Introduction to biological network analysis

Rui Benfeitas
NBIS - National Bioinformatics Infrastructure Sweden
Science for Life Laboratory, Stockholm
Stockholm University

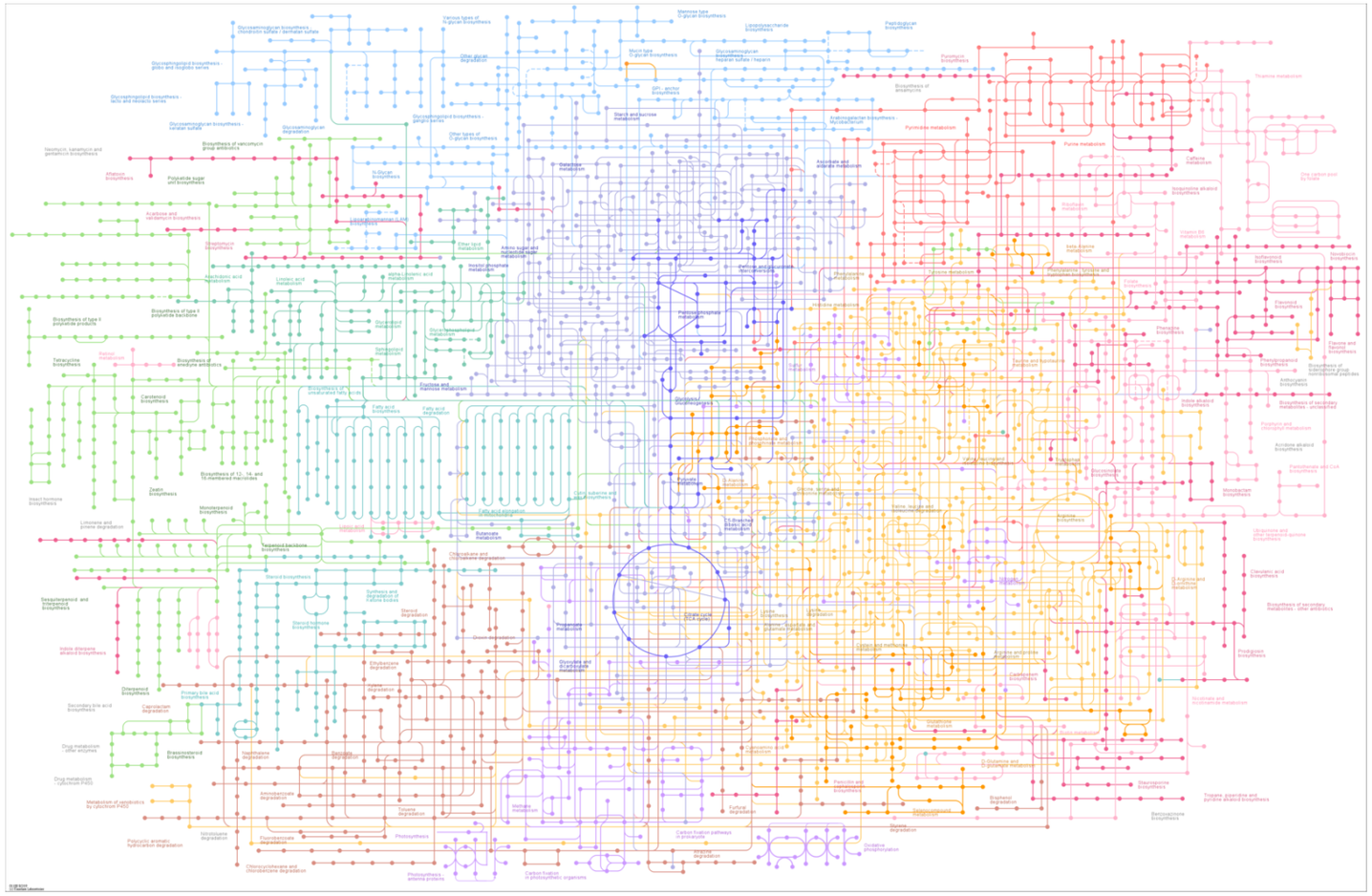
rui.benfeitas@scilifelab.se

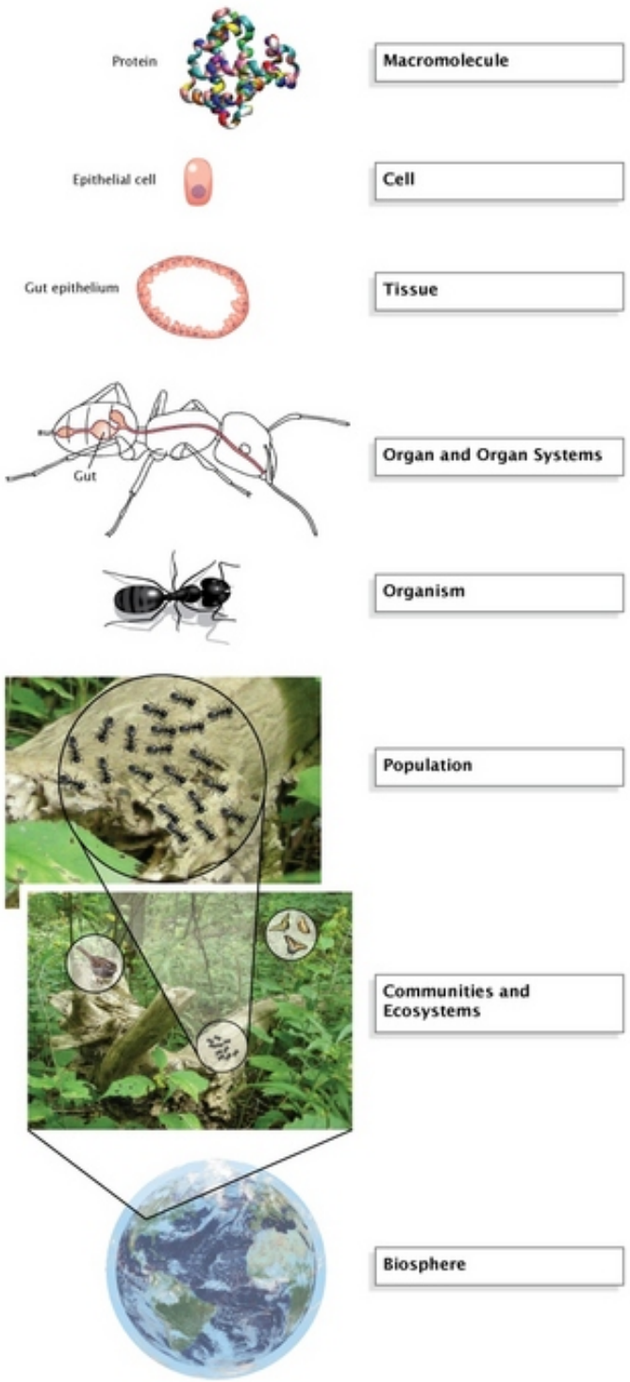
Overview

1. Introduction to network analysis
2. Terminology
3. Network construction
4. Key network properties
5. Community analysis
6. Visualization
7. Workshop

Introduction

Biological complexity under attack

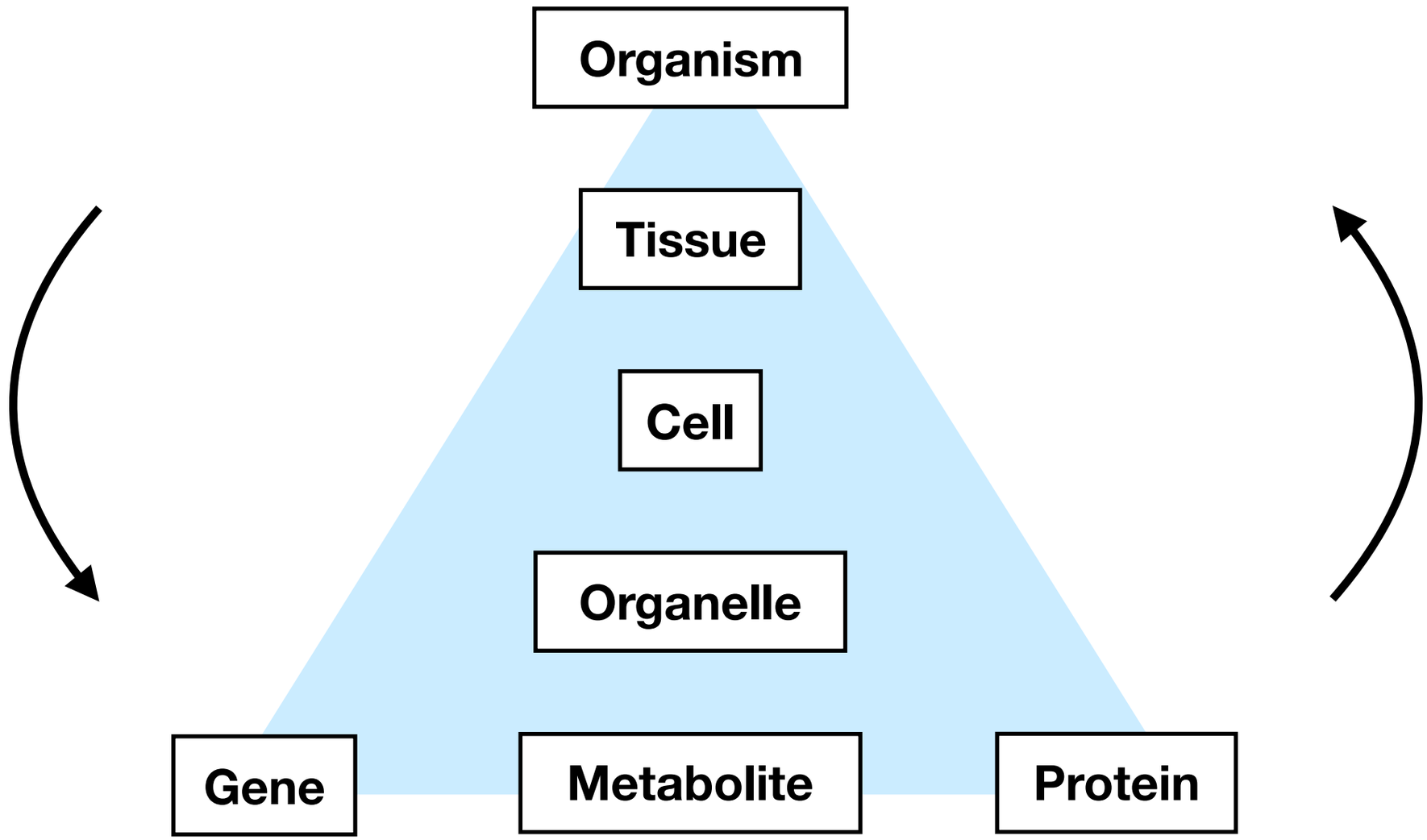




Lobo 2008
genome.jp/kegg/

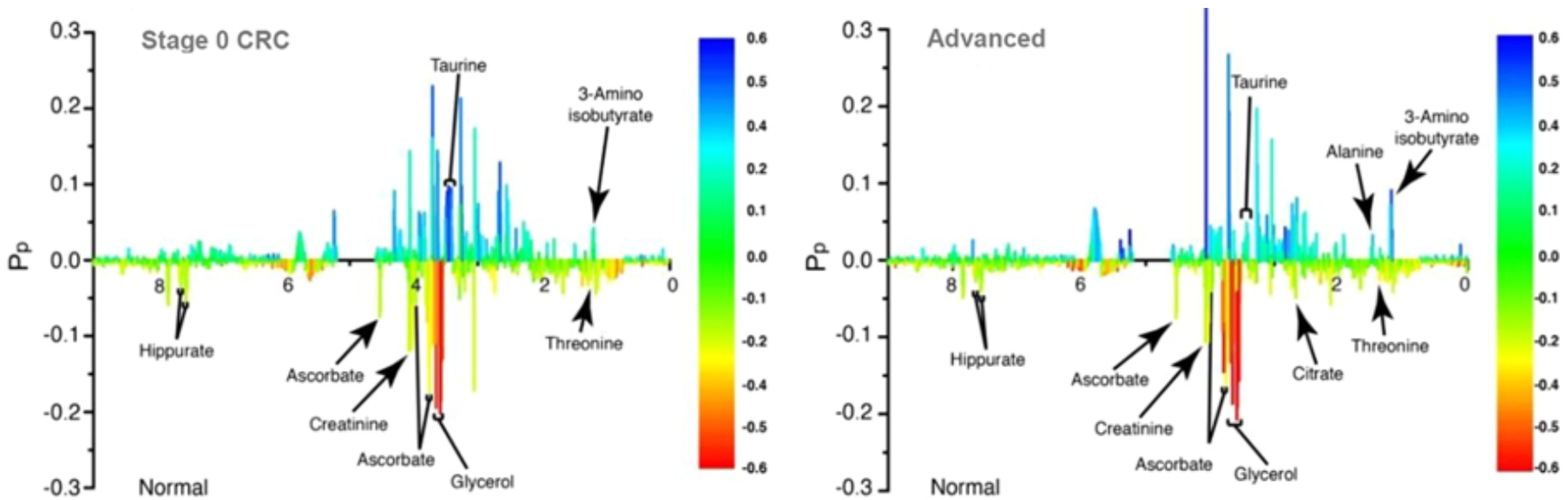
How to tackle biological complexity?

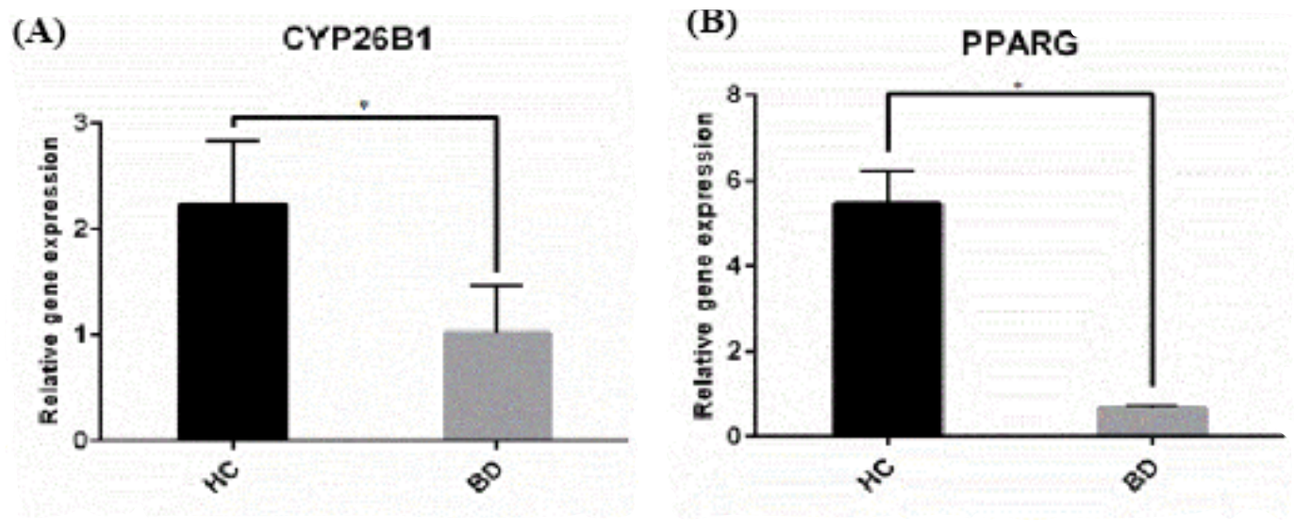
Moving from reductionist approaches towards global characterisations



How to tackle biological complexity?

Moving beyond traditional statistical analysis

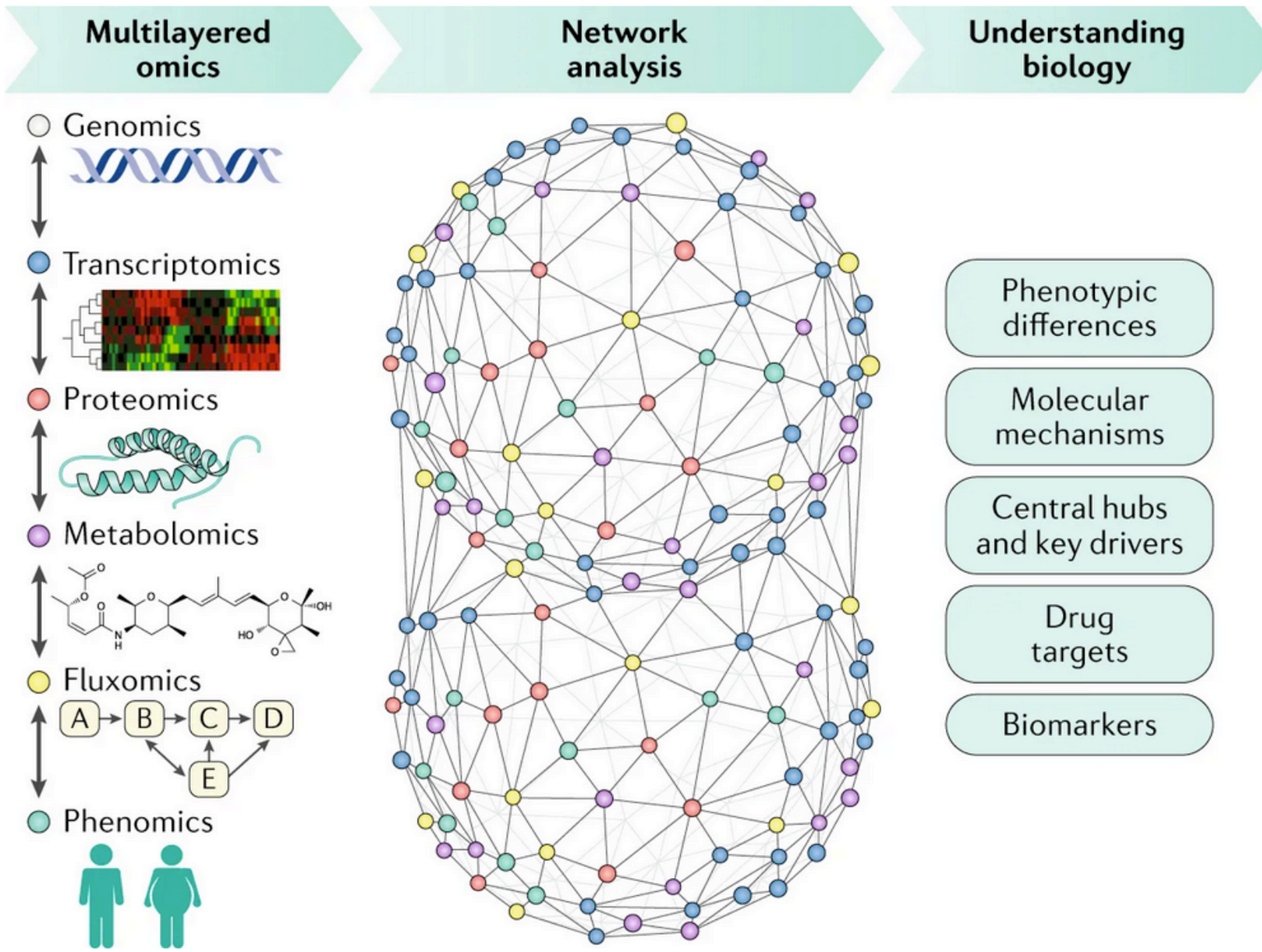




How to tackle biological complexity?

Integrative approaches, and global patterns

- Feature association
- Modeling
- Network analysis



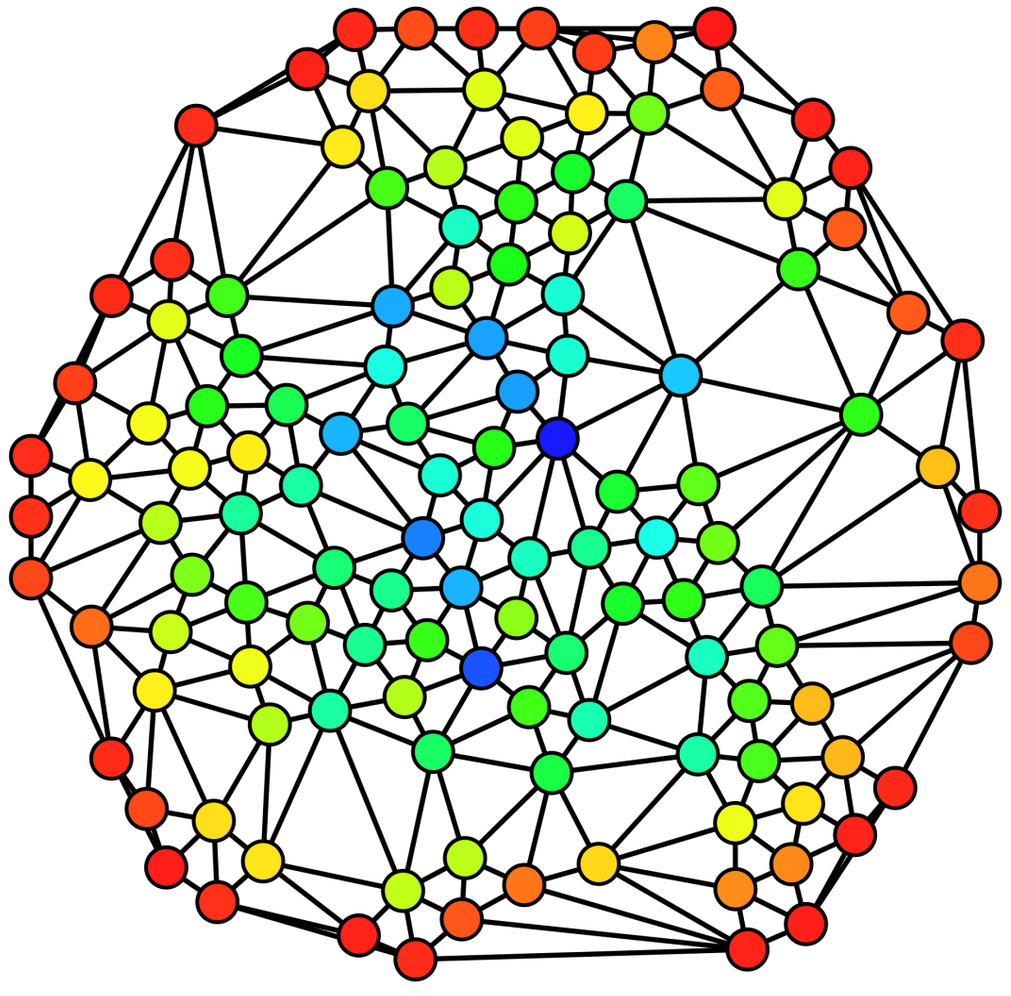
What are networks?

Networks are representations of complex systems

Permit defining and studying global properties of interacting components

Give us insight not easily achieved by other approaches:

- Comprehensive
- Coordinated



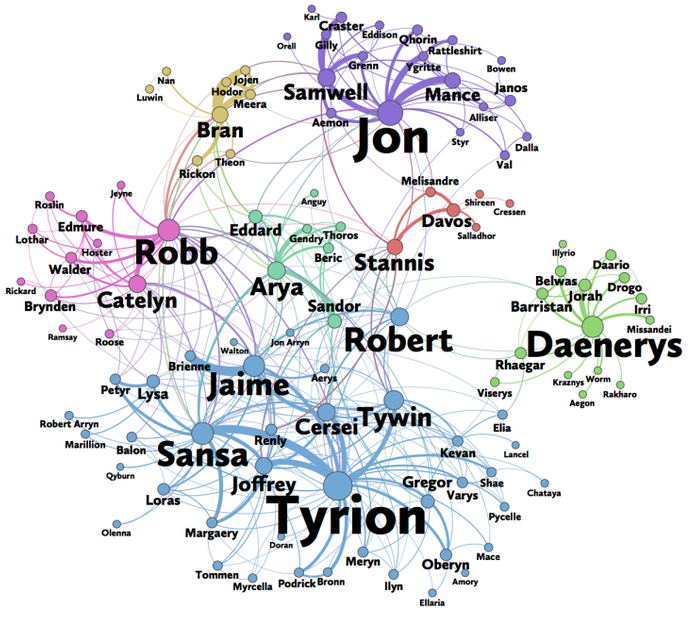
What are networks?

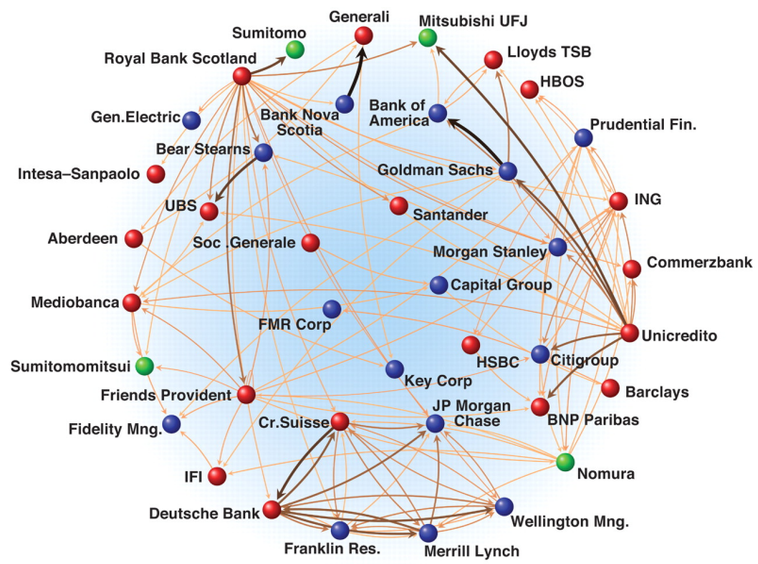
Social

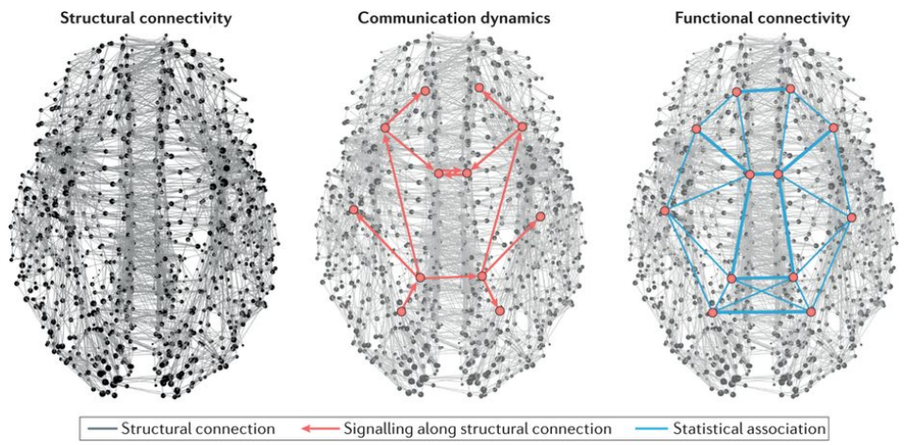
Economic

Communication

Neuronal







What are biological networks?

Protein - Protein interaction (PPI) networks

Transcription-factor regulatory networks

Gene - gene co-expression networks

Signal transduction networks

Transcription-Factor Regulatory networks

Drug-disease association networks

Aim
Functional characterisations

What are biological networks?

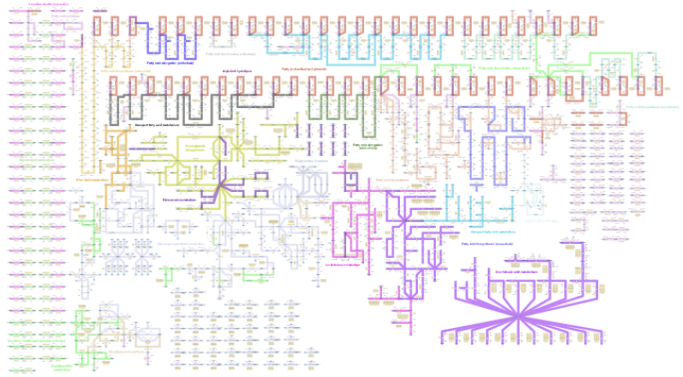
Metabolite - Enzyme - Signal - Genes (GEMs)

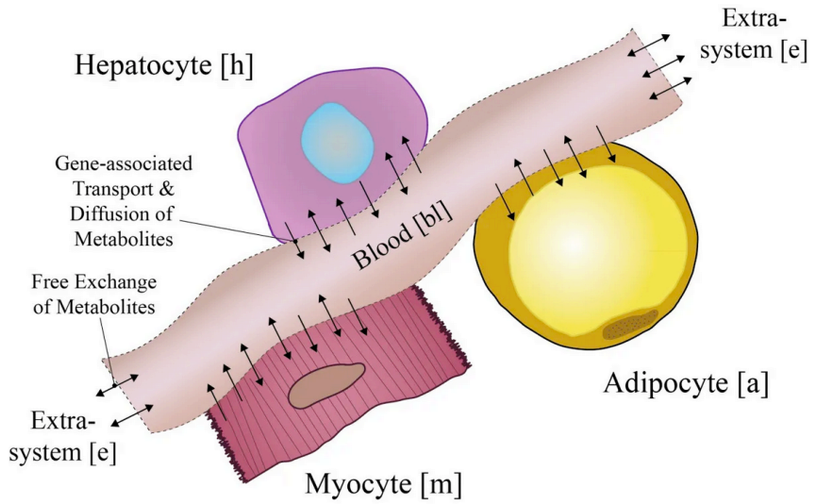
Multi-tissue networks

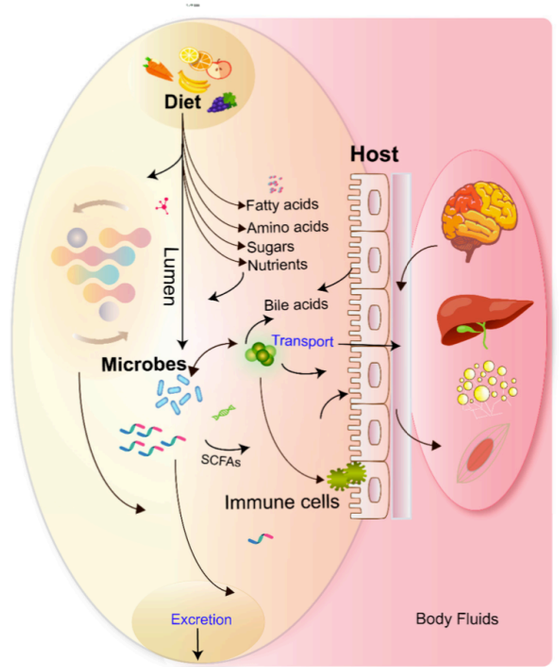
Multi-species networks

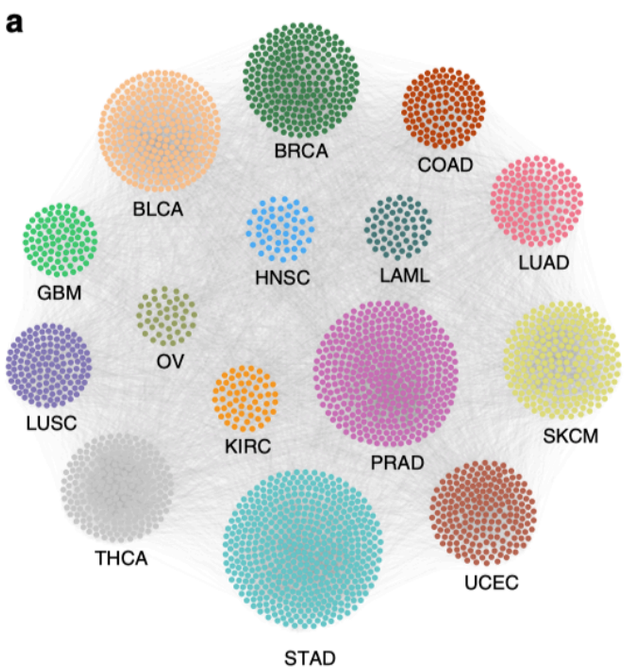
Cancer-disease networks

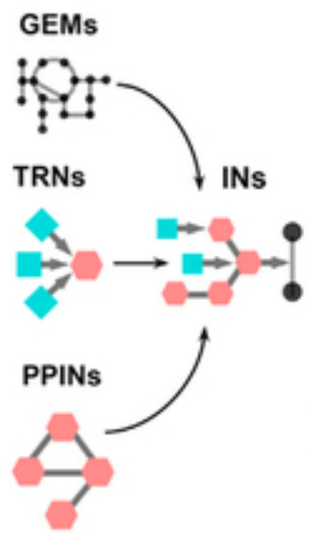
Integrated networks





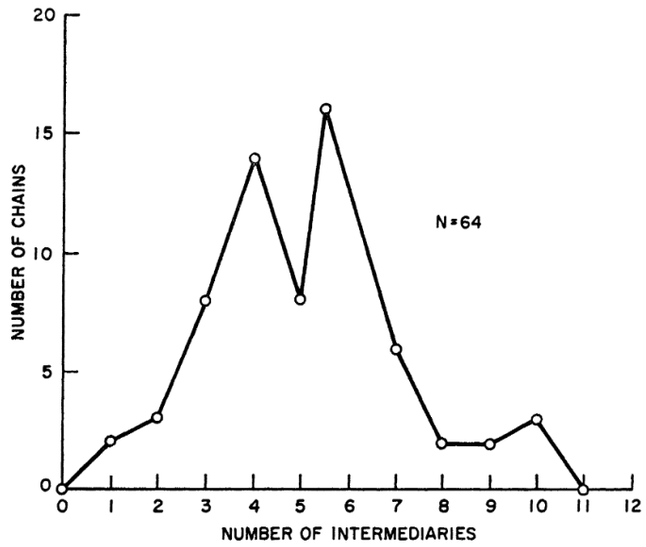




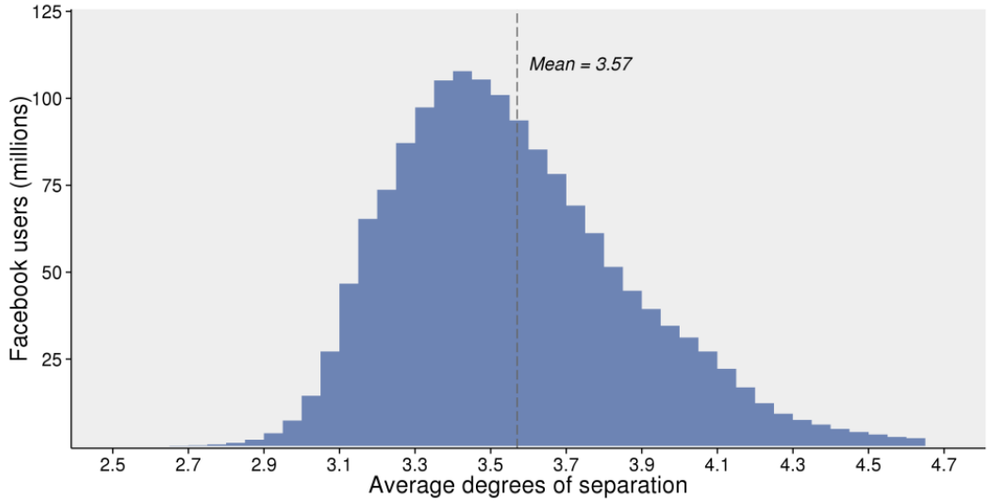


Small world

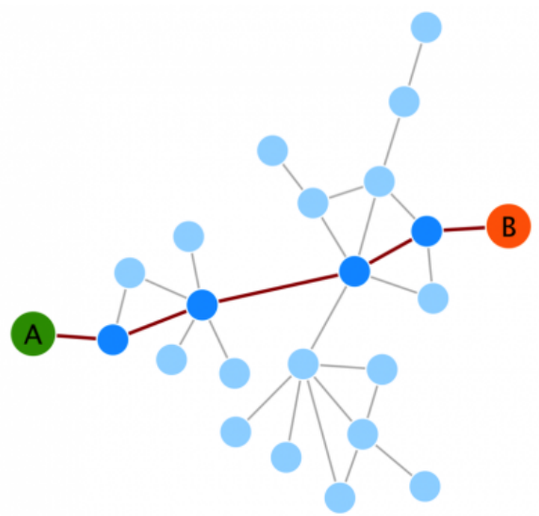
Stanley Milgram (1967) - 6 degrees



Backstrom et al. (2016) - 3.6 degrees



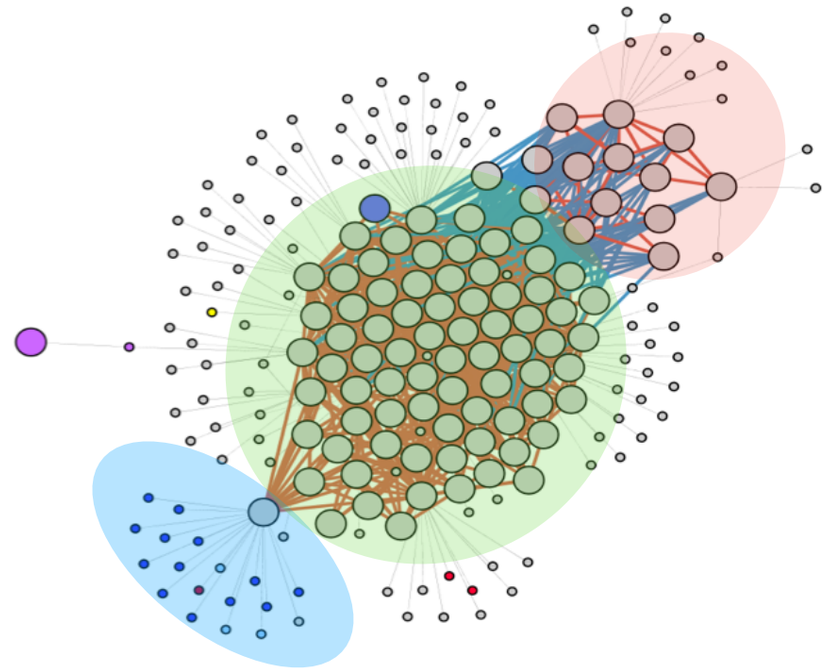
Biological Networks



Why look at network topology?

