $4\heartsuit$ (these cards form the set we will call \mathcal{P}) arranged into two lines of three cards. We will indicate the positions of the cards using numbers from 1 to 6, *i.e.* 1,2 and 3 for the first line from left to right and 4,5 and 6 for the second line. There are three special positions: 1 is called *colour* (CC), 2 is called *target* and 3 is the *number* (NN). At the beginning of the game we choose a starting point S between CC and NN, a target card (originally $2\heartsuit$) and we arrange cards face-up in position 1, 2, 3, 5 and face-down for 4 and 6. At each turn we can perform one of the following moves: pass (*PASS*), switch the card from S to 4, 5 or 6 without restriction ($N_4, C_4, N_5, C_5, N_6, C_6$), switch from S to 2 (N_2, C_2) only if the two cards agree (if S=CC then 2 must have the same colour of S, otherwise the same number). After we switch two cards (or we pass) we must change S (if S is CC then the next move starts with NN and viceversa). The goal is to put the target card in position 2 using the least number of moves. We will call $\mathcal{M} = \{PASS, N_2, C_2, N_4, C_4, N_5, C_5, N_6, C_6\}$ the set of **possible moves**.

6.2 Graph game construction

In this first part we will assume to play with the cards in position 4 and 6 face-up like the others. In this section we will see how to build the directed graph \mathcal{G} in which each node represents a configuration and each edge the move between two configurations.

We can represent a generic configuration using 7 symbols, six for the cards and one for S. It is easy to see that there are $2 \cdot 5 \cdot 5! + 2 \cdot 4! + 4! = 1272$ possible configurations:

- $2 \cdot 5 \cdot 5!$, considering $2\heartsuit$ not in position 2 there are 5 possibilities for position 2, and $5!$ for the others; moreover we must multiply these position by 2, the different values of S.
- $2 \cdot 4!$, considering $2\heartsuit$ in target and S=NN we must have a red card in 1 (there are 2 possibilities) and $4!$ for the others.

- 4!, considering $2\heartsuit$ in target and $S=CC$ we must have $2\clubsuit$ in 3 and 4! for the others.

All the configurations, composing the set indicated with \mathcal{C} , are stored in 7-uples in which the first six slots indicate the cards in the respective positions and the last indicates the status of S. As a matter of notation, if $c \in \mathcal{C}$ we will indicate with $c[i]$ the card in i -th position if $i \leq 6$ or the value of S if $i = 7$. Sometimes we will write also $c \in \mathcal{G}$.

Definition 12. A set $\mathcal{T} \subset \mathcal{C}$ is called **target set**. The most important target set is indicated with \mathcal{T}^* and defined as $\mathcal{T}^* = \{c \in \mathcal{C} \mid c[2] = 2\heartsuit\}$.

Let $m \in \mathcal{M}$ and $c \in \mathcal{C}$, with $m(c)$ we indicate the position reached from c performing the move m if this move is possible, the empty set otherwise. If $(m_1, \dots, m_k) = M \in \mathcal{M}^k$ for some $k \in \mathbb{N}$ and $c \in \mathcal{C}$ with the expression $M(c)$ we indicate the position $m_k(m_{k-1}(\dots(m_1(c)\dots)))$ if all moves are possible, the empty set otherwise.

The next step is to build the edge's set, indicate with $E(\mathcal{G})$. For each pair of different configurations, p_1 and p_2 , we can state if there is a move m between the two configurations or not following these steps:

1. if $p_1[2] = 2\heartsuit$: $m = \emptyset$ else
2. if $p_1[7] = p_2[7]$: $m = \emptyset$ else
3. if $p_1[i] = p_2[i] \forall i \leq 6$: $m = PASS$ else
4. if $\#\{i \mid p_1[i] = p_2[i]\} \neq 4$: $m = \emptyset$ else
5. if $p_1[1] = p_2[1]$: $m = \emptyset$ else
6. if $p_1[3] = p_2[3]$: $m = \emptyset$ else
7. if $p_1[7] = CC$:
 - if $p_1[1] = p_2[2]$ and $p_1[2], p_1[1]$ have a different colour: $m = \emptyset$ else
 - $m = C_x$ where x is such that $p_1[x] = p_2[1]$.
8. else:
 - if $p_1[3] = p_2[2]$ and $p_1[2], p_1[3]$ have a different number: $m = \emptyset$ else
 - $m = N_x$ where x is such that $p_1[x] = p_2[3]$.

Fact 1

The **game-graph** \mathcal{G} (in Fig. 6.1) is a digraph in which the node set $V(\mathcal{G}) = \mathcal{C}$ and $E(\mathcal{G})$ as howed above. This graph has 1272 nodes and 5159 edges.

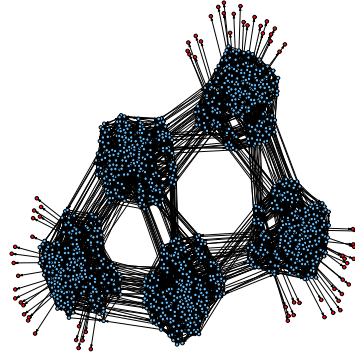


Fig. 6.1: Representation of \mathcal{G} . In red we can see the target set \mathcal{T}^* .

Definition 13. Let \mathcal{T} be a non empty target set. The set defined as $L_n = \{c \in \mathcal{C} \text{ s.t. } d(c, \mathcal{T}) = n\}$ is called **n-layer**; where $d(c, \mathcal{T}) = \min_{d \in \mathcal{T}} d(c, d)$ and $d(c, d)$ the standard geodesic distance.

Fact 2

Let $\bar{d} = \max_{c \in \mathcal{C}} d(c, \mathcal{T})$ then, for $c \in L_n$ and $t \in \mathcal{T}$, the following properties are verified:

- $L_{\bar{d}+n} = \emptyset$ for all $n > 0$ and $L_n \cap L_m = \emptyset$ if $m \neq n$.
- $d(c, L_{n-1}) = 1$ for all $2 \leq n \leq \bar{d}$ and $c \in L_n$.
- If (c, c_n, \dots, c_1, t) is the sequence of a shortest path then $d(c, \mathcal{T}) = n + 1$ and $c_i \in L_i$ for all i .
- with the target set \mathcal{T}^* we have $\bar{d} = 5$ and the cardinality of its layers:

$\#\mathcal{T}$	$\#L_1$	$\#L_2$	$\#L_3$	$\#L_4$	$\#L_5$
72	72	96	312	552	168

In Fig.6.2(a) we can see a representation of the graph \mathcal{G} in which the different layers have different colours.

Definition 14. Let \mathcal{G} and \mathcal{T}^* be as before. The **shortest path spanning graphs** \mathcal{G}^* is the graph such that $V(\mathcal{G}^*) = \mathcal{C}$ and in which there are only edges from L_m to L_{m-1} for all $m = 1, \dots, 5$.

If the value or the number of a card is not important, we will indicate it with the symbol #.

