Network Medicine

Deisy Morselli Gysi

2021-02-03

Contents

2nd Workshop in Advanced Bioinformatics: Network Medicine.

Chapter 1

Workshop on Network Medicine

Network medicine (**NetMed**) is a field of Network Science that uses Systems Biology to understand its impact on medicine, and it is mostly focused on network topology. In NetMed we often use biological networks to represent the topological space and it be used to identify disease modules, their relationship to other diseases, drug repurpusing, and drug combinations. The biological networks often used for it are based on protein-protein interactions (PPI), Gene-Disease-Associations (GDA), and Drug-Targets.

In this workshop, we will learn:

- how to identify disease modules (Session 3.1);
- how to predict disease comorbidities (Session 3.3);
- how to repurpuse drugs using a network approach (Session 4.1).

1.1 What are networks?

> Adapted from Gysi and Nowick (2020).

Network Science is broadly employed in many fields: from understanding *how friends bond in a party* to *how animals interact*; from *how superheroes appear in the same comic books* to *how genes can be related to a specific biological process*. Network analysis is especially beneficial for understanding complex systems, in all research fields. Examples of complex biological or medical systems include gene regulatory, ecological and neuropsychology networks. In this workshop, focus is given to applications of Network Science to the medical sciences.

Here, I will start by introducing the basic network terminologies and then explore how can we define and identify disease modules, identify disease commorbidities and lastly, we will learn how to repurpuse drugs for diseases with known modules. For each step, I will then present some classical and some new studies.

It is expected some degree of familiarity with `R`, `ggplot2`, `tidyr`, and `igraph`.

1.2 Terminology

While the nature of each system, i.e. what its entities are and what kind of interactions they have, is different, there are common notations. A short review of common network terms can be found in Session 1.2.1 and a brief review of biological terms can be found in Session 1.2.2.

The set of interactions among a set of entities is, in general, called a graph or a network (Newman, 2018; Barabási, 2016). In graph theory, each entity is called a vertex, while in network notation it is called a node

Table 1.1: Mathematical Representation of a Network: Adjacency Matrix

	A	C	B	D	H	F	J	I	E	G
A	0	1	1	1	0	0	0	1	1	0
C	1	0	0	0	1	0	1	0	0	0
B	1	0	0	1	0	1	0	0	0	1
D	1	0	1	0	0	0	0	0	0	0
H	0	1	0	0	0	0	0	0	0	0
F	0	0	1	0	0	0	0	0	0	0
J	0	1	0	0	0	0	0	0	0	0
I	1	0	0	0	0	0	0	0	0	0
E	1	0	0	0	0	0	0	0	0	0
G	0	0	1	0	0	0	0	0	0	0

Table 1.2: Mathematical Representation of a Network: Edge List

Source	Target
A	D
C	H
B	F
C	J
A	I
B	D
A	B
A	E
B	G
A	C

(Barabási, 2016). Accordingly, the connections between two entities are called edges or links, respectively (Barabási, 2016). In this workshop, I will always use the network notation, unless otherwise specified. The total number of nodes in a network is often denoted as **N** and the number of links in a network is denoted as **L**. While nodes can receive a label, links in general, are not labeled (Barabási, 2016) (although, in many cases, weights can also be perceived as a label). A network can be represented mathematically as an adjacency matrix (usually denoted as **A**) (Table 1.1), an edge-list (Table 1.2), or visually as a graph (Figure 1.1).

Links of a network can possess a direction (normally depicted by an arrow), which indicates that the interaction is asymmetric, e.g. one gene is regulating another gene, or a person follows somebody else in a social network. Networks with directed links are called directed networks, while networks without directed interactions or in which the direction is not known are referred to as undirected networks, e.g. collaboration in the same study or interactions between proteins. In NetMed - and in this workshop - most of the times we assume that networks do not posses a direction. The links can also have a weight to express the strength of the interaction, which results in a weighted network (Newman, 2018; Barabási, 2016). Usually, the weight is graphically displayed as the thickness or the length of the links.

Networks can also have different dimensions. These dimensions can be understood as layers (or different link types) of the same system (Kurant and Thiran, 2006; Kivela et al., 2014). For example, in a multi-omics multilayer system, each layer can be constructed using different -omics data (for example genomics, transcriptomics, proteomics, etc.) where the 'whole' biological system can be understood as a network of networks (De Domenico, 2017). The topology and the dynamic properties of the whole network can be changed by simply transforming the weights of the interactions, or by ignoring that nodes can interact in many ways (Mucha et al., 2010; Radicchi and Arenas, 2013) also ignoring the node's importance to the system. We will not deal with multi-layer networks in this workshop.

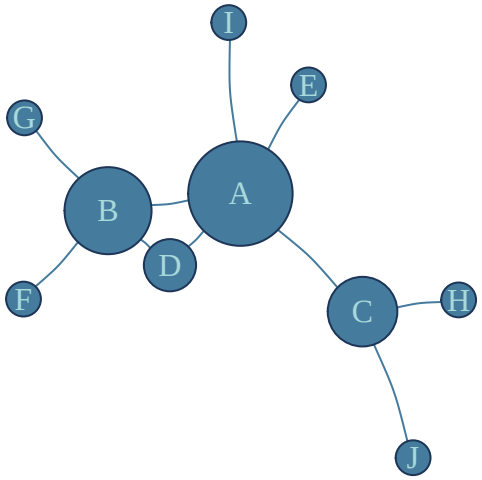


Figure 1.1: Visual Representation of a Network: Graph

1.2.1 Network Terminology

- A **network** is a pair **G = (N, L)** of a set **N** of nodes connected by a set **L** of links.
- Two nodes are **neighbours** if they are **connected**. The **degree** (d) of a node is the **number of nodes** it interacts with (Bondy and Murty, 2008).
- The **weight** is a measure of how strong a particular interaction is (Bondy and Murty, 2008).
- The **strength** of a node is the **sum of the weights** attached to links belonging to a node (Barrat et al., 2003).
- The **direction** of a link specifies the source (starting point) and a target (endpoint) where the interaction occurs (Barabási, 2016) .
- **Hubs** are nodes with a **much larger degree** compared to the average degree value (Barrat et al., 2003).
- A set of highly interconnected nodes is a **module** or **cluster** (Li and Horvath, 2009). Two nodes are connected in a network, if a sequence of adjacent nodes, a **path**, connects them (Barabási and Oltvai, 2004).
- The **shortest path length** is the number of links along the shortest path connecting two nodes (Barabási and Oltvai, 2004).
- The **average path length** is the average of the shortest paths between all pairs of nodes (Barabási and Oltvai, 2004).
- The **diameter** is the maximum distance between two nodes (Bondy and Murty, 2008).
- The **modularity index** is a measure of the strength of the network division into modules when this measure is maximized; it can be used for identifying nodes communities (Newman, 2018).
- **Preferential attachment** is the tendency of nodes to form new links preferentially to nodes with a high number of links (Barabási and Albert, 1999; Vázquez, 2003).
- The probability that a random node in the network has a particular degree is given by the **degree distribution** (Barabási and Oltvai, 2004).

- A **bipartide graph** is a network in which the nodes can be divided into two disjoint sets of nodes such that links connect nodes from the two sets to each other, but never inside the same set (Barabási, 2016). In those networks, most of the network measures are calculated differently than in a unipartide network.
- The **clustering coefficient** describes the degree with which a node is connected to all its neighbours (Barabási and Oltvai, 2004).
- The **global clustering** coefficient measures the total number of triangles in a network (Barabási, 2016).
- The **average clustering** coefficient is the average of the clustering coefficient of all nodes in a network (Barabási and Oltvai, 2004).
- **Centrality** is a set of measures that have been proposed to help to define the most central nodes. It has many interpretations for autonomy, control, risk, exposure, influence and power (Borgatti and Everett, 2006).
  - **Closeness centrality** is defined as the average distance from a single vertex to all other vertices(Newman, 2018).
  - **Betweenness centrality** is defined as the total number of shortest paths between pairs of nodes that pass through a particular node (Newman, 2018).
- **Global measures** are measures that describe the whole network, for example, *degree distribution; average clustering coefficient; path length; modularity index.*
- **Local measures** are characteristics of individual nodes of a network, such as their *degree* and *centrality.*

1.2.2 Biological Terminology

- **DNA** is the hereditary material of most organisms – usually all cells of an organism have the same DNA (Slack, 2013).
- **Genes** are the basic physical and functional units of heredity. They are parts of the DNA and contain the information for producing functional RNAs and proteins. (Slack, 2013).
- **Proteins** are large, complex molecules that play many critical roles in the body. The proteins are responsible for most of the work in cells and are necessary for structure, function, and regulation of the cells. They can act as enzymes, antibodies, transporters, transcription factors etc. (Slack, 2013).
- The **RNA** is synthesized from the DNA but has different properties and functions than the DNA. Some RNAs carry out biological functions in a cell, while others, messenger RNA (mRNA), are turned into proteins that fulfil biological functions (Slack, 2013).
- A **non-coding RNA (ncRNA)** is an RNA that does not encode a protein. NcRNAs often play a role in gene regulation (Mattick and Makunin, 2006).
- **microRNAs (miRNA)** are examples of ncRNA; they are involved in posttranscriptional regulation of protein expression (Tanase et al., 2012).
- **Gene expression** is, in short, the coupled process of transcription (from DNA to RNA) and translation (from RNA to proteins) to transform the stored information inside the DNA into proteins (Slack, 2013).
- **RNA-Seq** is a technique used to sequence the RNAs in a sample. The result is the snapshot abundance of all RNAs expressed in the sample at a particular time, often called the transcriptome (Metzker, 2010).
- **Microarrays**, or **gene chips**, are chips with thousands of tiny spots containing a known DNA sequence. It is used to measure the abundance of mRNAs by eminence of fluorescence (Slack, 2013).

- **Transcription Factors** are DNA binding proteins that activate or repress the transcription of particular target genes (Latchman, 1997).
- **Gene Regulatory Factors** are responsible for controlling the expression of genomic information and include transcription factors, co-factors, epigenetic modifiers, miRNAs and others (Hobert, 2008).
- **Systems Biology** examines the structures and dynamics of cellular and organismal function, instead of isolated characteristics of a cell or organism.
- **Drug repositioning** (or drug repurposing) is the process of redeveloping a compound for use in a different disease.
- **Yeast-Two-Hybrid (Y2H)** systems is a system to measure protein-protein interaction. Two proteins to be tested for interaction are expressed in yeast; one protein is fused to a DNA-binding domain from a transcription factor while another protein (Y) is fused to a transcription activation domain. If X and Y interact, there will be a formation of a colony on media used as evidence of the interaction of X and Y (Parrish et al., 2006).
- **Protein complex immunoprecipitation** is an alternative method for measuring protein interactions. It involves immunoprecipitation of the protein bait, purification of the complex, and the identification of the interacting partners.
- **High-throughput Mass Spectrometry** has the ability to detect a characteristic mass to charge ratio of different substances in a sample. It is used to identify the proteins present in a sample (Kempa et al., 2019).
- **Chromatin immunoprecipitation followed by sequencing (ChIP–Seq)** can be used to identify binding sites of transcription factors in the DNA or of histone modification in a genome-wide manner (Park, 2009).
- **Chromatin Isolation by RNA Purification followed by sequencing (ChIRP-seq)** maps lncRNA interactions to the chromatin (Park, 2009).
- **Genome-wide association studies (GWAS)** are studies where millions of SNPs are tested for association with a particular phenotype using hundreds or thousands of individuals. Those studies shed light on the genetic basis of complex traits.
- **Omics** is a term that refers to the study of different areas in biology, and indicates the totality of some kind, e.g. genome, transcriptome, proteome, etc.