Use networked systems to identify:

- Identify global / local patterns
- Identify functional properties
- Make predictions



Examples:

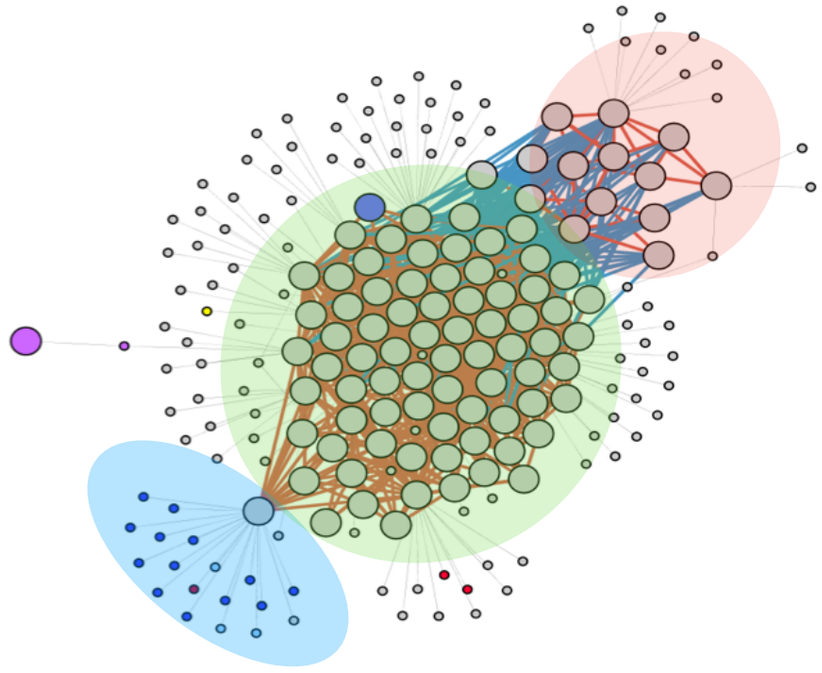
- How associated are the elements of my network?
- What are its first-hand associated elements?
- What are the groups of closely-associated elements in my network?
- What are their functional relationships?
- What are the "weakest" links in the network?

What is my biological network?

Any association matrix may be translated to a network format

Many standard analyses may be employed regardless of data type

…but care must be taken in generating the network



Some of the functional analyses depend on annotation

Limitations

Sample size

False discovery

~requires high throughput

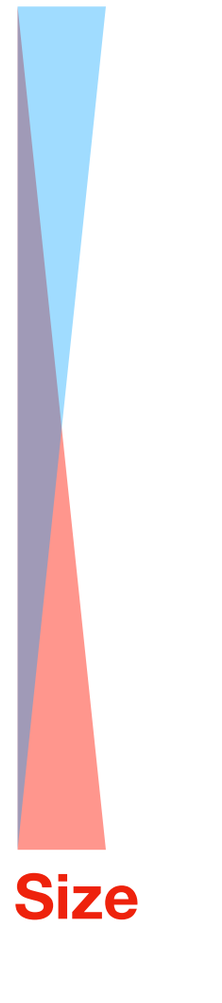
(further discussed in following lectures)

Terminology and initial properties in graph analysis

Motivation

What modeling formalism suits your data and biological question?

	Pros	Cons
Kinetic models	Detailed Quantitative Dynamic / Steady state	Small Requires detailed parameterization
Stoichiometric	Large Semi-quantitative Steady state	Static
Topological	Large Only topological information	No dynamic properties



Adapted from Hartmann et al.

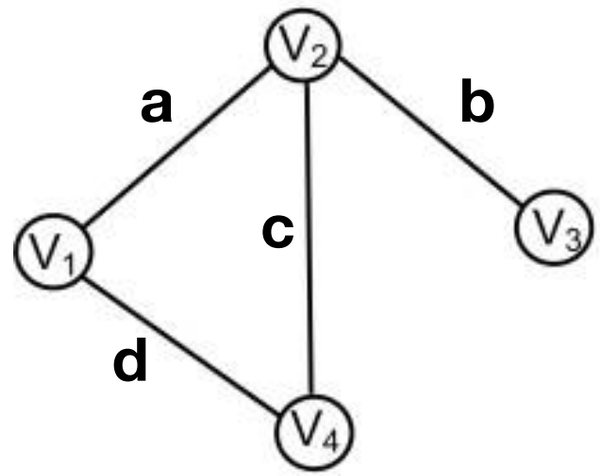
Graphs, nodes, edges

Graph *G* consists of a set of **nodes** (V) interconnected by **edges** (E)

$$G = (V, E)$$

$$V=\{v1,v2,v3,v4\}$$

$$E=\{a,b,c,d\}$$



Nodes sometimes called **vertices**

Two connected nodes are called **neighbours, adjacent,** or **end-nodes**

Simple vs multigraphs

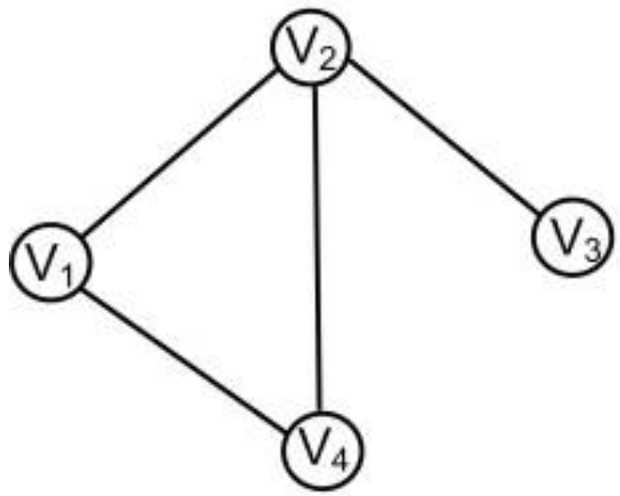
Multigraphs contain parallel edges

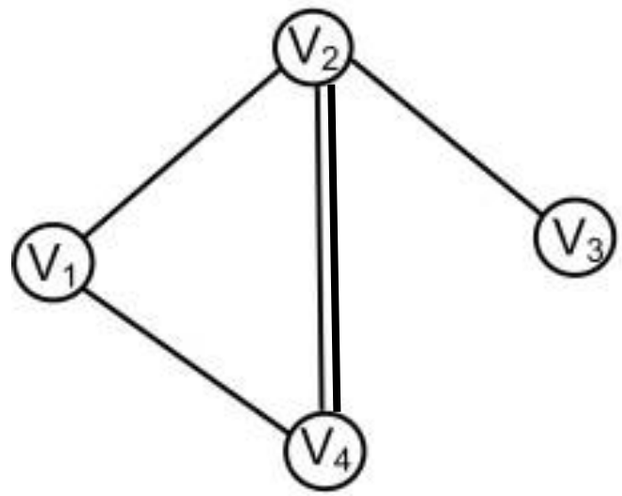
Multi-edged connections indicate different properties

Multigraphs vs **simple** graphs

Example: PPI

- Experimental evidence for interaction
- Co-expression



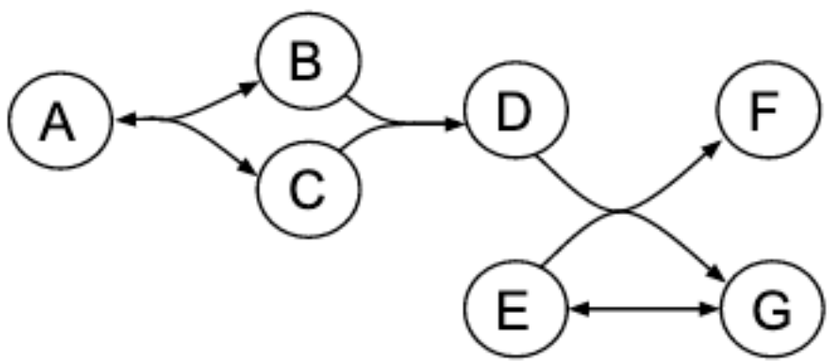


Hypergraphs

Hypergraphs contain edges that connect any number of nodes

Reaction 1: $A \rightarrow B + C$
Reaction 2: $B + C \rightarrow D$
Reaction 3: $D + E \rightarrow F + G$
Reaction 4: $E \rightarrow G$
Reaction 5: $B + C \rightarrow A$
Reaction 6: $G \rightarrow E$

(a) Reaction network



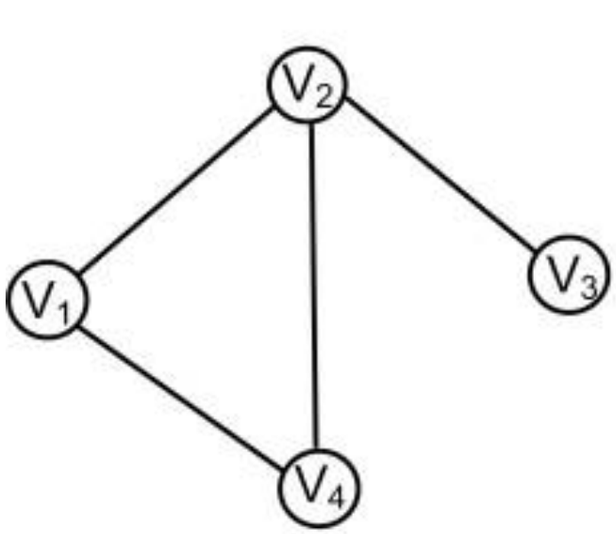
Directed vs undirected graphs

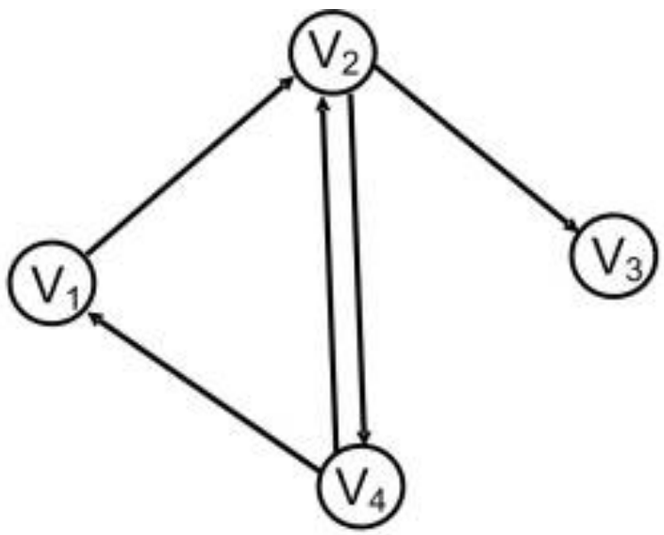
Directed graphs is given by an ordered triple

$G=(V,E,f)$

f : function mapping an element in ***E*** to the ordered pair of vertices in ***V***

E={ (v1, v2) , (v2, v3), (v2, v4), (v4, v1), (v4, v2)}





Directed vs undirected graphs

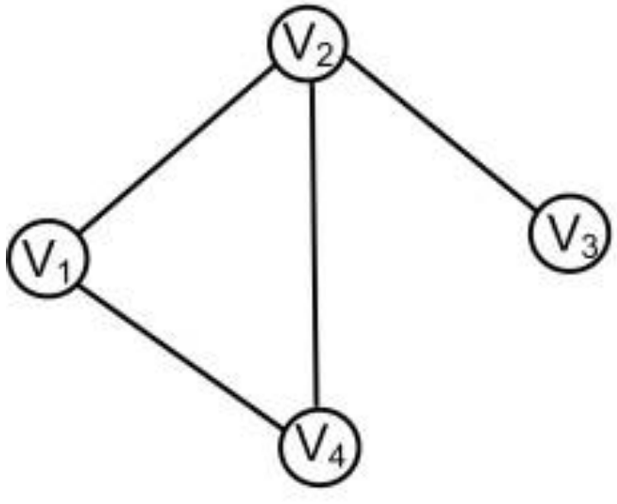
Directed graphs is given by an ordered triple

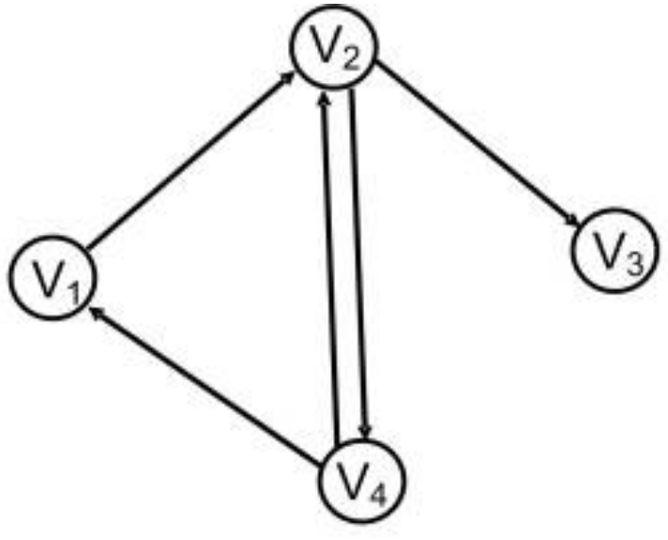
$$G=(V,E,f)$$

f : function mapping an element in ***E*** to the ordered pair of vertices in ***V***

Examples:

- **Undirected graphs:** co-expression networks
- **Directed graphs:** metabolic networks

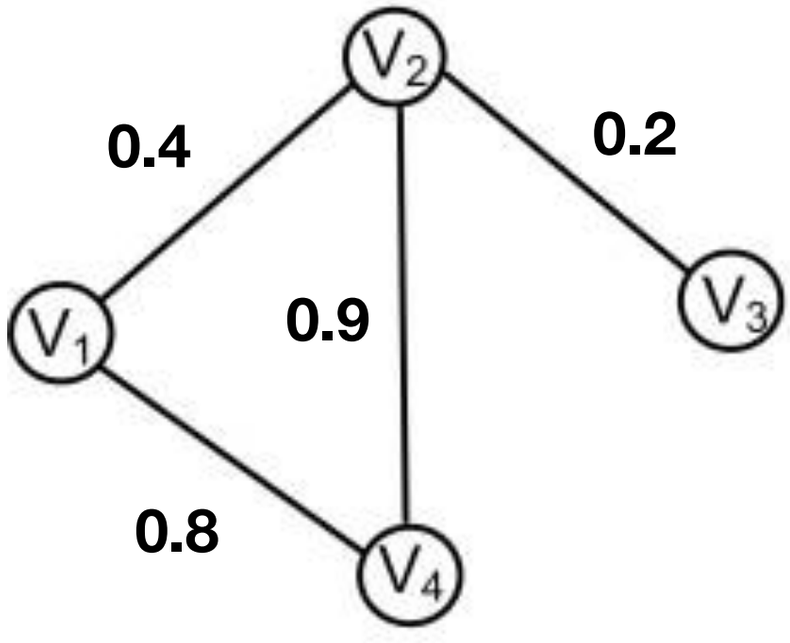




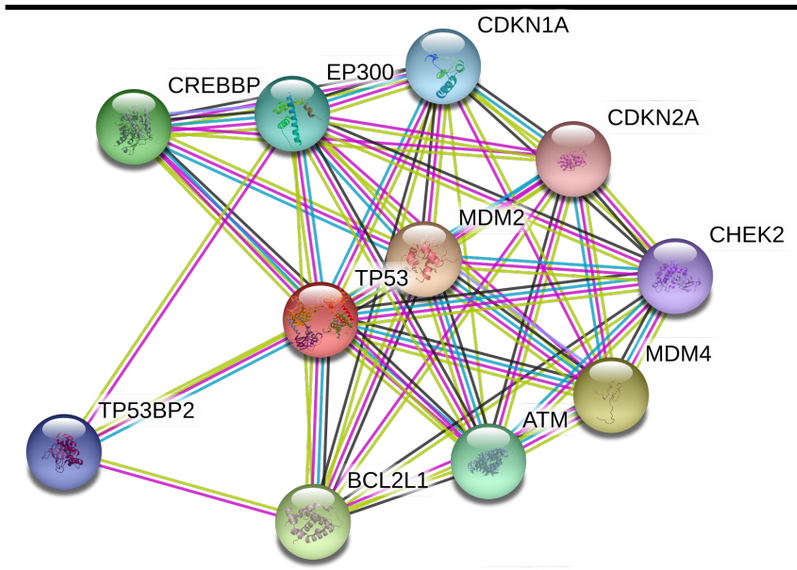
Weighted vs unweighted graphs

Weighted edges associate a value to an interaction between two nodes. Usually give the confidence in the interaction.

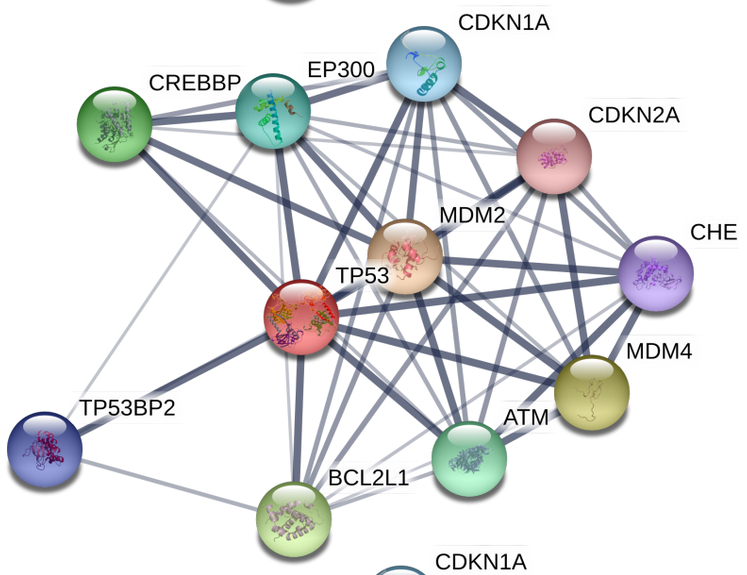
E.g. weighted co-expression networks



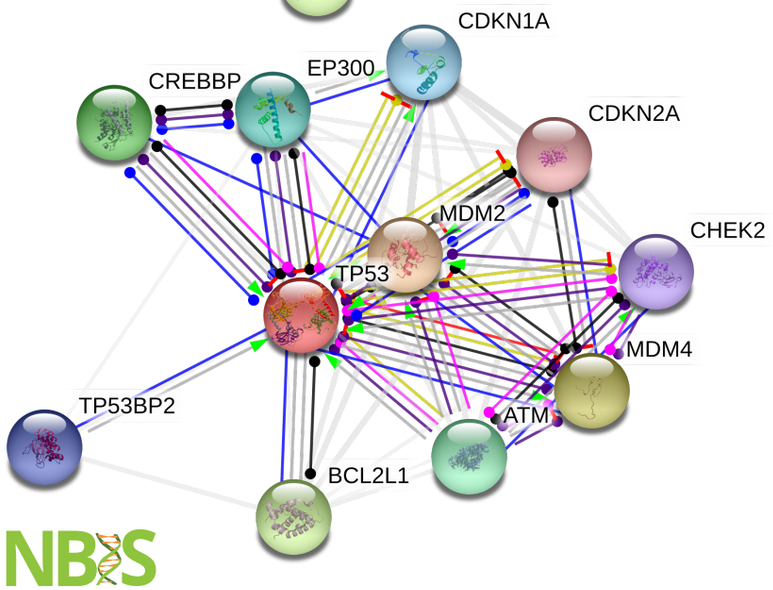
STRING-db.org: TP53



Multi-edged



Weighted multi-edged



Multi-edged directed

Bipartite graphs

A graph

$$G=(V,E)$$

may be partitioned into two sets of nodes (V_1, V_2) such that

$$u \in V_1 \text{ and } v \in V_2$$

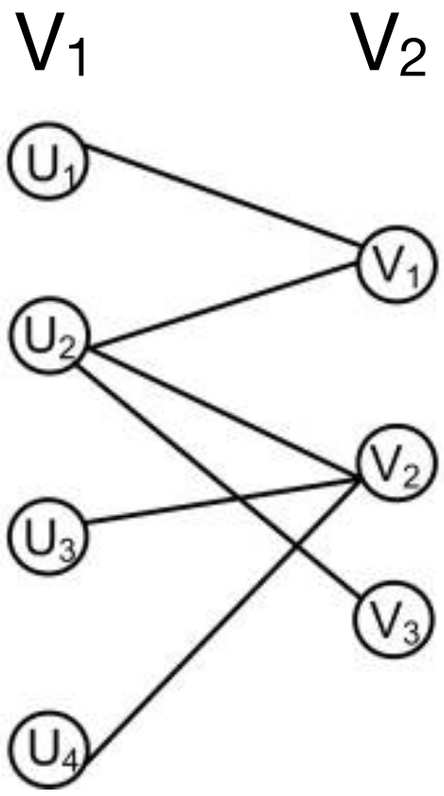
or

$$u \in V_2 \text{ and } v \in V_1$$

All e_i has end-nodes in V_1, V_2

A **subgraph** of G will thus be given by

$$G_1 = (V_1, E_1)$$



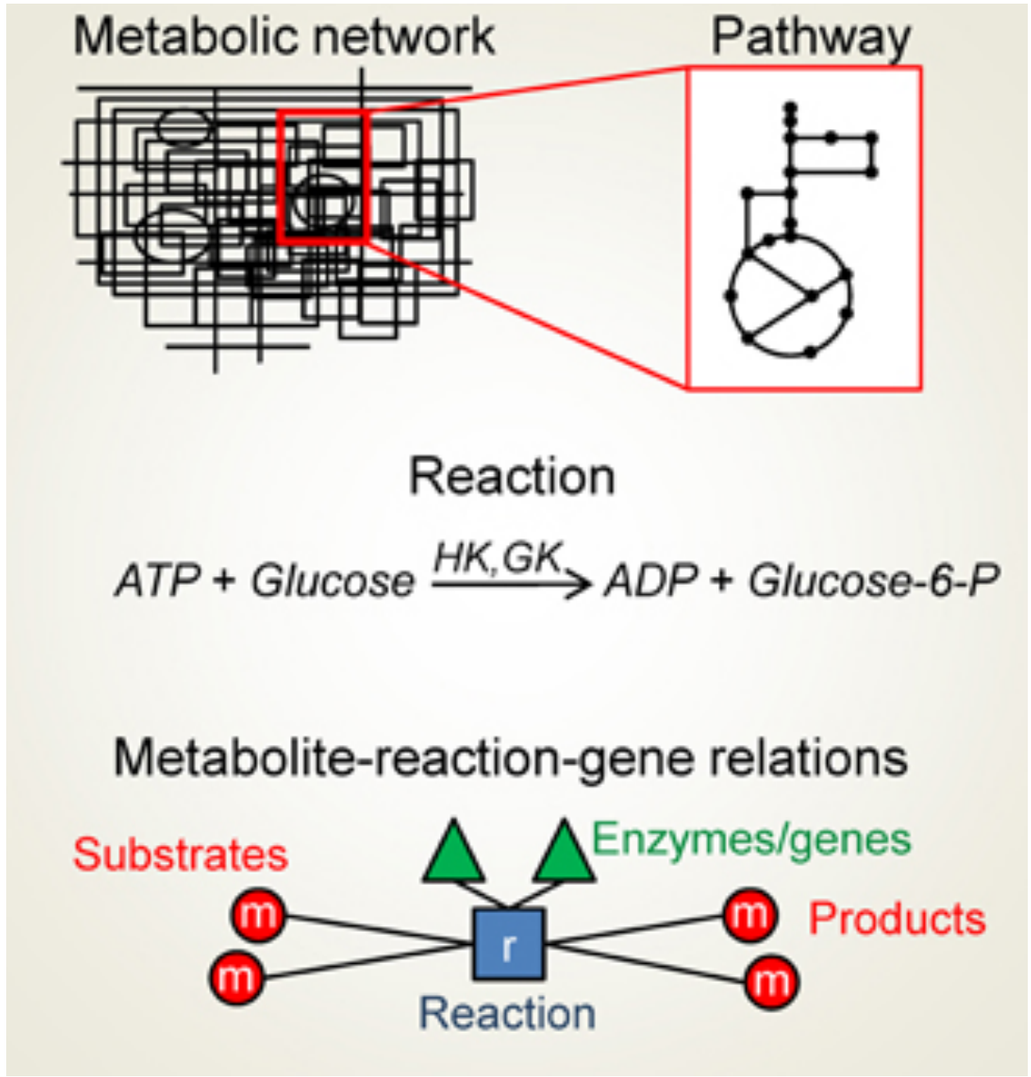
Bipartite and *k*-partite graphs

Example of bipartite graph:

Enzyme - Reaction

Metabolite - reaction - enzyme

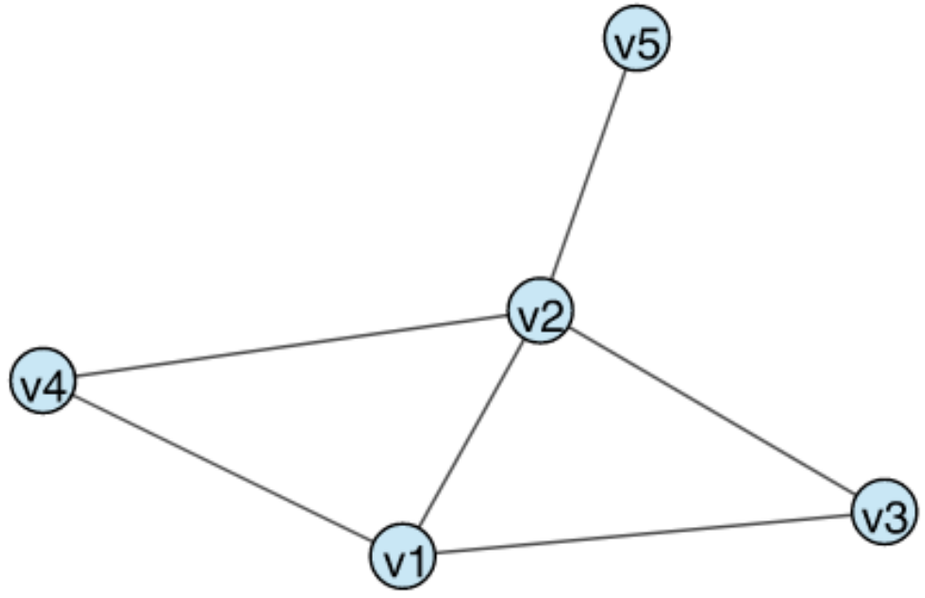
k-partite graphs display *k*-types of nodes



Adjacency matrix (undirected graphs)