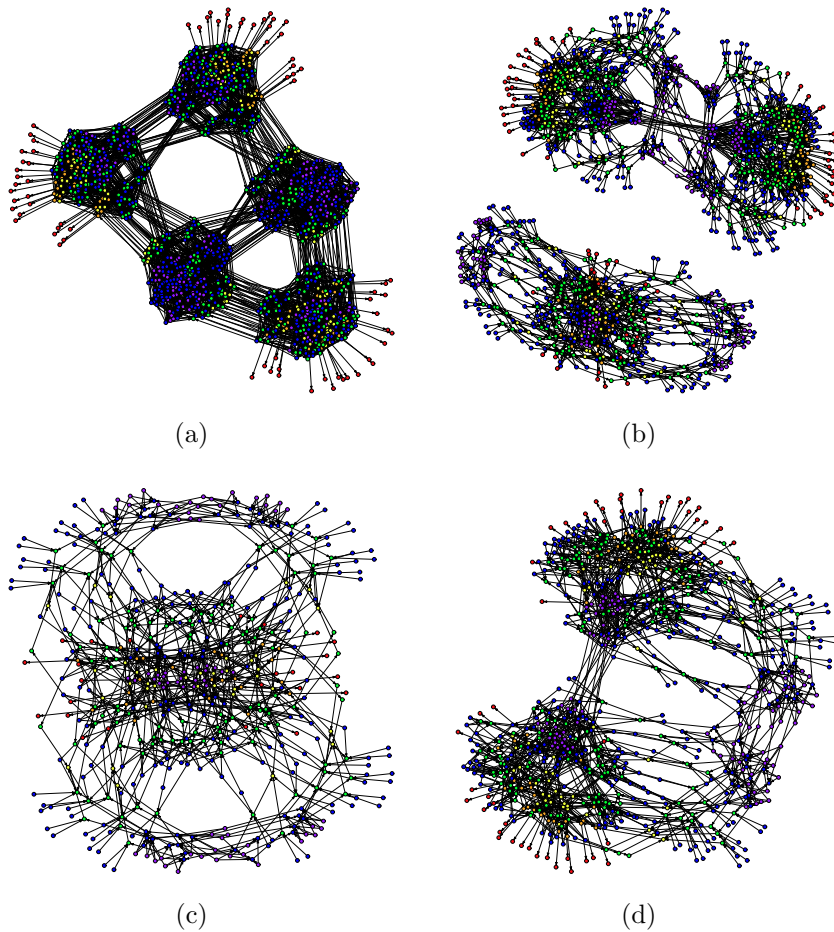


Fig. 6.2: (a) Representation of \mathcal{G} . In red we can see the target set \mathcal{T}^* , in orange L_1 , in yellow L_2 , in green L_3 , in blue L_4 and in purple L_5 . (b) Representation of \mathcal{G}^* . In (c) and (d) are showed the two connected component of \mathcal{G}^* .

In Fig.6.2(a) we can see a representation of the graph \mathcal{G}^* and in (b) we can recognise the two clusters of \mathcal{G}^* : the subgraph related to the final position ($\#\#, 2\heartsuit, 2\clubsuit, \#\#, \#\#, \#\#, CC$), (c), and the subgraph related to position

$(\#\heartsuit, 2\heartsuit, \#\#\heartsuit, \#\#\heartsuit, \#\#\heartsuit, \#\#\heartsuit, NN), (d)$. Moreover, it is easy to prove that the part of the cluster (c) related to $4\heartsuit$ is isomorphic to the part related to $3\heartsuit$.

The introduction of generic cards $\#\#\heartsuit$ leads us to introduce the concept of abstraction discussed in the following section.

6.3 Abstractions and strategies

Definition 15. Let $\mathcal{S} \subset \mathcal{P}$ be a subset of cards. Let \sim be an equivalence relation on \mathcal{C} defined as:

$$p \sim q \iff p[i] = q[i] \quad \forall i \text{ s.t. } p[i] \in \mathcal{S}.$$

Definition 16. Let \mathcal{T} be a target set. We defined a **subproblem** the couple $(\mathcal{S}, \mathcal{T})$. The induced quotient graph (whose edges can be created as before) is called **\mathcal{S} -abstraction** to \mathcal{T} .

Example 3. Let consider now $\mathcal{S} = \{2\heartsuit, 2\clubsuit\}$ and $\mathcal{T} = \mathcal{T}^*$. In this case the relative graph contains 60 nodes and 169 edges which topology is showed in Fig.6.3. For example position $(\#\#\heartsuit, \#\#\heartsuit, 2\clubsuit, \#\#\heartsuit, \#\#\heartsuit, 2\heartsuit, CC)$ can be connected to $(2\heartsuit, \#\#\heartsuit, 2\clubsuit, \#\#\heartsuit, \#\#\heartsuit, \#\#\heartsuit, NN)$ using the move C_6 . Notice that there are two clusters: one in which $p[2] \in \mathcal{S}$ (the interesting one) and the other in which there is a generic card in position 2.

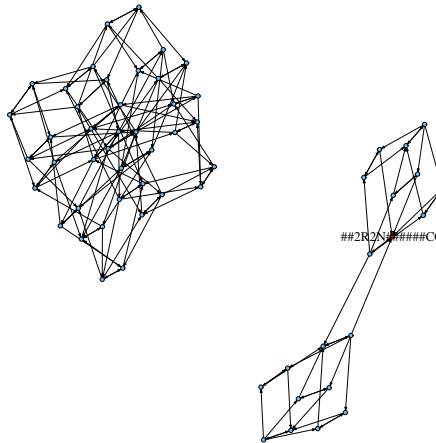


Fig. 6.3: Topology of of $\{2\heartsuit, 2\clubsuit\}$ -abstraction to \mathcal{T}^* .

For each $k = 2, 3, 4$ we can consider the subsets of \mathcal{P} of k elements. Since there are $\binom{6}{k}$ subset of k elements, we can build the $\sum_{k=2}^4 \binom{6}{k} = 50$ *basic* abstractions.

Definition 17. We would to define a strategy as a collection of subproblems coupled with a set of rules \mathbf{P} saying in which conditions we look at a particular abstraction. We will call this set of rules **decision path**.

Formally, a finite set $\mathbf{S} = \{(\mathcal{S}_i, \mathcal{T}_i), \mathbf{P} \mid i \leq \bar{n}\}$ is called **strategy** if $(\mathcal{S}_i, \mathcal{T}_i)$ is a subproblem, with \mathcal{G}_i as abstraction, for each $i = 1, \dots, \bar{n}$ and \mathbf{P} is a decision path such that:

1. $\forall p \in \mathcal{G} \exists! i \leq \bar{n} \text{ s.t. } \mathbf{P}(p) \in \mathcal{G}_i$
2. $d(\mathbf{P}(p), \mathcal{T}_i) < +\infty$
3. $\forall p \in \mathcal{G} \exists I = \{i_1, \dots, i_N\} \text{ s.t. :}$
 - $d(\mathbf{P}(p), \mathcal{T}_{i_1}) < +\infty,$
 - $d(\mathbf{P}(\mathcal{T}_{i_j}), \mathcal{T}_{i_{j+1}}) < +\infty \ 1 \leq j \leq N - 1,$
 - $d(\mathbf{P}(\mathcal{T}_{i_{N-1}}), \mathcal{T}^*) < +\infty.$

The first requirement says that for all starting position the decision path select just one abstraction, in which, using the second requirement, we can reach it's target. The last condition ensures that we can reach the target set starting from any initial position following a finite number of steps.

Example 4. Let \mathbf{S} be the strategy summarised in Tab.6.1:

index	\mathcal{S}_i	\mathcal{T}_i
1	$\{2\heartsuit, \#\heartsuit\}$	$\{(\#\heartsuit, 2\heartsuit, \#\#, \#\#, \#\#, \#\#, NN)\}$
2	$\{2\heartsuit, 2\clubsuit\}$	$\{(\#\#, 2\heartsuit, 2\clubsuit, \#\#, \#\#, \#\#, CC)\}$
3	$\{3\heartsuit, 3\clubsuit\}$	$\{(\#\#, 3\heartsuit, 3\clubsuit, \#\#, \#\#, \#\#, CC)\}$
3	$\{4\heartsuit, 4\clubsuit\}$	$\{(\#\#, 4\heartsuit, 4\clubsuit, \#\#, \#\#, \#\#, CC)\}$

Tab. 6.1: Abstractions and targets used in \mathbf{S}

We need also a decision path \mathbf{P} for a generic position $p \in \mathcal{G}^1$:

- if $p[2] = \#\heartsuit$: select 1 else
- if $p[2] = 2\clubsuit$: select 2 else
- if $p[2] = 3\clubsuit$: select 3 (then 1)
- if $p[2] = 4\clubsuit$: select 4 (then 1)

¹In this case p is a complete position.

This simple strategy show us a common decision process: we look at the target set, if there is a red card we try to put $2\heartsuit$ in position 1, if there is $2\clubsuit$ in target we try to put $2\heartsuit$ in position 3, if there is a black card, we first change it with its corresponding red card using **number** and then we apply the first abstraction. It is quite simple to show that **S** is actually a strategy. The third abstraction and the relative rule in **P** increase the distance of some configuration living in this abstraction. In Tab.6.2 we summarise the two distributions:

distance	# nodes in \mathcal{G}	# nodes in S	diff.	#
0	72	72	+0	892
1	72	72	+1	56
2	96	84	+2	140
3	312	228	+3	172
4	552	360	+4	0
5	168	156	+5	12
6	0	120		
7	0	144		
8	0	36		

Tab. 6.2: Comparison between real distances and distances though **S**.