Chapter 2

Data Commonly Used in Network Medicine

In NetMed we are often interested in understanding *how genes associated to a particular disease can influence each other*, *how two diseases are similar (or different)*, *and how a drug can be used in different set-ups.*

For that end, it is necessary to use data sets that are able to represent those associations: **Protein-Protein Interactions** are used as a map of the interactions inside our cells (Session 2.1); **Gene-Disease-Associations** are used for us to identify genes that were previously associated to diseases, often using a GWAS approach (Session 2.2); and Drug-Target interactions, often measured by identifying physical binding of a therapeutic compound (often a drug) and a protein (Session 2.3).

2.1 Protein Protein Interaction Networks

In PPI networks, the nodes represent proteins and they are connected by a link if they physically interact with each other (Rual et al., 2005). Typically, these interactions are measured experimentally, for instance with the Yeast-Two-Hybrid (Y2H) system (Uetz et al., 2000), or by protein complex immunoprecipitation followed by high-throughput Mass Spectrometry (Zhang et al., 2008; Koh et al., 2012), or inferred computationally based on sequence similarity (Fong et al., 2004). PPI can be used to infer gene functions and the association of sub-networks to diseases (Menche et al., 2015). In this type of network, a highly connected protein tends to interact with proteins that are less connected, probably to prevent unwanted cross-talk of functional modules. As mentioned, most of the methods in network medicine are based on PPI.

2.1.1 Measuring PPIs

Protein Protein Interactions can be measured mainly using three different techniques:

1. By the creation of protein protein interaction maps derived from existing scientific literature;
2. Using computational predictions of PPIs based on available orthogonal information; and
3. By systematic experimental mapping of proteins identify complex association and/or binary interactions. We will focus here only in the third.

Co-complex associations interrogate a protein composition of protein complex in one or several cell lines. The most common approach uses affinity purification to extract the proteins that associate with the *bait* proteins followed by mass expectometry in order to identify proteins that associate with the *bait.* This

approach is often used for simple organisms, however similar approaches have been reported for humans. Unfortunately, achieving stable expression of bait proteins is challenging. Co-complex map associations are composed by indirect and some direct binary associations. However, raw association data cannot distinguish the indirect from direct association and therefore, co-complex datasets have to be filtered and needs to have incorporated prior knowledge that might lead to bias towards super-start genes. On the other side, for experimental determination of binary interactions between proteins, all possible pair of proteins are systematically tested to generate a data set of all possible biophysical interactions.

Because the human genome is composed by ~20,000 unique genes - not even considering its isophorms - we would have ~200 million possible combinations in order to robust systematically identify interactions. To meet this requirement Yeast-to-Hybrid (Y2H) technology is the only one that can meet this requirement. This technology is able to interrogate hundreds of millions of human proteins pairs for binary interactions. In short the method works as follows: Protein of interest X and a DNA binding domain (DBD-X) fuse to form *bait*. Fusion of transcriptional activation domain (AD-Y) and a cDNA library Y results in *prey*. Those two form the basis of the protein–protein interaction detection system. Without bait–prey interaction, the activation domain is unable to restrict the gene-to-gene expression drive.

2.1.2 Commonly used data sources for PPIs

PPIs can be found from different sources. I list here some well known databases for that.

1. Binary PPIs derived from high-throughput yeast-two hybrid (Y2H) experiments:

- HI-Union (Luck et al., 2020)

2. Binary PPIs three-dimensional (3D) protein structures:

- Interactome3D (Mosca et al., 2013)
- Instruct (Meyer et al., 2013)
- Insider (Meyer et al., 2018)

3. Binary PPIs literature curation:

- PINA (Cowley et al., 2012)
- MINT (Licata et al., 2012)
- LitBM17 (Luck et al., 2020)
- Interactome3D
- Instruct
- Insider
- BioGrid (Chatr-Aryamontri et al., 2017)
- HINT (Das and Yu, 2012)
- HIPPIE (Alanis-Lobato et al., 2017)
- APID (Alonso-López et al., 2019)
- InWeb (Li et al., 2016)

4. PPIs identified by affinity purification followed by mass spectrometry:

- BioPlex (Huttlin et al., 2017)
- QUBIC (Hein et al., 2015)
- CoFrac (Wan et al., 2015)
- HINT
- HIPPIE

- APID
- LitBM17
- InWeb

5. Kinase substrate interactions:

- KinomeNetworkX (Cheng et al., 2014)
- PhosphoSitePlus (Hornbeck et al., 2015)

6. Signaling interactions:

- SignaLink (Fazekas et al., 2013)
- InnateDB (Breuer et al., 2013)

7. Regulatory interactions:

- ENCODE consortium.

2.1.3 Understanding a PPI

For this workshop, we will be using for this workshop is a combination of a manually curated PPI that combines all previous data sets. The data can be found here. This PPI was previously published in Gysi et al. (2020).

Before we can start any analysis using this interactome, let us first understand this data.

The PPI contains the EntrezID and the HGNC symbol of each gene, and some might not have a proper map, therefore, it should be removed from further analysis. Moreover, we might have loops, and those should also be removed.

Let us begin by preparing our environment and calling all libraries we will need at this point.

```
require(data.table)
require(tidyr)
require(igraph)
require(dplyr)
require(magrittr)
require(ggplot2)
```

Let's read in our data.

```
PPI = fread("./data/PPI_Symbol_Entrez.csv")
```

```
head(PPI)
```

GeneA_ID	GeneB_ID	Symbol_A	Symbol_B
9796	56992	PHYHIP	KIF15
7918	9240	GPANK1	PNMA1
8233	23548	ZRSR2	TTC33
4899	11253	NRF1	MAN1B1
5297	8601	PI4KA	RGS20
6564	8933	SLC15A1	RTL8C

Let's transform our edge-list into a network.

```
gPPI = PPI %>%
  select(starts_with("Symbol")) %>%
  filter(Symbol_A != "") %>%
  filter(Symbol_B != "") %>%
  graph_from_data_frame(., directed = F) %>%
  simplify()

gPPI
```

```
## IGRAPH 12d998e UN-- 18507 322289 --
## + attr: name (v/c)
## + edges from 12d998e (vertex names):
##  [1] PHYHIP--TTR      PHYHIP--NFE2     PHYHIP--DYRK1A   PHYHIP--HNRNPA1
##  [5] PHYHIP--COPS6    PHYHIP--SUPT5H   PHYHIP--SMARCC2  PHYHIP--EEF1A1
##  [9] PHYHIP--TRIP6    PHYHIP--NDUFV3   PHYHIP--CA10     PHYHIP--ERG28
## [13] PHYHIP--S100A13  PHYHIP--PPIE     PHYHIP--LIMD1    PHYHIP--ANKRD12
## [17] PHYHIP--ZZEF1    PHYHIP--PRMT5    PHYHIP--KIF15    PHYHIP--MED8
## [21] PHYHIP--PRKD2    PHYHIP--PAQR5    PHYHIP--MAGED4B  PHYHIP--NDRG1
## [25] PHYHIP--PTRH2    PHYHIP--HDAC11   PHYHIP--METTL18  PHYHIP--PNPLA2
## [29] PHYHIP--TMEM255B PHYHIP--WDR89    PHYHIP--FAM131A  GPANK1--TAF1
## + ... omitted several edges
```

How many genes do we have? How many interactions?

Next, let's check the degree distribution:

```
dd = degree(gPPI) %>% table() %>% as.data.frame()
names(dd) = c('Degree', "Nodes")
dd$Degree %<>% as.character %>% as.numeric()
dd$Nodes  %<>% as.character %>% as.numeric()

ggplot(dd) +
  aes(x = Degree, y = Nodes) +
  geom_point(colour = "#1d3557") +
  scale_x_continuous(trans = "log10") +
  scale_y_continuous(trans = "log10") +
  theme_minimal()
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

Most of the proteins have few connections, and very few proteins have lots of connections. Who's that protein?