Vertex association (undirected network)

n1	n2
v1	v2
v1	v4
v2	v4
v2	v3
v2	v5
v1	v3



Adjacency matrix is symmetric

	v1	v2	v3	v4	v5
v1	0	1	1	1	0
v2	1	0	1	1	1
v3	1	1	0	0	0
v4	1	1	0	0	0
v5	0	1	0	0	0

Upper triangular

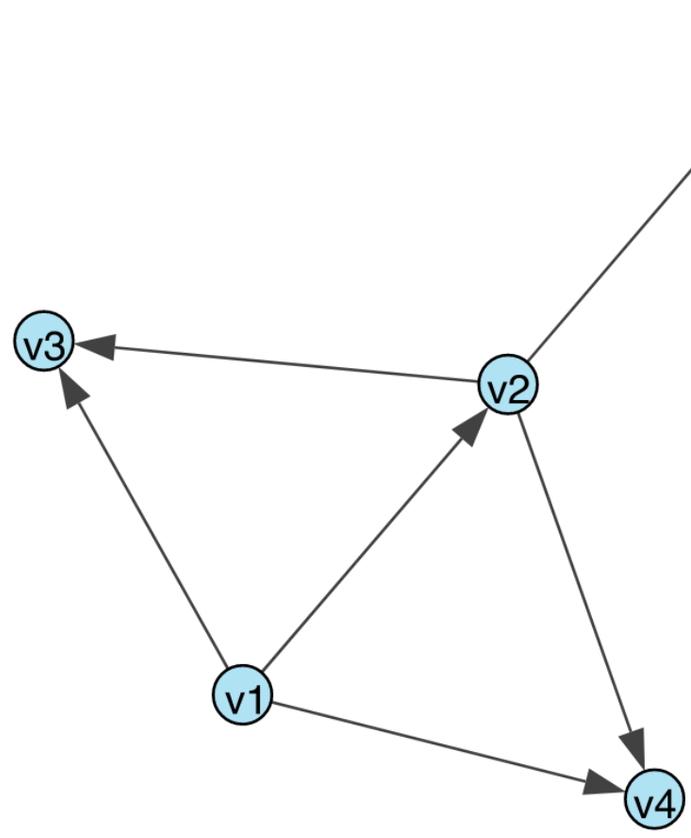
Diagonal

Lower triangular

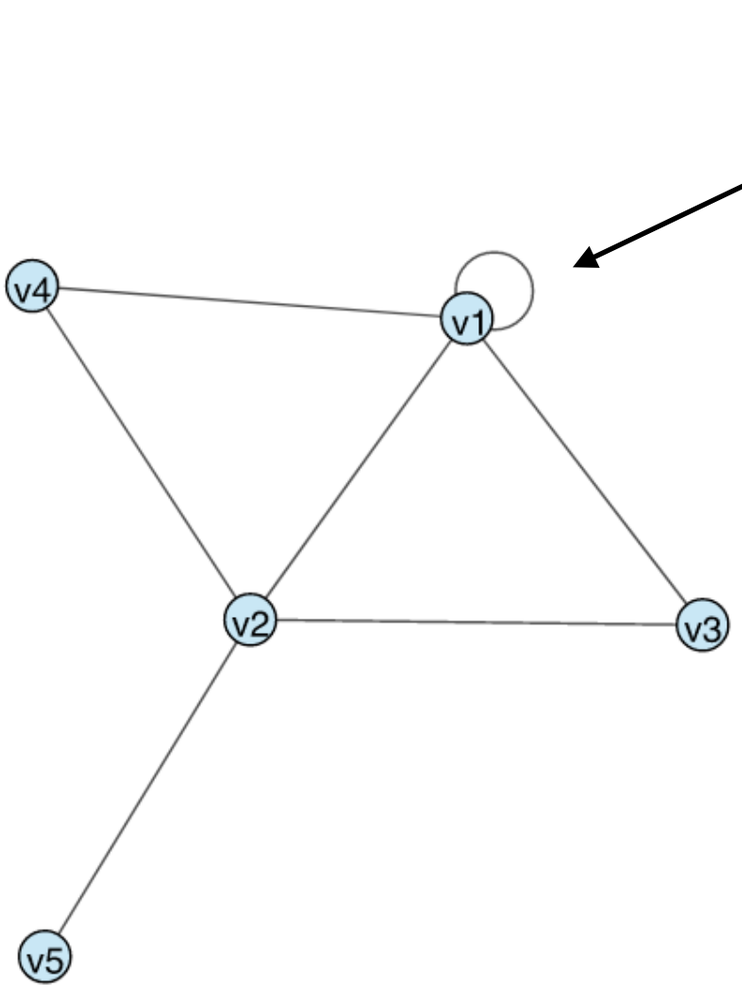
Adjacency matrix (directed graphs)

Source

	Target				
	v1	**v2**	**v3**	**v4**	**v5**
v1	0	1	1	1	0
v2	0	0	1	1	1
v3	0	0	0	0	0
v4	0	0	0	0	0
v5	0	0	0	0	0



Graphs may contain self-loops



Loop

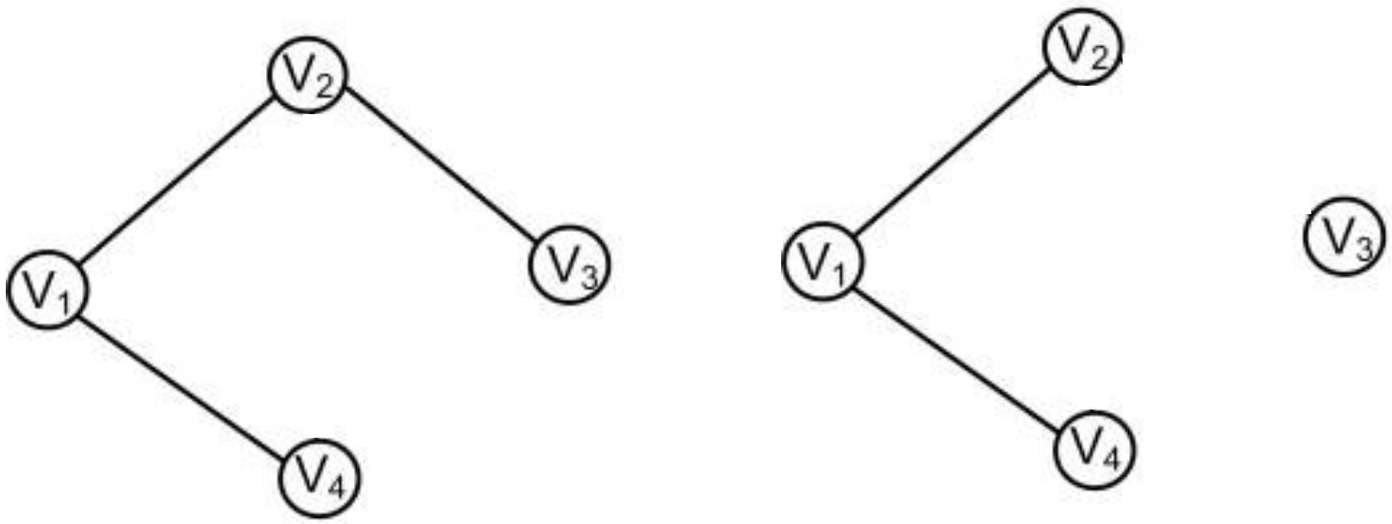
	v1	v2	v3	v4	v5
v1	**1**	1	1	1	0
v2	1	0	1	1	1
v3	1	1	0	0	0
v4	1	1	0	0	0
v5	0	1	0	0	0

Examples of **self-loops** are auto-regulatory mechanisms in Transcription Factor regulatory networks

Connected vs disconnected networks

Connected network: there is at least 1 path connecting all nodes in a network

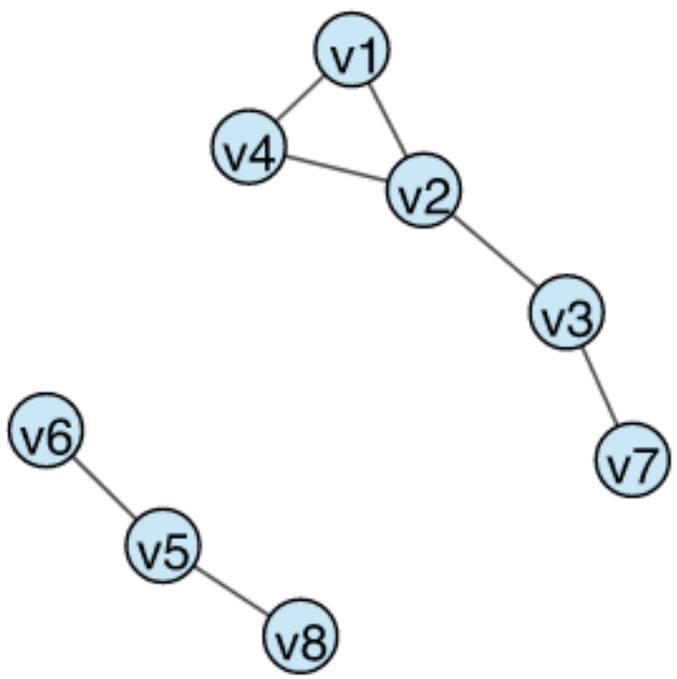
Disconnected network: some of the nodes are unreachable

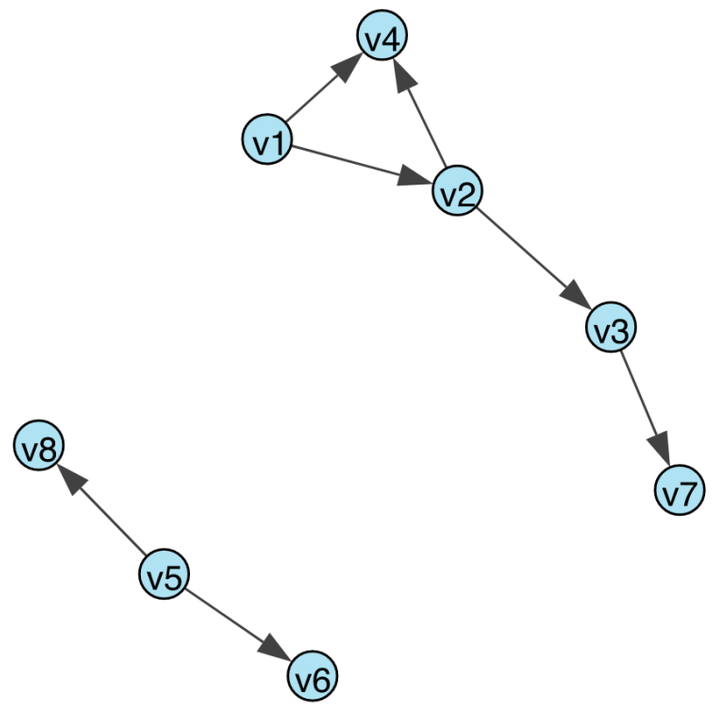


Connected components

Connected components are those where all nodes of each subgraph are connected.

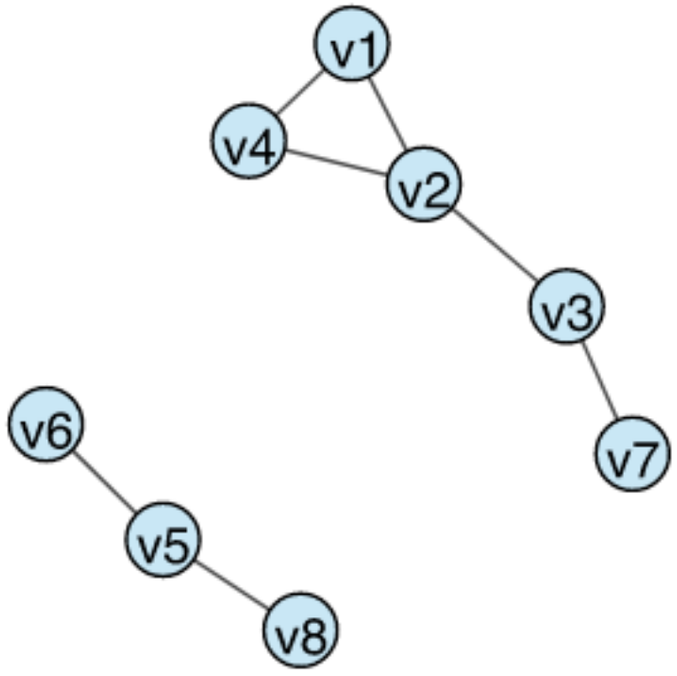
Weak vs **strong** components

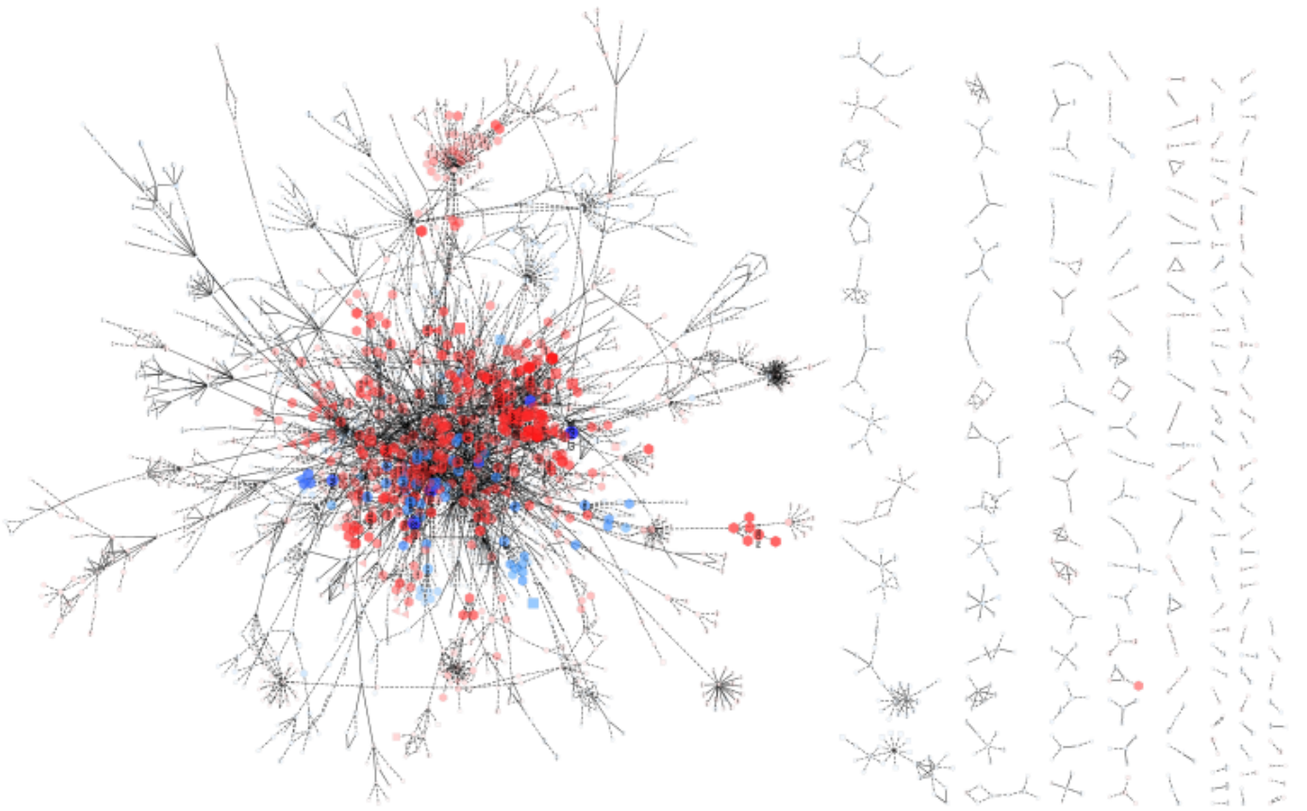




Connected components

In biological networks, often the most insightful properties come from the **largest connected component**

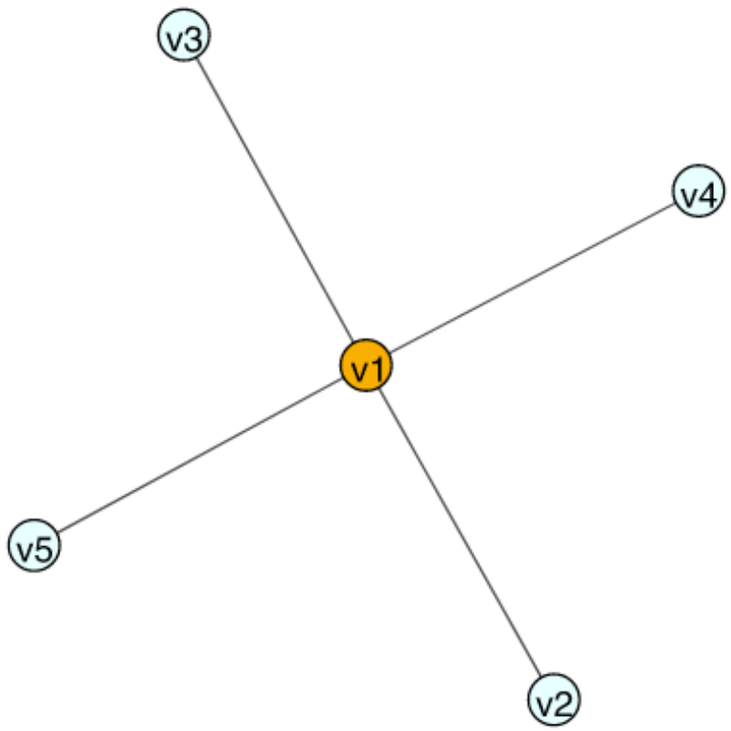


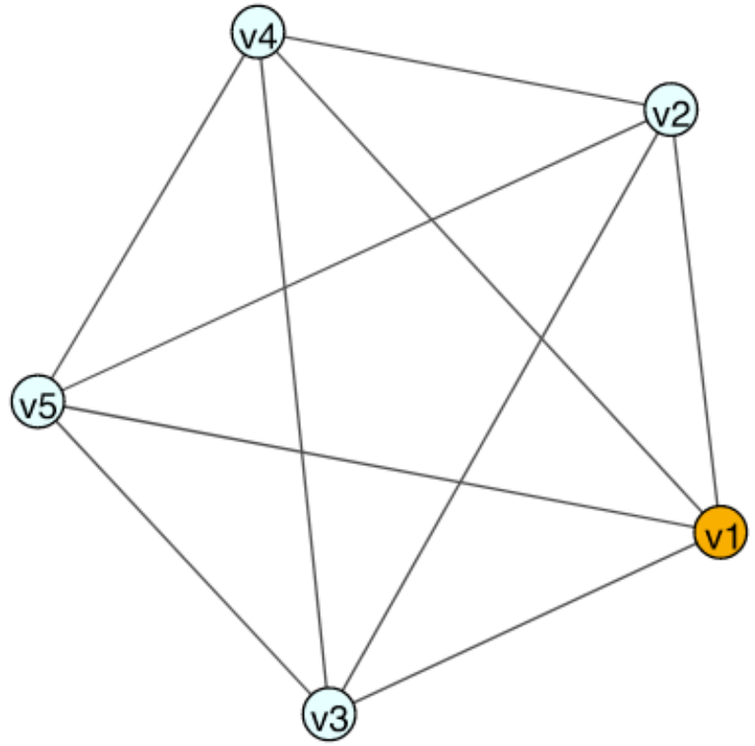


Stars vs cliques

A **star** forms a **complete bipartite subgraph**

A **clique** is a subgraph where all nodes are adjacent

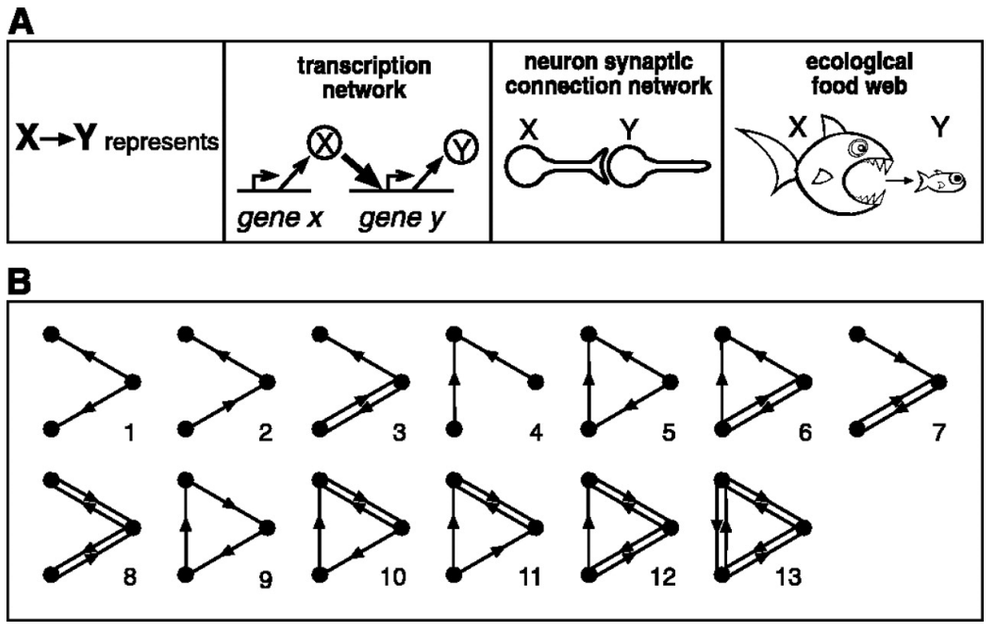


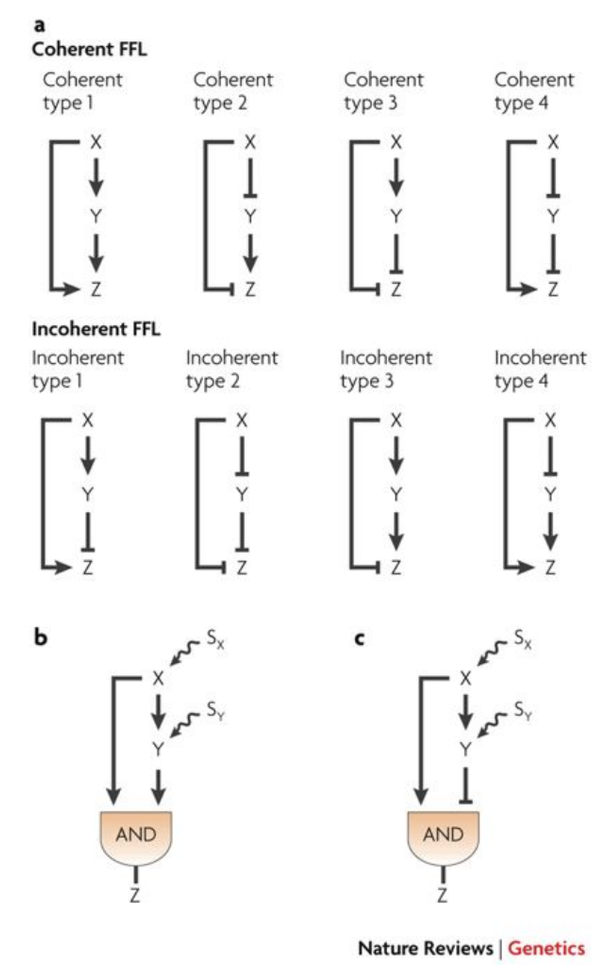


Motifs

Subgraphs are characterised by different motifs

Exploring prevalent motifs may allow us to understand the evolutionary advantage of a given architecture





Milo 2002
Alon 2007

Additional reading

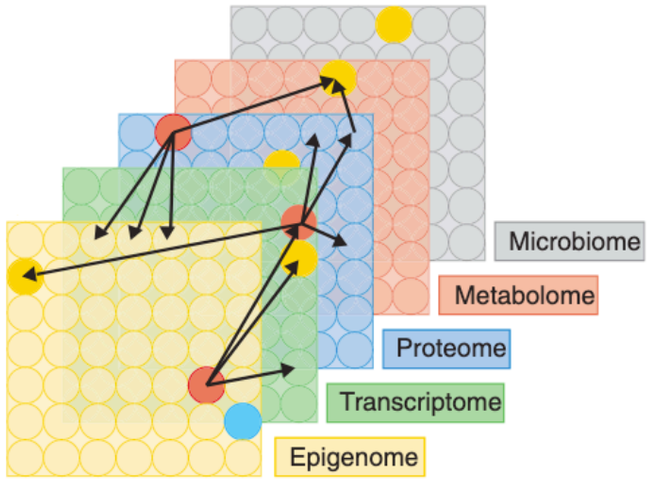
- Network Science - A fascinating textbook on graph theory and network analysis.
- Communication dynamics in complex brain networks - Interesting discussion about whether and how network topology may be applied to study the brain networks.
- A Systematic Evaluation of Methods for Tailoring Genome-Scale Metabolic Models - General review and discussion on methods to use in genome-scale metabolic models.
- Analysis of Biological Networks - General introduction into biological networks, network notation, and analysis, including graph theory.

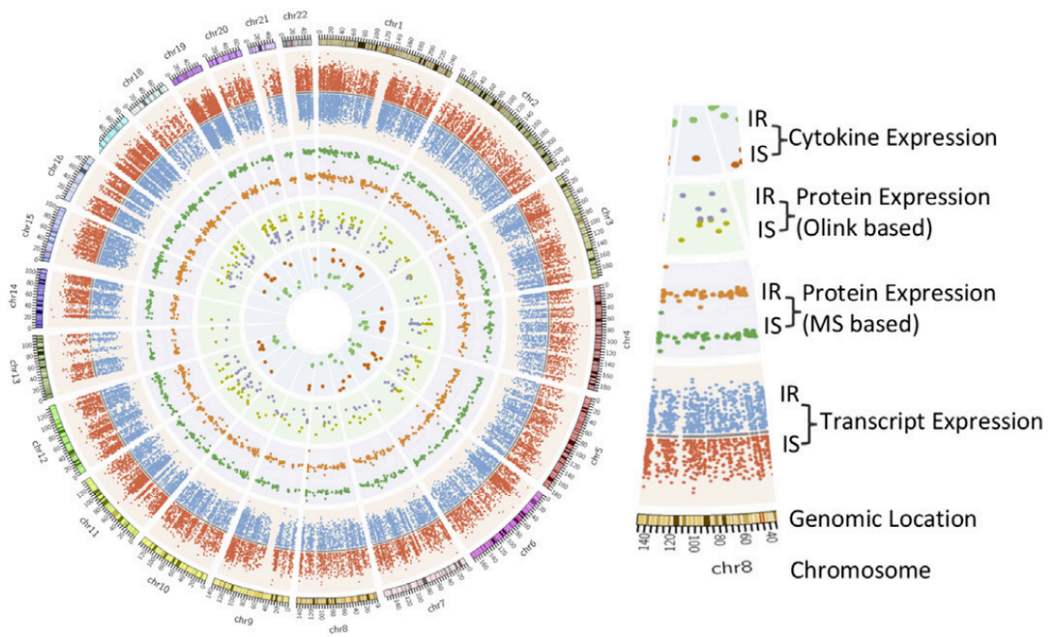
Additional references displayed as hyperlinks in each figure.

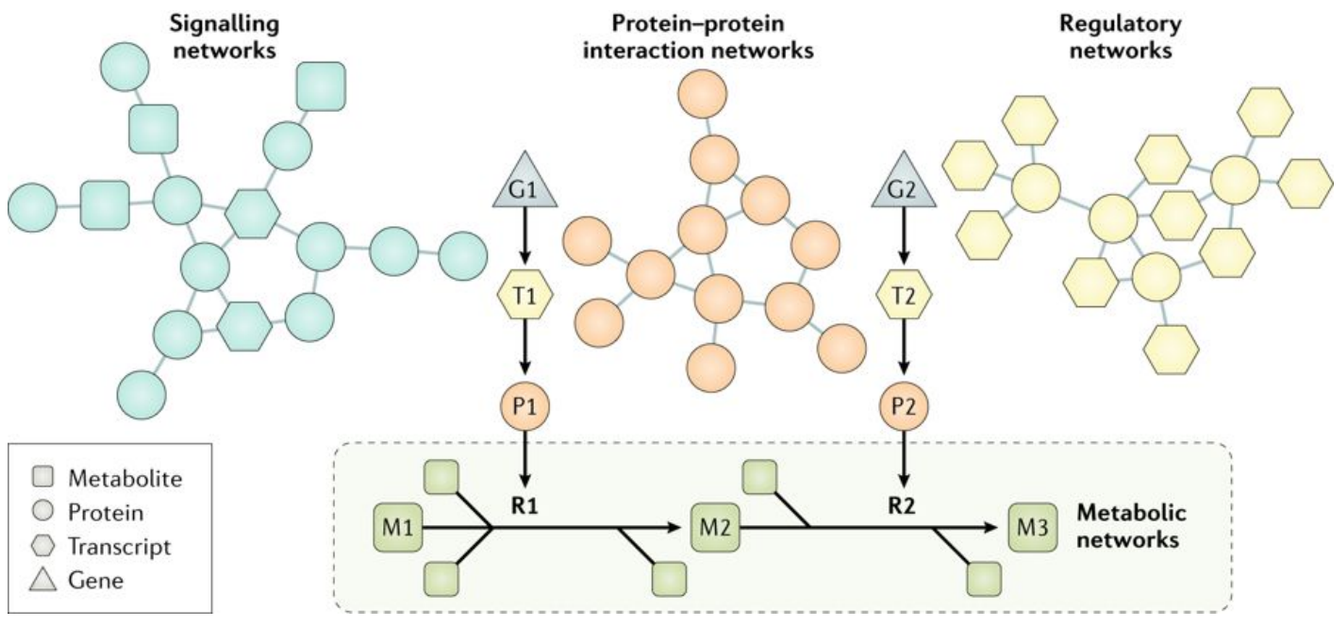
Biological network inference and analysis

Building networks

How to go from raw data to a graph tractable format?







Hasin 2017
Piening 2018
Mardinoglu 2018

Important considerations

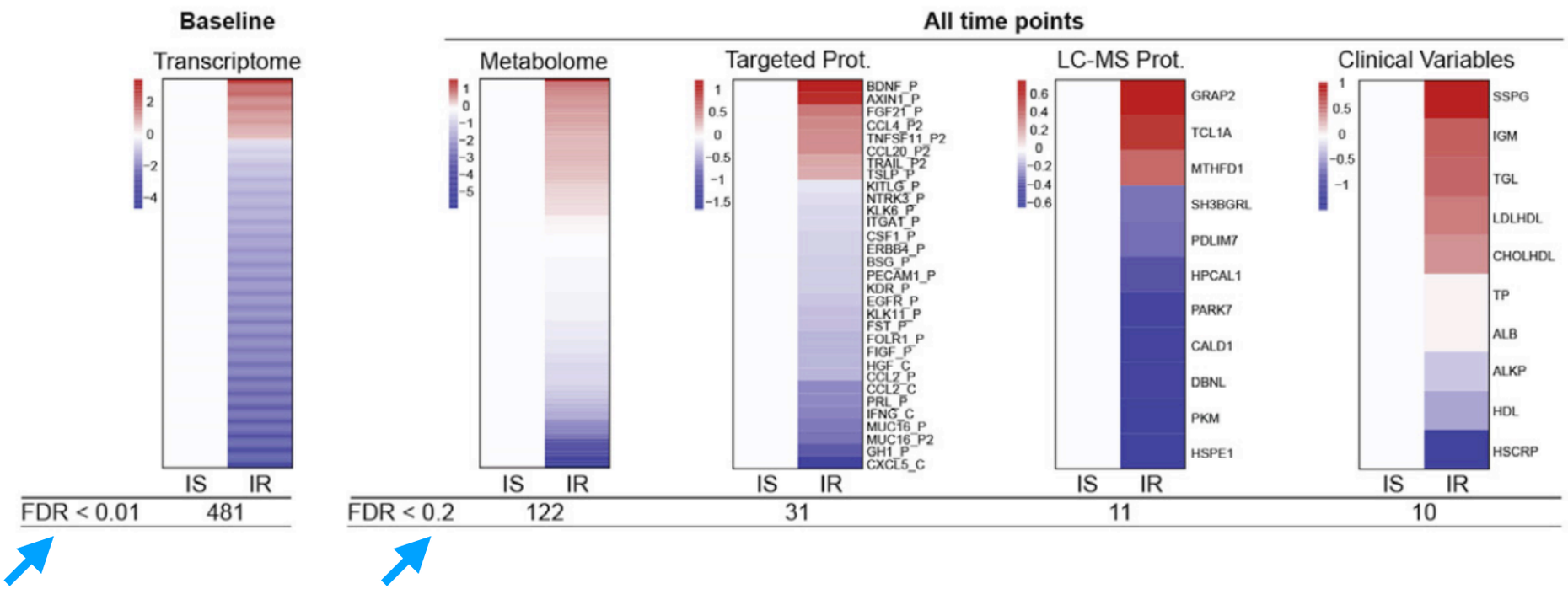
Standard statistics should not be ignored

Importance of power and sample size

Statistical power and significance in high throughput studies

Batch effect correction should be applied if needed, but avoided whenever possible

- Signal loss
- Overconfidence
- Alternative: include batch as a covariate
- Non-parametric analyses
- Limma's removeBatchEffects( ), NormalyzerDE( ), sva's ComBat( )



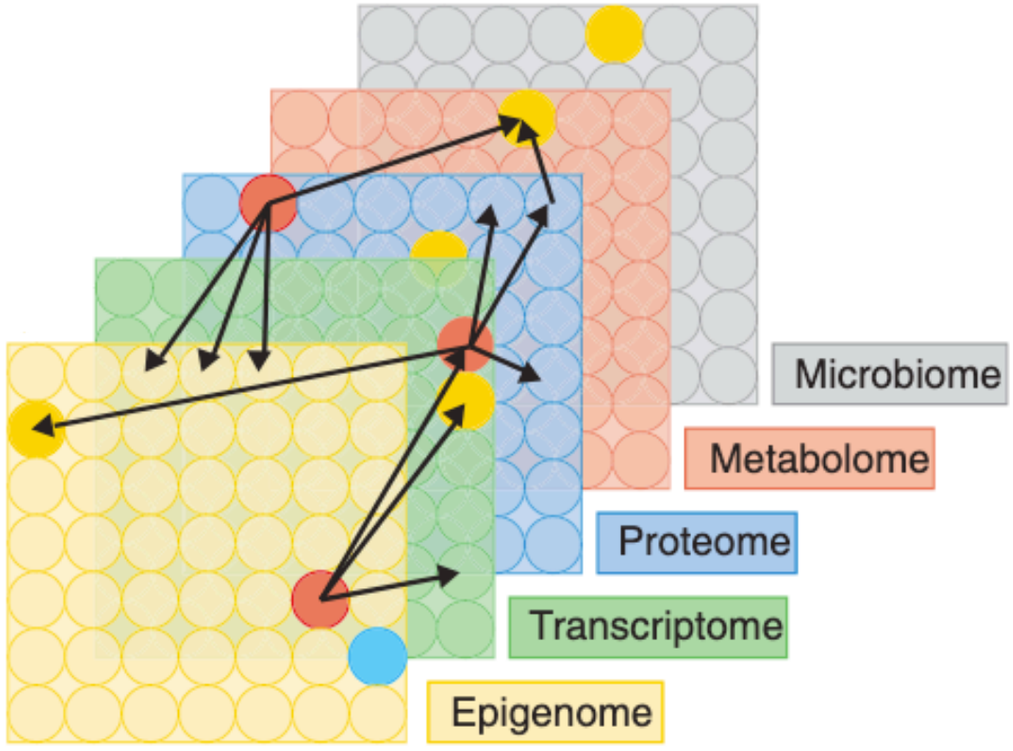
Clarke 2011
Krzywinski 2013
Sham 2014
Nygaard 2016
Piening 2018
other refs as links

Interomic vs Intraomic networks

Networks may be build for individual omics or for their integration

Should I even integrate different omics? What is my biological question?