Let now be $\mathbf{S}' = \mathbf{S} \cup (\{2\clubsuit, \#\clubsuit\}, \{(\#\clubsuit, 2\clubsuit, \#\#, \#\#, \#\#, \#\#, NN)\})$ and finally changing **P** as:

- if $p[2] = \#\heartsuit$: select 1
- if $p[2] = 2\clubsuit$: select 2
- if $p[2] = \#\clubsuit$ and $p[1] \neq 2\clubsuit$ select 3 (then 1)
- if $p[2] = \#\clubsuit$ and $p[1] = 2\clubsuit$ select 4 (then 2)

In Tab.6.3 we can see the last column of Tab.6.2 related to this new strategy:

+0	+1	+2	+3	+4	+5
1108	56	164	112	0	0

Tab. 6.3: Differences of distance between **S'** and the complete game.

The strategy **S'** is more complex than **S** but we need in terms of distance. Such strategy is not optimal, so the next question is: is there any *small* set of abstractions and a correlated optimal strategy?

Let $t = (\#\#, 2R, \#\#, \#\#, \#\#, \#\#, \#\#)$ be a generic target configuration. We can reach t from two generic position: $c = (2R, \#R, \#\#, \#\#, \#\#, \#\#, CC)$ and $n = (\#\#, 2N, 2R, \#\#, \#\#, \#\#, NN)$. We can find now all the generic configuration reaching c and n and so on, building a tree called **minimal backward tree** \mathcal{B} containing all generic position. With generic position we mean a position in which the cards (suit or/and number) appearing are the minimal information needed. In Fig.6.4 we plot the first 4 levels of the tree.

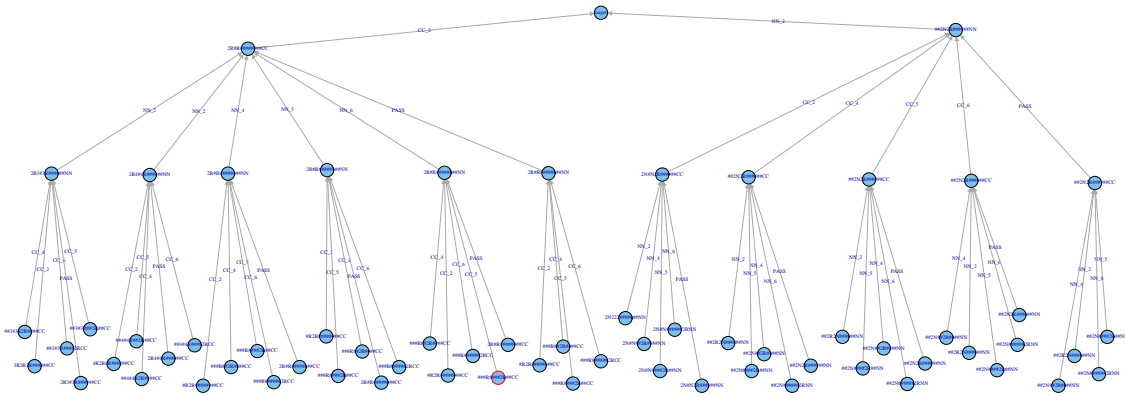


Fig. 6.4: First 4 levels of the minimal backward tree.

The following fact justifies the construction of \mathcal{B} :

Fact 3

$$\forall p \in \mathcal{C} \exists b \in \mathcal{B} \text{ s.t. } d(b, t) = d(p, \mathcal{T})$$

Which is the smallest set of symbols needed to describe an optimal solution? Obviously, the same position p can be seen in different abstraction in \mathcal{B} .

After some calculations, we find out that for each initial position there exists one or two possible abstraction satisfying Fact 6.3: 1116 positions admit only a (minimal) abstraction at desired distance, and 168 admit two abstractions. Considering only minimal abstractions does not allow us to create a strategy better than the complete graph \mathcal{G} , in other words, the union of all symbols present in all minimal abstraction is the entire set \mathcal{P} .

6.4 Extension to covered cards

In this section we want study the game when, at the beginning of the game, cards in position 4 and 6 are face-down. We will indicate this new set of configuration as $\mathcal{U} = \mathcal{C} \cup \mathcal{I}$ where $\mathcal{I} = \{(a, b, c, \#, d, \#, S) \mid a, b, c, d \in \mathcal{P} \text{ and } b \neq 2\heartsuit\}$. We assume the following property:

Fact 4

Let $c \in \mathcal{U}$ be a configuration, once an unknown card ($\#$) is flipped we can replace the unknown values with the real values and consider $c \in \mathcal{C}$.

In other words, once we have flipped an unknown card we can play with all face-up cards. We can split \mathcal{C} into three disjoint sets \mathcal{C}_{46} , \mathcal{C}_{64} and \mathcal{T} satisfying:

$$p \in \mathcal{C}_{46} \iff \exists q \in \mathcal{C}_{64} \text{ s.t. } p[i] = q[i] \text{ for } i \neq 4, 6, \quad p[6] = q[4], \quad p[2] \neq 2\heartsuit.$$

It is easy to see that $\#\mathcal{C}_{46} = \#\mathcal{C}_{64} = (1272 - \#\mathcal{T})/2 = 600$.

Definition 18. An element $c \in \mathcal{I}$ is called **superposition** and its value could be $p_i \in \mathcal{C}_{46}$ or $q_i \in \mathcal{C}_{64}$.

The set \mathcal{I} represent at the same time the two sets \mathcal{C}_{46} and \mathcal{C}_{64} in a quantistic sense: a superposition has the same role of the Schrödinger's cat, *i.e.* we can not know its real value until we observe it.

Fact 5

For each pair (p, q) we have that $d(p_i, \mathcal{T}) = d(q_i, \mathcal{T})$.

Let $c \in \mathcal{I}$ be a configuration, its distance $d_c = d(c, \mathcal{T})$ is well defined but with d_c we want to empathise that we do not know the shortest path; this is a sort of **quantistic distance**.

Can we reach the target set \mathcal{T} starting from c using d_p moves independently form the real expression of c ? If the answer to this question is negative, how much we have to increase the number of moves in order to solve the **quantistic effect** (independently from the real value of c)?

If $c \in \mathcal{I}$ could be $p \in \mathcal{C}_{46}$ or $q \in \mathcal{C}_{64}$ than holds:

Fact 6

We can reach \mathcal{T} using at most $d_p + 4$ moves.

Proof. We can use four moves to know the values of $\#$ cards and return back to the initial configuration. \square

In this way we can *resolve* the quantistic effect knowing the values of $\#$. The aim of the game is to reach the target using the least number of moves, and the strategy used in Fact 6 is not optimal.

Definition 19. Under the previous notation, we define the **orbit** of p to be $O(p) = \{M \in \mathcal{M}^{d_p} \mid M(p) = t \text{ with } t \in \mathcal{T}\}$. Let $v \in O(p)$ and $w \in O(q)$ be two elements in the respective orbits. We define the two paths v and w to be **equimovements** (indicate with $v \sim w$) if:

- $v = w$ or
- $v_i = w_i$ for all $i \leq d_p - 2$, $v_{d_p-1} = PASS$ and $w_{d_p-1} \notin \{N_2, C_2\}$ or viceversa, or
- $v_i = w_i$ for all $i < \bar{k}$ and, for some $k < \bar{k}$, happens that $v_k \in \{N_4, C_4, N_6, C_6\}$.

Fact 7

Under the previous notations, if $v \sim w$ than we can reach the target using exactly d_c moves.

Proof. Directly from the definition of equimovements. \square

In this case the quantistic effect can be solved without additional costs and the number of this kind of pairs is 304.