```
degree(gPPI) %>%
  as.data.frame() %>%
  arrange(desc(.)) %>%
  filter(. > 1000) %>%
  knitr::kable()
```

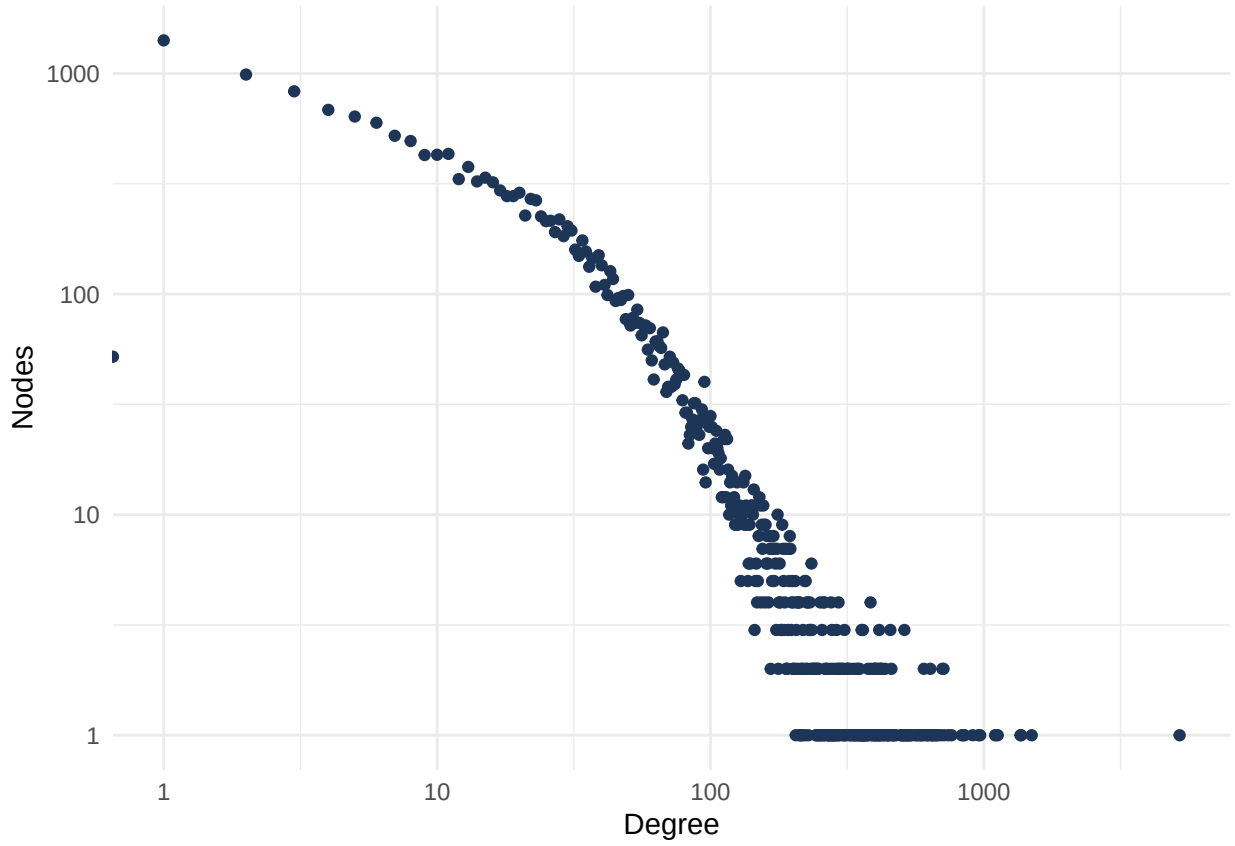


Figure 2.1: PPI Degree Distribution

	.
UBC	5199
ETS1	1496
GATA2	1369
CTCF	1361
EP300	1124
MYC	1107
AR	1099

2.1.4 Exercises

Now is your turn. Spend some minutes understanding the data and getting some familiarity with it.

1. What are the top 10 genes with higher degree?
2. Are those genes connected?

2.2 Gene Disease Association

A Gene-Disease-Association (GDA) database are typically used to understand the association of genes to diseases, and model the underlying mechanisms of complex diseases. Those associations often come from GWAS studies and knock-out studies.

2.2.1 Commonly used data sources for GDAs

As PPIs, GDAs can be found from different sources and with different evidences for each Gene-Disease association. I list here some well known databases for that.

- CTD – Curated scientific literature (Davis et al., 2020)
- OMIM – Curated scientific literature (McKusick, 2007)
- DisGeNet – Based on OMIM, ClinVar and other data bases (Piñero et al., 2019)
- Orphanet – Validated - and non validated - GDAs
- ClinGen – Validated - and non validated - GDAs (Rehm et al., 2015)
- ClinVar – Different levels of evidence (Landrum et al., 2019)
- GWAS catalogue – GWAS associations to diseases (Buniello et al., 2018)
- PheGenI – GWAS associations to diseases (Ramos et al., 2013)
- lncRNADisease – Experimental validated lncRNAs in diseases (Chen et al., 2012)
- HMDD – Experimental validated miRNAs in diseases (Huang et al., 2018)

2.2.2 Understanding a GDA dataset

We will use in this workshop Gene-Disease-Association from DisGeNet. And can be found here.

Similar to the PPI, let us first get some familiarity with the data, before we are able to perform any analysis.

Let's read in the data and again, do some basic statistics.

```
GDA = fread(file = 'data/curated_gene_disease_associations.tsv', sep = '\t')

head(GDA)
```

geneId	geneSymbol	DSI	DPI	diseaseId	diseaseName	diseaseType	diseaseClass	diseaseSema
1	A1BG	0.700	0.538	C0019209	Hepatomegaly	phenotype	C23;C06	Finding
1	A1BG	0.700	0.538	C0036341	Schizophrenia	disease	F03	Mental or B
2	A2M	0.529	0.769	C0002395	Alzheimer's Disease	disease	C10;F03	Disease or S
2	A2M	0.529	0.769	C0007102	Malignant tumor of colon	disease	C06;C04	Neoplastic P
2	A2M	0.529	0.769	C0009375	Colonic Neoplasms	group	C06;C04	Neoplastic P
2	A2M	0.529	0.769	C0011265	Presenile dementia	disease	C10;F03	Mental or B

The first thing to notice is the inconsistency with the disease names, in order to be able to work with it, let's first put every disease to lower-case.

```
Cleaned_GDA = GDA %>% filter(diseaseType == 'disease') %>%
  mutate(diseaseName = tolower(diseaseName)) %>%
  select(geneSymbol, diseaseName, diseaseSemanticType) %>%
  unique()

dim(Cleaned_GDA)
```

```
## [1] 60478     3
```

```
dim(GDA)
```

```
## [1] 84038    16
```

```
numGenes = Cleaned_GDA %>%
  group_by(diseaseName) %>%
  summarise(numGenes = n()) %>%
  ungroup() %>%
  group_by(numGenes) %>%
  summarise(numDiseases = n())
```

Let's also understand the degree distribution of the diseases.

```
ggplot(numGenes) +
  aes(x = numGenes, y = numDiseases) +
  geom_point(colour = "#1d3557") +
  scale_x_continuous(trans = "log10") +
  scale_y_continuous(trans = "log10") +
  labs(x = "Genes", y = "Diseases")+
  theme_minimal()
```

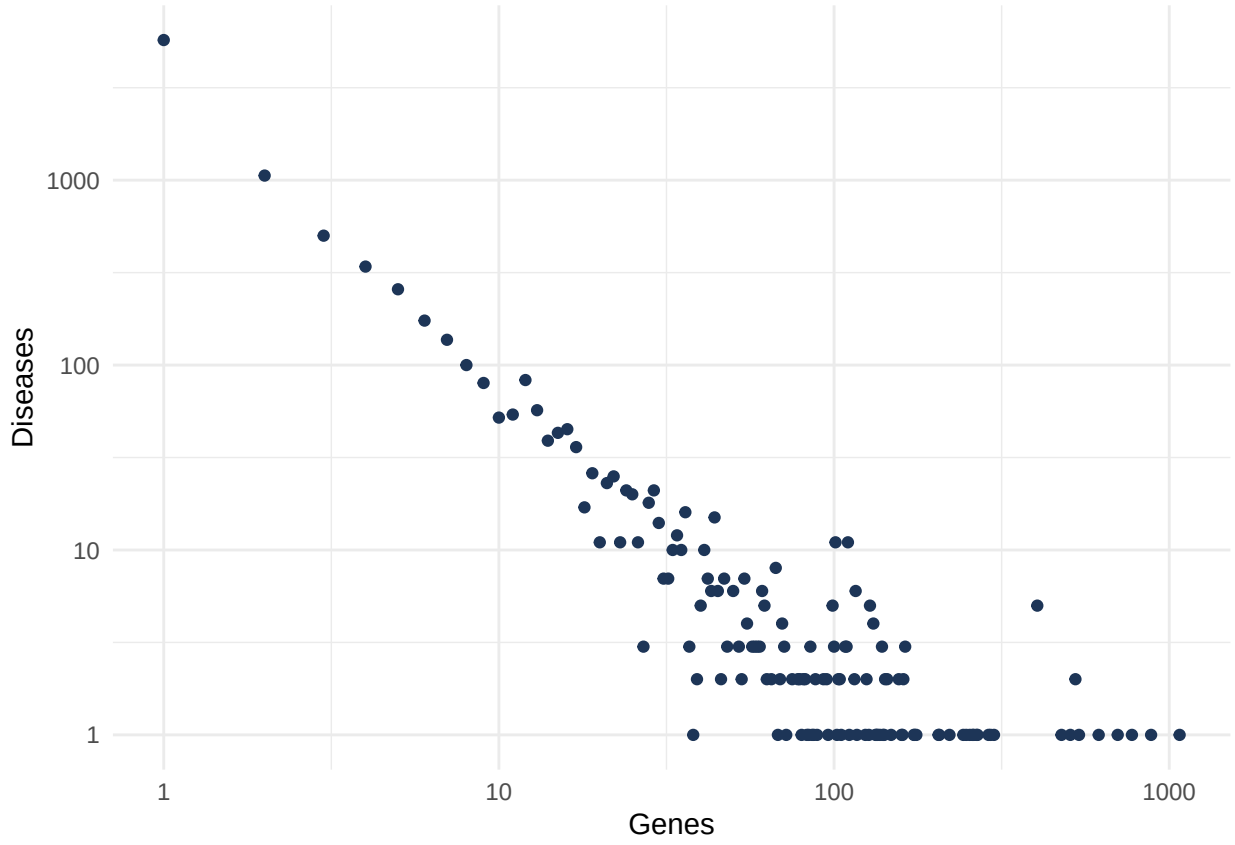


Figure 2.2: Gene-Disease degree distribution

Because we want to focus in well studied diseases, and also that are known to be complex diseases, let's filter for diseases with at least 10 genes.

```
Cleaned_GDA %<>%
  group_by(diseaseName) %>%
  mutate(numGenes = n()) %>%
  filter(numGenes > 10)

Cleaned_GDA$diseaseName %>% unique() %>%  length()
```

```
## [1] 920
```

2.2.3 Exercises

Now is your turn. Spend some minutes understanding the data and getting some familiarity with it.

1. What are the top 10 genes mostly involved with diseases? What are those diseases?
2. What are the top 10 highly polygenic diseases?
3. What are the top 10 highly polygenic disease classes?

2.3 Drug-Targets

A *druggable* target is a protein, peptide or nucleic acid that has activity which can be modulated by a drug. A *drug* can be any small molecular weight chemical compound (SMOL) or a biologic (BIOL), such as an antibody or a recombinant protein that can treat a disease or a sympthom.

2.3.1 Properties of an ideal drug target:

A drug-target has a couple of proprieties that are highly desired when constructing the drug (Gashaw et al., 2011):

- Target is disease-modifying and/or has a proven function in the pathophysiology of a disease.
- Modulation of the target is less important under physiological conditions or in other diseases.
- If the druggability is not obvious (e.g. as for kinases) a 3D-structure for the target protein or a close homolog should be available for a druggability assessment.
- Target has a favorable 'assayability' enabling high throughput screening.
- Target expression is not uniformly distributed throughout the body.
- A target/disease-specific biomarker exists to monitor therapeutic efficacy.
- Favorable prediction of potential side effects according to phenotype data (e.g. in k.o. mice or genetic mutation databases).
- Target has a favorable IP situation (no competitors on target, freedom to operate).

2.3.2 Commonly used data sources for GDAs

There are a couple of really good data sets that report drug-target interactions, I list here three good examples:

1. DrugBank (Wishart et al., 2006; Wishart et al., 2017)
2. CTD (Davis et al., 2020)
3. Broad Institute Drug Repositioning Hub (Corsello et al., 2017)

2.3.3 Understanding a Drug-Target dataset

For this workshop we will use the drug bank drug-target dataset, and can be found here. This dataset is from Drug-Bank, and has been previously parsed for your convenience. The original file is an XML file, and needs to be carefully handled to get information needed.

Similar to the PPI and the GDA, let us understand a little bit of the data set, and what kind of information we have here.