- Do I want to analyse vertical relationship between features?
- Why integrate omics with different coverage such as transcriptomic and proteomic data
- Do I want to extract functional properties?
- Am I predicting biomarkers?

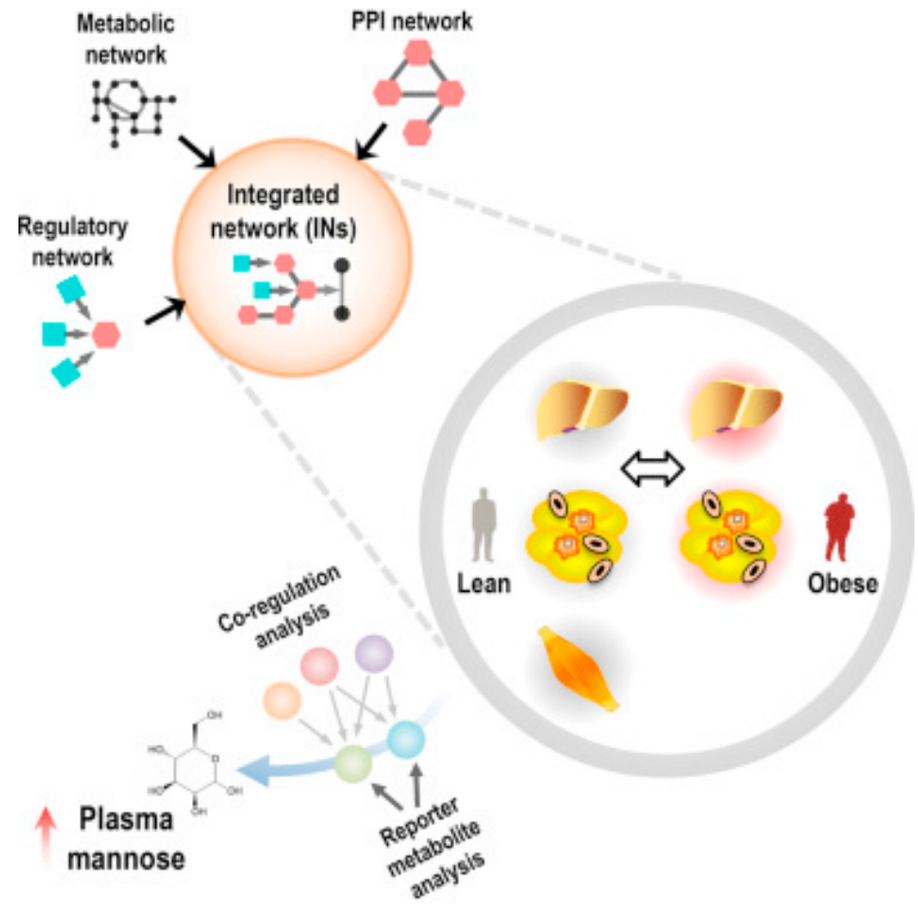


Interomic vs Intraomic networks

Multi-modal (*k*-partite) networks may be generated from different sources

- Transcription-factor - Gene (DNAseq)
- Gene-gene (Co-expression, PPI, GEMs)
- Gene-metabolite (GEM)
- Metabolite-metabolite (GEM)

Integrated Networks



Different approaches for network inference

1. Feature association
2. K-nearest neighbour graph (k-NNG) construction

No prior graph structure

3. Pathway-based
4. Genome-scale metabolic models

Based on available information

5. Network deconvolution

Filter indirect effects

1. Association analysis

Balanced dataset, standard pre-processing of each omics needed

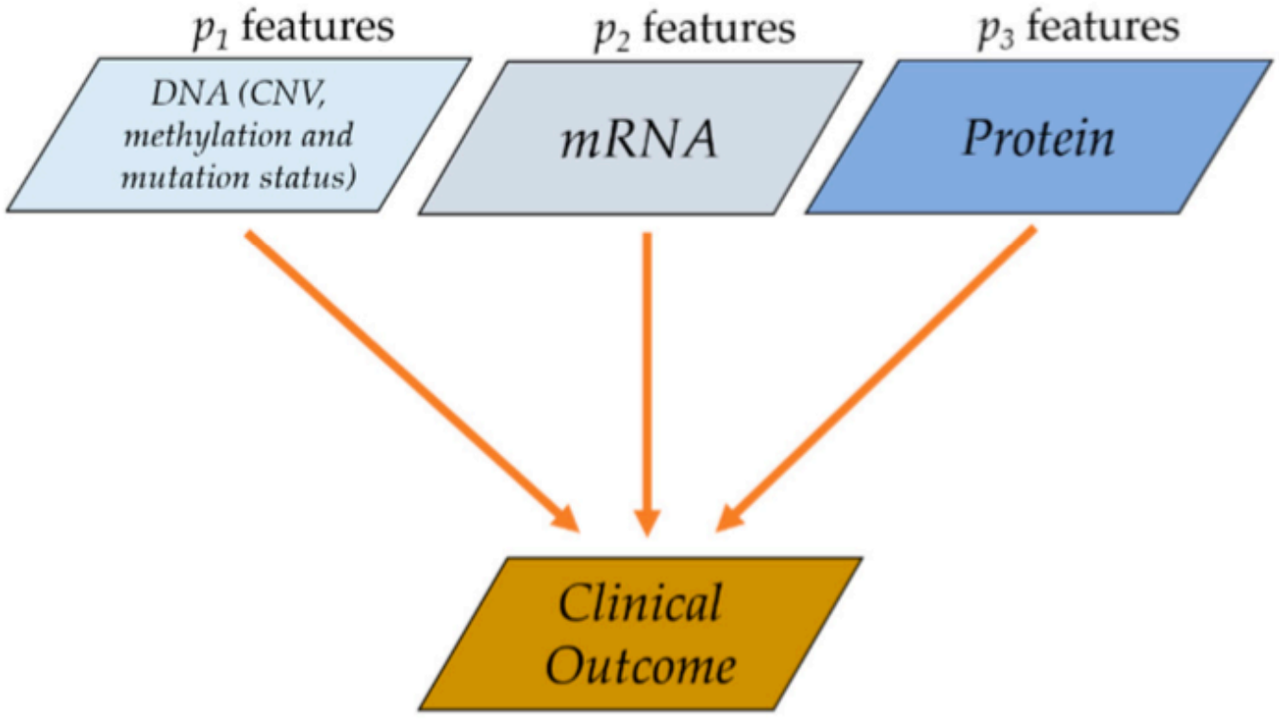
Normalization *may* be needed to make omic datasets comparable (e.g. standardization)

- depending on the following steps

Common approach: compute correlations between different features

- Spearman
- Pearson

Extend known associations



Wu 2019

1. Association analysis

Easy to interpret

Unweighted vs weighted ($-1 \leq \rho \leq 1$)

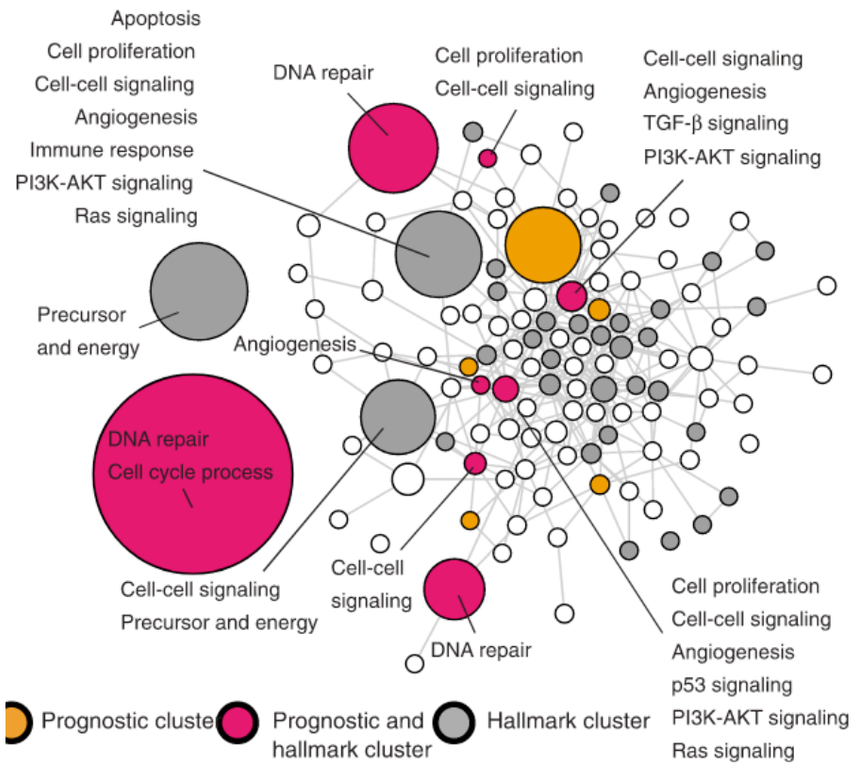
Unbalanced networks

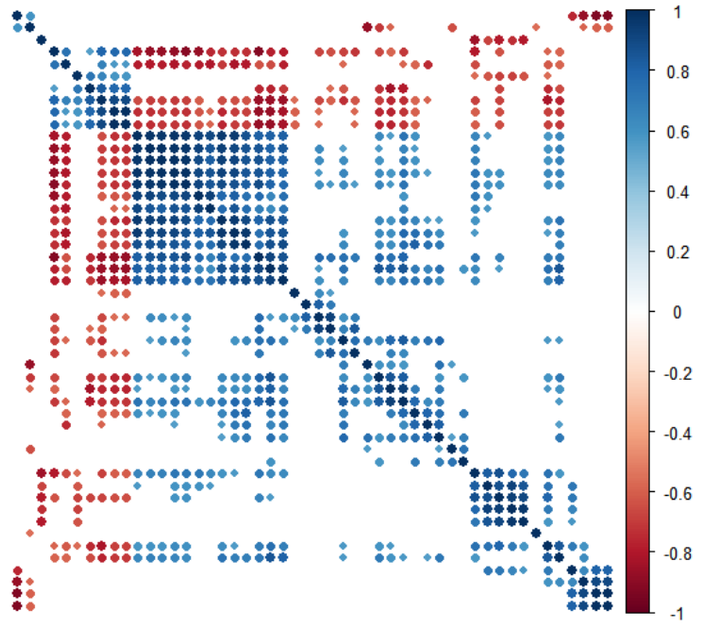
Prone to type I errors

Being conservative in the statistic methods is essential

- FDR vs Bonferroni
- LFC cutoff

Need adjustment to possible confounding factors





1. Association analysis

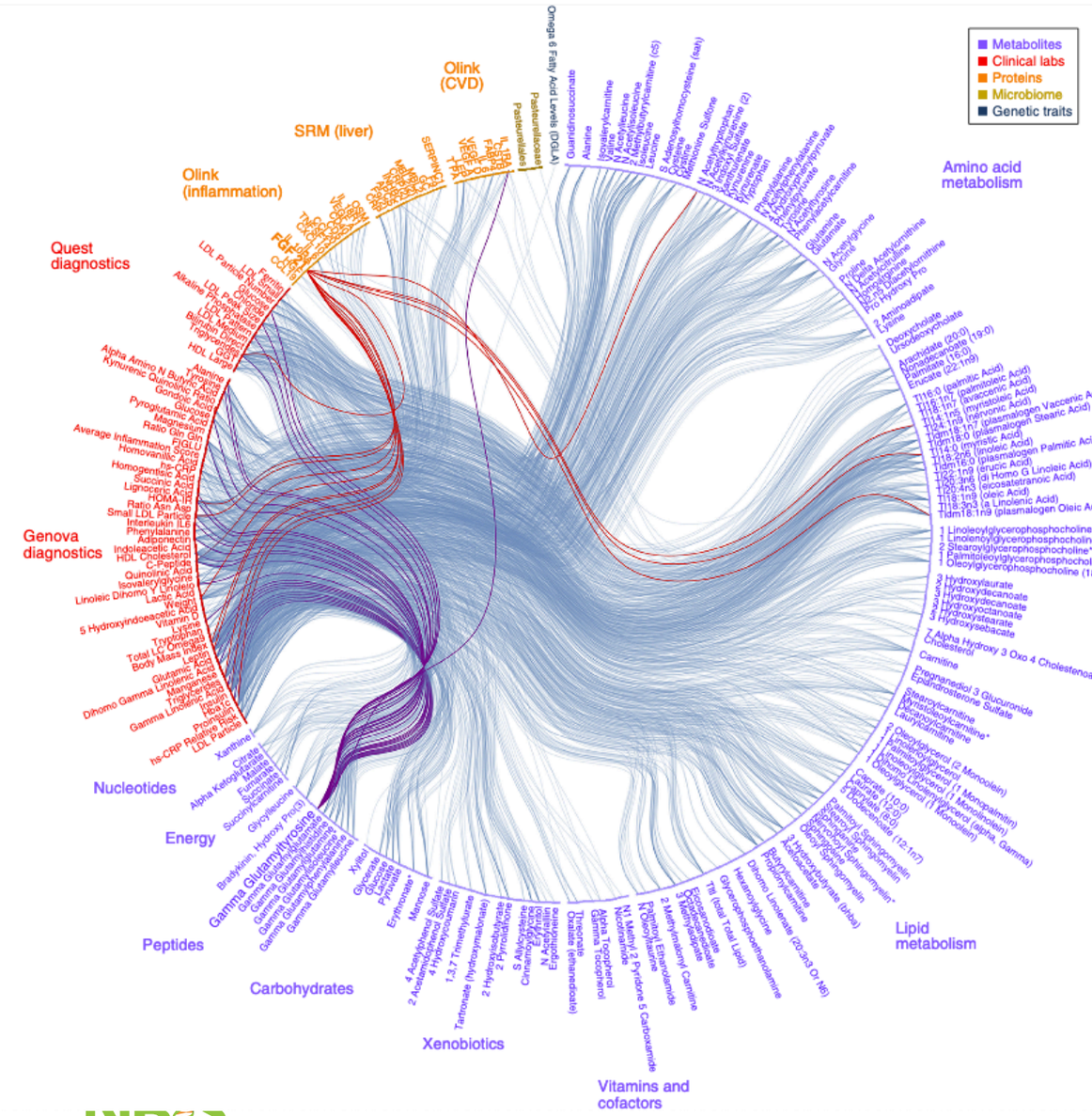
Adjusting for confounding factors: partial correlation analysis

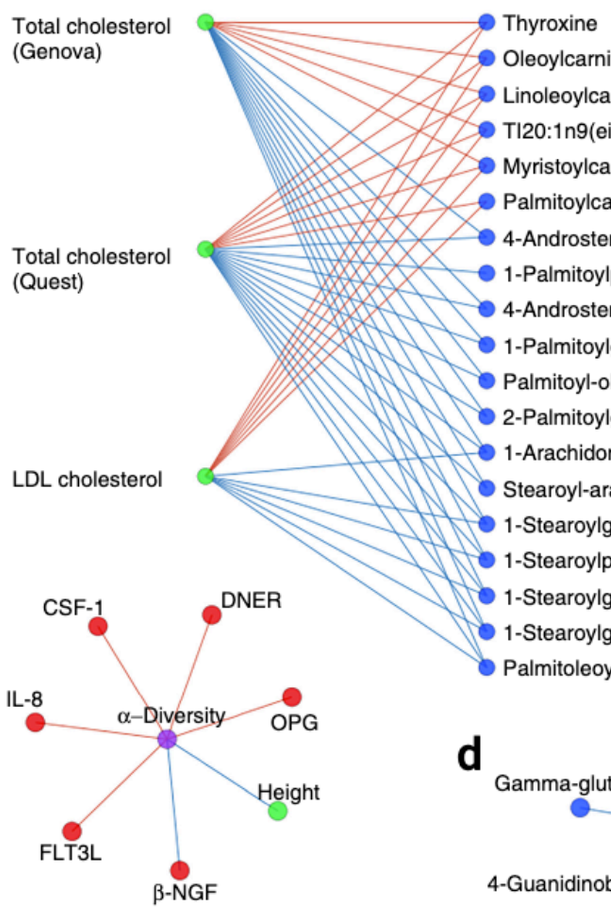
Still considers linear relationship between variables

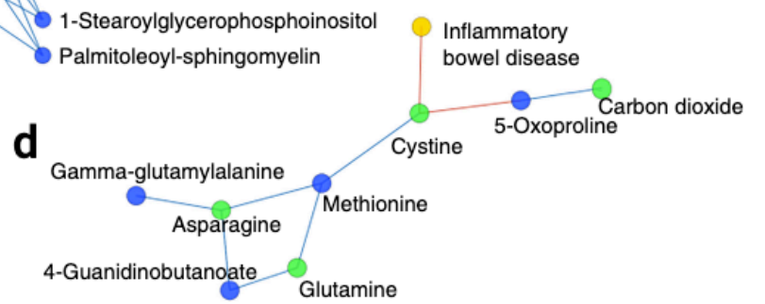
Below:

- gender and age are known confounding factors

- feature regression on confounding factors, followed by correlation on the residuals of each model





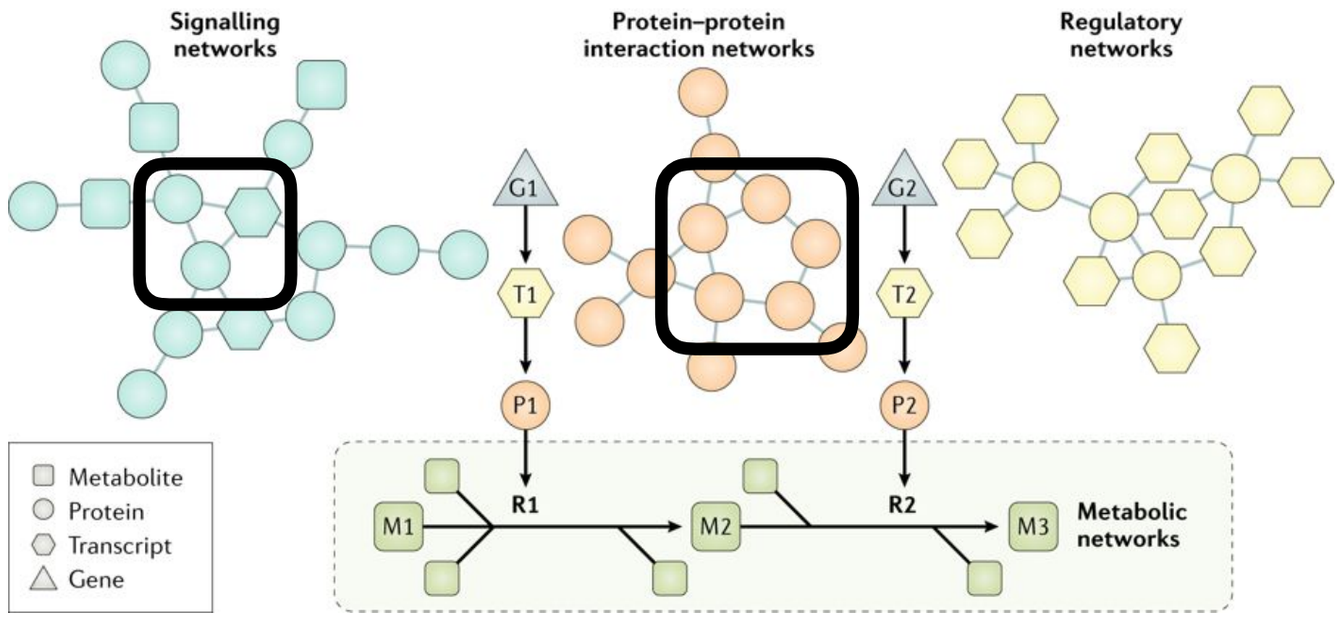


Computationally expensive representations may be simplified

Does your graph have many cliques? **Possibly noisy**

Graph contraction simplifies the graph by successively grouping cliques

Problem: reduces information and prevents studying many properties of the graph



Clarke 2011
Krzywinski 2013
Sham 2014
Nygaard 2016
Piening 2018
other refs as links

2. *k*-nearest neighbour graph

1. For each pair of features (u, v), compute a distance metric:

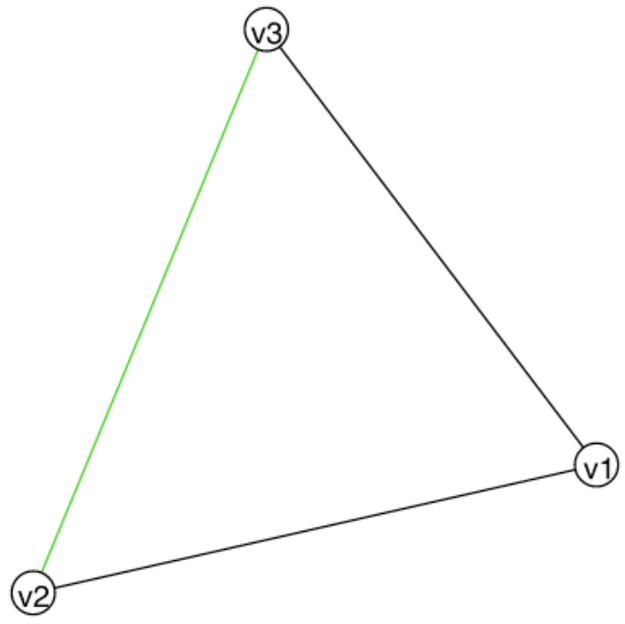
- Correlation

- Euclidean

- Jaccard

- ...

2. For each feature, select the *closest k* neighbours



Efficiency (not scalable, compute all neighbours for every node)

Generates well-structured graph

Simple as it reduces the number of features

Loses potentially important information because *k* is fixed

Dong 2011

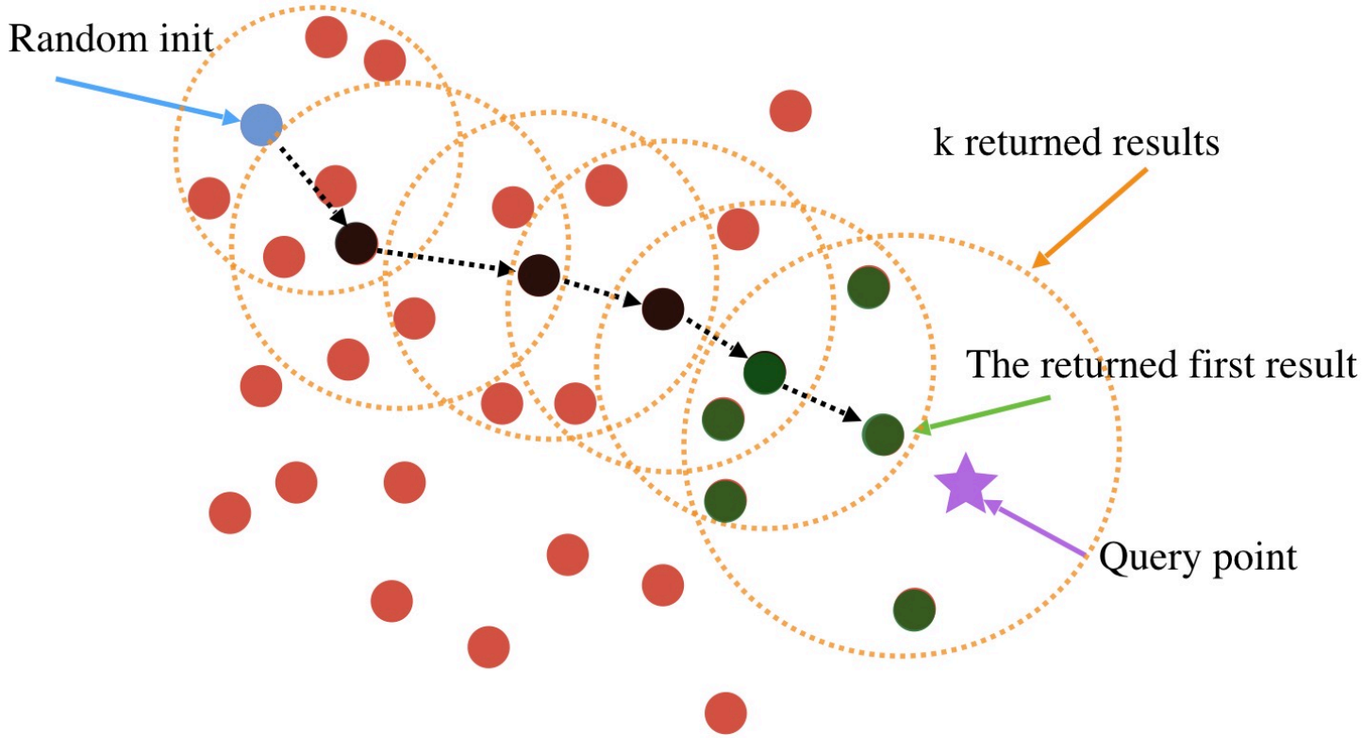
2. *k*-nearest neighbour graph

Brute force vs **NN-Descent**

A neighbour of a neighbour is likely to also be a neighbour

Given a query point *B*:

1. Start with random node *v*;
2. For each *v*, find the *k* nearest objects;
3. Compare its neighbours' neighbours
4. Repeat until no further improvement is possible
5. Improve local neighbourhood



3. Knowledge-based graph creation

Database-derived

- PPI
- TFRN
- Metabolic Atlas
- …

Many reference databases

KEGG

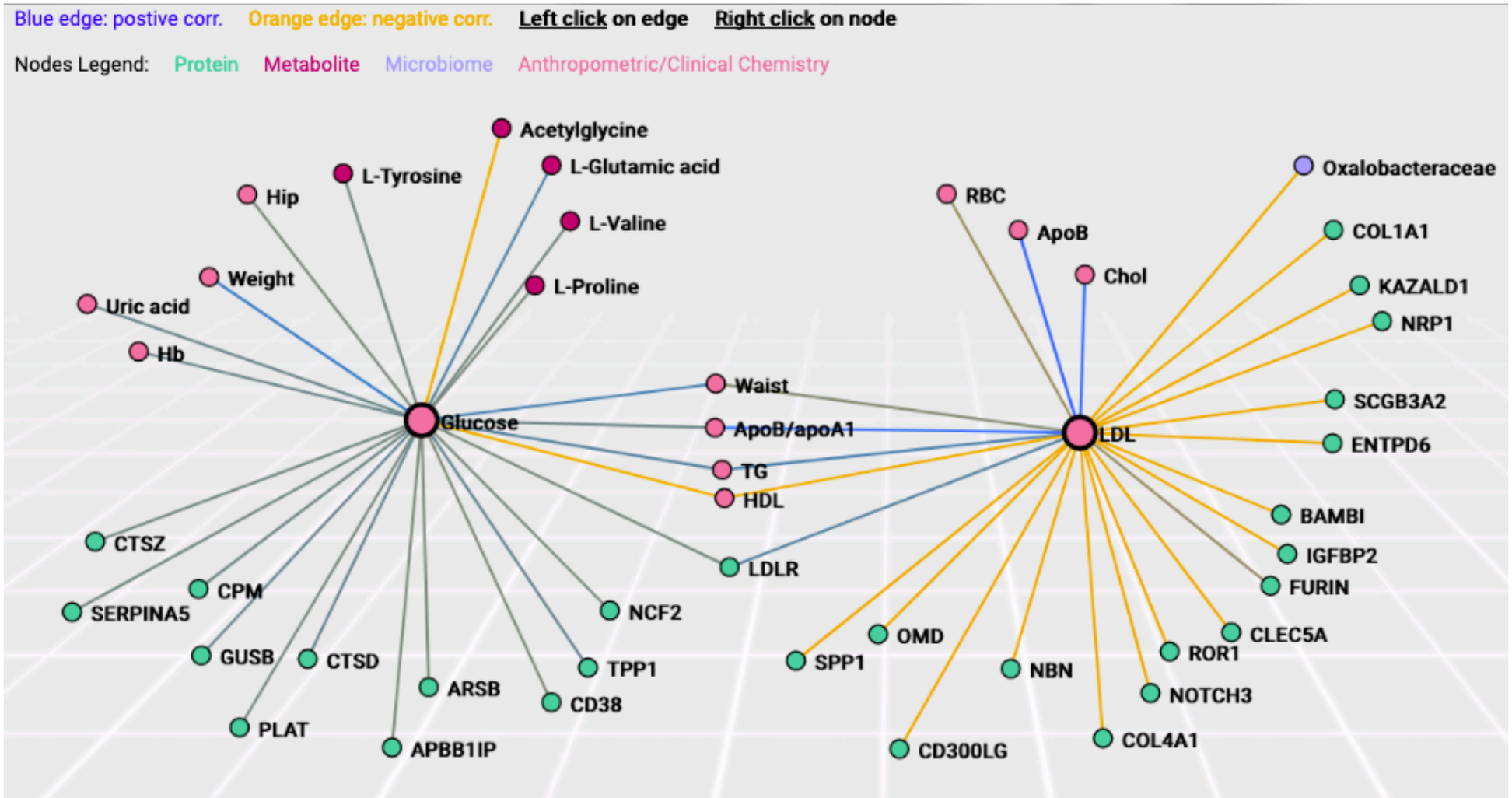
Reactome

WikiPathways

MSigDB

3. Knowledge-based graph creation

Multi-omic biological networks



3. Knowledge-based graph creation

NAR December 2018: 1613 databases

OXFORD ACADEMIC | Journals

You are here: NAR Journal Home » Database Summary Paper Categories

NAR Database Summary Paper Category List

- Compilation Paper
- Category List
- Alphabetical List
- Category/Paper List
- Search Summary Papers

Nucleotide Sequence Databases
RNA sequence databases
Protein sequence databases
Structure Databases
Genomics Databases (non-vertebrate)
Metabolic and Signaling Pathways
Human and other Vertebrate Genomes
Human Genes and Diseases
Microarray Data and other Gene Expression Databases
Proteomics Resources
Other Molecular Biology Databases
Organelle databases
Plant databases
Immunological databases
Cell biology

- Compilation Paper
- Category List
- Alphabetical List
- Category/Paper List
- Search Summary Papers

Collection of databases on our website

3. Knowledge-based graph creation

Overlap among reference pathways

3. Knowledge-based graph creation

How to overlay your data based on known interactions?

- Filter your predicted interactions based on known information? (intersection)

- Add interactions that are not found in the reference networks?

- Simply consider interactions based on physical presence considering the reference networks?