Definition 20. Let us generalise the definition of orbit: for every $n \in \mathbf{N}$ let $O_n(p)$ be the **n -th orbit** of p define as $O_n(p) = \{M \in \mathcal{M}^{d_p+n} \mid M(p) = t \text{ with } t \in \mathcal{T}\}$. We can also generalise the definition of equimovements: two paths $v \in O_n(p)$ and $w \in O_m(q)$ are **$\frac{m+n}{2}$ -equimovements** (indicate with $v \sim_{(m+n)/2} w$) if:

- $v_i = w_i$ for all $i = 1, \dots, \min\{n, m\}$ or
- $m = n$ and $v_i = w_i$ for all $i \leq d_p + n - 2$, $v_{d_p+n-1} = PASS$ and $w_{d_p+n-1} \notin \{N_2, C_2\}$ or viceversa, or
- $v_i = w_i$ for all $i < \bar{k}$ and, for some $k < \bar{k}$, happens that $v_k \in \{N_4, C_4, N_6, C_6\}$.

The number $d_p + (m + n)/2$ indicates the mean of the distances from c to \mathcal{C} : for example suppose that $d_p = 5, n = 1, m = 0$ and that there exists v and w such that $v \sim_{1/2} w$, since $c \in \mathcal{I}$ it could be at the same time the configuration p and q . This means that we can reach the target using the path v with a length $d_p + 1$ or using the path w with a length d_p . Since the two positions are equiprobable, the mean of the length to the target is $(d_c + 1 + d_c)/2 = 1/2 + d_c = (m + n)/2 + d_c$.

For all superpositions, using a computer, we can calculate the minimum cost to resolve the quantistic effect. The results are summarised in Tab.6.4.

$\frac{m+n}{2}$	number of pairs	rate
0	304	50.7%
$\frac{1}{2}$	104	17.3%
1	172 ($m = 0, n = 2$) + 20 ($m = n = 1$)	32%

Tab. 6.4: Minimum costs to solve quantistic effect.

Among the 20 couples with $m = 1$ and $n = 1$ there are 16 couples that have also $m = 0$ and $n = 2$.

We can create a network \mathcal{J} using as nodes the set \mathcal{I} and creating edges using a set of moves $\mathcal{N} = \{PASS, N_2, C_2, N_5, C_5\} \subset \mathcal{M}$. Now we can create the final graph $\mathcal{H} = \mathcal{G} \sqcup \mathcal{J}$ adding some extra links between \mathcal{I} and \mathcal{C} .

More precisely, $p \in \mathcal{I}$ is connected to $q \in \mathcal{C}$ if the move m such that $m(p) = q$ belongs to any equimovements².

In other words for each starting position $p \in \mathcal{J}$ we can stay in \mathcal{J} using the set \mathcal{N} or decide to discover a covered card. We are allow to flip an uncovered card if the resulting position q ($p_i \in \mathcal{C}_{46}$ or $q_i \in \mathcal{C}_{64}$) belongs to a equimovement starting from p .

Notice that once we arrive into \mathcal{G} , we can follow the minimum path to the target.

6.5 Conclusion

We show that concepts of strategy, abstraction and subproblem can be easily formalise using basic elements of network theory. In particular quotient graphs and shortest constrained paths results to be useful tools for our analysis. Further studies can be made both in order to understand how a strategy is created (or selected) and to apply these ideas to a (particular) class of games.

²In this case $m(p)$ represents both configurations: $p_i \in \mathcal{C}_{46}$ and $q_i \in \mathcal{C}_{64}$

Bibliography

- [1] A.-L. Barabási. The network takeover. *Nature Physics*, 8:14–16, 2012.
- [2] P. O. Perry and P. J. Wolfe. Null models for network data. January 2012.
- [3] P. Erdős and A. Rényi. On Random Graphs. I. *Publicationes Mathematicae*, 6:290–297, 1959.
- [4] M.E.J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69:026113, 2004.
- [5] M. Girvan and M.E.J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Science U.S.A.*, 99:7821–7826, 2002.
- [6] T. Poisot and D. Gravel. When is a network complex? Connectance drives degree distribution and emerging network properties. *PeerJ PrePrints* — <https://peerj.com/preprints/50v1>, 2013.
- [7] X. Liu, T. Murata, and K. Wakita. Extending modularity by capturing the similarity attraction feature in the null model. *arXiv:1210.4007v3 [cs.SI]*, 2013.
- [8] W. Ulrich and N.J. Gotelli. Null model analysis of species associations using abundance data. *Ecology*, 91(11):3384–3397, 2010.
- [9] W.P. Kelly, T. Thorne, and M.P.H. Stumpf. Statistical Null Models for Biological Network Analysis. In M.P.H. Stumpf and C. Wiuf, editors, *Statistical and Evolutionary Analysis of Biological Networks*, pages 145–166. Imperial College Press, 2010.
- [10] T. Milenkovic, I. Filippisi, M. Lappe, and N. Prülj. Optimized Null Model for Protein Structure Networks. *PLoS ONE*, 4(6):e5967, 2009.

-
- [11] F. Ruzzenenti, F. Picciolo, R. Basosi, and D. Garlaschelli. Spatial effects in real networks: Measures, null models, and applications. *Physical Review E*, 86:066110, 2012.
- [12] G. Fagiolo, T. Squartini, and D. Garlaschelli. Null models of economic networks: the case of the world trade web. *Journal of Economic Interaction and Coordination*, 8(1):75–107, 2013.
- [13] M. Dueñas and G. Fagiolo. Global Trade Imbalances: A Network Approach. SSRN: <http://ssrn.com/abstract=2266320>, 2013.
- [14] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the National Academy of Sciences*, 107(14):6286–6291, 2010.
- [15] R.J. Prill, D. Marbach, J. Saez-Rodriguez, P.K. Sorger, L.G. Alexopoulos, X. Xue, N.D. Clarke, G. Altan-Bonnet, and G. Stolovitzky. Towards a Rigorous Assessment of Systems Biology Models: The DREAM3 Challenges. *PLoS ONE*, 5(2):e9202, 02 2010.
- [16] D. Marbach, T. Schaffter, C. Mattiussi, and D. Floreano. Generating realistic in silico gene networks for performance assessment of reverse engineering methods. *Journal of Computational Biology*, 16(2):229–239, 2009.
- [17] R. K. Thomas, A. C. Baker, R. M. DeBiasi, and W. Winckler. High-throughput oncogene mutation profiling in human cancer. *Nature*, 2007.
- [18] G. Ciriello, E. Cerami, C. Sander, and N. Schultz. Mutual exclusivity analysis identifies oncogenic network modules. *Genome Research*, 22(2):398–406, February 2012.
- [19] M. D. Cohen and P. Bacdayan. Organizational Routines Are Stored As Procedural Memory: Evidence from a Laboratory Study. *Organization Science*, 5(4):pp. 554–568, 1994.
- [20] M. Egidi. Chapter 1 Decomposition Patterns in Problem Solving . In Richard Topol and Bernard Walliser, editors, *Cognitive Economics New Trends*, volume "280 of *Contributions to Economic Analysis*, pages 15 – 46. Elsevier, 2006.
- [21] M. Egidi and A. Narduzzo. The Emergence of Path-dependent Behaviors in Cooperative Contexts. *International Journal of Industrial Organization*, 15(6):677 – 709, 1997.

- [22] C. Michael, M. Francis, F. Raphael, and E. Damien. Learning exploration/exploitation strategies for single trajectory reinforcement learning. In *EWRL*, pages 1–10, 2012.
- [23] C. Berge. *Graphs and Hypergraphs*. Elsevier Science Ltd, 1985.
- [24] M. Dickison, S. Havlin, and H. E. Stanley. Epidemics on interconnected networks. *CoRR*, abs/1201.6339, 2012.
- [25] A. Lancichinetti, M. Kivelä, J. Saramäki, and S. Fortunato. Characterizing the Community Structure of Complex Networks. *PLoS ONE*, 5(8):e11976+, 2010.
- [26] P. Erdős and A. Rnyi. *On the Evolution of Random Graphs*. 1960.
- [27] A. Gobbi, F. Iorio, D. Wedge, K. Dawson, A. F. Ludmil, G. Jurman, and J. Saez-Rodriguez. Next-generation-sequencing data randomisation preserving genomic-event distributions. *submitted*, 37:547–579, 2013.
- [28] D.J. de Solla Price. Networks of Scientific Papers. *Science*, 149(3683):510–515, 1965.
- [29] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- [30] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.
- [31] D. Volchenkov and P. Blanchard. An algorithm generating random graphs with power law degree distributions. *Physica A: Statistical Mechanics and its Applications*, 315(3-4):677–690, 2002.
- [32] G. Stolovitzky, D. Monroe, and A. Califano. Dialogue on reverse-engineering assessment and methods: the DREAM of high-throughput pathway inference. *Annals of the New York Academy of Sciences*, 1115(1):1–22, 2007.
- [33] sbv IMPROVER project team. On Crowd-verification of Biological Networks. *Bioinformatics and biology insights*, 7:307–325, 2013.
- [34] I. Miklós and J. Podani. Randomization of presence-absence matrices: comments and new algorithms. *Ecology*, 85(1):86–92, 2004.
- [35] Arif Zaman and Daniel Simberloff. Random binary matrices in biogeographical ecology Instituting a good neighbor policy. *Environmental and Ecological Statistics*, 9(4):405–421, 2002.