```
DT = fread(file = 'data/DB_DrugTargets_1201.csv')

head(DT)
```

i	ID	Name	Started_commer	Ended_commer	ATC	State	Approved	Gene_Target	DB_id
1	DB00001	Lepirudin	1997-03-13	2012-07-27	B01AE	liquid	approved	F2	BE0000
2	DB00002	Cetuximab	2004-06-29	NA	L01XC	liquid	approved	EGFR	BE0002
2	DB00002	Cetuximab	2004-06-29	NA	L01XC	liquid	approved	FCGR3B	BE0002
2	DB00002	Cetuximab	2004-06-29	NA	L01XC	liquid	approved	C1QA	BE0002
2	DB00002	Cetuximab	2004-06-29	NA	L01XC	liquid	approved	C1QB	BE0002
2	DB00002	Cetuximab	2004-06-29	NA	L01XC	liquid	approved	C1QC	BE0002

```
Cleaned_DT = DT %>%
  filter(organism == 'Humans') %>%
  select(Gene_Target, Name,ID, Type, known_action) %>%
  unique()

dim(Cleaned_DT)
```

```
## [1] 22931     5
```

```
dim(DT)
```

```
## [1] 26817    16
```

```
head(Cleaned_DT)
```

```
##    Gene_Target      Name      ID        Type known_action
## 1:          F2 Lepirudin DB00001 Polypeptide          yes
## 2:        EGFR Cetuximab DB00002 Polypeptide      unknown
## 3:      FCGR3B Cetuximab DB00002 Polypeptide      unknown
## 4:        C1QA Cetuximab DB00002 Polypeptide      unknown
## 5:        C1QB Cetuximab DB00002 Polypeptide      unknown
## 6:        C1QC Cetuximab DB00002 Polypeptide      unknown
```

```
TargetDist = Cleaned_DT %>%
  group_by(Gene_Target) %>%
  summarise(numDrugs = n())

DrugDist = Cleaned_DT %>%
  group_by(ID) %>%
  summarise(numTargets = n())
```

```
ggplot(TargetDist) +
  aes(x = numDrugs) +
  geom_histogram(colour = "#1d3557", fill = "#a8dadc" ) +
  labs(x = "Targets", y = "Drugs")+
  theme_minimal()
```

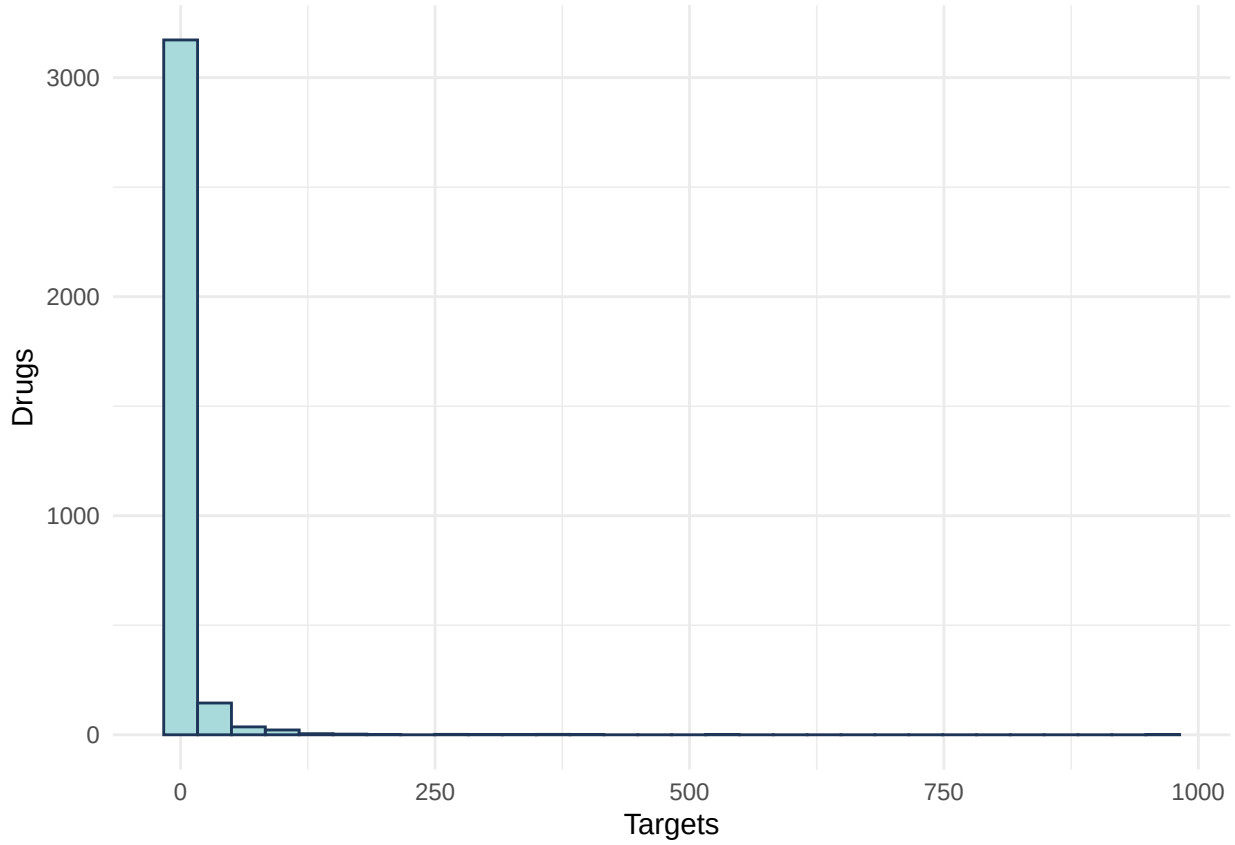


Figure 2.3: Target distribution

Which Target is the most targetted gene?

```
TargetDist %>%
  arrange(desc(numDrugs)) %>%
  filter(numDrugs > 400)
```

```
## # A tibble: 2 x 2
##   Gene_Target numDrugs
##   <chr>          <int>
```

```
## 1 CYP3A4           966
## 2 ABCB1            524
```

```
ggplot(DrugDist) +
  aes(x = numTargets) +
  geom_histogram(colour = "#1d3557", fill = "#a8dadc" ) +
  labs(y = "Targets", x = "Drugs")+
  theme_minimal()
```

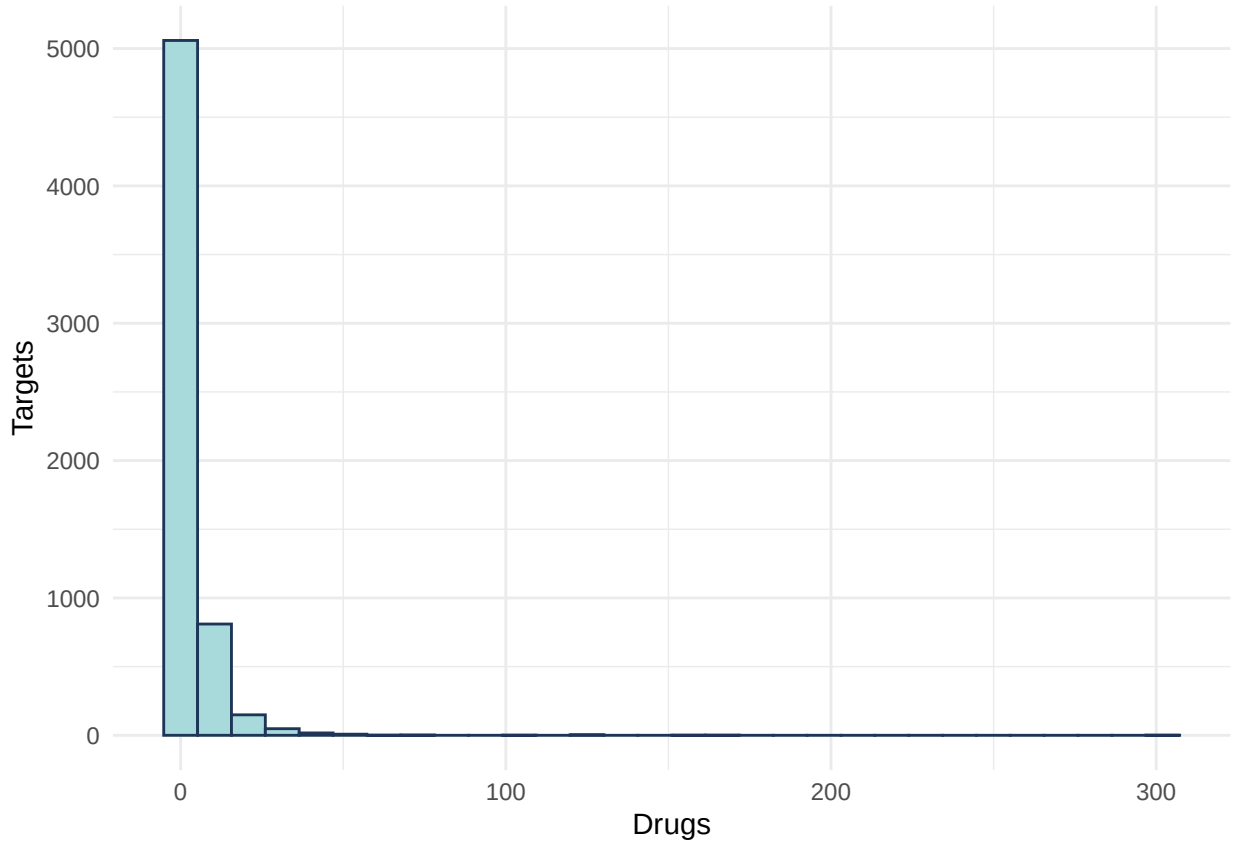


Figure 2.4: Drug distribution

2.3.4 Exercises

Let us understand a little bit more about the data.

1. What are the top 10 genes mostly targeted by drugs? Are they types are they mostly?
2. What are the top 10 most promiscuous drugs? What are their indication?

Chapter 3

Methods fo Disease Module Identification and Disease Similarity

In this chapter, I will introduce the main methods used in Network Medicine. We will start by understanding what a Disease Module is (Session 3.1), how can we calculate its significance and also understand its importance. We next will explore the disease separation (Session 3.3), how to calculate, make interpretations.

3.1 Disease Module

In biological networks often genes involved in the same topological communities are also associated with similar biological processes (Ahn et al., 2010). It also reflects on *how diseases localized themselves in the interaction*; meaning that disease modules are highly localized in specific network neighborhoods (Menche et al., 2015) (Figure 3.1).

3.1.1 Largest connected component

The size of the largest connected component (LCC) is the number of nodes that form a connected subgraph (in our case, it is the number of proteins that are interconnected in the PPI). Many properties of this quantity allow us to understand how a particular disease interacts with the interactome. It is important to note here that this measure is highly dependent on the completeness of an interactome. If a link between a protein and their counterparts is unknown – therefore missing – we might say that that particular node is not involved in a disease module (or that the LCC is not significant).

However, just computing this number might not be informative, and it is expected a randomness. To calculate this randomness, we often calculate the significance of the LCC by selecting proteins in the interactome with similar degrees (aka degree preserving randomization).

To calculate the significance of the LCC one can calculate its Z-Score or simply calculate the empirical probability under the curve from the empirical distribution. The Z-score is given by:

$$Z - Score_{LCC} = \frac{LCC - \mu_{LCC}}{\sigma_{LCC}}$$

3.1.2 Example in real data

Our first task now, is to understand if some diseases, from our `Cleaned_GDA` are able to form a Disease-Module. Let's start doing it for Schizophrenia and later we will add some more diseases.

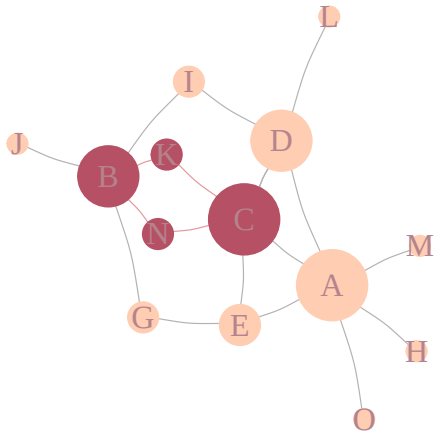


Figure 3.1: Disease-Module. A schematic of a PPI, in pink we see genes associated with a disease, forming a connected component of size 4.

The idea now is: Gather the genes associated to our disease in the data, find them in the PPI, check if they form a connected component, check the significance of the component and visualize the Disease-Module.