4. Using genome-scale metabolic models for graph creation

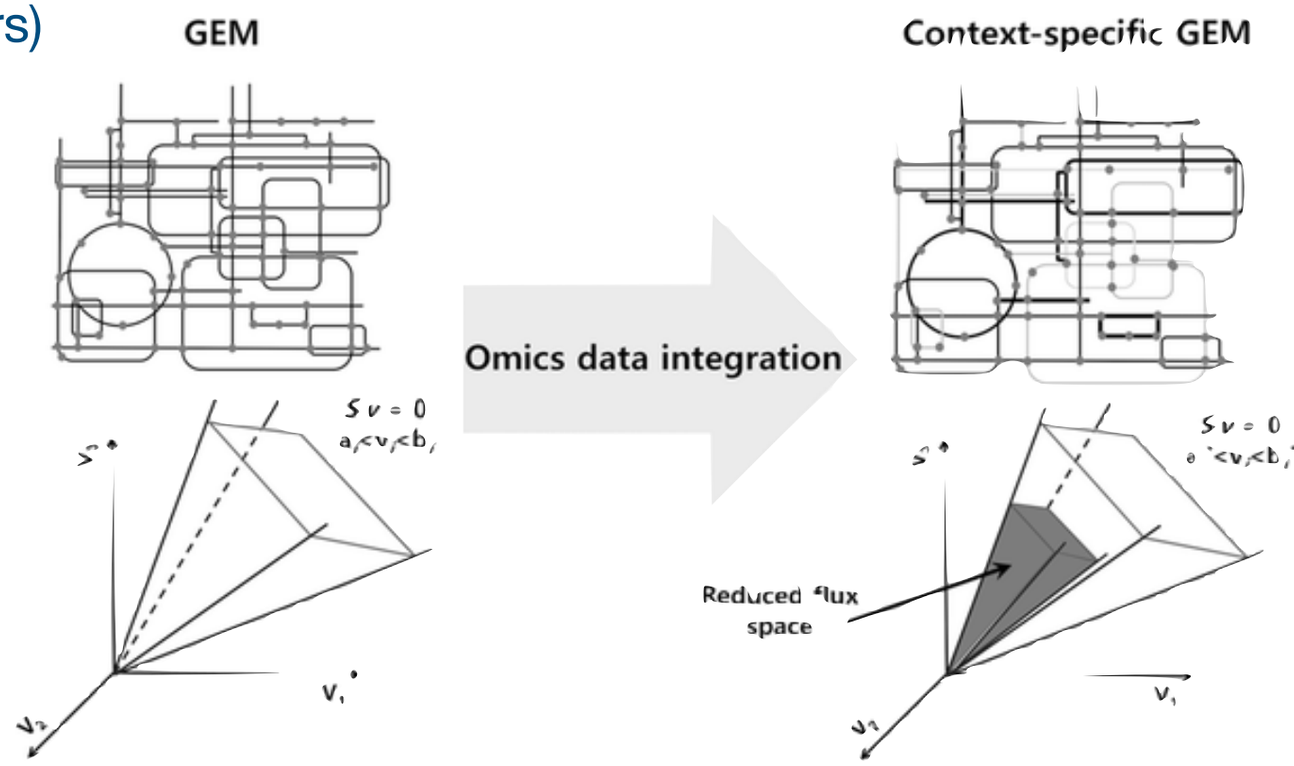
GEMs may be used to find such missing relationships, but there is a coverage issue

The overall strategy follows

1. Integrate proteomic, transcriptomic, metabolomic, fluxomic
2. Flux distribution
3. Compute metabolite-reaction-gene relationships
4. Extract relevant relationships (met-met, gene-gene)

4b. Exclude unnecessary interactions (e.g. cofactors)

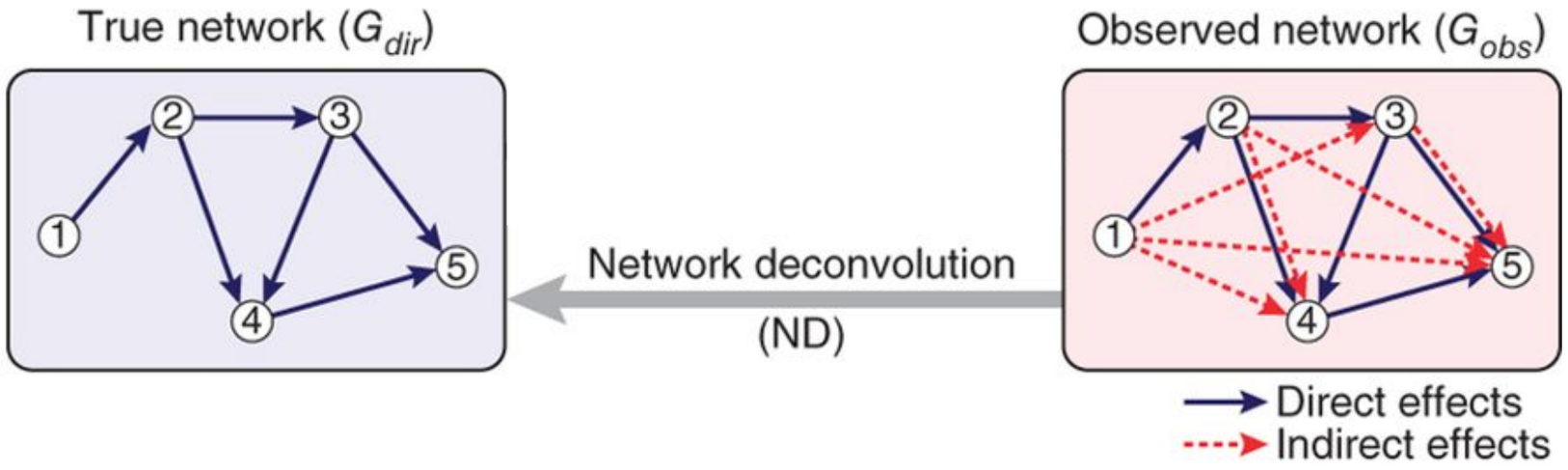
5. Downstream analysis (e.g. topology)



5. Network deconvolution

Biology is ***noisy***, which may result in edges that are not true

- 1 → 3
- 2 → 3
- 1 ⇢ 3



Direct and indirect effects

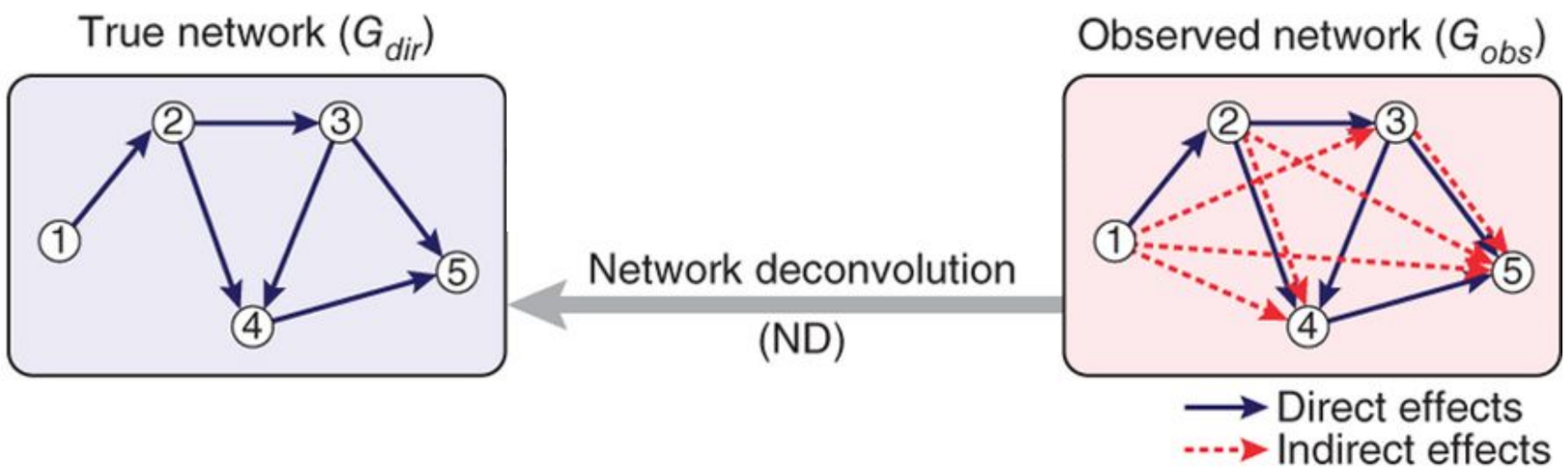
Indirect effects can be of arbitrary length:

- 2 → 3 → 5
- 1 → 2 → 3 → 5
- …

Decreasing effect with increasing path length

Indirect effects derived from algebraic manipulation for $l = 2, 3, \ldots$

5. Network deconvolution



$$G_{obs} = G_{dir} + G_{indir}$$

$$G_{indir} = G_{dir}^2 + G_{dir}^3 + G_{dir}^4 + \ldots$$

3rd order effects (paths of length 3)

2nd order effects (paths of length 2)

$$G_{obs} = G_{dir} + \underbrace{G_{dir}^2 + G_{dir}^3 + G_{dir}^4 + \ldots}_{\text{Indirect effects}}$$

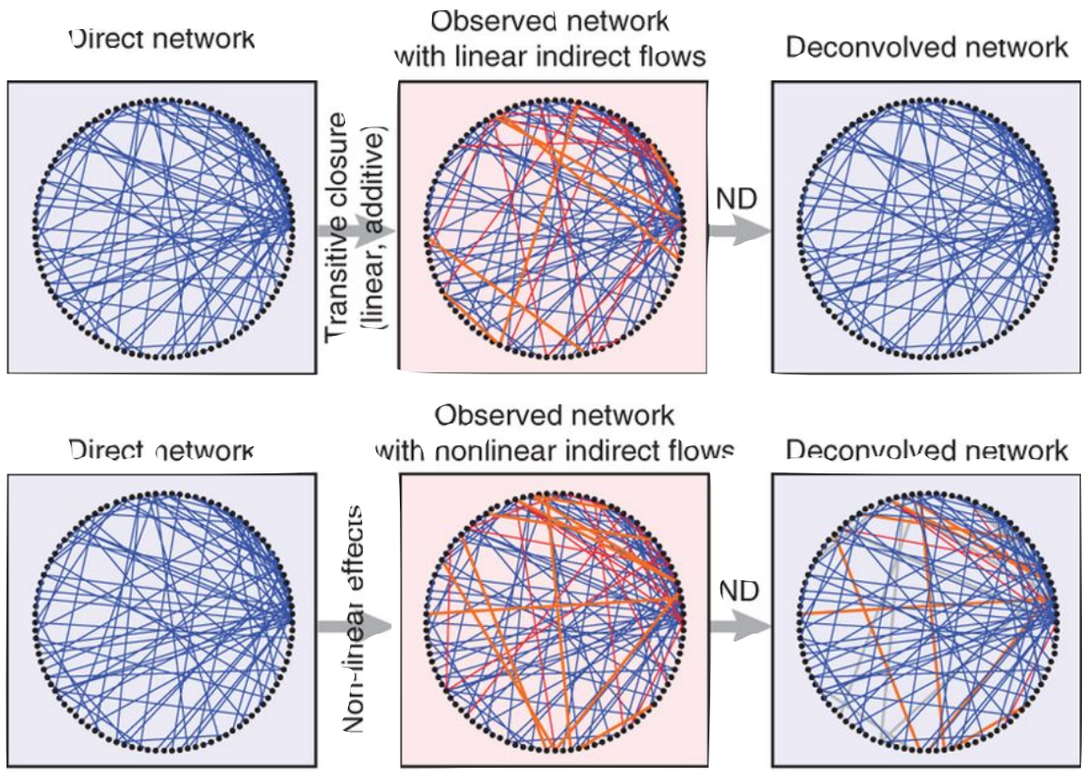
5. Network deconvolution

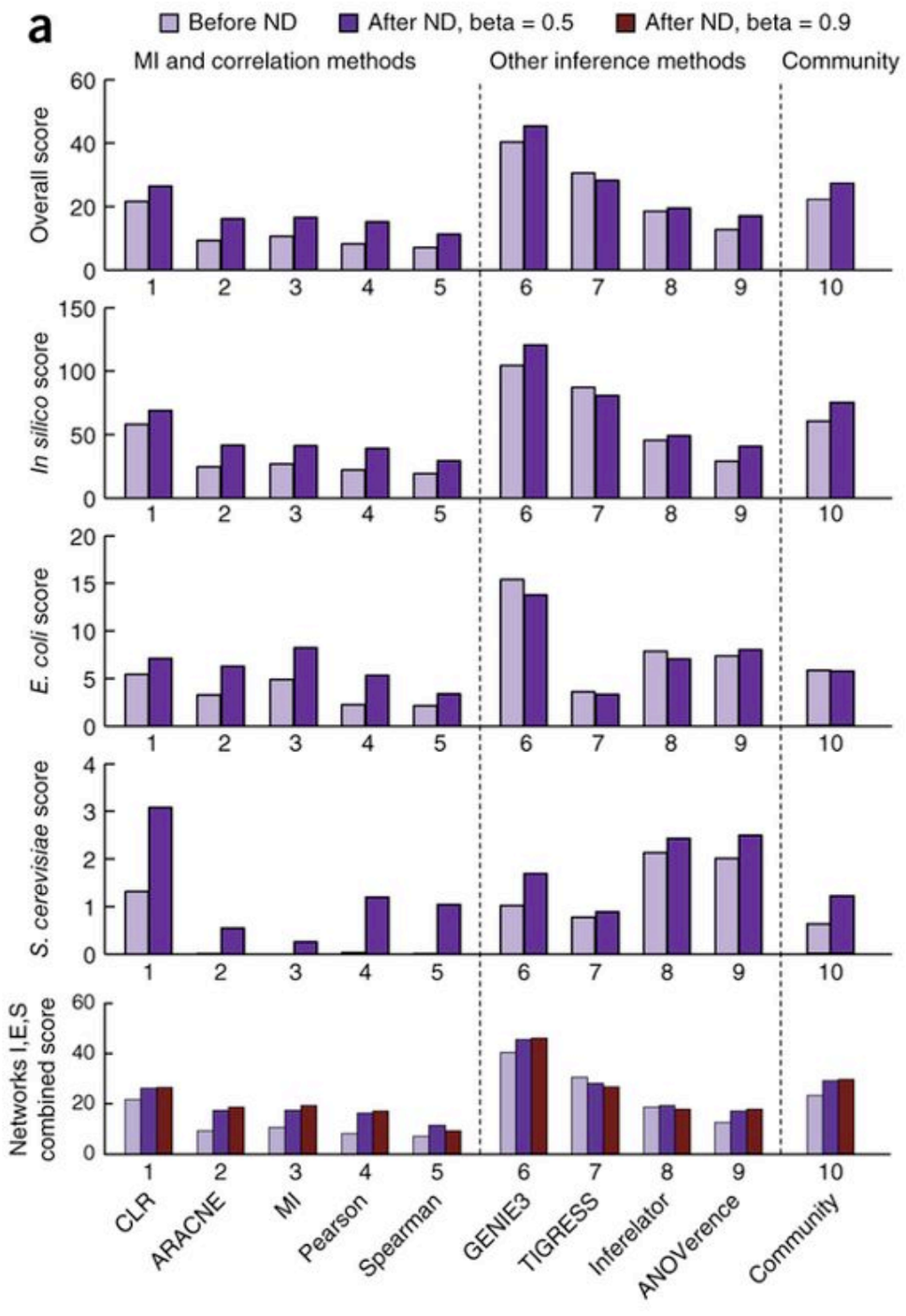
Nonlinear indirect effects are not always captured

May remove some direct interactions

Does not take into consideration edge weight

Still improve predictions (true edges: experimental)





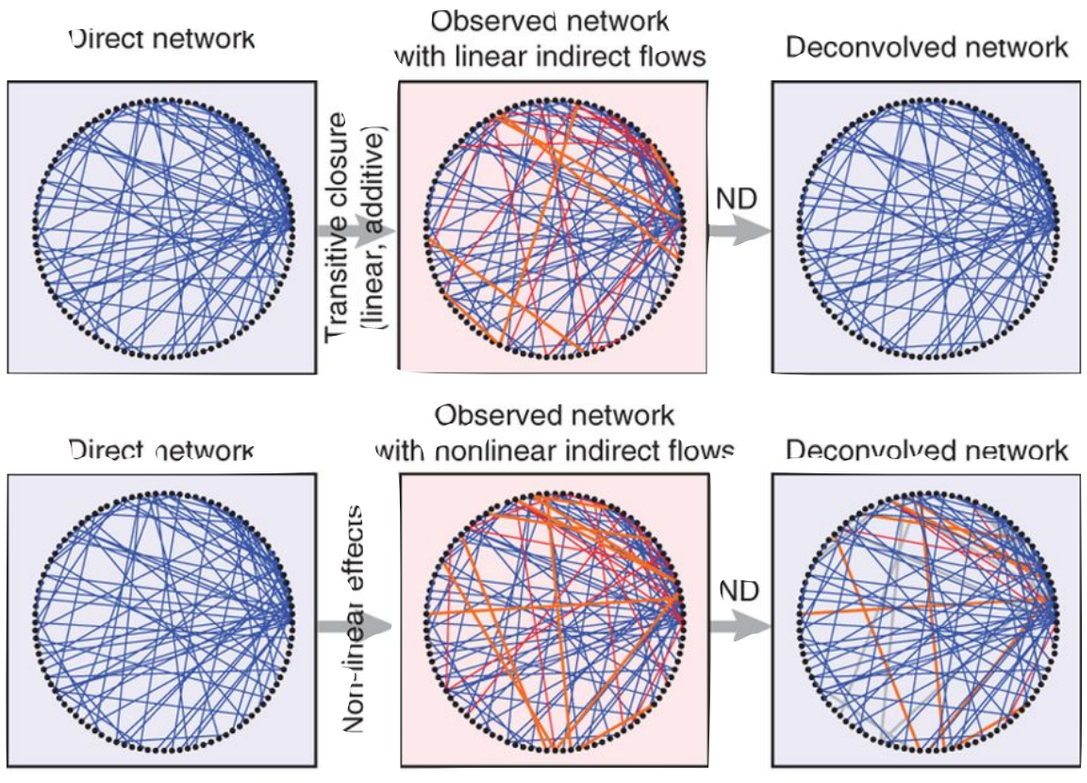
5. Network deconvolution

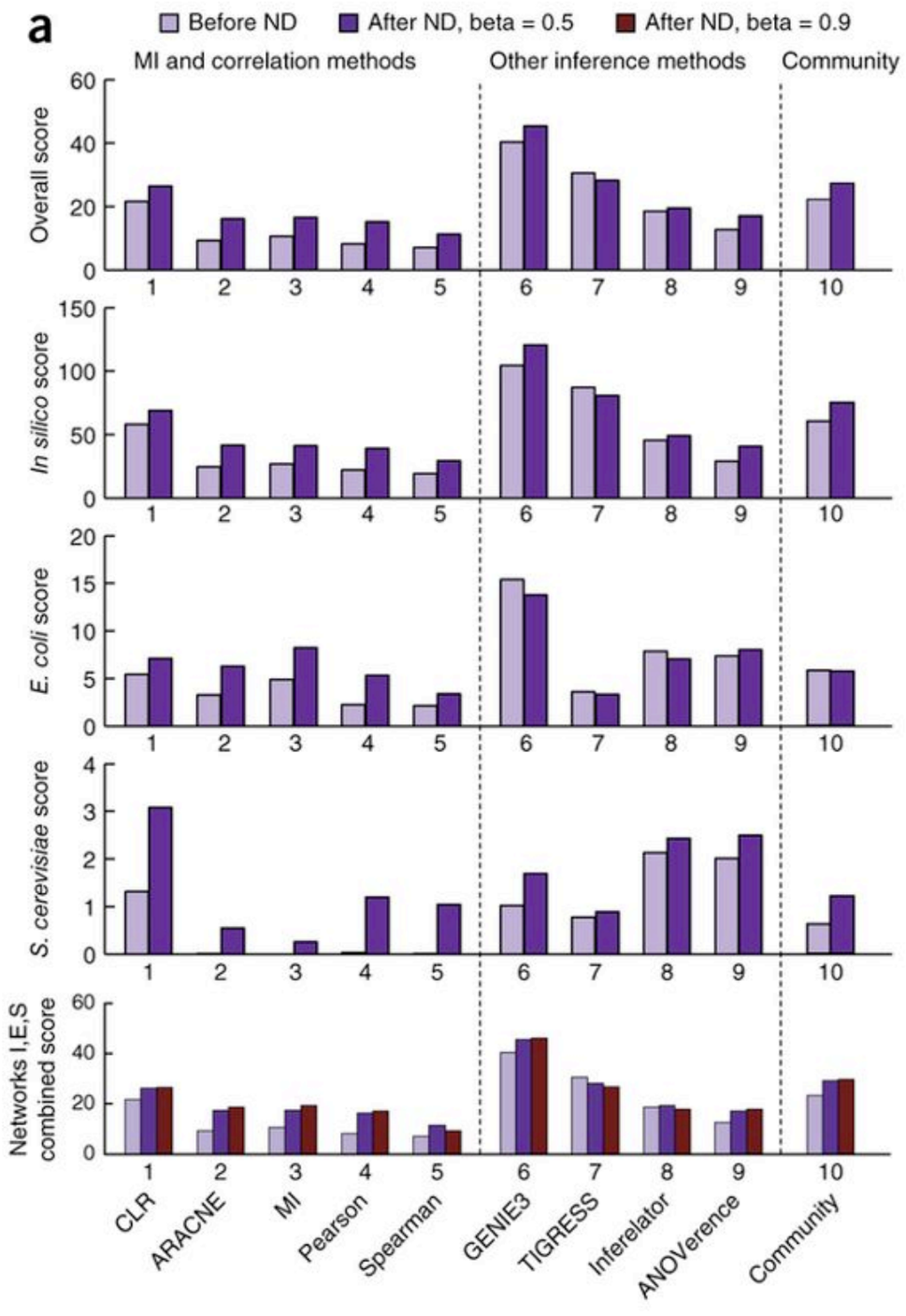
Nonlinear indirect effects are not always captured

May remove some direct interactions

Does not take into consideration edge weight

Still improve predictions (true edges: experimental)

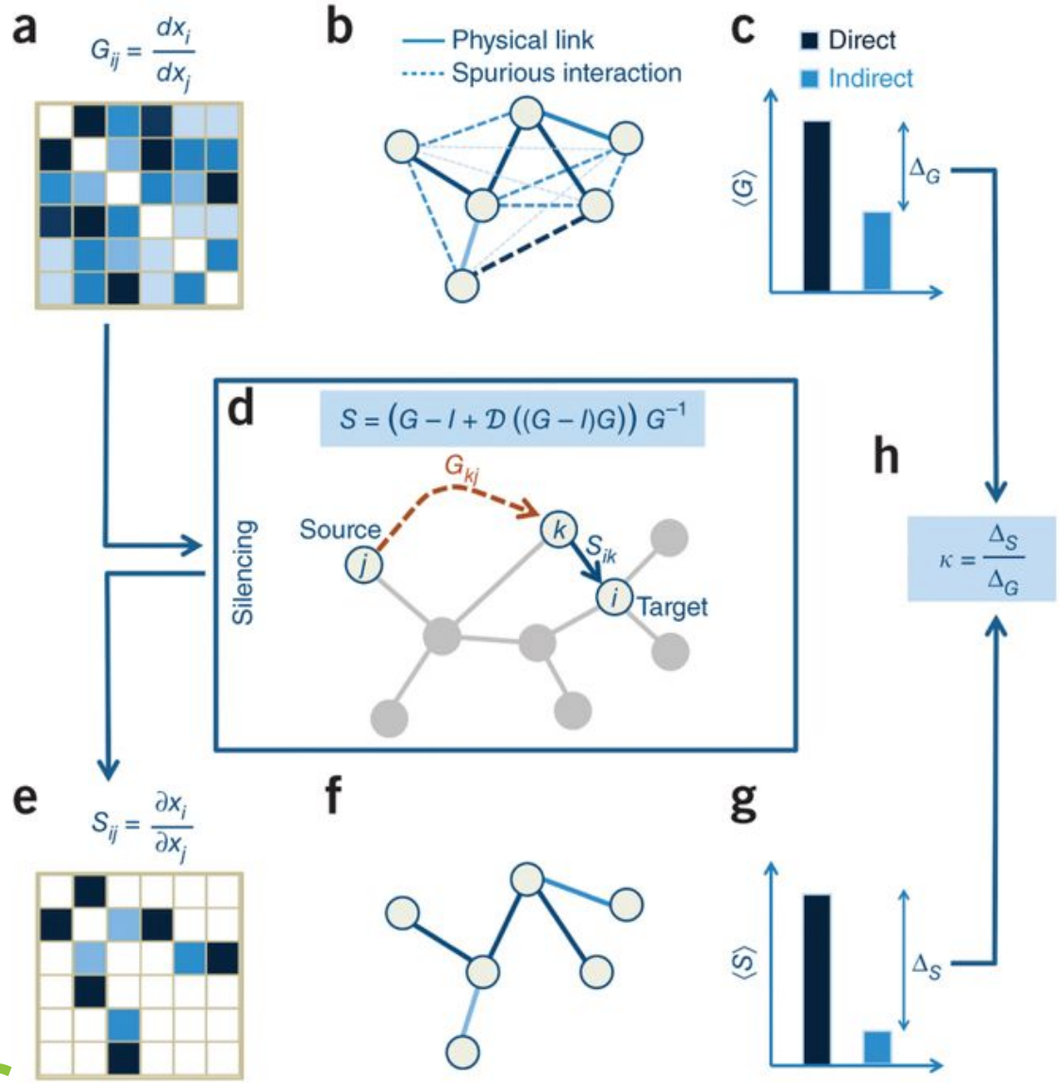


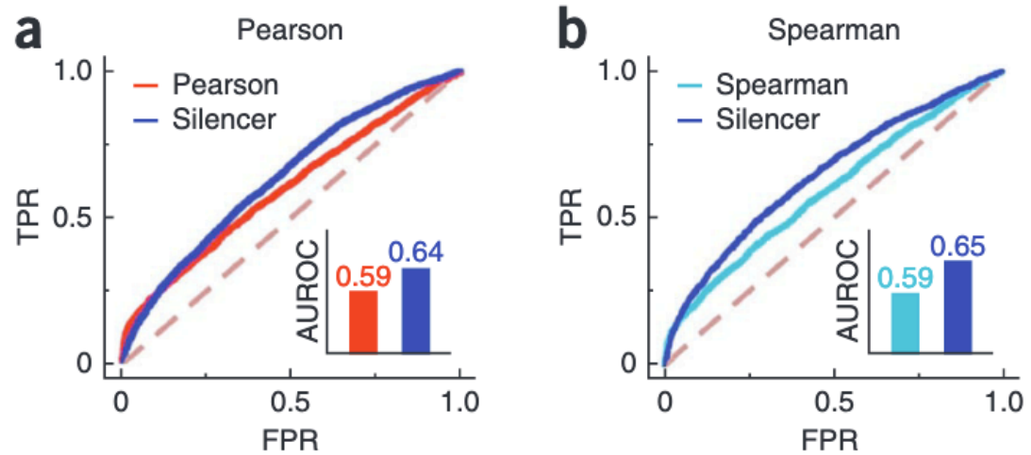


5. Network deconvolution

Similar approach involving perturbation of individual nodes to identify indirect interactions

Test case in *E. Coli* co-expression network (4511 nodes)



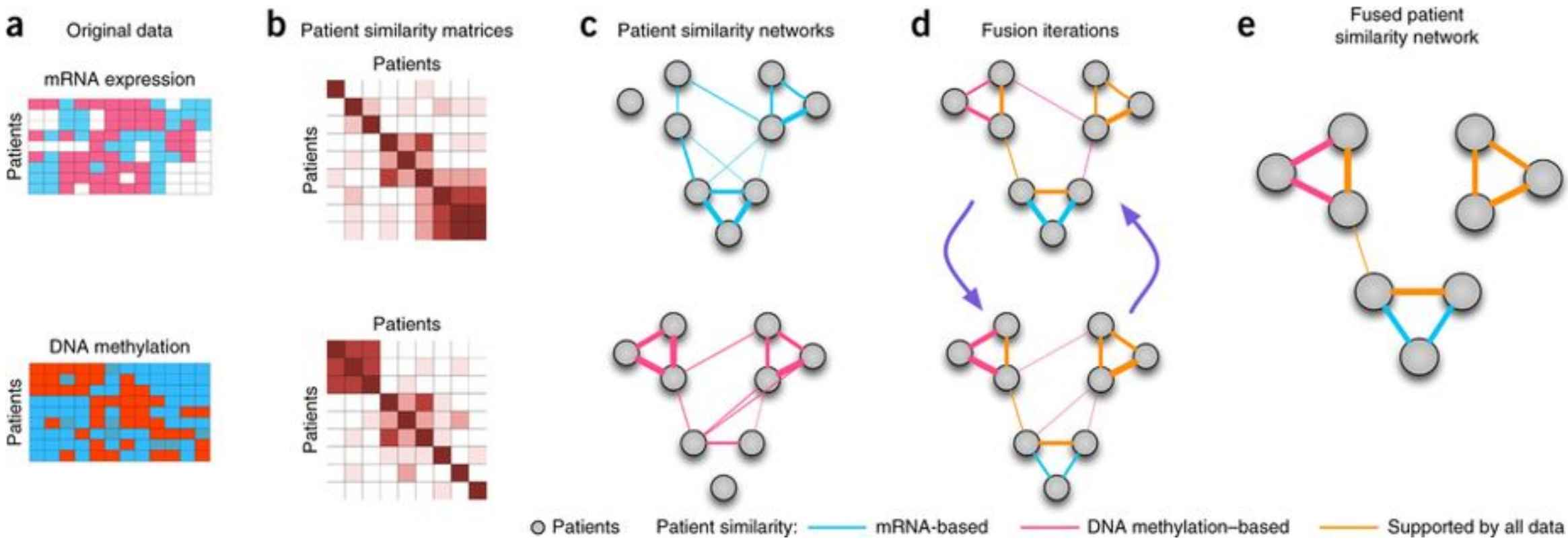


Barzel 2013

Bonus: similarity network fusion

Sample-sample clustering based on multi-comic data improves single-omics present complementary (non-redundant) information

Enables further comparisons between clusters



Additional reading

- Multi-omics approaches to disease - Introduction to how integrative approaches may be applied in disease

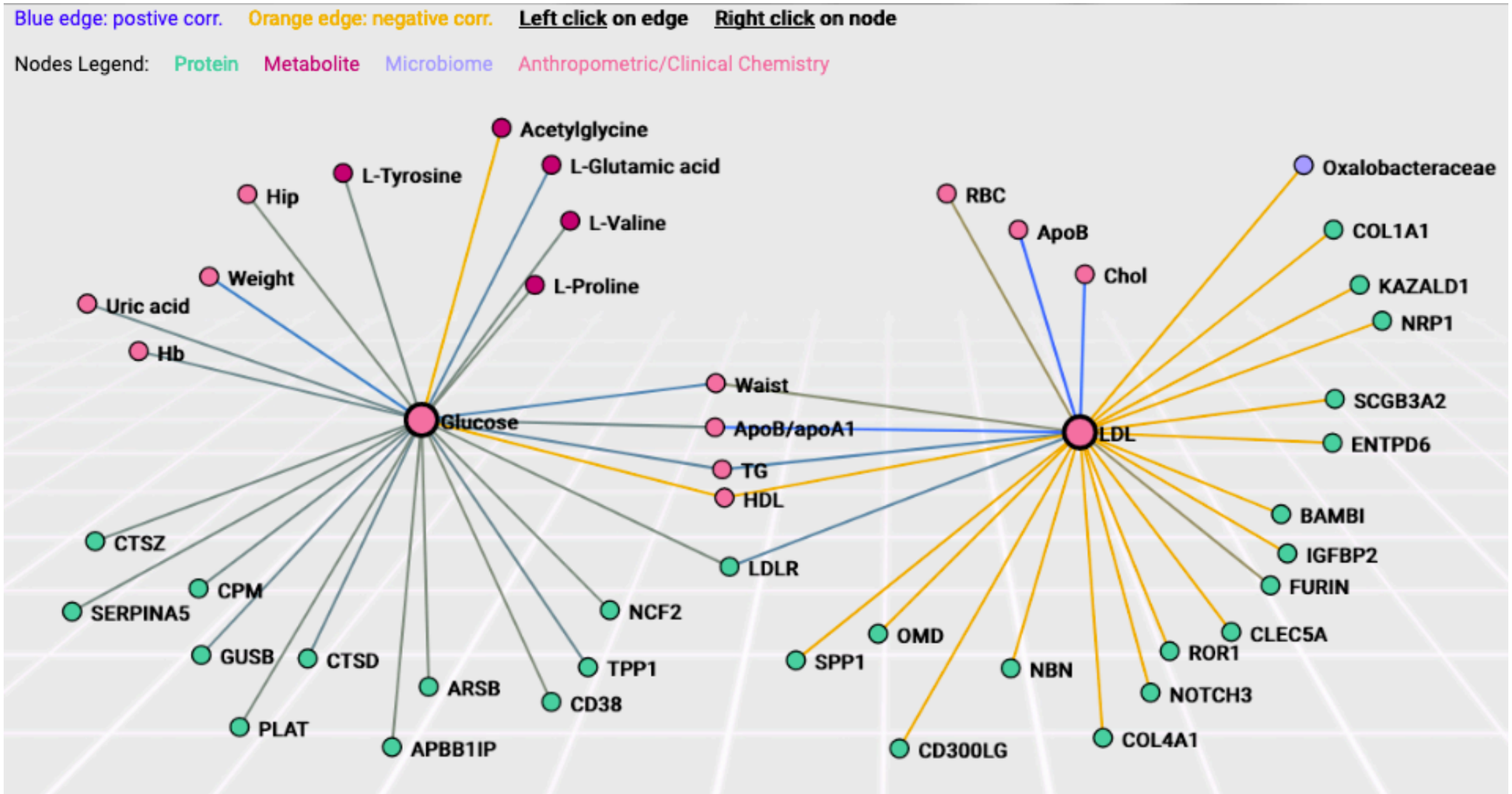
Additional references displayed as hyperlinks in each slide.

Key network properties

1. Introduction
2. Terminology
3. Network construction
4. **Key properties**
5. Community analysis
6. Visualization
7. Workshop

Motivation

You have built an association network (e.g. PPI, multi-omic, GEM-derived). How to identify pivotal features, their organization, and biological characteristics?