- [36] I. Stanton and A. Pinar. Constructing and sampling graphs with a prescribed joint degree distribution. *J. Exp. Algorithmics*, 17:3.5:3.1–3.5:3.25, 2012.
- [37] A. R. Rao, R. Jana, and S. Bandyopadhyay. A Markov Chain Monte Carlo Method for Generating Random $(0, 1)$ -Matrices with Given Marginals. *Sankhy: The Indian Journal of Statistics, Series A*, 58(2), 1996.
- [38] C. H. Yeang, F. McCormick, and A. Levine. Combinatorial patterns of somatic gene mutations in cancer. *The FASEB Journal*, 22(8):2605–2622, April 2008.
- [39] A. G. Uren, J. Kool, K. Matentzoglou, and J. de Ridder. Large-Scale Mutagenesis in p19ARF-and p53-Deficient Mice Identifies Cancer Genes and Their Collaborative Networks. *Cell*, 2008.
- [40] Q. Cui. A network of cancer genes with co-occurring and anti-co-occurring mutations. *PLoS ONE*, 5(10):e13180, 2010.
- [41] C. A. Miller, S. H. Settle, E. P. Sulman, K. D. Aldape, and A. Milosavljevic. Discovering functional modules by identifying recurrent and mutually exclusive mutational patterns in tumors. *BMC Medical Genomics*, 4(1):34, April 2011.
- [42] Y. Gu, D. Yang, J. Zou, W. Ma, R. Wu, W. Zhao, Y. Zhang, H. Xiao, X. Gong, M. Zhang, J. Zhu, and Z. Guo. Systematic Interpretation of Co-mutated Genes in Large-Scale Cancer Mutation Profiles. *Molecular Cancer Therapeutics*, 9(8):2186–2195, August 2010.
- [43] F. Vandin, E. Upfal, and B. J. Raphael. De novo discovery of mutated driver pathways in cancer. *Genome Research*, 22(2):375–385, February 2012.
- [44] A. Bauer-Mehren, M. Bundschuh, M. Rautschka, M. A. Mayer, F. Sanz, and L. I. Furlong. Gene-disease network analysis reveals functional modules in mendelian, complex and environmental diseases. *PLoS ONE*, 6(6):e20284, 2011.
- [45] M. A. Yildirim, K. I. Goh, M. E. Cusick, and A. L. Barabási. Drug-target network. *Nature*, 2007.
- [46] I. Vogt and J. Mestres. DrugTarget Networks. *Molecular Informatics*, 2010.
- [47] G. Hu and P. Agarwal. Human disease-drug network based on genomic expression profiles. *PLoS ONE*, 4(8):e6536, 2009.

- [48] N. J. Gotelli. Null model analysis of species co-occurrence patterns. *Ecology*, 2000.
- [49] N. J. Gotelli and G. L. Entsminger. Swap and fill algorithms in null model analysis: rethinking the knight’s tour. *Oecologia*, 2001.
- [50] P. Jordano, J. Bascompte, and J. M. Olesen. Invariant properties in coevolutionary networks of plant-animal interactions. *Ecology Letters*, 6(1):69–81, December 2002.
- [51] J. A Dunne, R. J Williams, and N. D. Martinez. Food-web structure and network theory: The role of connectance and size. *Proceedings of the National Academy of Sciences of the United States of America*, 99(20):12917–12922, October 2002.
- [52] W.M. Patefield. An efficient method of generating random rxc tables with given row and column totals. (algorithm as 159.). *Journal of the Royal Statistical Society. Series C. Applied Statistics*, 30:91–97, 1981.
- [53] R. Milo, N. Kashtan, S. Itzkovitz, M. E. J. Newman, and U. Alon. On the uniform generation of random graphs with prescribed degree sequences. May 2004.
- [54] J. Ray, A. Pinar, and C. Seshadhri. Are we there yet? When to stop a Markov chain while generating random graphs. 2012.
- [55] C. I. Del Genio, H. Kim, Z. Toroczkai, and K. E. Bassler. Efficient and Exact Sampling of Simple Graphs with Given Arbitrary Degree Sequence. *PLoS ONE*, 5(4):e10012+, 2010.
- [56] P. Jaccard. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.
- [57] S. P. Brooks and A. Gelman. General Methods for Monitoring Convergence of Iterative Simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, 1998.
- [58] T. J. Hudson, W. Anderson, A. Aretz, A. D. Barker, C. Bell, R. R. Bernabé, M. K. Bhan, F. Calvo, I. Eerola, D. S Gerhard, et al. International network of cancer genome projects. *Nature*, 464(7291):993–998, 2010.
- [59] M. J. Garnett, E. J. Edelman, S. J. Heidorn, C. D. Greenman, A. Dastur, K. W. Lau, P. Greninger, I. R. Thompson, X. Luo, J. Soares, Q. Liu, F. Iorio, D. Surdez, L. Chen, Randy J Milano, Graham R Bignell, Ah T Tam,

- Helen Davies, Jesse A Stevenson, Syd Barthorpe, S. R Lutz, F. Kogera, K. Lawrence, A. McLaren-Douglas, X. Mitropoulos, T. Mironenko, H. Thi, L. Richardson, W. Zhou, F. Jewitt, T. Zhang, P. O'Brien, J. L Boisvert, S. Price, W. Hur, W. Yang, X. Deng, A. Butler, H. Geun Choi, J. W. Chang, J. Baselga, I. Stamenkovic, J. A. Engelman, S. V. Sharma, O. Delattre, J. Saez-Rodriguez, N. S Gray, J. Settleman, P. A. Futreal, D. A Haber, M. R Stratton, S. Ramaswamy, U. McDermott, and C. H. Benes. Systematic identification of genomic markers of drug sensitivity in cancer cells. *Nature*, 483(7391):570–575, March 2012.
- [60] Cancer Genome Atlas Research Network. Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*, 455(7216):1061–1068, October 2008.
- [61] J. Barretina, G. Caponigro, N. Stransky, K. Venkatesan, A. A. Margolin, S. Kim, C. J. Wilson, J. Lehár, G. V. Kryukov, D. Sonkin, A. Reddy, M. Liu, L. Murray, M. F. Berger, J. E. Monahan, P. Morais, J. Meltzer, A. Korejwa, J. Jané-Valbuena, F. A. Mapa, J. Thibault, E. Bric-Furlong, P. Raman, A. Shipway, I. H. Engels, J. Cheng, G. K. Yu, J. Yu, P. Aspesi, M. de Silva, K. Jagtap, M. D. Jones, L. Wang, C. Hatton, E. Palesscandolo, S. Gupta, S. Mahan, C. Sougnez, R. C. Onofrio, T. Liefeld, L. MacConaill, W. Winckler, M. Reich, N. Li, J. P. Mesirov, S. B. Gabriel, G. Getz, K. Ardlie, V. Chan, V. E. Myer, B. L. Weber, J. Porter, M. Warmuth, P. Finan, J. L. Harris, M. Meyerson, T. R. Golub, M. P. Morrissey, W. R. Sellers, R. Schlegel, and L. A. Garraway. The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature*, 483(7391):603–607, March 2012.
- [62] M. R. Stratton, P. J. Campbell, and P. A. Futreal. The cancer genome. *Nature*, 2009.
- [63] C. Greenman, P. Stephens, R. Smith, G. L. Dalgliesh, C. Hunter, G. Bignell, H. Davies, J. Teague, A. Butler, C. Stevens, S. Edkins, S. O'Meara, I. Vastrik, E. E. Schmidt, T. Avis, S. Barthorpe, G. Bhamra, G. Buck, B. Choudhury, J. Clements, J. Cole, E. Dicks, S. Forbes, K. Gray, K. Halliday, R. Harrison, K. Hills, J. Hinton, A. Jenkinson, D. Jones, A. Menzies, T. Mironenko, J. Perry, K. Raine, D. Richardson, R. Shepherd, A. Small, C. Tofts, J. Varian, T. Webb, S. West, S. Widaa, A. Yates, D. P. Cahill, D. N. Louis, P. Goldstraw, A. G. Nicholson, F. Brasseur, L. Looijenga, B. L. Weber, Y. Chiew, A. DeFazio, M. F. Greaves, A. R. Green, P. Campbell, E. Birney, D. F. Easton, G. Chenevix-Trench, M. Tan, S. K. Khoo, B. T. Teh, S. T. Yuen, S. Y.