```
# First, let's attach all packages we will need.
require(NetSci)
require(magrittr)
require(dplyr)
require(igraph)
```

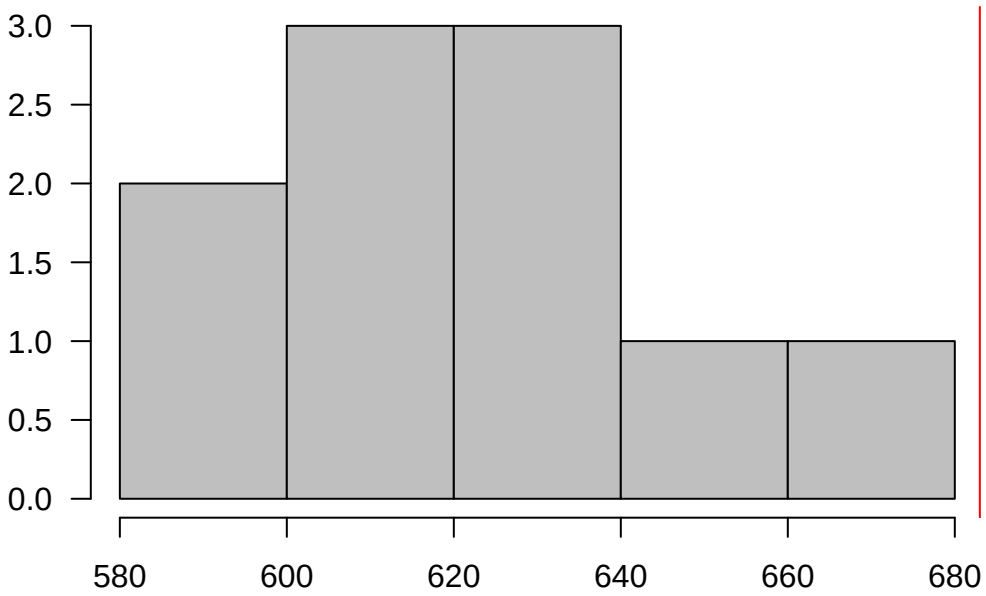
```
#First, let's select genes that are associated with Schizophrenia.

SCZ_Genes =
  Cleaned_GDA %>%
  filter(diseaseName %in% 'schizophrenia') %>%
  pull(geneSymbol) %>%
  unique()

# Next, let's see how they are localized in the PPI.
# Fist, we have to make sure all genes are in the PPI.
# Later, we calculate the LCC.
# And lastly, let's visualize it.

SCZ_PPI = SCZ_Genes[SCZ_Genes %in% V(gPPI)$name]
gScz = gPPI %>% induced.subgraph(., SCZ_PPI)

components(gScz)
```

```
components(gScz)$csize %>% max
```

```
## [1] 683
```

```
# The size of the LCC is 683. But... How does it compare to a random selection genes?

LCC_scz = LCC_Significance(N = 10, Targets = SCZ_PPI,
```

```
                          G = gPPI)
Histogram_LCC(LCC_scz)
```



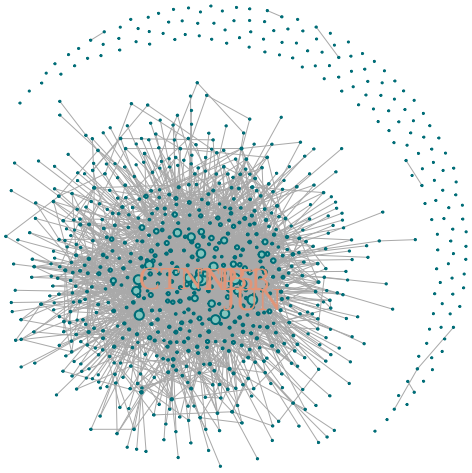
.

LCC: 683 (621.8 +− 24.79; p: 0)

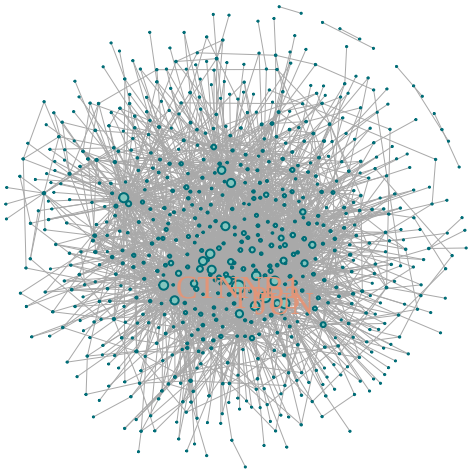
```
gScz
```

```
## IGRAPH c564e9d UN-- 846 2564 --
## + attr: name (v/c)
## + edges from c564e9d (vertex names):
##  [1] PI4KA  --SP1    F2     --SP1    DNM1   --CTCF   GSK3B  --HSPA1A
##  [5] DNM1   --GRB2   SP1    --GRB2   MET    --GRB2   GSK3B  --MAPK14
##  [9] SP1    --MAPK14 CTCF   --MAPK14 GRB2   --MAPK14 MET    --ACTB
## [13] GRB2   --ACTB   MAPK14 --ACTB   GSK3B  --SOX10  SP1    --SOX10
## [17] PAX6   --SOX10  SP1    --CCNA2  ACTB   --MTNR1A ACTB   --GSN
## [21] PI4KA  --JUN    GSK3B  --JUN    SMARCA2--JUN    SP1    --JUN
## [25] MAPK14 --JUN    SOX10  --JUN    GSK3B  --ESR1   SMARCA2--ESR1
## [29] SP1    --ESR1   HSPA1A --ESR1   FMR1   --ESR1   MAPK14 --ESR1
## + ... omitted several edges
```

```
V(gScz)$size = degree(gScz) %>%
  CoDiNA::normalize()
V(gScz)$size = (V(gScz)$size + 0.1)*5
V(gScz)$color = '#83c5be'
V(gScz)$frame.color = '#006d77'
V(gScz)$label = ifelse(V(gScz)$size  > 4, V(gScz)$name, NA )
V(gScz)$label.color = '#e29578'

E(gScz)$width = edge.betweenness(gScz, directed = F) %>% CoDiNA::normalize()
E(gScz)$width = E(gScz)$width + 0.01
E(gScz)$weight = E(gScz)$width
plot(gScz)
```



```
gScz %<>% delete.vertices(., degree(.) == 0)

V(gScz)$size = degree(gScz) %>%
  CoDiNA::normalize()
V(gScz)$size = (V(gScz)$size + 0.1)*5
V(gScz)$color = '#83c5be'
V(gScz)$frame.color = '#006d77'
V(gScz)$label = ifelse(V(gScz)$size  > 4, V(gScz)$name, NA )
V(gScz)$label.color = '#e29578'

E(gScz)$width = edge.betweenness(gScz, directed = F) %>% CoDiNA::normalize()
E(gScz)$width = E(gScz)$width + 0.01
E(gScz)$weight = E(gScz)$width
plot(gScz)
```



3.1.3 Exercises

1. Calculate the LCC, and visualize the modules for the following diseases:

    - Autistic Disorder;
    - Obesity;

- Hyperlipidemia;
- Rheumatoid Arthritis.

2. Now choose any disease you are interested in and do the same thing.

3.2 Gene Overlap

A first intuitive way to measure the overlap of two gene sets is by calculating its overlap, or its normalized overlap, the **Jaccard Index**. The Jaccard index is calculated by taking the ratio of **Intersection of two sets over Union of those sets**. The Jaccard coefficient measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Note that by design, $0 \leq J(A,B) \leq 1$. If A and B are both empty, define $J(A,B) = 1$

Let's calculate the Jaccard Index for the 5 diseases we calculated its LCCs.

```
Dis_Ex1 = c('schizophrenia',
            "autistic disorder",
            'obesity',
            'hyperlipidemia',
            'rheumatoid arthritis')
GDA_Interest = Cleaned_GDA %>%
  filter(diseaseName %in% Dis_Ex1) %>%
  select(diseaseName, geneSymbol) %>%
  unique()

Jaccard_Ex2 = Jaccard(GDA_Interest)
```

```
##   |                                                                      |
```

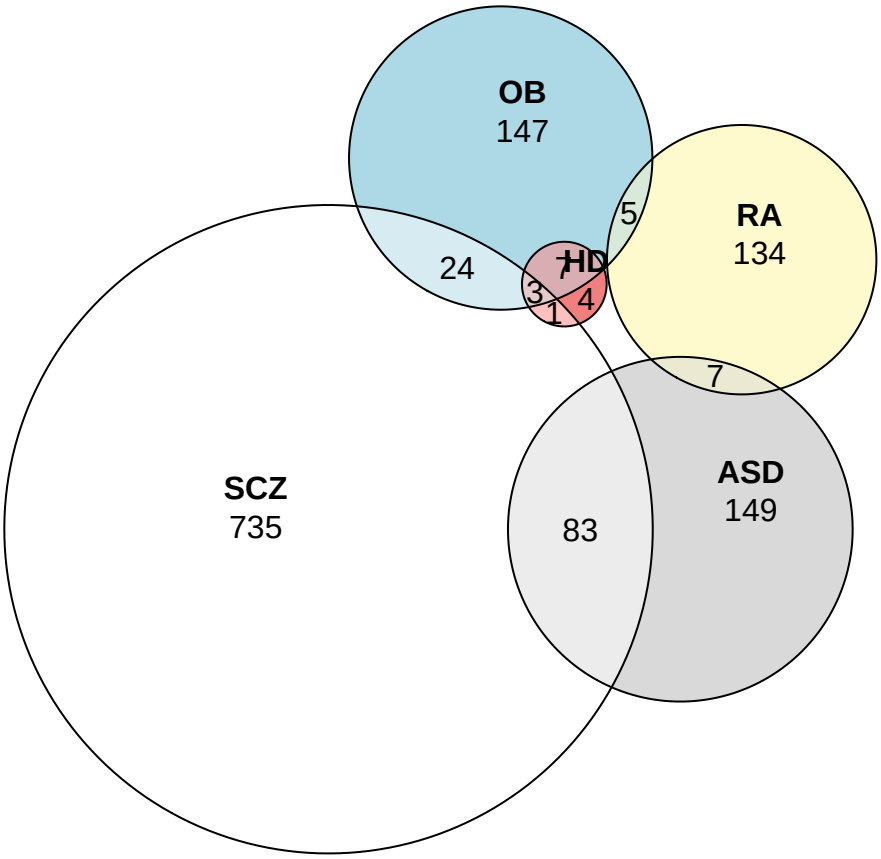
```
Jaccard_Ex2
```