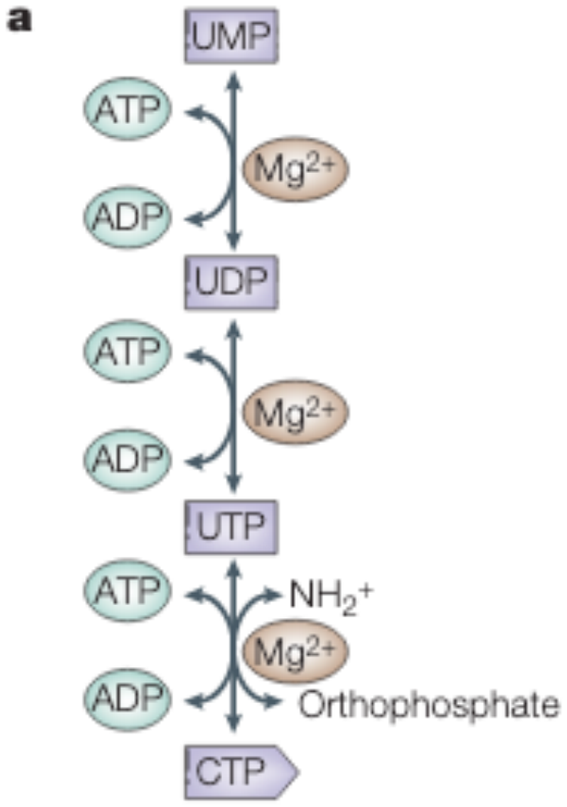
Key network properties to discuss

1. Network representations
2. Network density
3. Shortest path, mean path length and walks
4. Connectivity
5. Centrality
6. Clustering coefficient
7. Degree and connectivity distributions

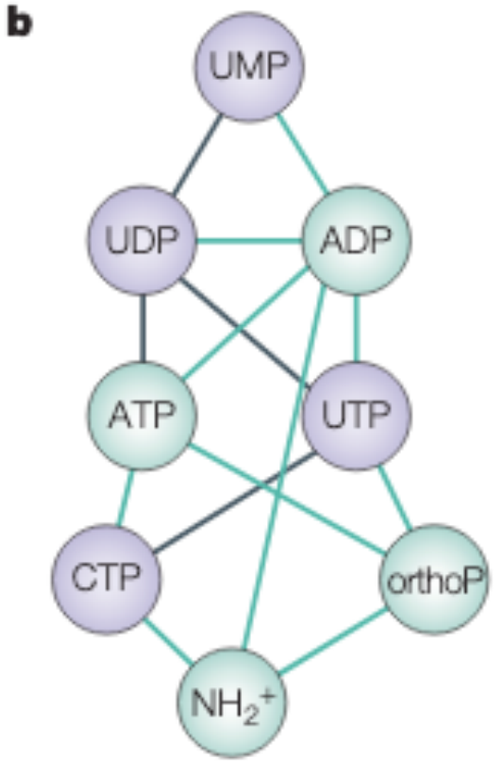
1. Network representations

Representations of a metabolic network: pyrimidine metabolism

Metabolism



Graph representation: metabolites and co-factors



metabolite-metabolite association

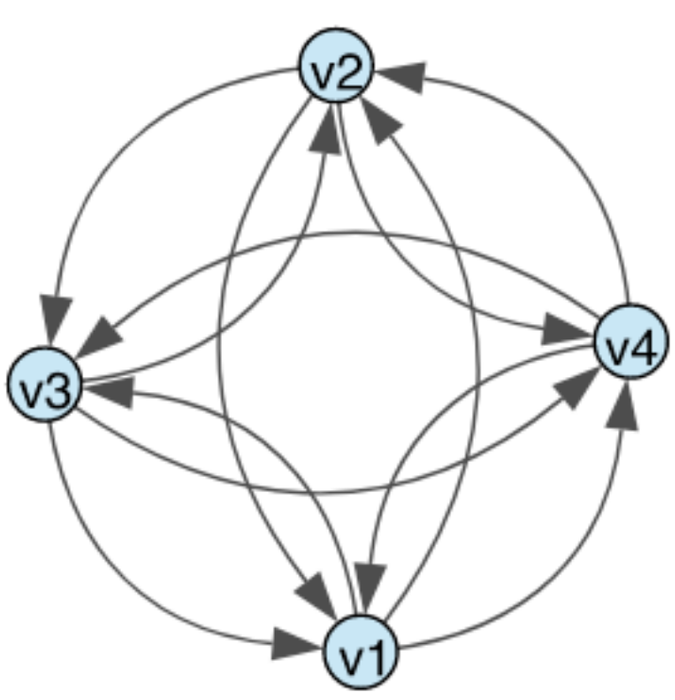
Other representations: Protein-Protein, Protein-Metabolite, …

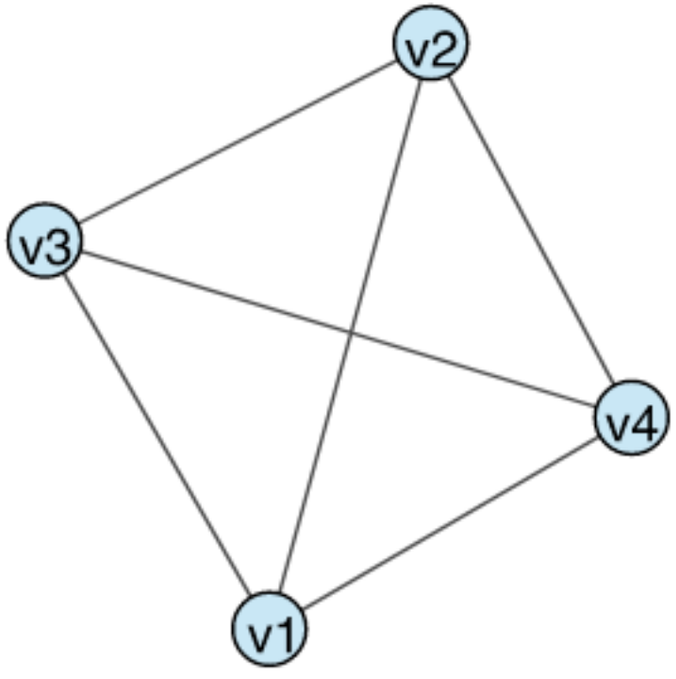
2. Network density

For a graph with *N* nodes, the total number of possible edges is given by

Directed graphs: $|N| \cdot (|N| - 1)$

Undirected graphs: $\frac{|N| \cdot (|N| - 1)}{2}$





2. Network density

A **dense graph** is a graph where the number of edges approximates the maximum possible number of edges for the given node number.

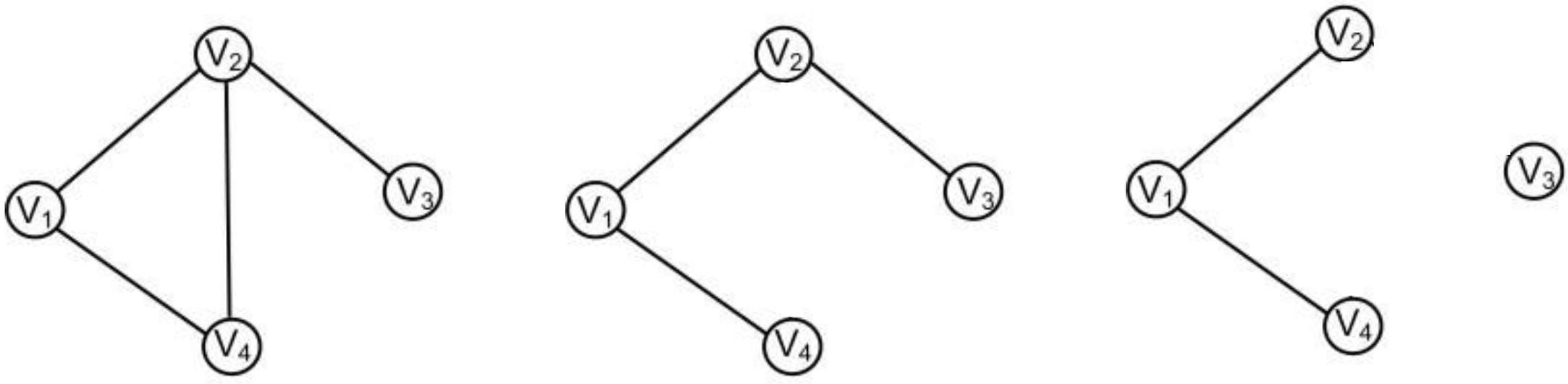
We can thus compute the network **density** (or **global connectivity**) as

Undirected graphs: $D = \frac{E}{V \cdot (V-1)}$

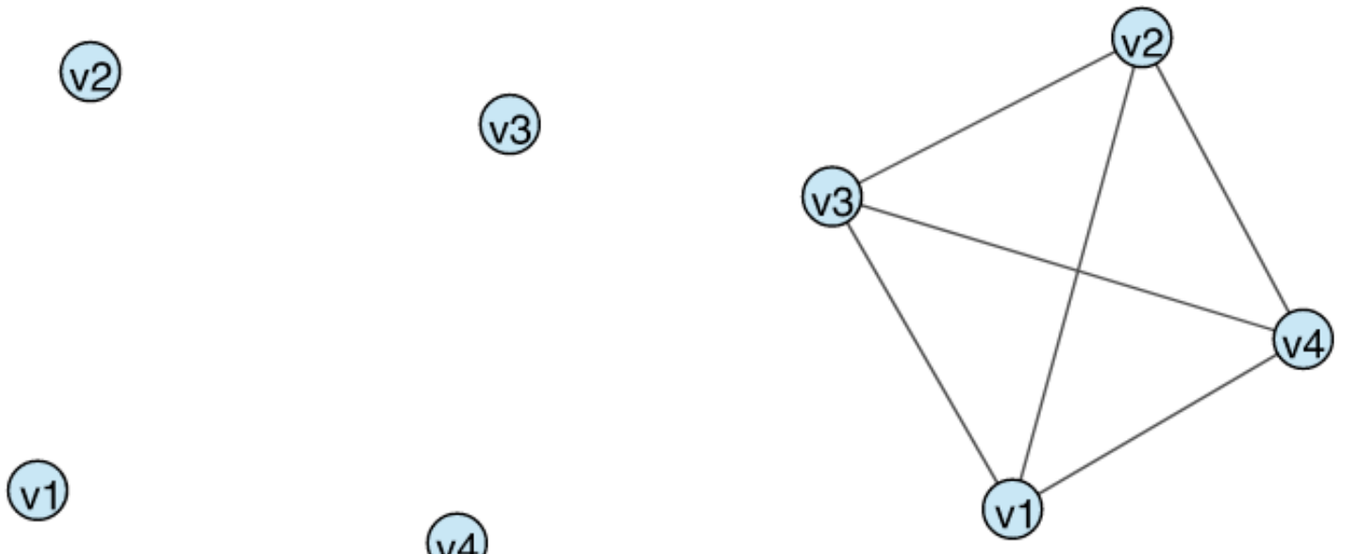
Directed graphs: $D = \frac{2 \cdot E}{V \cdot (V-1)}$

E : number of edges

V : number of vertices

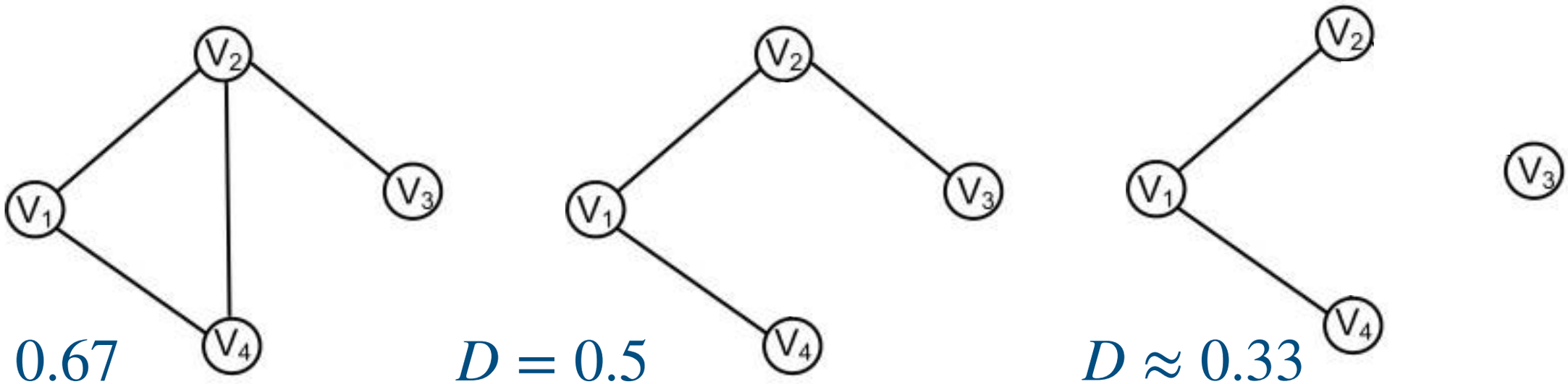


2. Network density



$$0 \leq D \leq 1$$

Higher density indicates higher associations in the network, which implies lower resilience to changes.



$D \approx 0.67$ $D = 0.5$ $D \approx 0.33$

2. Biological network density

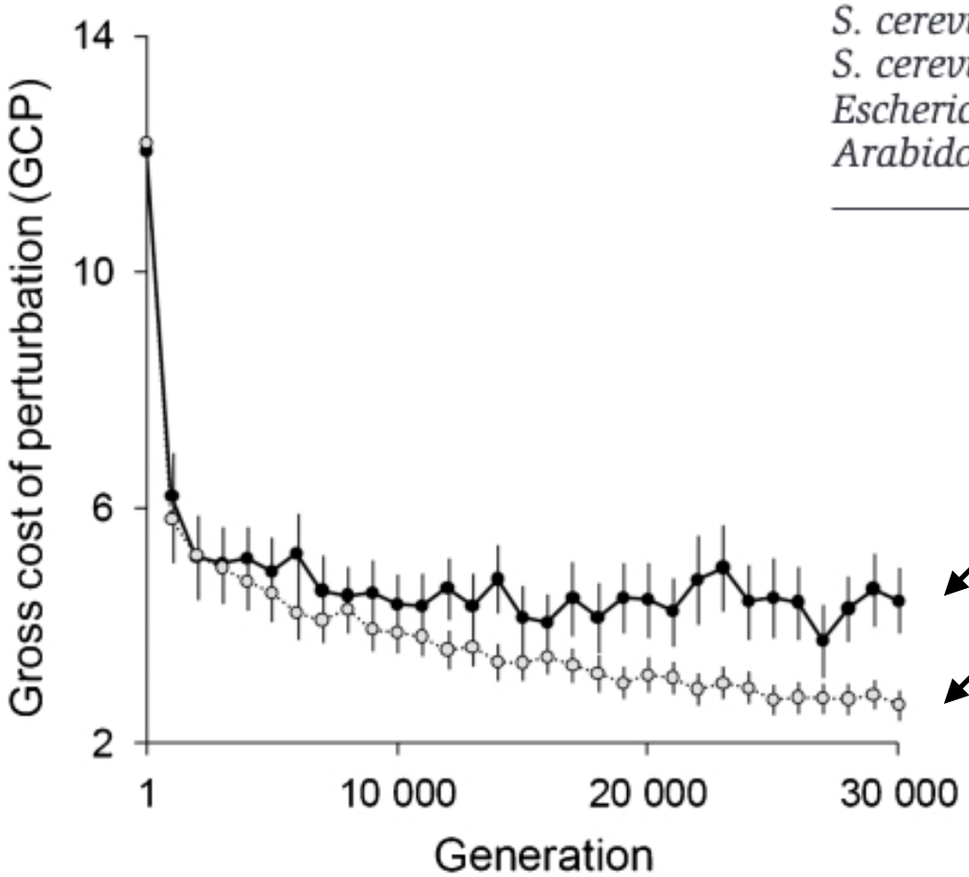
Evolutionary analysis of biological networks indicates general sparsity

Network structure must balance robustness to mutation, stochasticity and environmental queues

Sparse networks show higher robustness when accounting for costs and benefits of complexity

Table I Biological networks are sparsely connected

Organism	Interactions	Genes	D	K
Drosophila melanogaster	29	14	0.148	2.07
D. melanogaster	45	25	0.072	1.8
Sea urchin	82	44	0.0065	1.86
Saccharomyces cerevisiae	1052	678	0.0023	1.55
S. cerevisiae	3969	2341	0.0007	1.7
S. cerevisiae	106	56	*0.0338*	1.9
Escherichia coli[a]	578	423	0.0032	1.37
Arabidopsis thaliana[b]	18 625	6760	0.0004	2.75



dense network (density 0.9)

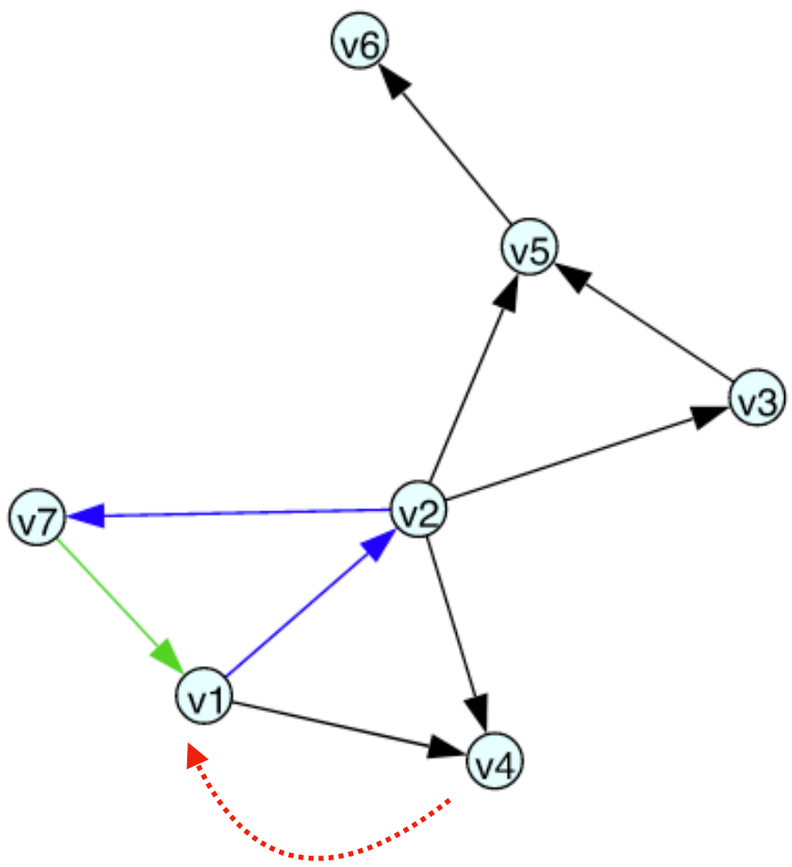
sparse network (density 0.1)

Leclerc 2008

3. Paths and walks

Distance is measured in path length, i.e. how many edges to walk between two nodes.

In directed graphs, the shortest path between $(a, b) \neq (b, a)$



	v1	v2	v4	v3	v5	v7	v6
v1	0.0	1.0	1.0	2.0	2.0	2.0	3.0
v2	2.0	0.0	1.0	1.0	1.0	1.0	2.0
v4	inf	inf	0.0	inf	inf	inf	inf
v3	inf	inf	inf	0.0	1.0	inf	2.0
v5	inf	inf	inf	inf	0.0	inf	1.0
v7	1.0	2.0	2.0	3.0	3.0	0.0	4.0
v6	inf	inf	inf	inf	inf	inf	0.0

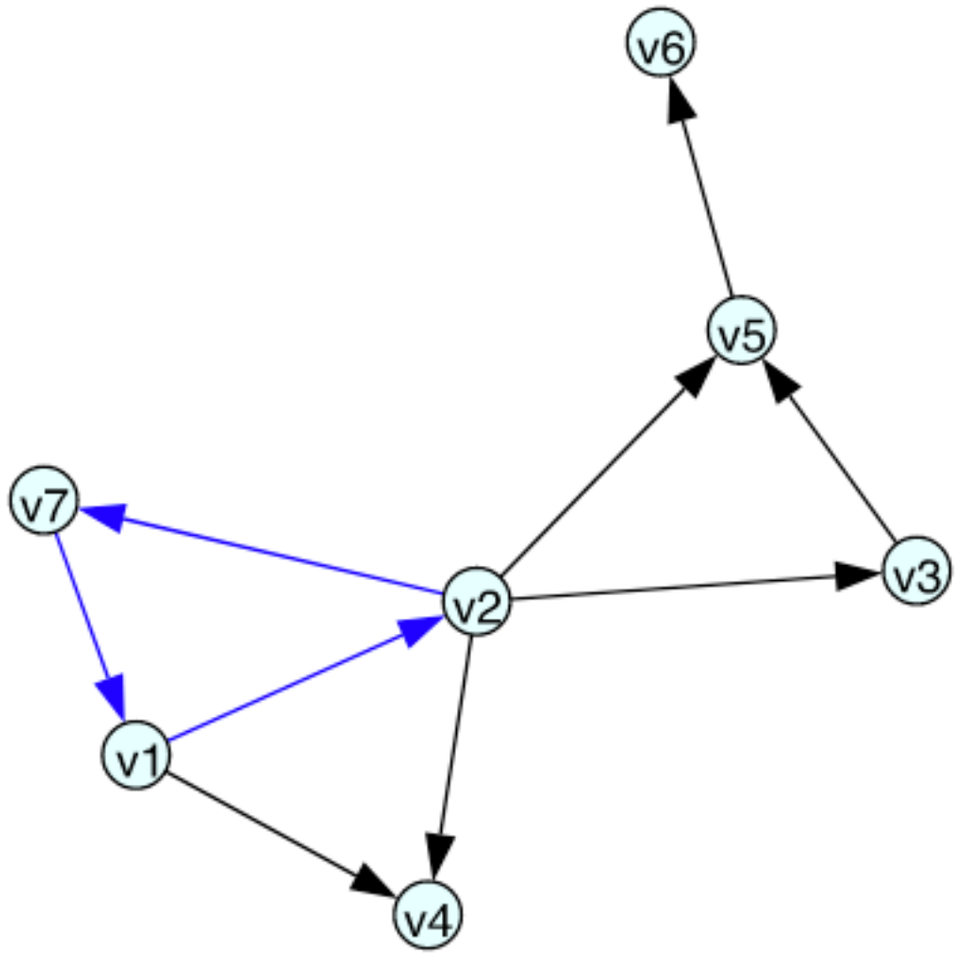
3. Paths and walks

Cycles and acyclic graphs

The **average path** gives a measure of network navigability

Algorithms are selected based on graph features (negative weights? Edge number)

- Breath First Search
- Dijkstra
- Bellman-Ford
- Floyd-Warshall
- Johnson's

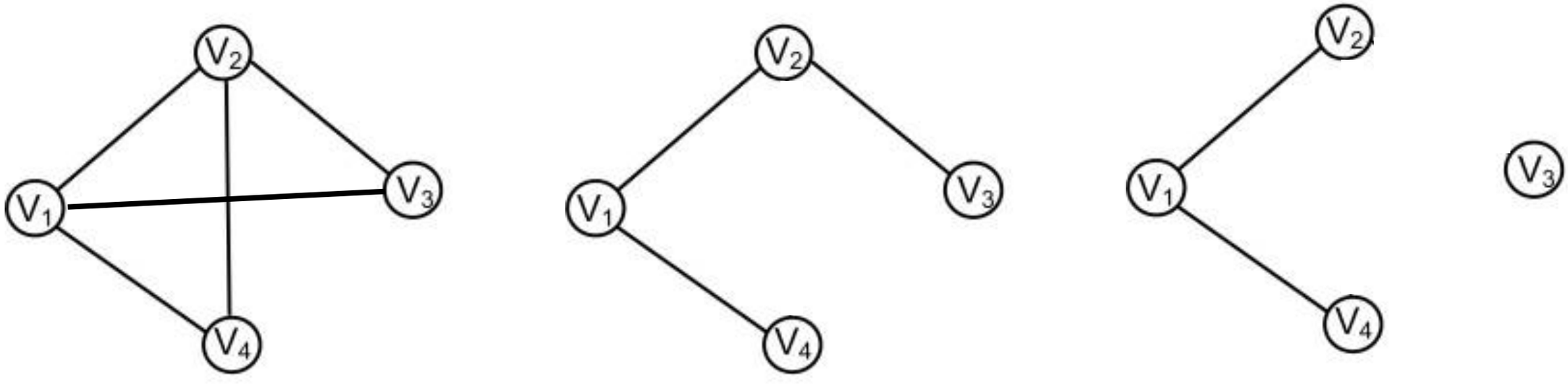


Average path length = 1.8

4. Connectivity

Node connectivity $\kappa(G)$: minimum number of **nodes** whose removal renders the network disconnected

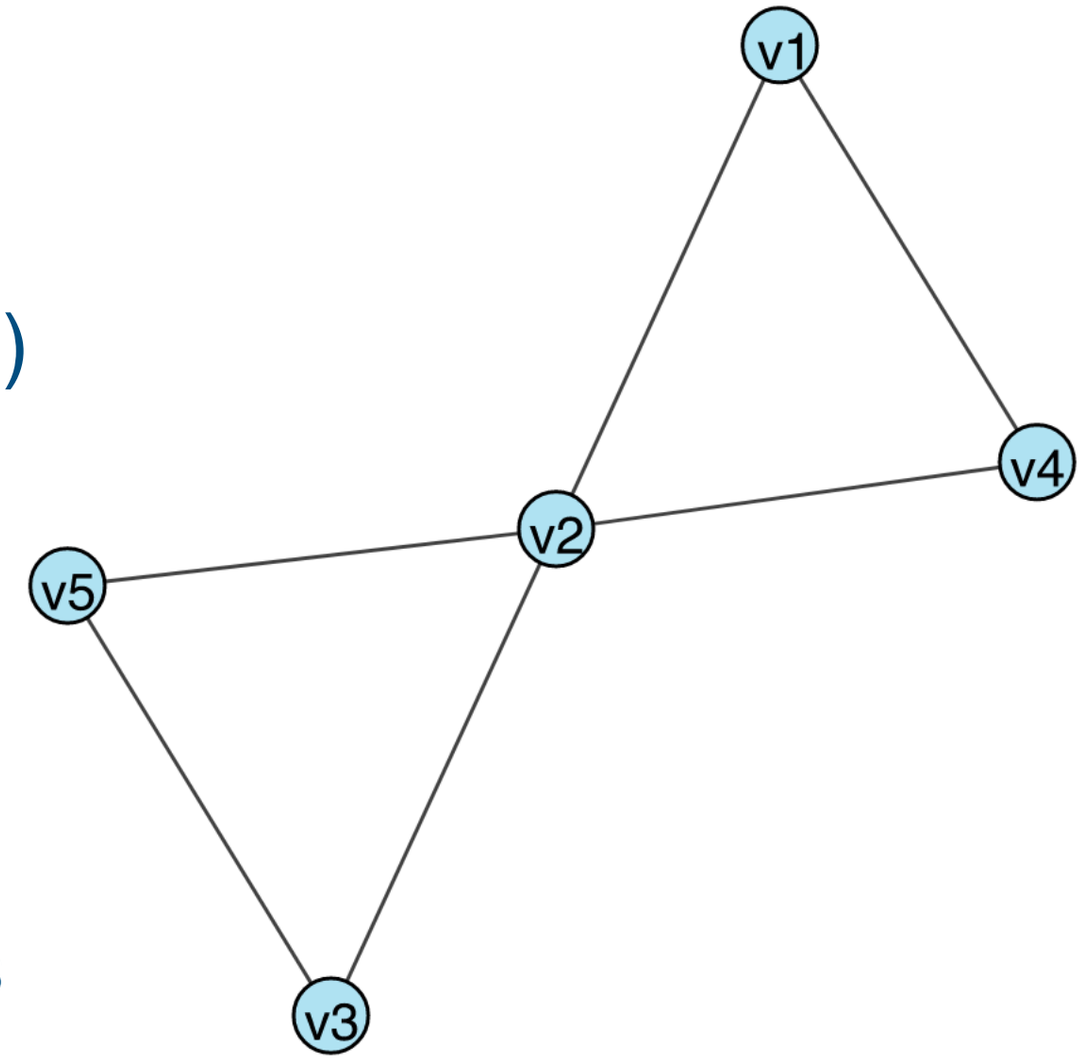
Edge connectivity $\lambda(G)$: minimum number of **edges** whose removal renders the network disconnected



4. Connectivity

$\kappa(G) = 1$; **cut:** v_2

$\lambda(G) = 2$; **bridge:** ((v_2 , v_1) & (v_2 , v_4))



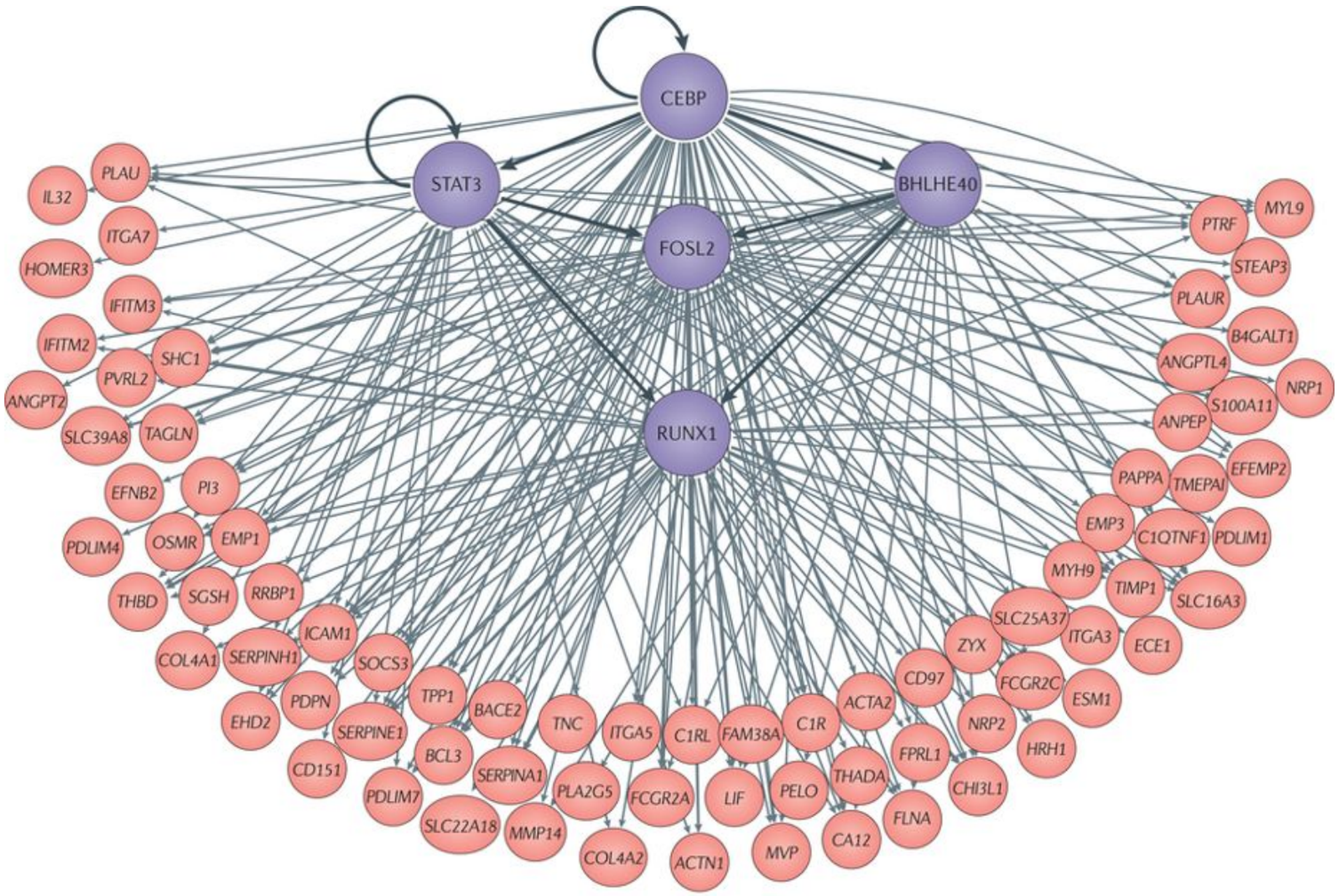
Local connectivity may also be computed for any given pair of vertices

5. Centrality

Indicate the most central nodes in a network

Central nodes **possibly** most important in the network, act as **hubs**

Example: Transcription Factor Master Regulators



Nature Reviews | Cancer

5. Centrality

Indicate the most central nodes in a network

Central nodes **possibly** most important in the network

There are many different measures of centrality:

- **Degree**
- **Eccentricity**
- Closeness
- Betweenness
- Eigenvector
- Katz
- PageRank
- Percolation
- Cross-clique

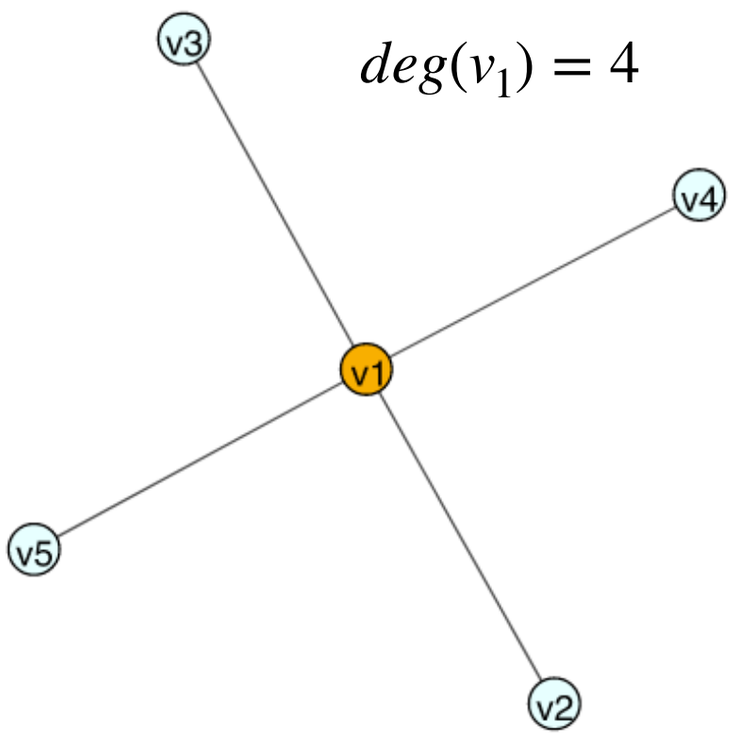
…

5. Centrality: degree and degree centrality

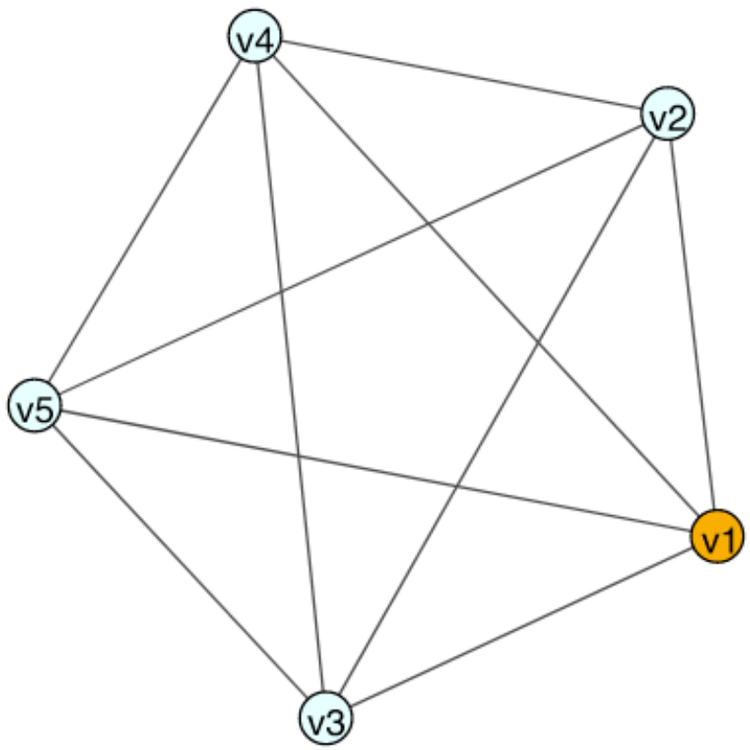
Degree indicates the number of connections with a node

$$d(v) = |N(i)|$$

where *N(i)* is the number of 1st neighbours of a node.