- Leung, R. Wooster, P. A. Futreal, and M. R. Stratton. Patterns of somatic mutation in human cancer genomes. *Nature*, 446(7132):153–8, 2007.
- [64] G. R. Bignell, C. D. Greenman, H. Davies, A. P. Butler, S. Edkins, J. M. Andrews, G. Buck, L. Chen, D. Beare, C. Latimer, S. Widaa, J. Hinton, C. Fahey, B. Fu, S. Swamy, G. L. Dalgliesh, B. T. Teh, P. Deloukas, F. Yang, P. J. Campbell, P. A. Futreal, and M. R. Stratton. Signatures of mutation and selection in the cancer genome. *Nature*, 463(7283):893–898, 2010.
- [65] L. Spans, L. Clinckemalie, C. Helsen, D. Vanderschueren, S. Boonen, E. Lerut, S. Joniau, and F. Claessens. The genomic landscape of prostate cancer. *International Journal of Molecular Sciences*, 14(6):10822–10851, 2013.
- [66] D. Hanahan and R. A. Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, March 2011.
- [67] J. B. Wilson. Methods for detecting non-randomness in species co-occurrences: a contribution. *Oecologia*, 1987.
- [68] E. F. Connor and D. Simberloff. The assembly of species communities: chance or competition? *Ecology*, 1979.
- [69] J. L. Gross and J. Yellen. *Graph Theory and Its Applications*. Chapman & Hall/CRC, 2006.
- [70] K. Misue. Drawing bipartite graphs as anchored maps. In *Proceedings of the 2006 Asia-Pacific Symposium on ...*, 2006.
- [71] A. Palacín, C. Gómez-Casado, L. A. Rivas, J. Aguirre, L. Tordesillas, J. Bartra, C. Blanco, T. Carrillo, J. Cuesta-Herranz, C. de Frutos, G. G. Alvarez-Eire, F. J. Fernández, P. Gamboa, R. Muñoz, R. Sánchez-Monge, S. Sirvent, M. J. Torres, S. Varela-Losada, R. Rodríguez, V. Parro, M. Blanca, G. Salcedo, and A. Díaz-Perales. Graph based study of allergen cross-reactivity of plant lipid transfer proteins (LTPs) using microarray in a multicenter study. *PLoS ONE*, 7(12):e50799, 2012.
- [72] A. Brousseau. *Linear Recursion and Fibonacci Sequences*. Fibonacci Assoc., 1971.
- [73] G. Csardi and T. Nepusz. The igraph Software Package for Complex Network Research. *InterJournal*, Complex Systems:1695, 2006.

- [74] G. Szederkenyi, J. Banga, and A. Alonso. Inference of complex biological networks: distinguishability issues and optimization-based solutions. *BMC Systems Biology*, 5(1):177, 2011.
- [75] F. He, R. Balling, and A.-P. Zeng. Reverse engineering and verification of gene networks: Principles, assumptions, and limitations of present methods and future perspectives. *Journal of Biotechnology*, 144:190–203, 2009.
- [76] P. Meyer, L.G. Alexopoulos, T. Bonk, A. Califano, C.R. Cho, A. de la Fuente, D. de Graaf, A.J. Hartemink, J. Hoeng, N.V. Ivanov, H. Koepl, R. Linding, D. Marbach, R. Norel, M.C. Peitsch, J.J. Rice, A. Royyuru, F. Schacherer, J. Sprengel, K. Stolle, D. Vitkup, and G. Stolovitzky. Verification of systems biology research in the age of collaborative competition. *Nature Biotechnology*, 29(9):811–815, 2011.
- [77] R. De Smet and K. Marchal. Advantages and limitations of current network inference methods. *Nature Reviews Microbiology*, 8(10):717–729, 2010.
- [78] H.K. Lee, A.K. Hsu, J. Sajdak, J. Qin, and P. Pavlidis. Coexpression Analysis of Human Genes Across Many Microarray Data Sets. *Genome Research*, 14(6):1085–1094, 2004.
- [79] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.
- [80] N.A. Furlotte, H.M. Kang, C. Ye, and E. Eskin. Mixed-model coexpression: calculating gene coexpression while accounting for expression heterogeneity. *Bioinformatics*, 27(13):i288–i294, 2011.
- [81] J.D. Allen, Y. Xie, M. Chen, L. Girard, and G. Xiao. Comparing Statistical Methods for Constructing Large Scale Gene Networks. *Plos ONE*, 7(1):e29348, 2012.
- [82] L. Song, P. Langfelder, and S. Horvath. Comparison of co-expression measures: mutual information, correlation, and model based indices. *BMC Bioinformatics*, 13:328, 2012.
- [83] P. Madhamshettiwar, S. Maetschke, M. Davis, A. Reverter, and M. Ragan. Gene regulatory network inference: evaluation and application to ovarian cancer allows the prioritization of drug targets. *Genome Medicine*, 4(5):41, 2012.

- [84] A. Baralla, W.I. Mentzen, and A. de la Fuente. Inferring Gene Networks: Dream or Nightmare? *Annals of the New York Academy of Science*, 1158:246–256, 2009.
- [85] S.L. Carter, C.M. Brechbühler, M. Griffin, and A.T. Bond. Gene co-expression network topology provides a framework for molecular characterization of cellular state. *Bioinformatics*, 20(14):2242–2250, 2004.
- [86] L.D. Wood, D.W. Parsons, S. Jones, J. Lin, T. Sjöblom, R.J. Leary, D. Shen, S.M. Boca, T. Barber, J. Ptak, N. Silliman, S. Szabo, Z. Dezsó, V. Ustyanksky, T. Nikolskaya, Y. Nikolsky, R. Karchin, P.A. Wilson, J.S. Kaminker, Z. Zhang, R. Croshaw, J. Willis, D. Dawson, M. Shipitsin, J.K.V. Willson, S. Sukumar, K. Polyak, B.H. Park, C.L. Pethiyagoda, P.V.K. Pant, D.G. Ballinger, A.B. Sparks, J. Hartigan, D.R. Smith, E. Suh, N. Papadopoulos, P. Buckhaults, S.D. Markowitz, G. Parmigiani, K.W. Kinzler, V.E. Velculescu, and B. Vogelstein. The Genomic Landscapes of Human Breast and Colorectal Cancers. *Science*, 318(5853):1108–1113, 2007.
- [87] M. Carlson, B. Zhang, Z. Fang, P. Mischel, S. Horvath, and S. Nelson. Gene connectivity, function, and sequence conservation: predictions from modular yeast co-expression networks. *BMC Genomics*, 7(1):40, 2006.
- [88] R. Chen, G.I. Mias, J. Li-Pook-Than, L. Jiang, H.Y.K. Lam, R. Chen, E. Miriami, K.J. Karczewski, M. Hariharan, F.E. Dewey, Y. Cheng, M.J. Clark, H. Im, L. Habegger, S. Balasubramanian, M. O’Huallachain, J.T. Dudley, S. Hillenmeyer, R. Haraksingh, D. Sharon, G. Euskirchen, P. Lacroute, K. Bettinger, A.P. Boyle, M. Kasowski, F. Grubert, S. Seki, M. Garcia, M. Whirl-Carrillo, M. Gallardo, M.A. Blasco, P.L. Greenberg, P. Snyder, T.E. Klein, R.B. Altman, A.J. Butte, E.A. Ashley, M. Gerstein, K.C. Nadeau, H. Tang, and M. Snyder. Personal Omics Profiling Reveals Dynamic Molecular and Medical Phenotypes. *Cell*, 148(6):1293–1307, 2012.
- [89] J. Friedman and E.J. Alm. Inferring Correlation Networks from Genomic Survey Data. *PLoS Computational Biology*, 8(9):e1002687, 2012.
- [90] A.J. Butte and I.S. Kohane. Mutual Information Relevance Networks: Functional Genomic Clustering Using Pairwise Entropy Measurements. *Pacific Symposium on Biocomputing*, 5:415–426, 2000.
- [91] H.-Q. Wang and C.J. Tsai. CorSig: A General Framework for Estimating Statistical Significance of Correlation and Its Application to Gene Co-Expression Analysis. *PLoS ONE*, 8(10):e77429, 2013.