```
##                   Node.1               Node.2 Jaccard.Index
##  1:        schizophrenia    autistic disorder   0.095785441
##  2:        schizophrenia              obesity   0.039159503
##  3:    autistic disorder              obesity   0.033259424
##  4:        schizophrenia rheumatoid arthritis   0.027210884
##  5:    autistic disorder rheumatoid arthritis   0.035714286
##  6:              obesity rheumatoid arthritis   0.029891304
##  7:        schizophrenia       hyperlipidemia   0.005586592
##  8:    autistic disorder       hyperlipidemia   0.007246377
##  9:              obesity       hyperlipidemia   0.052132701
## 10: rheumatoid arthritis       hyperlipidemia   0.000000000
```

```
# Let's visualize the Venn diagram (Euler Diagram) of those overlaps.

require(eulerr)
```

```
## Loading required package: eulerr
```

```
Euler_List = list (
  SCZ = GDA_Interest$geneSymbol[GDA_Interest$diseaseName == 'schizophrenia'],

  ASD = GDA_Interest$geneSymbol[GDA_Interest$diseaseName == 'autistic disorder'],

  OB = GDA_Interest$geneSymbol[GDA_Interest$diseaseName == 'obesity'],

  HD = GDA_Interest$geneSymbol[GDA_Interest$diseaseName == 'hyperlipidemia'],

  RA = GDA_Interest$geneSymbol[GDA_Interest$diseaseName == 'rheumatoid arthritis'])

EULER = euler(Euler_List)
plot(EULER, quantities = TRUE)
```



3.3 Disease Separation

When looking into the Jaccard Index, we have a sense of how similar two diseases are based on genes that are **known** to be associated to both diseases. The main problem with this is that we assume that all genes associated with a disease is known, and we do not take the topology of the underlying network into account.

The **separation** is a complementary quantity that is a bit less sensitive to the incompleteness of the PPI, we can measure the distances d_s of each disease associated node to all other disease associated nodes. Taking into account only the shortest distance among the result among them results in a distribution $P(d_s)$. The

mean value $< d_s >$ can be interpreted as the diameter of the disease model. **Note** the diameter here is the average distance instead of the maximal distance.

The **concept of network localization** can be further generalized to exam the relationship between any different sets of nodes, for example, proteins associated with two different diseases.

The network serves as a **map**, where diseases are represented by different neighborhoods.

How close and degree of overlap of two network neighborhoods can be found to be highly predictive of the pathological similarity of those diseases (Menche et al., 2015) (Figure 3.2).

To quantify the distance of two sets of nodes A and B we first compute the distribution $P(d_{AB})$ of all shortest distances d_{AB} between nodes A and B and the respective mean distance $< d_{AB} >$.

The network based separation S_{AB} can be obtained by comparing the mean shortest distance **within** the respective node sets and the mean shortest distance **between** them.

$$S_{AB} =< d_{AB} > -\frac{< d_{AA} > + < d_B B >}{2}$$

Note: negative S_{AB} indicates topological overlap of the two node sets, while a positive S_{AB} indicates topological separation of the two node sets.

The size of the overlap is highly predictive of pathological and functional similarity, elevated co-expression, symptoms similarity and high comorbidity diseases.

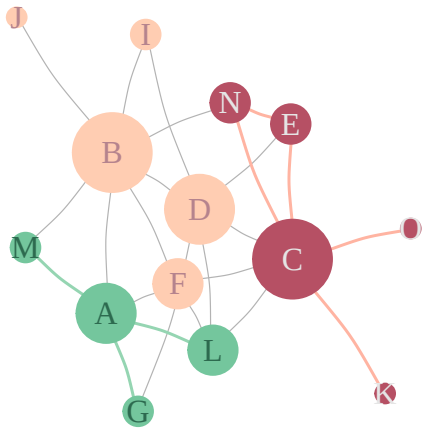


Figure 3.2: Disease-Separation. A schematic of a PPI, in pink we see genes associated with a disease A, and in green genes associated to disease B.

The separation of diseases A and B is given by:

$$< d_{AA} >= 1.5$$

$$< d_{BB} >= 1.5$$

$$< d_{AB} >= 2.7$$

$$S_{AB} = 2.7 - \frac{1.5 + 1.5}{2} = 1.2$$

3.3.1 Example in real data

```
sab = separation(gPPI, GDA_Interest)
```

```
##   |                                                                      |
```

```
## Calculating S_ab..
```

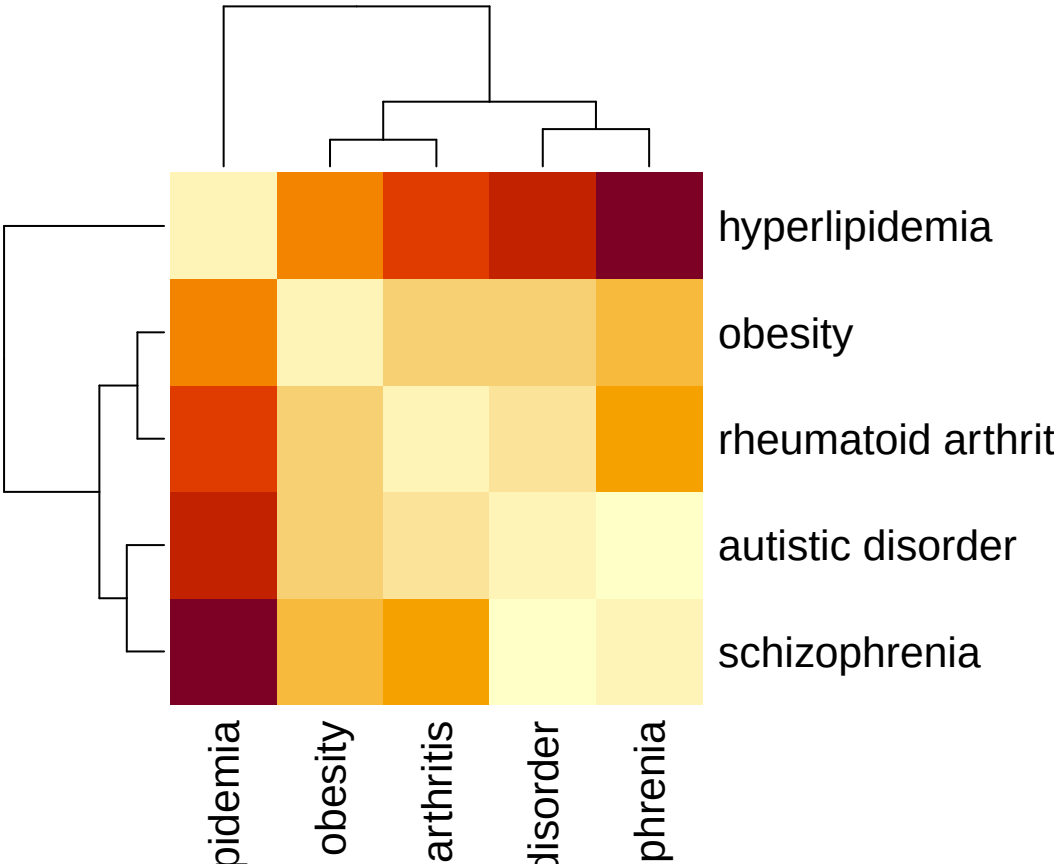
```
##   |                                                                      |
```

```
## Done..
```

```
Sep_ex2 = sab$Sab %>% as.matrix()

Sep_ex2[lower.tri(Sep_ex2)] = t(Sep_ex2)[lower.tri(Sep_ex2)]
```

We can visualize the network separation of the diseases using a heatmap.

```
Sep_ex2 %>% heatmap(., symm = T)
```



3.4 Exercises

1. If we go back to our PPI, can we identify that the modules are indeed close or separated? Plot the network for those diseases.
2. Calculate the **Jaccard Index** and the **Separation** for the following diseases:
   - Schizophrenia, Bipolar Disorder, Intellectual Disability, Depressive disorder, Autistic Disorder, Unipolar Depression, Mental Depression, Major Depressive Disorder, Mood Disorders, Cocaine Dependence, Cocaine Abuse, Cocaine-Related Disorders, Substance abuse problem, Drug abuse, Drug Dependence, Drug habituation, Drug Use Disorders, Substance-Related Disorders, Psychotic Disorders, Obesity, hyperlipidemia, Rheumatoid Arthritis, Prostatic Neoplasms, Mammary Neoplasms, Mammary Neoplasms, Human Malignant neoplasm of stomach, Stomach Neoplasms, Colorectal Neoplasms, Malignant neoplasm of lung, Lung Neoplasms, Malignant neoplasm of prostate.
3. Optional: Try to make the network visualization for the heatmap of `Sep_ex2`. Use diseases as nodes, and their weight as links.
4. Optional: Plot the PPI with genes selected in `GDA_Interest` where each node is a piechart representing which diseases are associated to that particular gene. Tip: Check `vertex.shape.pie` for help.

Chapter 4

Method for drug-repurpusing

In this Chapter we will learn how to calculate the proximity of a drug to a disease - and infer drug repurpusing (Session 4.1)- based on network methodologies.

There are different methods that are used for drug-repurpusing based on networks, such as the **diffusion state distance (DSD)** (Cao et al., 2013), that uses a **graph diffusion property** to derive a similarity metric for pairs of nodes that takes into account how similarly they affect the rest of the network; and **AI-based methods**, where a heterogeneous graph $G = (V, R)$ with N nodes $v_i \in V$ representing distinct types of biomedical entities and labeled edges representing semantically distinct types of edges between the entities (i.e., protein-protein interactions, drug-target associations, disease-protein associations, and drug-disease indications) and are tasked to predict drugs for a particular disease (Zitnik et al., 2018). Due to the limited time, we will focus only on the proximity based method.

For this, we will be using the `R` package `NetSci` and to make the appropriate visualizations we will use `igraph`.

4.1 Proximity

Given G, the set of Disease-Genes, T, the set of drug targets, and d(g,t), the shortest path length between nodes $g \in G$ and $t \in T$ in the network, the proximity can be defined as (Guney et al., 2016):

$$d(g,t) = \frac{1}{||T||} \sum_{t \in T} \min_{v \in V} d(g,t)$$

A visual representation of the method can be seen in Figure 4.1.

The proximity for drug 2 to the disease is calculated by the average of the shortest path from its targets to the disease genes. The shortest path from N to D is 1, from F to D is 3, the average is 2.

For Drug 1 we have:

$$d(Drug_1, disease) = \frac{2+2+1}{3} = 1.66$$

Similarly to the LCC (3.1) it is important to calculate a measure of randomness associate to the proximity. In the same sense, it is important that the nodes being randomized, they are not simply randomly selected from the pool of proteins in the PPI, but rather selected from matching degree proteins. To calculate the significance of the proximity one can calculate its Z-Score or simply calculate the empirical probability under the curve from the empirical distribution. Similartly as before, the Z-score is given by:

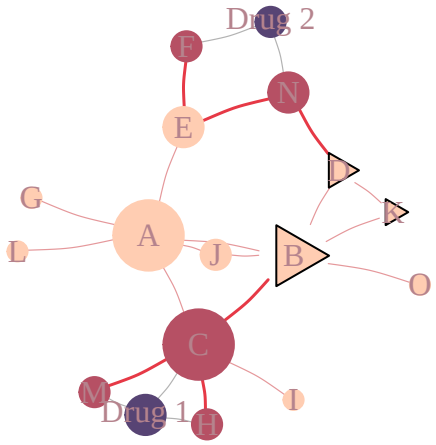


Figure 4.1: Drug-Target & Disease-Module Proximity. Triangles represents Disease Associated Genes, while circles represent non-associated genes. In dark purple, we see the drugs and light purple, its targets.

$$Z - Score_{d(g,t)} = \frac{d(g,t) - \mu_{d(g,t)}}{\sigma_{d(g,t)}}$$

4.2 Example in real data

Let's try it to identify drugs that could work for our disease sets. Let's focus on hyperlipidemia and focus on 5 drugs at first.

- Asenapine,
- Phentermine,
- Simvastatin,
- Pizotifen,
- Eprotirome.