$deg(v_1) = 4$



$deg(v_1) = 4$

5. Centrality: degree and degree centrality

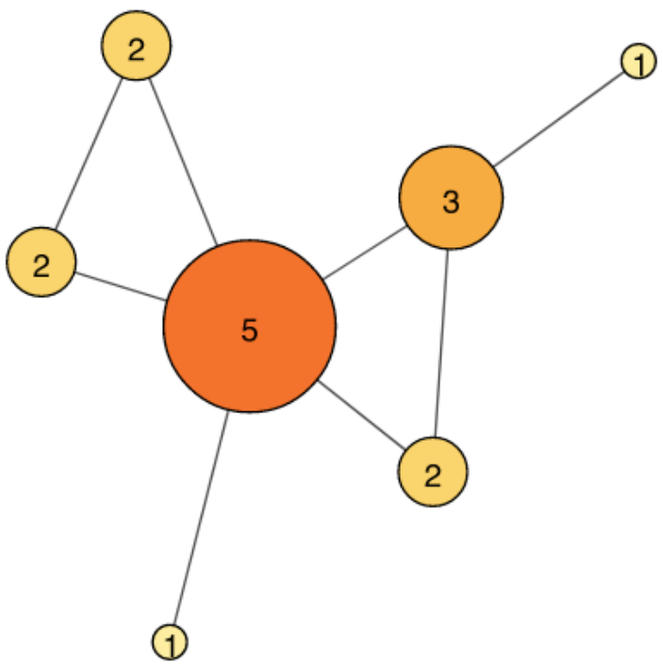
Undirected networks vs directed networks

In-degree vs **Out-degree**

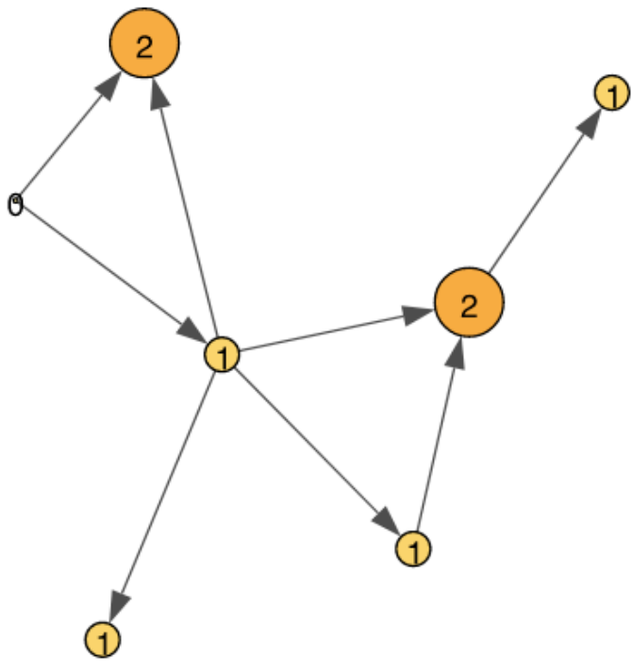
$$C_D(v_i) = \sum_{j=1}^{N} e_{ij}$$

Numbers indicate degree:

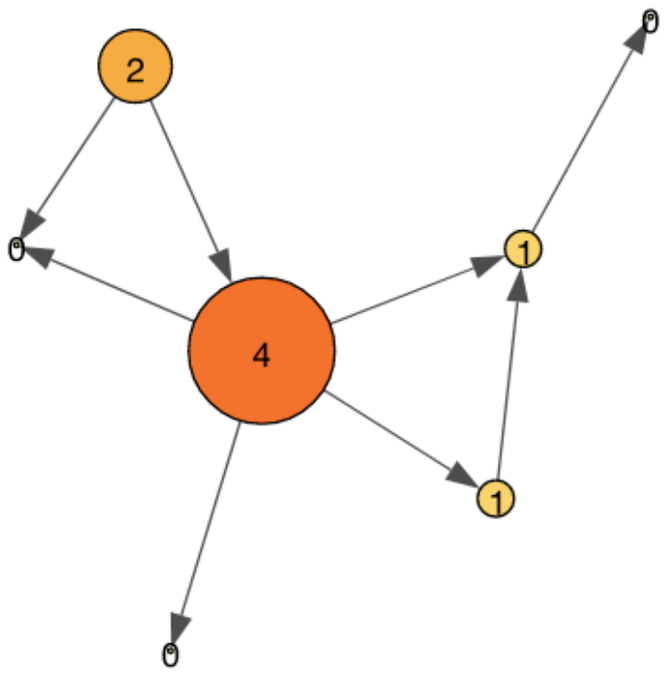
Undirected



In-degree



Out-degree



5. Centrality: degree and degree centrality

Degree centrality

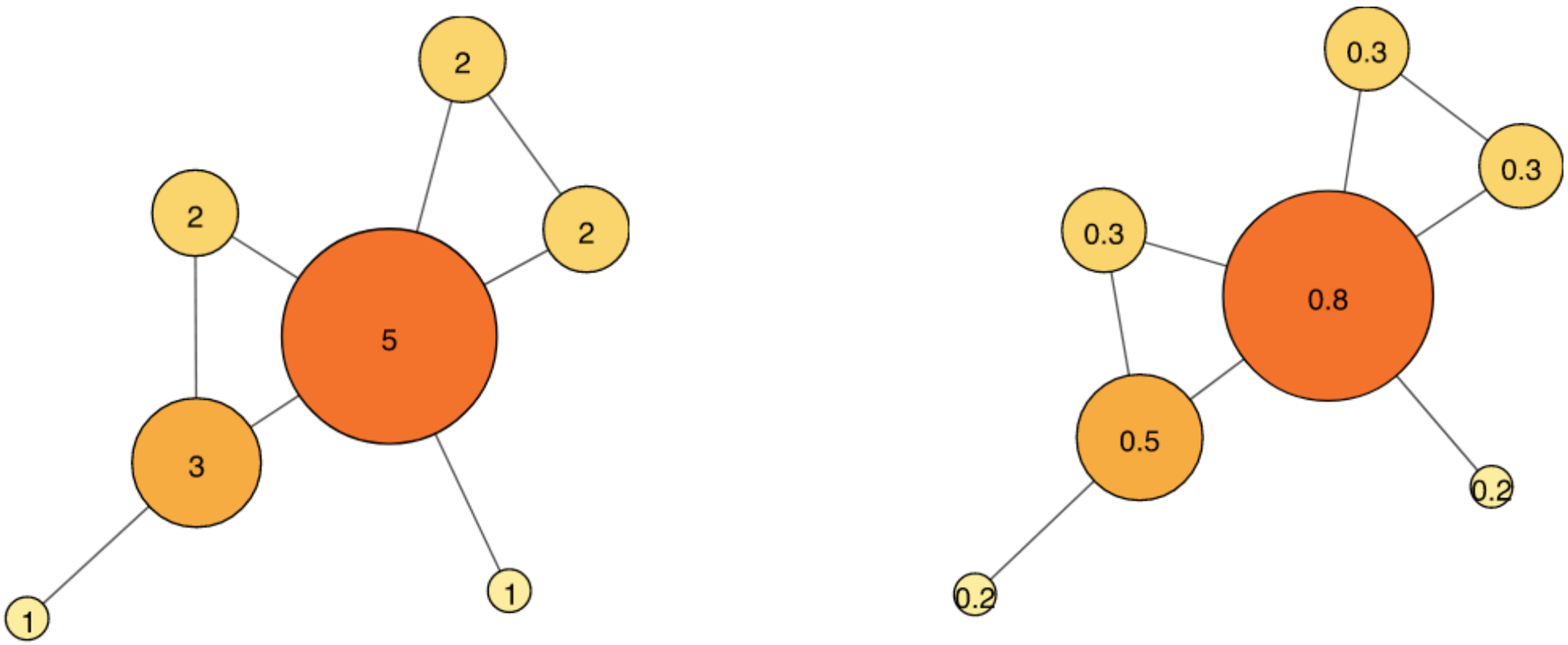
$$C_D(v_i) = \sum_{j=1}^{N} e_{ij}$$

Normalized degree centrality

$$C_D(v_i) = \frac{\sum_{j=1}^{N} e_{ij}}{N-1}$$

Normalized degree centrality accounts for the total possible number of connections

Centrality normalization allows for comparison between networks of different sizes



5. Centrality: eccentricity centrality

Network Diameter is the maximum distance

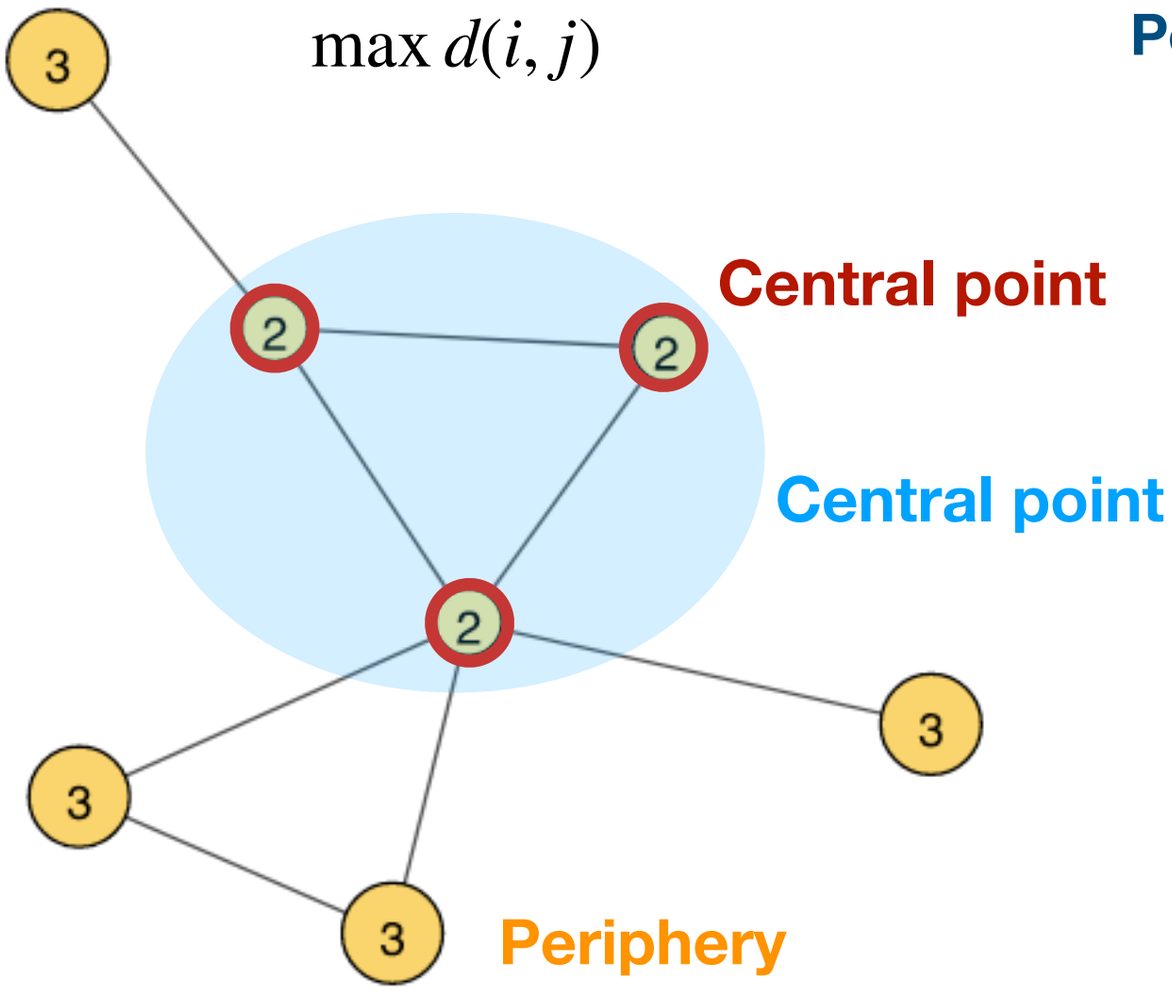
Radius is the minimum distance

Central point is that which has $d(i, j) =$ radius

Graph center: set of nodes with $d(i, j) =$ radius

Periphery: set of nodes with $d(i, j) =$ diameter

$$\max d(i, j)$$



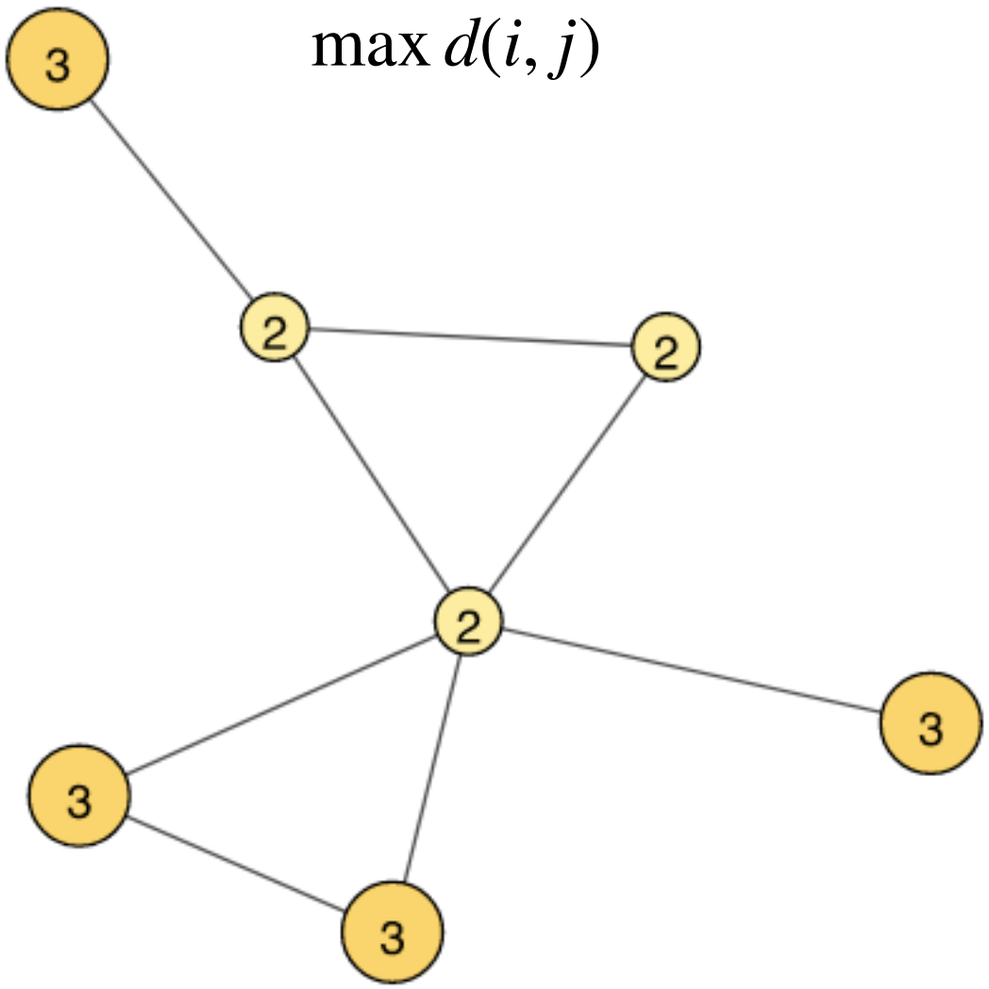
Diameter = 3

Radius = 2

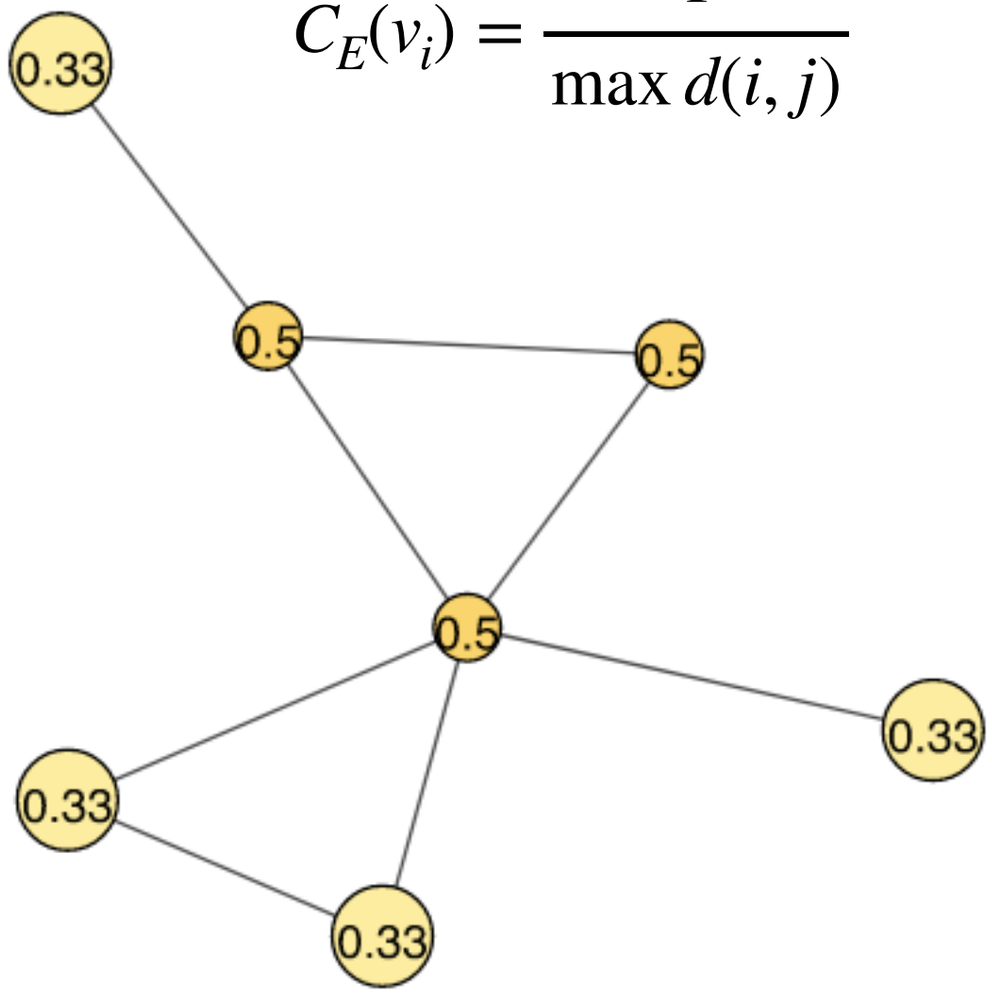
5. Centrality: eccentricity centrality

Eccentricity considers a node's max path to all other nodes

$$\max d(i, j)$$



$$C_E(v_i) = \frac{1}{\max d(i, j)}$$



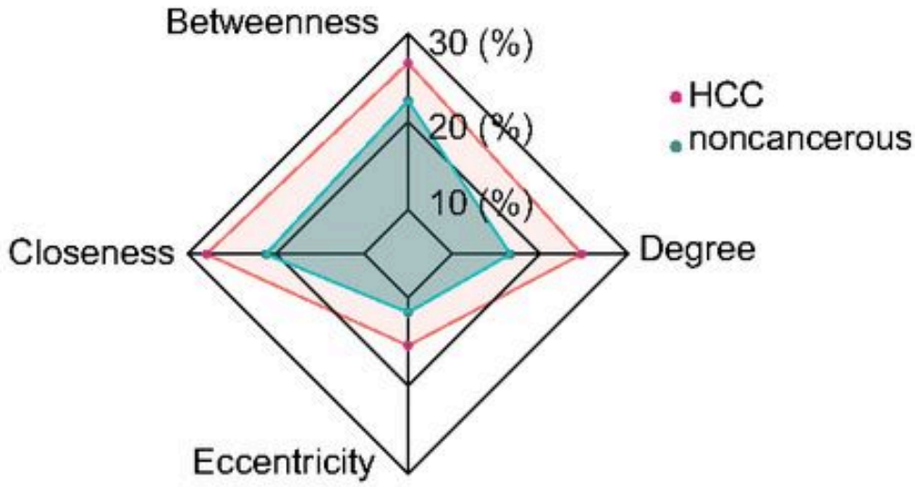
5. Centrality: limitations & influence

Node centrality does not necessarily imply **importance**

How to tackle this?

1. Complement with experimental observations
2. Compute multiple metrics and summarise joint observations
3. Compute node **influence,** modifications of centrality

- **Accessibility**
- **Dynamic influence**
- **Impact**
- **Expected force**

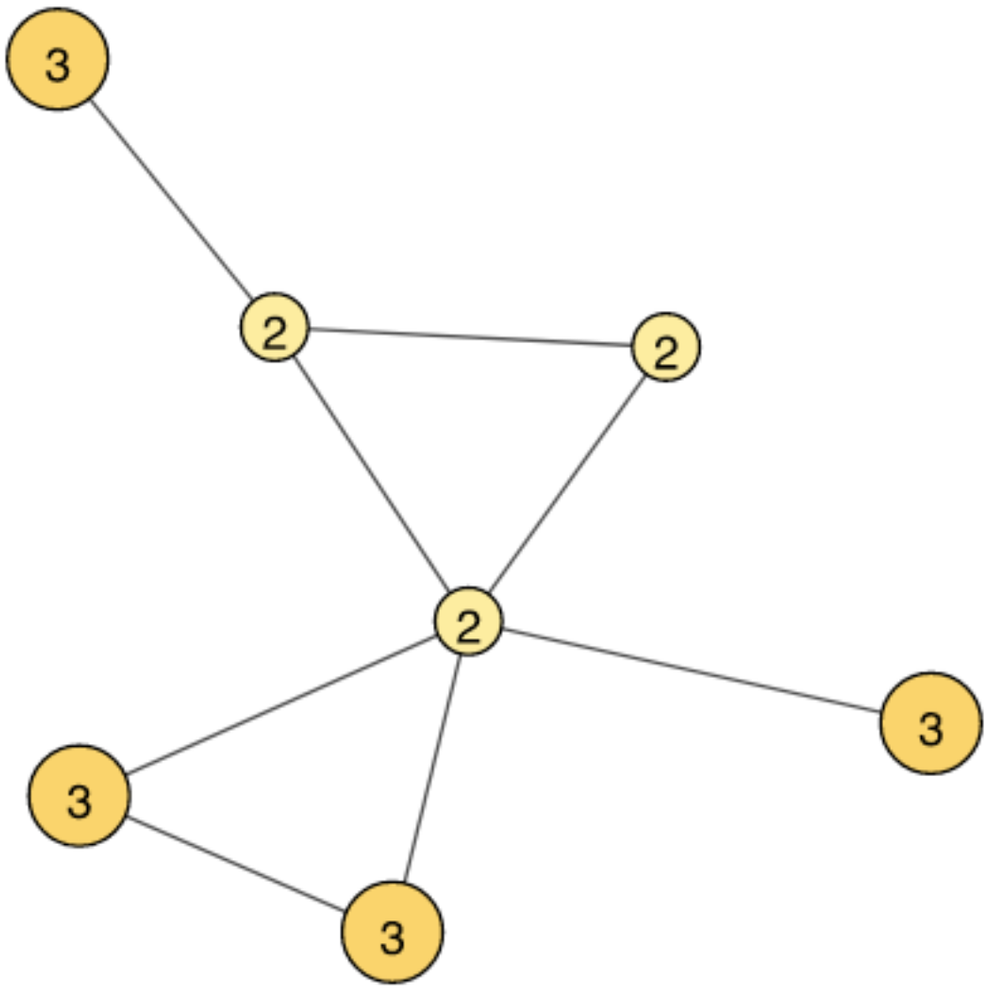


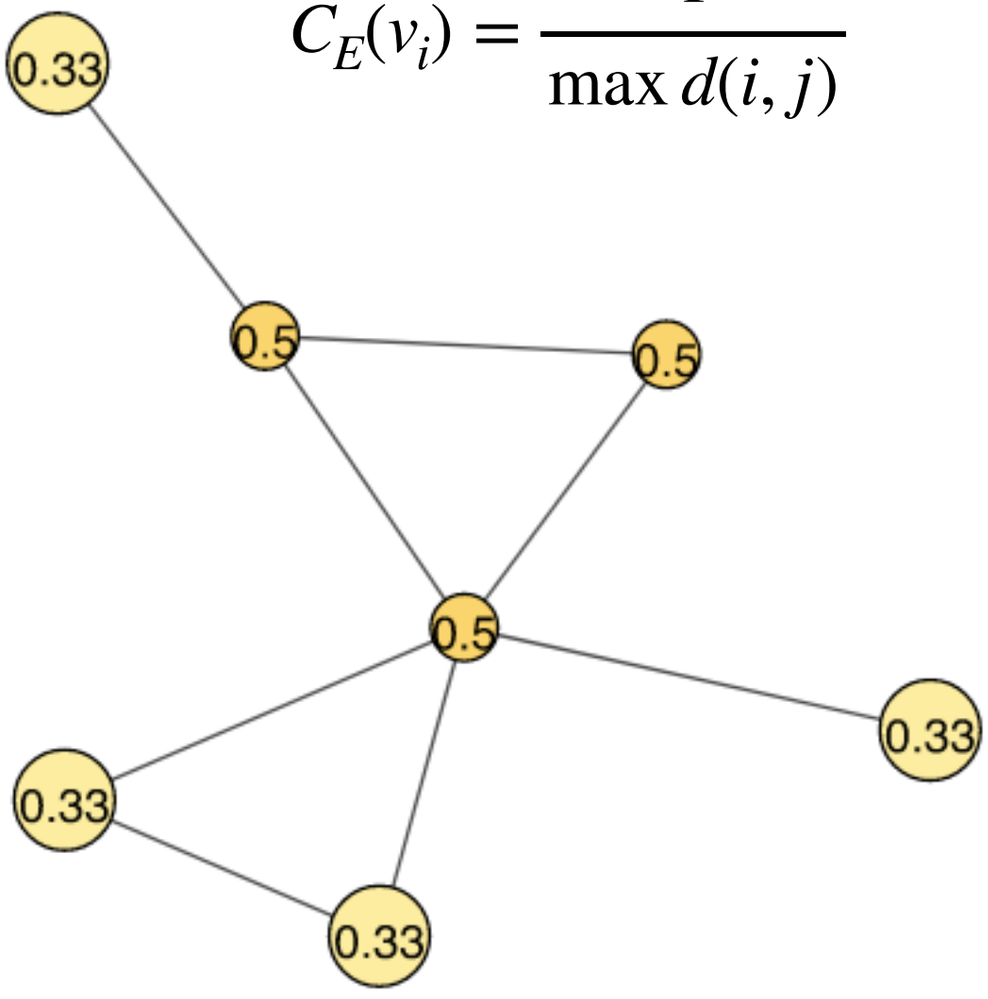
Measure **information transmission** rather than *connectiveness*

Bidkhori 2018

Bonus: node influence

Centrality does not imply import





$$C_E(v_i) = \frac{1}{\max d(i,j)}$$

6. Clustering coefficient

How likely is it that two connected nodes are part of a highly connected group of nodes?

If node v_1 is connected with v_2 and v_3, it is very likely that v_2 and v_3 are also connected.