- [92] D.-Y. Cho, Y.-A. Kim, and T.M. Przytycka. Chapter 5: Network Biology Approach to Complex Diseases. *PLoS Computational Biology*, 8(12):e1002820, 2012.
- [93] B. Zhang and S. Horvath. A General Framework for Weighted Gene Co-Expression Network Analysis. *Statistical Applications in Genetics and Molecular Biology*, 4(1):Article 17, 2005.
- [94] J. Zhang, K. Lu, Y. Xiang, M. Islam, S. Kotian, Z. Kais, C. Lee, M. Arora, H.-W. Liu, J.D. Parvin, and K. Huang. Weighted Frequent Gene Co-expression Network Mining to Identify Genes Involved in Genome Stability. *PLoS Computational Biology*, 8(8):e1002656, 2012.
- [95] D. Gibbs, A. Baratt, R. Baric, Y. Kawaoka, R. Smith, E. Orwoll, M. Katze, and S. McWeeney. Protein co-expression network analysis (ProCoNA). *Journal of Clinical Bioinformatics*, 3(1):11, 2013.
- [96] G.S. Davidson, B.N. Wylie, and K.W. Boyack. Cluster Stability and the Use of Noise in Interpretation of Clustering. In *Proceedings of the IEEE Symposium on Information Visualization 2001 INFOVIS'01*, page 23. IEEE Computer Society, 2001.
- [97] D. Zhu, A.O. Hero, Z.S. Qin, and A. Swaroop. High throughput screening of co-expressed gene pairs with controlled false discovery rate (FDR) and minimum acceptable strength (MAS). *Journal of Computational Biology*, 12(7):1029–1045, 2005.
- [98] H. Chen. *Clustering and Network Analysis with Single Nucleotide Polymorphism (SNP)*. PhD thesis, Stony Brook University, 2011.
- [99] J. Numata, O. Ebenhöf, and E.W. Knapp. Measuring correlations in metabolomic networks with mutual information. *Genome Informatics*, 20:112–122, 2008.
- [100] A. Fukushima. DiffCorr: An R package to analyze and visualize differential correlations in biological networks. *Gene*, 518(1):209–214, 2013.
- [101] R. Opgen-Rhein and K. Strimmer. From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC Systems Biology*, 1:37, 2007.
- [102] T. Obayashi, S. Hayashi, M. Shibaoka, M. Saeki, H. Ohta, and K. Kinoshita. Coxpresdb: a database of coexpressed gene networks in mammals. *Nucleic Acids Research*, 36(suppl 1):D77–D82, 2008.

- [103] J. Ruan, A. Dean, and W. Zhang. A general co-expression network-based approach to gene expression analysis: comparison and applications. *BMC Systems Biology*, 4(1):8, 2010.
- [104] M. Mistry, J. Gillis, and P. Pavlidis. Meta-analysis of gene coexpression networks in the post-mortem prefrontal cortex of patients with schizophrenia and unaffected controls. *BMC Neuroscience*, 14(1):105, 2013.
- [105] F. Luo, Y. Yang, J. Zhong, H. Gao, L. Khan, D. Thompson, and J. Zhou. Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory. *BMC Bioinformatics*, 8(1):299, 2007.
- [106] M. Scholz. *Approaches to analyse and interpret biological profile data*. PhD thesis, Potsdam University, 2006.
- [107] C. Ma and X. Wang. Application of the Gini Correlation Coefficient to Infer Regulatory Relationships in Transcriptome Analysis. *Plant Physiology*, 160(1):192–203, 2012.
- [108] P. Caraiani. Using Complex Networks to Characterize International Business Cycles. *PLoS ONE*, 8(3):e58109, 2013.
- [109] M. Inouye, K. Silander, E. Hamalainen, V. Salomaa, K. Harald, P. Jousilahti, S. Männistö, J.G. Eriksson, J. Saarela, S. Ripatti, M. Perola, G.J. van Ommen, M.R. Taskinen, A. Palotie, E.T. Dermitzakis, and L. Peltonen. An immune response network associated with blood lipid levels. *PLoS Genetics*, 6(9):e1001113, 2010.
- [110] F.M. Giorgi. *Expression-based Reverse Engineering of Plant Transcriptional Networks*. PhD thesis, Potsdam University, 2011.
- [111] B. Usadel, T. Obayashi, M. Mutwil, F.M. Giorgi, G.W. Bassel, M. Tanimoto, A. Chow, D. Steinhauser, S. Persson, and N.J. Provart. Co-expression tools for plant biology: opportunities for hypothesis generation and caveats. *Plant, Cell & Environment*, 32(12):1633–1651, 2009.
- [112] A. Yuan, Q. Yue, V. Apprey, and G.E. Bonney. Global pattern of pairwise relationship in genetic network. *Journal of Biomedical Science and Engineering*, 3:977–985, 2010.
- [113] G.W. Bassel, H. Lan, E. Glaab, D.J. Gibbs, T. Gerjets, N. Krasnogor, A.J. Bonner, M.J. Holdsworth, and N.J. Provart. Genome-wide network model capturing seed germination reveals coordinated regulation of plant cellular phase transitions. *Proceedings of the National Academy of Sciences*, 108(23):9709–9714, 2011.

- [114] Z.-L. Zheng and Y. Zhao. Transcriptome comparison and gene coexpression network analysis provide a systems view of citrus response to " *Candidatus Liberibacter asiaticus*" infection. *BMC Genomics*, 14:27, 2013.
- [115] J. Stöckel, E.A. Welsh, M. Liberton, R. Kunnvakkam, R. Aurora, and H.B. Pakrasi. Global transcriptomic analysis of *Cyanospora* 51142 reveals robust diurnal oscillation of central metabolic processes. *Proceedings of the National Academy of Sciences*, 105(16):6156–6161, 2008.
- [116] K. Dempsey, S. Bonasera, D. Bastola, and H. Ali. A Novel Correlation Networks Approach for the Identification of Gene Targets. In *Proceedings of the 44th Hawaii International Conference on System Sciences - HICSS 2011*, pages 1–8. IEEE, 2011.
- [117] L.L. Elo, H. Järvenpää, M. Orešič, R. Lahesmaa, and T. Aittokallio. Systematic construction of gene coexpression networks with applications to human T helper cell differentiation process. *Bioinformatics*, 23(16):2096–2103, 2007.
- [118] S.M. Gibson, S.P. Ficklin, S. Isaacson, F. Luo, F.A. Feltus, and M.C. Smith. Massive-Scale Gene Co-Expression Network Construction and Robustness Testing Using Random Matrix Theory. *PLoS ONE*, 8(2):e55871, 2013.
- [119] F.A. Feltus, S.P. Ficklin, S.M. Gibson, and M.C. Smith. Maximizing capture of gene co-expression relationships through pre-clustering of input expression samples: an *Arabidopsis* case study. *BMC Systems Biology*, 7:44, 2013.
- [120] A.D. Perkins and M.A. Langston. Threshold selection in gene co-expression networks using spectral graph theory techniques. *BMC Bioinformatics*, 10(Suppl 11):1–11, 2009.
- [121] B. Borate, E. Chesler, M. Langston, A. Saxton, and B. Voy. Comparison of threshold selection methods for microarray gene co-expression matrices. *BMC Research Notes*, 2(1):240, 2009.
- [122] P.R. Bevington and D.K. Robinson. *Data Reduction and Error Analysis for the Physical Sciences*. McGraw-Hill, 2002.
- [123] A. Zhang. *Advanced Analysis of Gene Expression Microarray Data*. World Scientific, 2006.
- [124] J. Casellas and L. Varona. Modeling Skewness in Human Transcriptomes. *PLoS ONE*, 7(6):e38919, 2012.