```
hyperlipidemia_genes = Cleaned_GDA %>% filter(diseaseName == 'hyperlipidemia') %>% pull(geneSymbol) %>%

Asenapine_t = DT %>%
  filter(Name == 'Asenapine') %>%
  pull(Gene_Target)

Asenapine_t
```

```
##  [1] "HTR1A"  "HTR1B"  "HTR2A"  "HTR2B"  "HTR2C"  "HTR5A"  "HTR6"   "HTR7"
##  [9] "DRD2"   "DRD3"   "DRD4"   "DRD1"   "ADRA1A" "ADRA2A" "ADRA2B" "ADRA2C"
## [17] "HRH1"   "HRH2"   "ADRB1"  "ADRB2"  "UGT1A4" "CYP1A2" "CYP2D6" "CYP3A4"
## [25] "ALB"    "ORM1"
```

```
proximity_average(gPPI,
                  source = hyperlipidemia_genes,
                  targets = Asenapine_t)
```

```
## [1] 1.961538
```

Let's do it in a loop:

```
drugs = c("Asenapine",
          'Phentermine',
          'Simvastatin',
          'Pizotifen',
          'Eprotirome')

p = list()
for(i in 1:length(drugs)){
  d = drugs[i]
  Drug_targets = DT %>%
    filter(Name %in% d) %>%
    pull(Gene_Target)

  prox = proximity_average(gPPI,
                           source = hyperlipidemia_genes,
                           targets = Drug_targets)

  p[[i]] = data.frame(prox = prox,
                      ntargets = length(Drug_targets),
                      drug = d)
}

p %<>% bind_rows()
```

Now, let's do the same, but also calculating the significance of the proximity.

```
Drug_Target = DT %>%
  filter(Name %in% drugs) %>%
  select(Name, Gene_Target) %>%
  unique()

names(Drug_Target) = c('ID', "Target" )

proximity_significance = avr_proximity_multiple_target_sets(
  set = drugs,
  G = gPPI,
  ST = Drug_Target,
  source = hyperlipidemia_genes,
  N = 10,
  bins = 100,
  min_per_bin = 20
)
```

```
##   |                                                                      |
```

Which are the drugs that we can use for hyperlipidemia?

```
proximity_significance
```

```
##                   Drug targets targets_G proximity  IC_97.5   IC_2.5 p_gt p_lt
```

```
## Asenapine     Asenapine      26        26  1.961538 2.251923 1.978846  0.9  0.1
## Phentermine Phentermine       7         7  2.000000 2.792857 2.142857  1.0  0.0
## Simvastatin Simvastatin      20        19  2.157895 2.619737 2.327632  1.0  0.0
## Pizotifen     Pizotifen      17        17  2.058824 2.294118 1.876471  0.7  0.3
## Eprotirome   Eprotirome       2         2  1.500000 2.000000 2.000000  1.0  0.0
##                       Z
## Asenapine   -1.7084389
## Phentermine -1.8973666
## Simvastatin -2.8460499
## Pizotifen   -0.5217939
## Eprotirome        -Inf
```

Now, let us check those drug indications:

```
Indication = DT %>%
  filter(Name %in% drugs) %>%
  select(Name, Indication) %>%
  unique()

Indication
```

```
##           Name
## 1: Phentermine
## 2: Simvastatin
## 3:  Eprotirome
## 4:   Pizotifen
## 5:   Asenapine
##
## 1:
## 2: Simvastatin is indicated for the treatment of hyperlipidemia to reduce elevated total cholesterol
## 3:
## 4:
## 5:
```

4.3 Exercises

1. Test the same drugs for all the 5 other diseases we are interested in. How those values compare?

    - Autistic Disorder;
    - Obesity;
    - Hyperlipidemia;
    - Rheumatoid Arthritis.

2. Choose one disease and visualize the disease module along with each of the drugs we tested.

Chapter 5

Summary

In this course we learned how to identify disease modules, disease separation and how to repurpuse drugs using a network medicine approach.

Bibliography

Ahn, Y. Y., Bagrow, J. P., and Lehmann, S. (2010). Link communities reveal multiscale complexity in networks. *Nature*.

Alanis-Lobato, G., Andrade-Navarro, M. A., and Schaefer, M. H. (2017). HIPPIE v2.0: enhancing meaningfulness and reliability of protein-protein interaction networks. *Nucleic acids research*, 45(D1):D408–D414.

Alonso-López, D., Campos-Laborie, F. J., Gutiérrez, M. A., Lambourne, L., Calderwood, M. A., Vidal, M., and De Las Rivas, J. (2019). APID database: Redefining protein-protein interaction experimental evidences and binary interactomes. *Database*, 2019(i):1–8.

Barabási, A.-L. (2016). *Network Science*. Cambridge University Press, Cambridge, UK, 1 edition.

Barabási, A. L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.

Barabási, A. L. and Oltvai, Z. N. (2004). Network biology: Understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113.

Barrat, A., Barthelemy, M., Pastor-Satorras, R., and Vespignani, A. (2003). The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences*, 101(11):3747–3752.

Bondy, J. A. and Murty, U. S. R. (2008). *Graph Theory*. Springer.

Borgatti, S. P. and Everett, M. G. (2006). A Graph-theoretic perspective on centrality. *Social Networks*, 28(4):466–484.

Breuer, K., Foroushani, A. K., Laird, M. R., Chen, C., Sribnaia, A., Lo, R., Winsor, G. L., Hancock, R. E. W., Brinkman, F. S. L., and Lynn, D. J. (2013). InnateDB: systems biology of innate immunity and beyond–recent updates and continuing curation. *Nucleic acids research*, 41(Database issue):D1228–33.

Buniello, A., MacArthur, J., Cerezo, M., Harris, L., Hayhurst, J., Malangone, C., McMahon, A., Morales, J., Mountjoy, E., Sollis, E., Suveges, D., Vrousgou, O., Whetzel, P., Amode, R., Guillen, J., Riat, H., Trevanion, S., Hall, P., Junkins, H., Flicek, P., Burdett, T., Hindorff, L., Cunningham, F., and Parkinson, H. (2018). The nhgri-ebi gwas catalog of published genome-wide association studies, targeted arrays and summary statistics 2019. *Nucleic Acids Research*, 47(D1):D1005–D1012.

Cao, M., Zhang, H., Park, J., Daniels, N. M., Crovella, M. E., Cowen, L. J., and Hescott, B. (2013). Going the Distance for Protein Function Prediction: A New Distance Metric for Protein Interaction Networks. *PLoS ONE*.

Chatr-Aryamontri, A., Oughtred, R., Boucher, L., Rust, J., Chang, C., Kolas, N. K., O'Donnell, L., Oster, S., Theesfeld, C., Sellam, A., Stark, C., Breitkreutz, B.-J., Dolinski, K., and Tyers, M. (2017). The BioGRID interaction database: 2017 update. *Nucleic acids research*, 45(D1):D369–D379.

Chen, G., Wang, Z., Wang, D., Qiu, C., Liu, M., Chen, X., Zhang, Q., Yan, G., and Cui, Q. (2012). Lncrnadisease: a database for long-non-coding rna-associated diseases. *Nucleic Acids Research*, 41(D1):D983–D986.

Cheng, F., Jia, P., Wang, Q., and Zhao, Z. (2014). Quantitative network mapping of the human kinome interactome reveals new clues for rational kinase inhibitor discovery and individualized cancer therapy. *Oncotarget*, 5(11):3697–3710.

Corsello, S., Bittker, J., Liu, Z., Gould, J., McCarren, P., Hirschman, J., Johnston, S., Vrcic, A., Wong, B., Khan, M., Asiedu, J., Narayan, R., Mader, C., Subramanian, A., and Golub, T. (2017). The drug repurposing hub: a next-generation drug library and information resource. *Nature Medicine*, 23(4):405–408.

Cowley, M. J., Pinese, M., Kassahn, K. S., Waddell, N., Pearson, J. V., Grimmond, S. M., Biankin, A. V., Hautaniemi, S., and Wu, J. (2012). PINA v2.0: mining interactome modules. *Nucleic acids research*, 40(Database issue):D862–5.

Das, J. and Yu, H. (2012). HINT: High-quality protein interactomes and their applications in understanding human disease. *BMC Systems Biology*, 6.

Davis, A., Grondin, C., Johnson, R., Sciaky, D., Wiegers, J., Wiegers, T., and Mattingly, C. (2020). Comparative toxicogenomics database (ctd): update 2021. *Nucleic Acids Research*, 49(D1):D1138–D1143.

De Domenico, M. (2017). Multilayer modeling and analysis of human brain networks. *GigaScience*, 6(5).

Fazekas, D., Koltai, M., Türei, D., Módos, D., Pálfy, M., Dúl, Z., Zsákai, L., Szalay-Bekő, M., Lenti, K., Farkas, I. J., Vellai, T., Csermely, P., and Korcsmáros, T. (2013). SignaLink 2 - a signaling pathway resource with multi-layered regulatory networks. *BMC systems biology*, 7:7.

Fong, J., Keating, A., and Singh, M. (2004). *Genome Biology*, 5(2):R11.

Gashaw, I., Ellinghaus, P., Sommer, A., and Asadullah, K. (2011). What makes a good drug target?

Guney, E., Menche, J., Vidal, M., and Barábasi, A.-L. L. (2016). Network-based in silico drug efficacy screening. *Nature Communications*, 7(1):10331.

Gysi, D. M., Do Valle Zitnik, M., Í., Ameli, A., Gan, X., Varol, O., Sanchez, H., Baron, R. M., Ghiassian, D., Loscalzo, J., and Barabási, A. L. (2020). Network Medicine Framework for Identifying Drug Repurposing Opportunities for COVID-19. *ArXiv*.

Gysi, D. M. and Nowick, K. (2020). Construction, comparison and evolution of networks in life sciences and other disciplines. *Journal of the Royal Society, Interface*, 17(166):20190610.

Hein, M. Y., Hubner, N. C., Poser, I., Cox, J., Nagaraj, N., Toyoda, Y., Gak, I. A., Weisswange, I., Mansfeld, J., Buchholz, F., Hyman, A. A., and Mann, M. (2015). A Human Interactome in Three Quantitative Dimensions Organized by Stoichiometries and Abundances. *Cell*, 163(3):712–723.

Hobert, O. (2008). Gene regulation by transcription factors and MicroRNAs.

Hornbeck, P. V., Zhang, B., Murray, B., Kornhauser, J. M., Latham, V., and Skrzypek, E. (2015). PhosphoSitePlus, 2014: mutations, PTMs and recalibrations. *Nucleic acids research*, 43(Database issue):D512–20.

Huang, Z., Shi, J., Gao, Y., Cui, C., Zhang, S., Li, J., Zhou, Y., and Cui, Q. (2018). Hmdd v3.0: a database for experimentally supported human microrna–disease associations. *Nucleic Acids Research*, 47(D1):D1013–D1017.

Huttlin, E. L., Bruckner, R. J., Paulo, J. A., Cannon, J. R., Ting, L., Baltier, K., Colby, G., Gebreab, F., Gygi, M. P., Parzen, H., Szpyt, J., Tam, S., Zarraga, G., Pontano-Vaites, L., Swarup, S., White, A. E., Schweppe, D. K., Rad, R., Erickson, B. K., Obar, R. A., Guruharsha, K. G., Li, K., Artavanis-Tsakonas, S., Gygi, S. P., and Wade Harper, J. (2017). Architecture of the human interactome defines protein communities and disease networks. *Nature*, 545(7655):505–509.

Kempa, E. E., Hollywood, K. A., Smith, C. A., and Barran, P. E. (2019). High throughput screening of complex biological samples with mass spectrometry-from bulk measurements to single cell analysis. *Analyst*, 144(3):872–891.

Kivela, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., and Porter, M. A. (2014). Multilayer networks. *Journal of Complex Networks*, 2(3):203–271.

Koh, G., Porras, P., Aranda, B., Hermjakob, H., and Orchard, S. (2012). Analyzing protein–protein interaction networks. *Journal of Proteome Research*, 11(4):2014–2031.

Kurant, M. and Thiran, P. (2006). Layered Complex Networks. *Physical Review Letters*, 96(13):138701.

Landrum, M., Chitipiralla, S., Brown, G., Chen, C., Gu, B., Hart, J., Hoffman, D., Jang, W., Kaur, K., Liu, C., Lyoshin, V., Maddipatla, Z., Maiti, R., Mitchell, J., O'Leary, N., Riley, G., Shi, W., Zhou, G., Schneider, V., Maglott, D., Holmes, J., and Kattman, B. (2019). Clinvar: improvements to accessing data. *Nucleic Acids Research*, 48(D1):D835–D844.

Latchman, D. S. (1997). Transcription factors: An overview. *International Journal of Biochemistry and Cell Biology*, 29(12):1305–1312.

Li, A. and Horvath, S. (2009). Network module detection: Affinity search technique with the multi-node topological overlap measure. *BMC Research Notes*, 2(1):142.

Li, T., Wernersson, R., Hansen, R., Horn, H., Mercer, J., Slodkowicz, G., Workman, C., Rigina, O., Rapacki, K., Stærfeldt, H., Brunak, S., Jensen, T., and Lage, K. (2016). A scored human protein–protein interaction network to catalyze genomic interpretation. *Nature Methods*, 14(1):61–64.

Licata, L., Briganti, L., Peluso, D., Perfetto, L., Iannuccelli, M., Galeota, E., Sacco, F., Palma, A., Nardozza, A. P., Santonico, E., Castagnoli, L., and Cesareni, G. (2012). MINT, the molecular interaction database: 2012 update. *Nucleic Acids Research*, 40(D1):D857–D861.

Luck, K., Kim, D. K., Lambourne, L., Spirohn, K., Begg, B. E., Bian, W., Brignall, R., Cafarelli, T., Campos-Laborie, F. J., Charloteaux, B., Choi, D., Coté, A. G., Daley, M., Deimling, S., Desbuleux, A., Dricot, A., Gebbia, M., Hardy, M. F., Kishore, N., Knapp, J. J., Kovács, I. A., Lemmens, I., Mee, M. W., Mellor, J. C., Pollis, C., Pons, C., Richardson, A. D., Schlabach, S., Teeking, B., Yadav, A., Babor, M., Balcha, D., Basha, O., Bowman-Colin, C., Chin, S. F., Choi, S. G., Colabella, C., Coppin, G., D'Amata, C., De Ridder, D., De Rouck, S., Duran-Frigola, M., Ennajdaoui, H., Goebels, F., Goehring, L., Gopal, A., Haddad, G., Hatchi, E., Helmy, M., Jacob, Y., Kassa, Y., Landini, S., Li, R., van Lieshout, N., MacWilliams, A., Markey, D., Paulson, J. N., Rangarajan, S., Rasla, J., Rayhan, A., Rolland, T., San-Miguel, A., Shen, Y., Sheykhkarimli, D., Sheynkman, G. M., Simonovsky, E., Taşan, M., Tejeda, A., Tropepe, V., Twizere, J. C., Wang, Y., Weatheritt, R. J., Weile, J., Xia, Y., Yang, X., Yeger-Lotem, E., Zhong, Q., Aloy, P., Bader, G. D., De Las Rivas, J., Gaudet, S., Hao, T., Rak, J., Tavernier, J., Hill, D. E., Vidal, M., Roth, F. P., and Calderwood, M. A. (2020). A reference map of the human binary protein interactome. *Nature*.

Mattick, J. S. and Makunin, I. V. (2006). Non-coding RNA. *Human Molecular Genetics*, 15(suppl_1):R17–R29.

McKusick, V. (2007). Mendelian inheritance in man and its online version, omim. *The American Journal of Human Genetics*, 80(4):588–604.

Menche, J., Sharma, A., Kitsak, M., Ghiassian, S. D., Vidal, M., Loscalzo, J., and Barabasi, A.-L. (2015). Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347(6224).

Metzker, M. L. (2010). Sequencing technologies the next generation.

Meyer, M. J., Beltrán, J. F., Liang, S., Fragoza, R., Rumack, A., Liang, J., Wei, X., and Yu, H. (2018). Interactome INSIDER: a structural interactome browser for genomic studies. *Nature methods*, 15(2):107–114.

Meyer, M. J., Das, J., Wang, X., and Yu, H. (2013). INstruct: a database of high-quality 3D structurally resolved protein interactome networks. *Bioinformatics (Oxford, England)*, 29(12):1577–9.

Mosca, R., Céol, A., and Aloy, P. (2013). Interactome3D: adding structural details to protein networks. *Nature methods*, 10(1):47–53.

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., and Onnela, J.-P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *Science (New York, N.Y.)*, 328(5980):876–8.

Newman, M. E. J. (2018). *Networks*.

Park, P. J. (2009). ChIP-seq: Advantages and challenges of a maturing technology.

Parrish, J. R., Gulyas, K. D., and Finley, R. L. (2006). Yeast two-hybrid contributions to interactome mapping.

Piñero, J., Ramírez-Anguita, J., Saüch-Pitarch, J., Ronzano, F., Centeno, E., Sanz, F., and Furlong, L. (2019). The disgenet knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research*.

Radicchi, F. and Arenas, A. (2013). Abrupt transition in the structural formation of interconnected networks. *Nature Physics*, 9(11):717–720.

Ramos, E., Hoffman, D., Junkins, H., Maglott, D., Phan, L., Sherry, S., Feolo, M., and Hindorff, L. (2013). Phenotype–genotype integrator (phegeni): synthesizing genome-wide association study (gwas) data with existing genomic resources. *European Journal of Human Genetics*, 22(1):144–147.

Rehm, H., Berg, J., Brooks, L., Bustamante, C., Evans, J., Landrum, M., Ledbetter, D., Maglott, D., Martin, C., Nussbaum, R., Plon, S., Ramos, E., Sherry, S., and Watson, M. (2015). Clingen — the clinical genome resource. *New England Journal of Medicine*, 372(23):2235–2242.

Rual, J., Venkatesan, K., Hao, T., Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G., Gibbons, F., Dreze, M., Ayivi-Guedehoussou, N., Klitgord, N., Simon, C., Boxem, M., Milstein, S., Rosenberg, J., Goldberg, D., Zhang, L., Wong, S., Franklin, G., Li, S., Albala, J., Lim, J., Fraughton, C., Llamosas, E., Cevik, S., Bex, C., Lamesch, P., Sikorski, R., Vandenhaute, J., Zoghbi, H., Smolyar, A., Bosak, S., Sequerra, R., Doucette-Stamm, L., Cusick, M., Hill, D., Roth, F., and Vidal, M. (2005). Towards a proteome-scale map of the human protein–protein interaction network. *Nature*, 437(7062):1173–1178.

Slack, J. M. (2013). Molecular Biology of the Cell. In *Principles of Tissue Engineering: Fourth Edition*.

Tanase, C. P., Ogrezeanu, I., and Badiu, C. (2012). MicroRNAs. In *Molecular Pathology of Pituitary Adenomas*, pages 91–96. Elsevier.

Uetz, P., Giot, L., Cagney, G., Mansfield, T., Judson, R., Knight, J., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M., Fields, S., and Rothberg, J. (2000). A comprehensive analysis of protein–protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–627.

Vázquez, A. (2003). Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Physical Review E*, 67(5):056104.

Wan, C., Borgeson, B., Phanse, S., Tu, F., Drew, K., Clark, G., Xiong, X., Kagan, O., Kwan, J., Bezginov, A., Chessman, K., Pal, S., Cromar, G., Papoulas, O., Ni, Z., Boutz, D., Stoilova, S., Havugimana, P., Guo, X., Malty, R., Sarov, M., Greenblatt, J., Babu, M., Derry, W., R. Tillier, E., Wallingford, J., Parkinson, J., Marcotte, E., and Emili, A. (2015). Panorama of ancient metazoan macromolecular complexes. *Nature*, 525(7569):339–344.

Wishart, D., Feunang, Y., Guo, A., Lo, E., Marcu, A., Grant, J., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., Pon, A., Knox, C., and Wilson, M. (2017). Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic Acids Research*, 46(D1):D1074–D1082.

Wishart, D. S., Knox, C., Guo, A. C., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z., and Woolsey, J. (2006). DrugBank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic acids research*, 34(Database issue):D668–72.

Zhang, B., Park, B., Karpinets, T., and Samatova, N. (2008). From pull-down data to protein interaction networks and complexes with biological relevance. *Bioinformatics*, 24(7):979–986.

Zitnik, M., Agrawal, M., and Leskovec, J. (2018). Modeling polypharmacy side effects with graph convolutional networks. In *Bioinformatics*.