- [125] T. Doig, D. Hume, T. Theocharidis, J. Goodlad, C. Gregory, and T. Freeman. Coexpression analysis of large cancer datasets provides insight into the cellular phenotypes of the tumour microenvironment. *BMC Genomics*, 14(1):469, 2013.
- [126] R.W. Tothill, A.V. Tinker, J. George, R. Brown, S.B. Fox, S. Lade, D.S. Johnson, M.K. Trivett, D. Etemadmoghadam, B. Locandro, N. Traficante, S. Fereday, J.A. Hung, Y.-E. Chiew, I. Haviv, Australian Ovarian Cancer Study Group, D. Gertig, A. deFazio, and D.D.L. Bowtell. Novel Molecular Subtypes of Serous and Endometrioid Ovarian Cancer Linked to Clinical Outcome. *Clinical Cancer Research*, 14(16):5198–5208, 2008.
- [127] G. Sergeant, R. van Eijnsden, T. Roskams, V. Van Duppen, and B. Topal. Pancreatic cancer circulating tumour cells express a cell motility gene signature that predicts survival after surgery. *BMC Cancer*, 12(1):527, 2012.
- [128] O.C. Maes, H.M. Schipper, H.M. Chertkow, and E. Wang. Methodology for Discovery of Alzheimer’s Disease Blood-Based Biomarkers. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 64A(6):636–645, 2009.
- [129] O.C. Maes, S. Xu, B. Yu, H.M. Chertkow, E. Wang, and H.M. Schipper. Transcriptional profiling of Alzheimer blood mononuclear cells by microarray. *Neurobiology of Aging*, 28(12):1795–1809, 2007.
- [130] V. Krishnamurthy, N.S. Issac, and J. Natarajan. Computational Identification of Alzheimer’s Disease Specific Transcription Factors using Microarray Gene Expression Data. *Journal of Proteomics & Bioinformatics*, 2(12):505–508, 2009.
- [131] M.G. Kendall and A. Stuart. *The Advanced Theory of Statistics: Distribution theory*. Griffin, 1977.
- [132] S. Li. Concise Formulas for the Area and Volume of a Hyperspherical Cap. *Asian Journal of Mathematics & Statistics*, 4:66–70, 2011.
- [133] A.K. Gayen. The Frequency Distribution of the Product-Moment Correlation Coefficient in Random Samples of Any Size Drawn from Non-Normal Universes. *Biometrika*, 38(1–2):219–247, 1951.
- [134] J.B.S. Haldane. A note on non-normal correlation. *Biometrika*, 36:467–468, 1949.

- [135] G.B. Hey. A new method for experimental sampling illustrated in certain non-normal populations. *Biometrika*, 30:68–80, 1938.
- [136] C.J. Kowalski. On the Effects of Non-Normality on the Distribution of the Sample Product-Moment Correlation Coefficient. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 21(1):1–12, 1972.
- [137] G. Jurman, R. Visintainer, S Riccadonna, M. Filosi, and C. Furlanello. The HIM glocal metric and kernel for network comparison and classification. arXiv:1201.2931 [math.CO], 2013.
- [138] M. Filosi, R. Visintainer, S. Riccadonna, G. Jurman, and C. Furlanello. Stability Indicators in Network Reconstruction. arXiv:1209.1654 [q-bio.MN], submitted, 2013.
- [139] T. Ideker and N.J. Krogan. Differential network biology. *Molecular Systems Biology*, 8:565, 2012.
- [140] M. Bockmayr, F. Klauschen, B. Györffy, C. Denkert, and J. Budczies. New network topology approaches reveal differential correlation patterns in breast cancer. *BMC Systems Biology*, 7(1):78, 2013.
- [141] A. Barla, G. Jurman, R. Visintainer, M. Squillario, M. Filosi, S. Riccadonna, and C. Furlanello. A Machine Learning Pipeline for Discriminant Pathways Identification. In N.K. Kasabov, editor, *Springer Handbook of Bio-/Neuroinformatics*, chapter 53, page 1200. Springer, Berlin, 2013.
- [142] Castro, C. and Krumsiek, J. and Lehrbach, N.J. and Murfitt, S.A. and Miska, E.A. and Griffin, J.L. A study of *Caenorhabditis elegans* DAF-2 mutants by metabolomics and differential correlation networks. *Molecular BioSystems*, 9:1632–1642, 2013.
- [143] M.F. Folstein, S.E. Folstein, and P.R. McHugh. "Mini-mentalstate". A practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research*, 12(3):189–198, 1975.
- [144] D. Amar, H. Safer, and R. Shamir. Dissection of Regulatory Networks that Are Altered in Disease via Differential Co-expression. *PLoS Computational Biology*, 9(3), 2013.
- [145] M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28:27–30, 2000.

- [146] M. Kanehisa, S. Goto, Y. Sato, M. Furumichi, and M. Tanabe. KEGG for integration and interpretation of large-scale molecular datasets. *Nucleic Acids Research*, 40:D109–D114, 2012.
- [147] T.H. Hwang, G. Atluri, M.Q. Xie, S. Dey, C. Hong, V. Kumar, and R. Kuang. Co-clustering phenomegenome for phenotype classification and disease gene discovery. *Nucleic Acids Research*, 40(19):e146, 2012.
- [148] D. Warde-Farley, S.L. Donaldson, O. Comes, K. Zuberi, R. Badrawi, P. Chao, M. Franz, C. Grouios, F. Kazi, C.T. Lopes, A. Maitland, S. Mostafavi, J. Montojo, Q. Shao, G. Wright, G.D. Bader, and Q. Morris. The GeneMANIA prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Research*, 38(suppl 2):W214–W220, 2010.
- [149] K.-C. Li, C.-T. Liu, W. Sun, S. Yuan, and T. Yu. A system for enhancing genome-wide coexpression dynamics study. *Proceedings of the National Academy of Sciences of the United States of America*, 101(44):15561–15566, 2004.
- [150] L. Crews, C. Patrick, A. Adame, E. Rockenstein, and E. Masliah. Modulation of aberrant CDK5 signaling rescues impaired neurogenesis in models of Alzheimer’s disease. *Cell Death & Disease*, 2(2):e120, 2011.
- [151] J.C. Cruz and L.-H. Tsai. Cdk5 deregulation in the pathogenesis of Alzheimers disease. *Trends in Molecular Medicine*, 10(9):452–458, 2004.
- [152] D. Lazer, A. S. Pentland, . Adamic, S. Aral, A. L. Barabasi, D. Brewer, N. Christakis, N. Contractor, J. Fowler, and M. Gutmann. Life in the network: the coming age of computational social science. *Science (New York, NY)*, 323(5915):721, 2009.
- [153] G. S. Becker. Irrational behavior and economic theory. *Journal of Political Economy*, 70(1):1–13, 1962.
- [154] D. K. Gode and S. Sunder. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101(1):119–37, 1993.
- [155] V. L. Smith. An experimental study of competitive market behavior. *Journal of Political Economy*, 70:111, 1962.
- [156] S. Parsons, M. Marcinkiewicz, J. Niu, and S Phelps. Everything you wanted to know about double auctions, but were afraid to (bid or) ask. *Technical report, Brooklyn College*, 2006.

-
- [157] J. Doyne Farmer, Paolo Patelli, and Ilija I. Zovko. The predictive power of zero intelligence in financial markets. *Proceedings of the National Academy of Sciences of the United States of America*, 102(6):2254–2259, 2005.
- [158] V. L. Smith. Microeconomic systems as an experimental science. *American Economic Review*, 72(5):923–55, 1982.