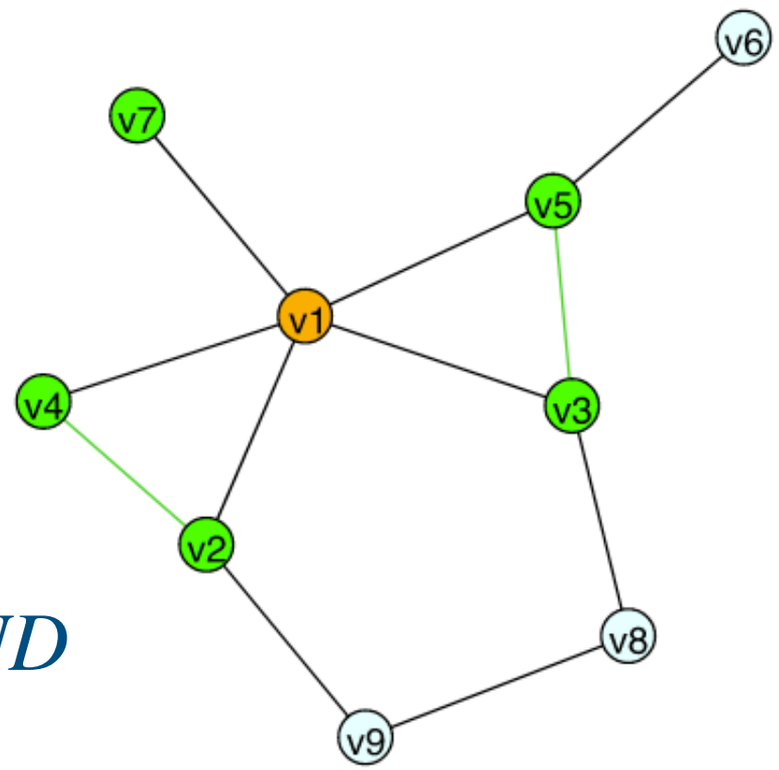
Takes into account degree of a node and the degree of its 1st neighbours

For node v_2

- $deg(v_1) = k = 4$
- n connections between 1st neighbours of $v1 = 2$

$$C_i = \frac{2 \cdot n}{k \cdot (k-1)}$$

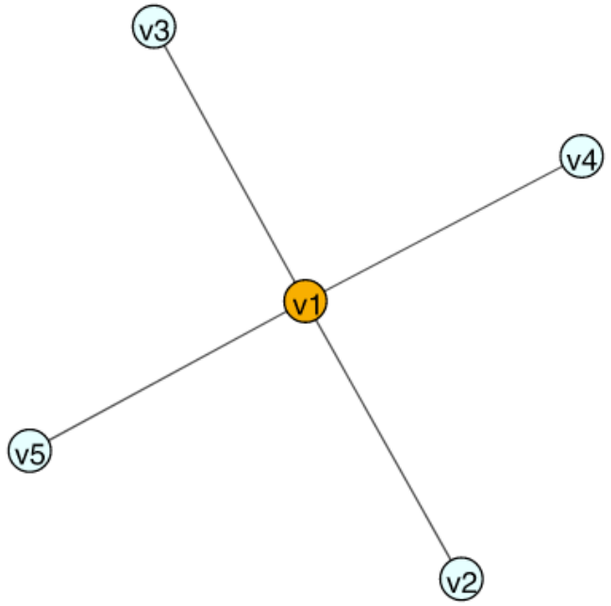
$$C(v_1) = \frac{2 \cdot 2}{5 \cdot 4} = 0.2 \qquad C(v_7) = \frac{2 \cdot 0}{1 \cdot 0} = 0 \text{ or } ND$$



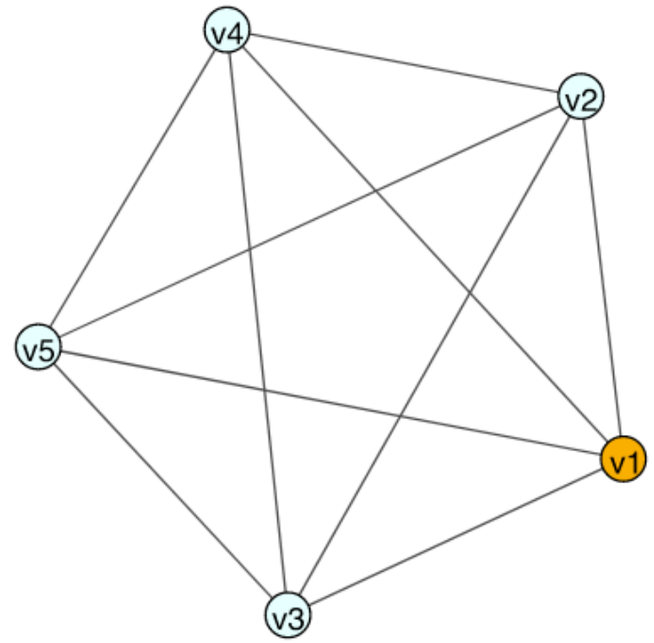
6. Clustering coefficient

$C_i = \frac{2 \cdot n}{k \cdot (k-1)}$ gives the **fraction of possible interconnections** for neighbours of node i

where $\frac{k \cdot (k-1)}{2}$ is the maximum number of connections for k nodes



$$0 \leq C_i \leq 1$$



The global clustering coefficient $C(G)$ is simply the average of its clustering coefficients

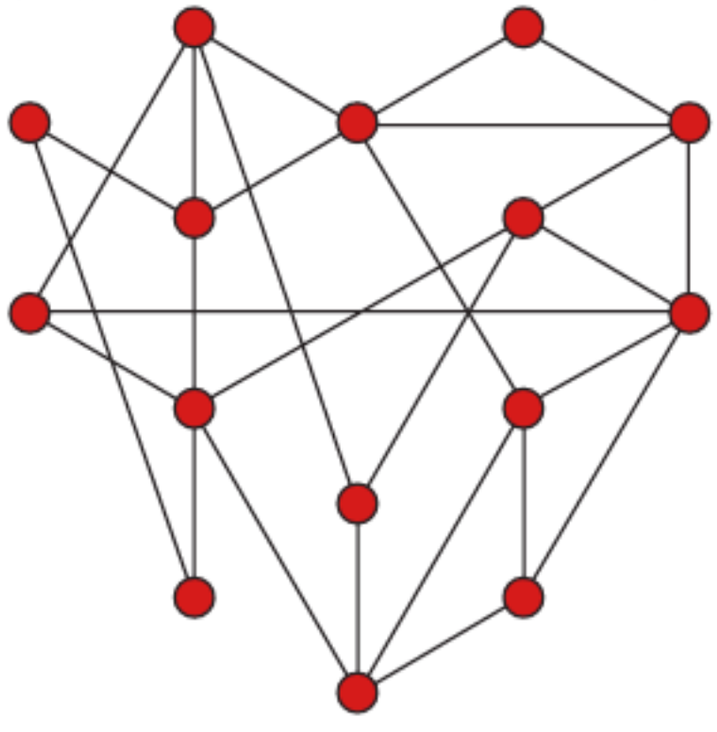
7. Degree and clustering coefficient distribution

P(k) gives the probability that a selected node has exactly *k* edges

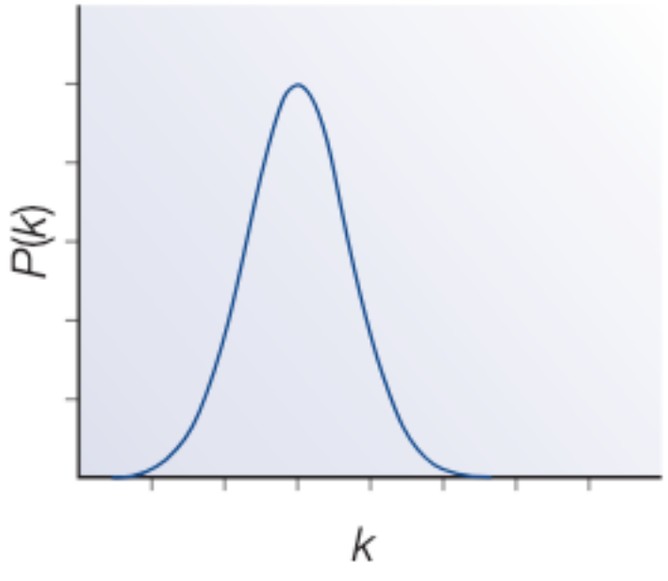
Obtained by counting number of nodes *N(k)*, for *k* = 1, 2, …

Allows distinguishing different kinds of networks

Random network
(e.g. Erdös-Rényi model)



Poisson degree distribution
shows no highly connected nodes

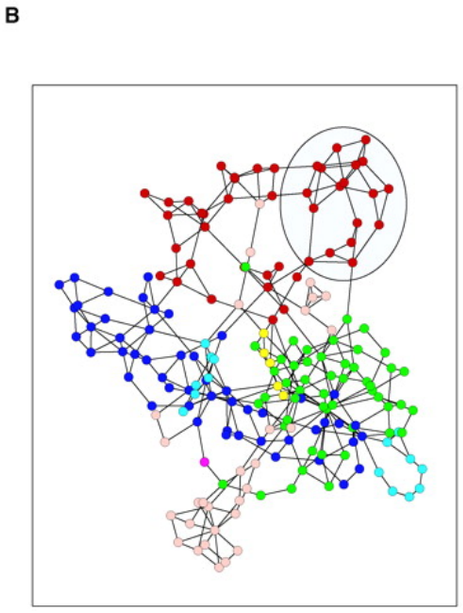


Most nodes have near <*k*>

Metabolic networks show hierarchical topology

Metabolic networks of 43 organisms are organised into
small, tightly connected modules

Their combination shows a hierarchical structure



Ravasz 2002

7. Degree distribution

Biological networks do not follow topology features of random networks.

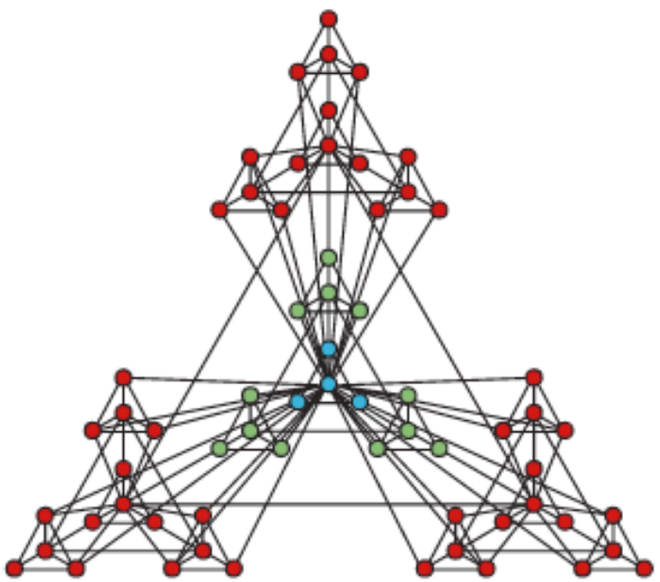
Analysis of metabolic networks of 43 organisms shows common patterns

Degree distribution *follows* the power-law $P(k) \propto k^{-\gamma}$, and is termed **scale-free**

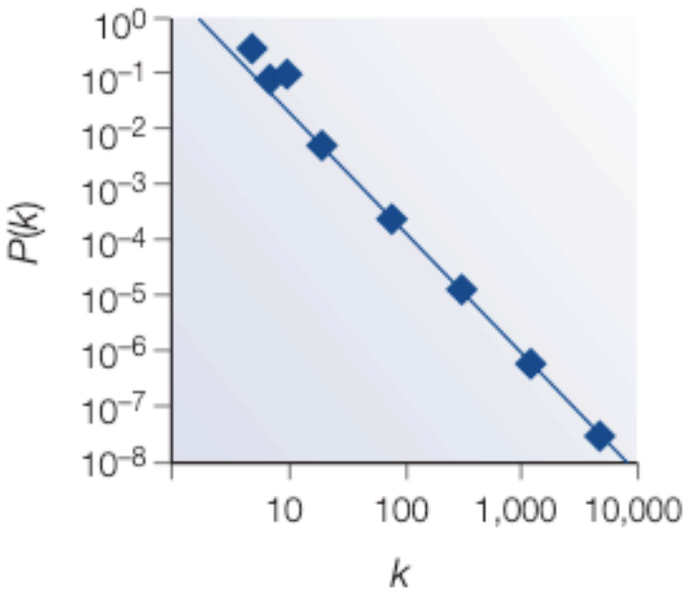
Degree exponent $2 < \gamma < 3$, where smaller γ indicates larger importance

Scale-free and Hierarchical networks tend to display high robustness to node failure: removal of <80% nodes still retains paths between any two nodes

Hierarchical network



Degree distribution
shows many with low degrees
a few highly connected nodes



In practice:

$\gamma < 3$:
$P(k) \propto k^{-\gamma}$ and $P(k) \propto N$

$\gamma > 3$:
network behaves like random

Barabasi 2004
Jeong 2000

7. Degree and clustering coefficient distribution

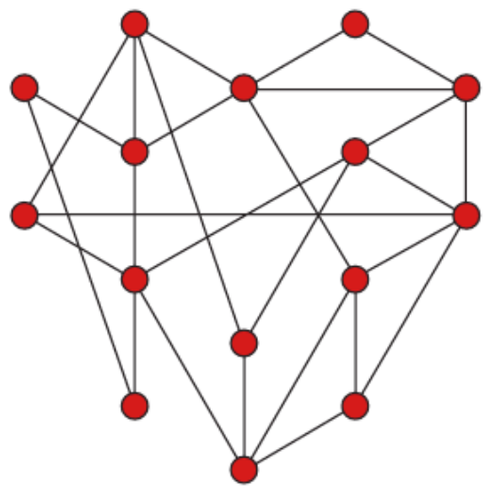
$C(k)$ shows no relationship with *k* in random networks: no modular organisation

$C(k) = k^{-1}$ in hierarchical networks

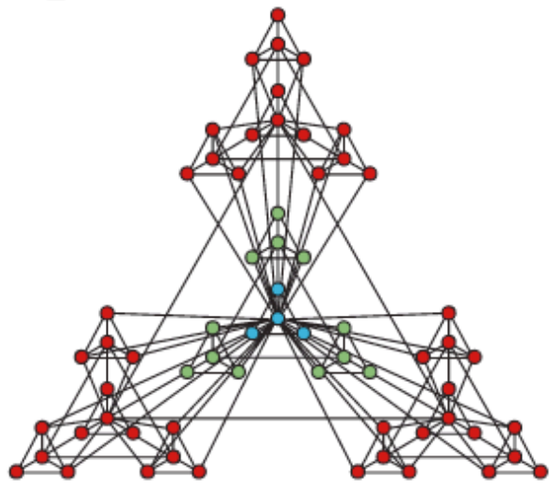
Sparsely connected nodes are part of highly modular areas

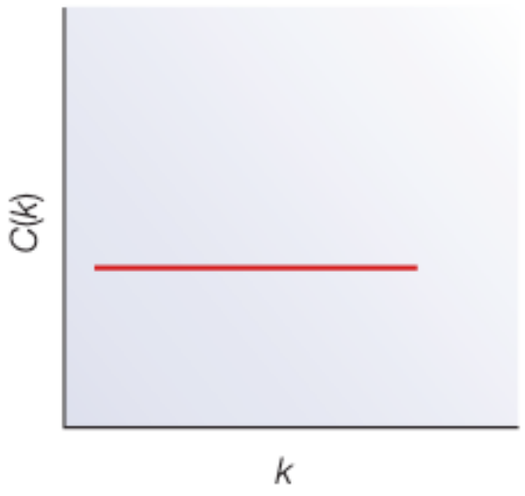
Communication between highly clustered neighbourhoods maintained by a few hubs

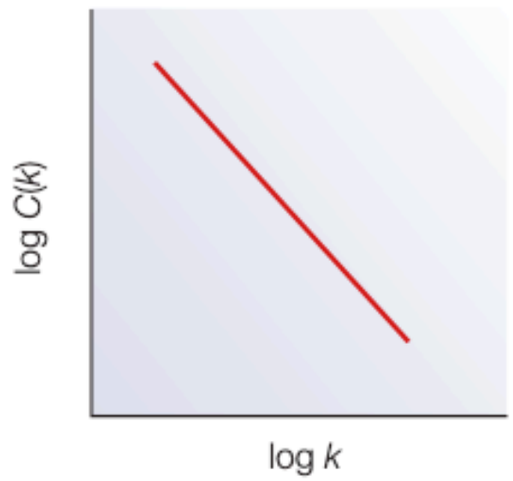
Random network



Hierarchical network







Barabasi 2004
Jeong 2000

7. Small world

Any two nodes can be connected in a small number of steps.

This is a property see in **random networks** where the mean path length

$$l(G) \approx logN$$ for a network of size *N*

Scale-free networks show **ultra-small world**:

$$l(G) \approx log(logN)$$

In practice, this indicates that perturbations may quickly spread throughout the network

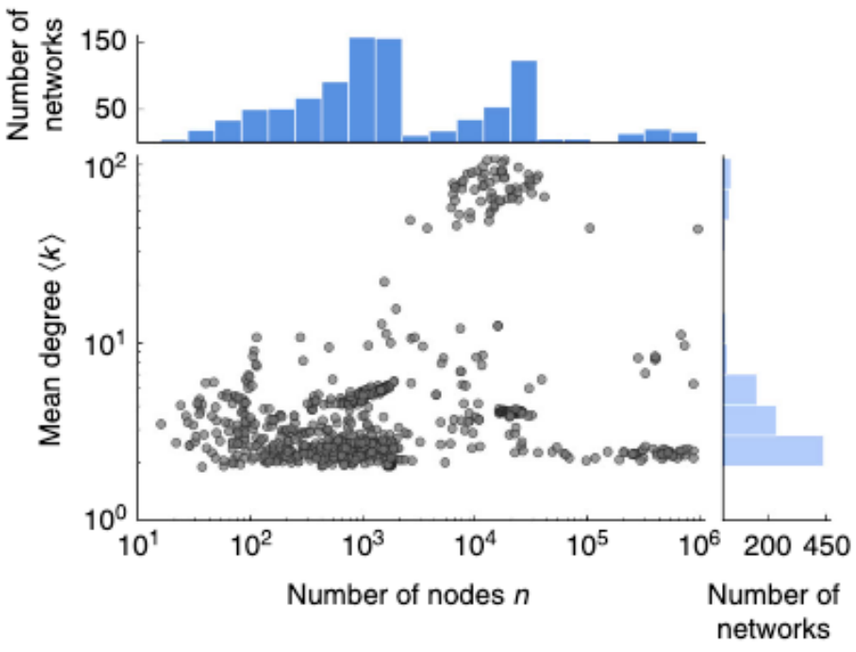
Highly central hubs tend **not** to be connected in biological networks: they are **disassortative**

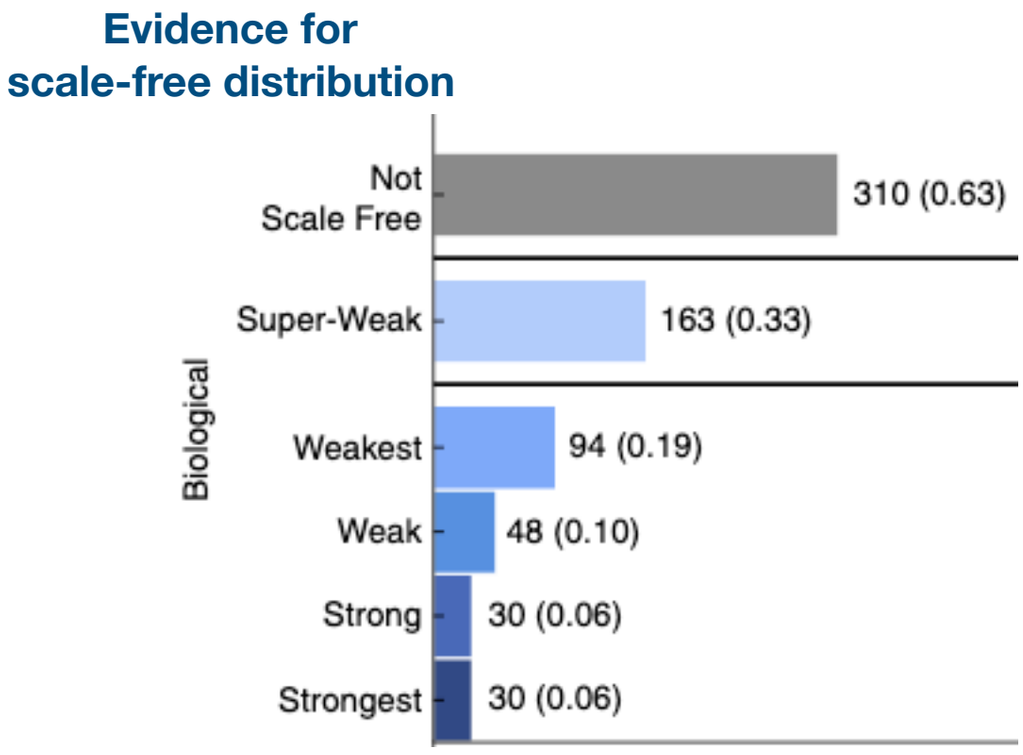
(social networks: **assortative**)

Barabasi 2004

7. Not all networks show scale-free behavior

Analysis of many types of networks shows that scale-free distributions are rare





Broido 2019
Stumpf 2012

Additional reading

- Analysis of Biological Networks - General introduction into biological networks, network notation, and analysis, including graph theory.
- Using graph theory to analyze biological networks - overview of the usage of graph theory in biological network analysis
- Survival of the sparsest: robust gene networks are parsimonious - analysis of network complexity and robustness.
- Network biology: understanding the cell's functional organization - Overview of key concepts in biological network structure
- Graph Theory and Networks in Biology - extended perspective on how graph analysis is applied in biology
- Scale free networks are rare
- Modularity and community structure in networks

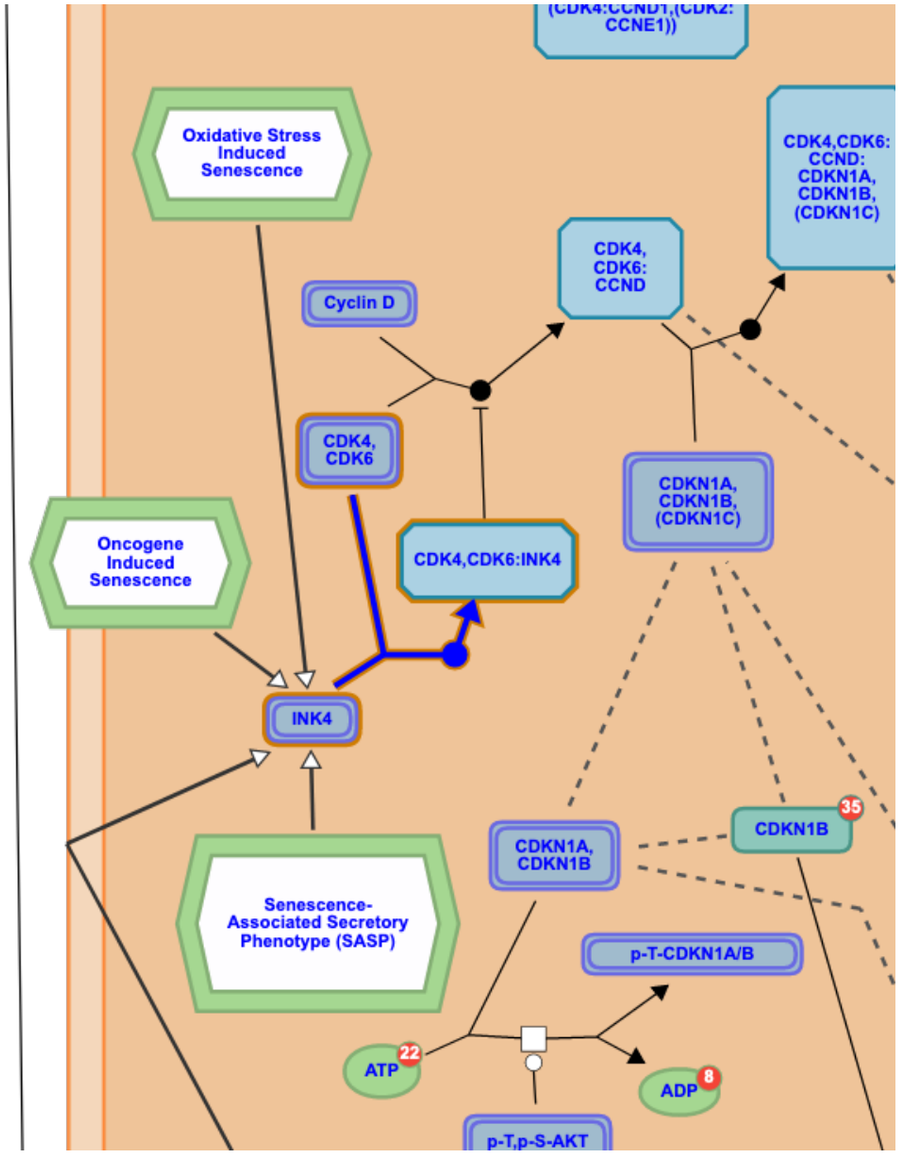
Additional references displayed as hyperlinks in each figure.

Community and functional analysis

What are modules?

Modules are physically or functionally associated nodes that work together to achieve a distinct function

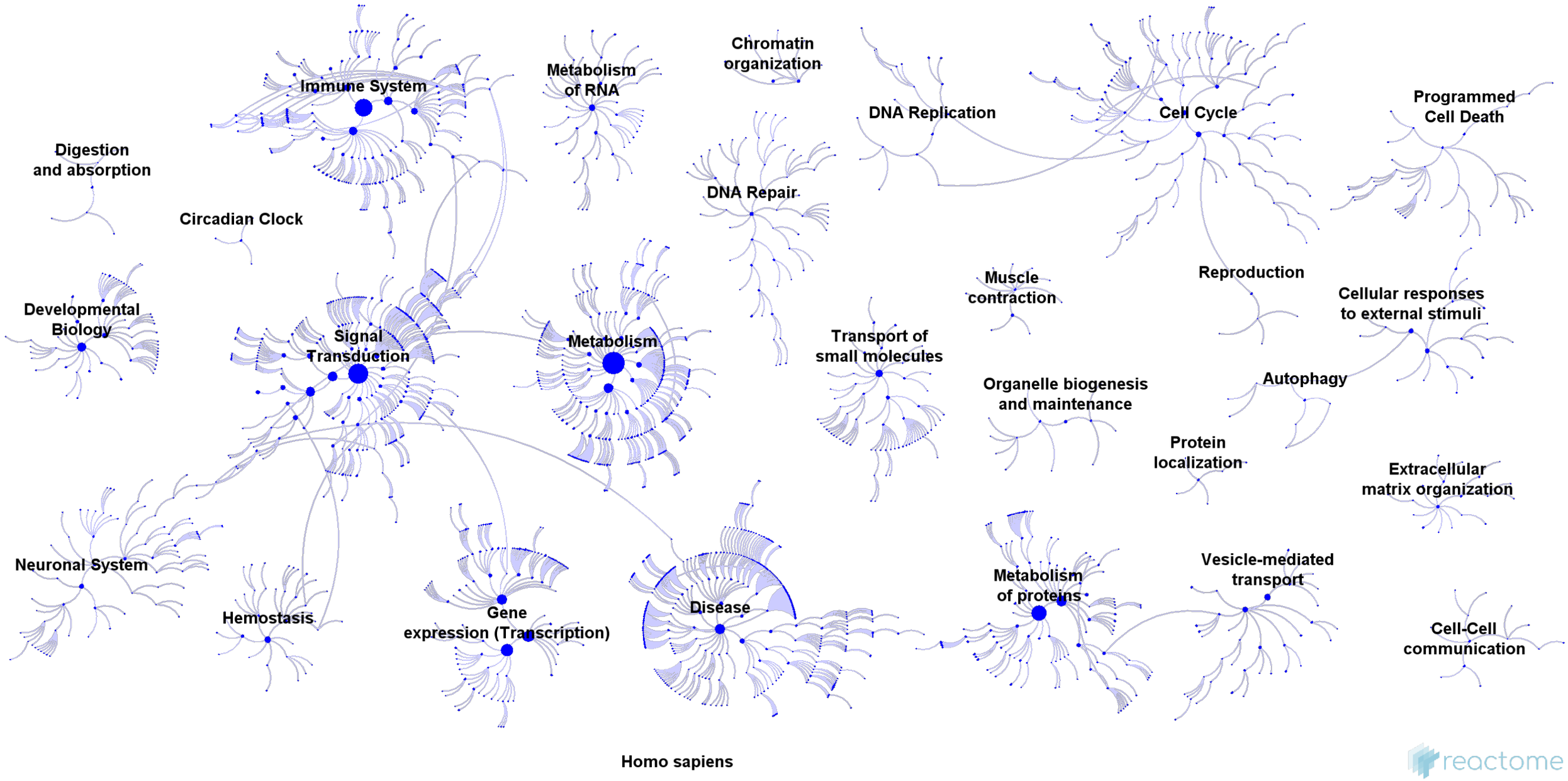
Protein complexes are physical modules



What are modules?

Pathway-associated proteins ***may*** represent functional modules

Gene Ontology



What are modules?

In addition to physical or functional modules, one may identify other types of modules

Topological: derived from their high within-module degree

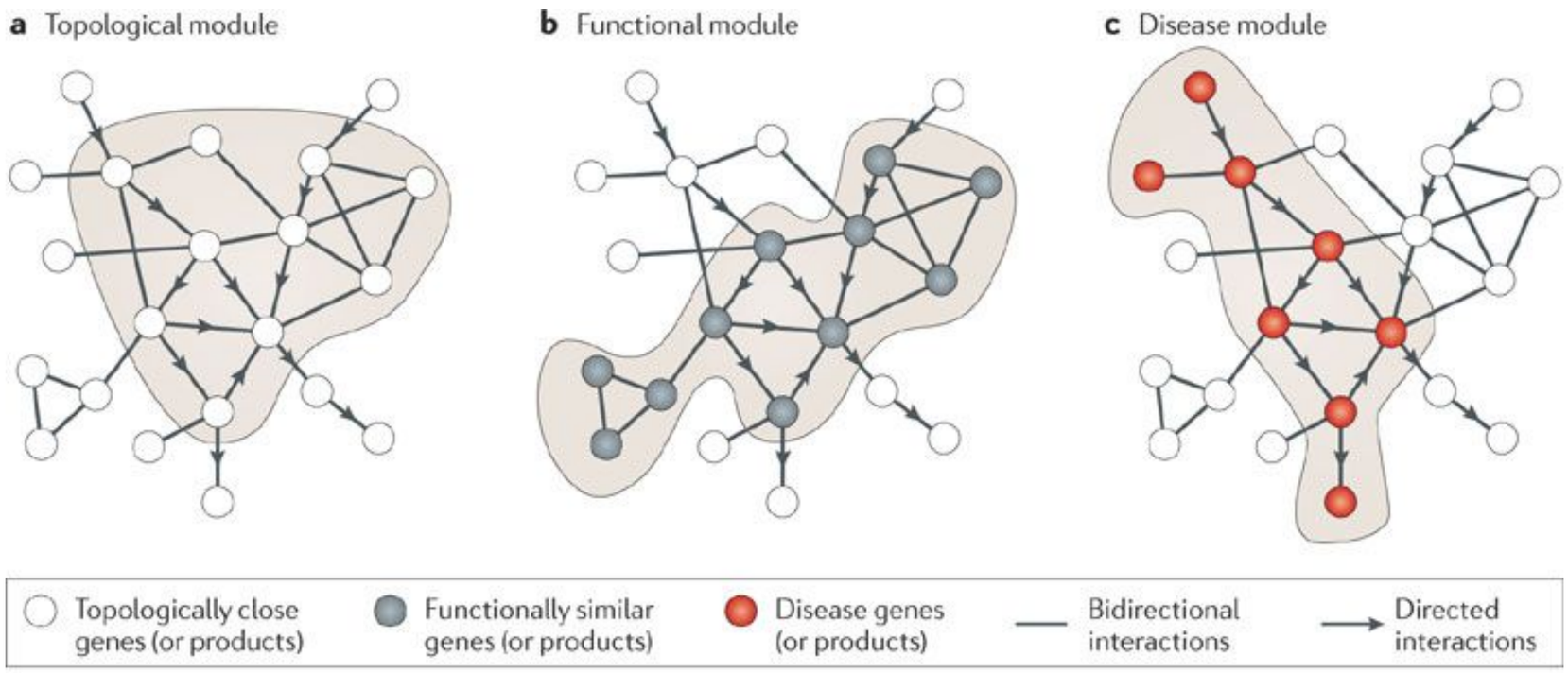
Disease: highly interconnected nodes associated with a disease response

Drug: highly interconnected nodes associated with a drug response

Subgroup: highly interconnected nodes associated with a sample subgroup (e.g. cancer subtype)

Tissue-, cell-type-specific: highly interconnected nodes associated with a specific tissue or cell type

Highly interlinked local regions of a network



Barabasi 2011

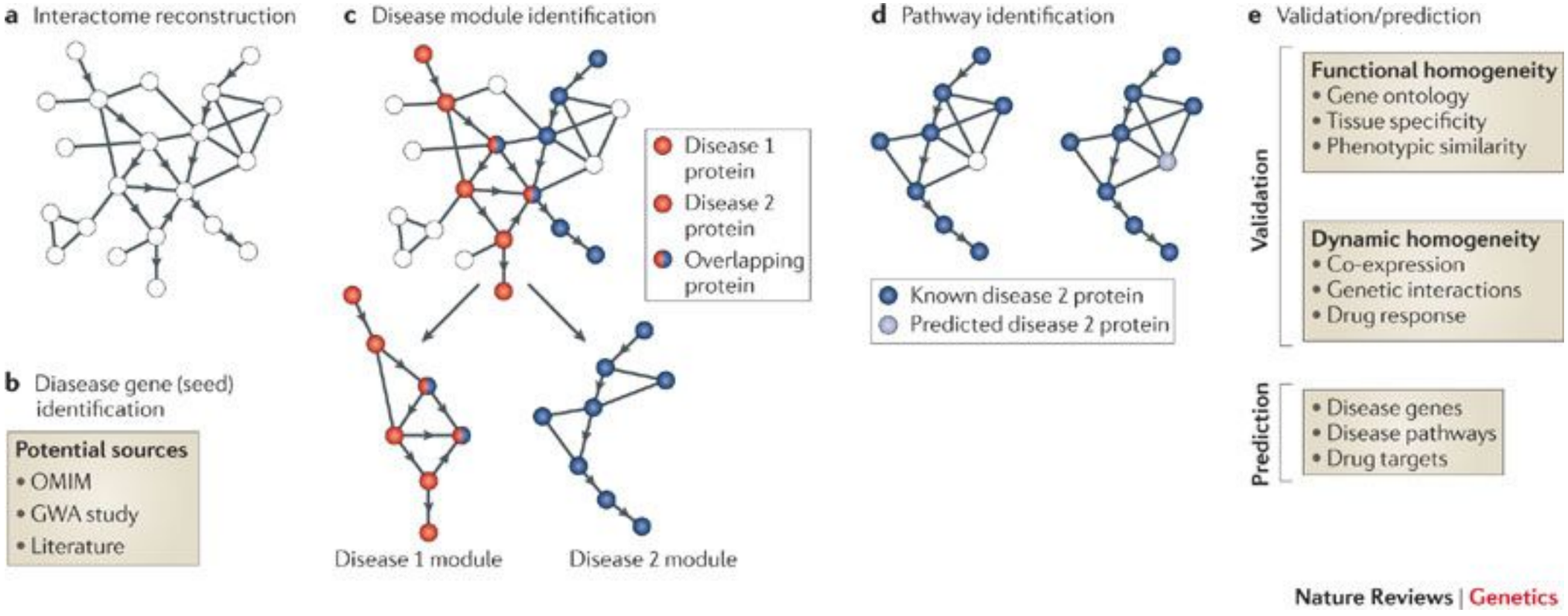
The challenge: identify and characterise modules

Moving from full network to modular characterisation

Different features (diseases, biological processes, etc.) may be associated with the same module

Prediction: *in silico*, relies on available knowledge

Validation: experimental responses



Barabasi 2011

Modularity

Modularity is a property of the network

Modularity (Q) measures the tendency of a graph to be organised into modules

Modules computed by comparing probability that an edge is in a module vs what would be expected in a random network

For a given partitioning of the network into individual groups s, compute

$$Q \propto \sum_{s \in S} [(e_s) - (\text{expected } e_s)]$$

edges in group s

Random network with
same number of nodes, edges and
degree per node

Modularity

Number of expected edges e if network is random, given the degree for its nodes

$$\frac{K_s^2}{2m}$$

Sum(degrees of nodes in community)

Total number of edges in community

$$Q \propto \sum_{s \in S} [(e_s) - (\text{expected } e_s)]$$

edges in group s

Random network with
same number of nodes, edges

$$-1 < Q < 1$$

Q = 1: much higher number of edges than expected by chance

Q = -1: lower number of edges than expected by chance

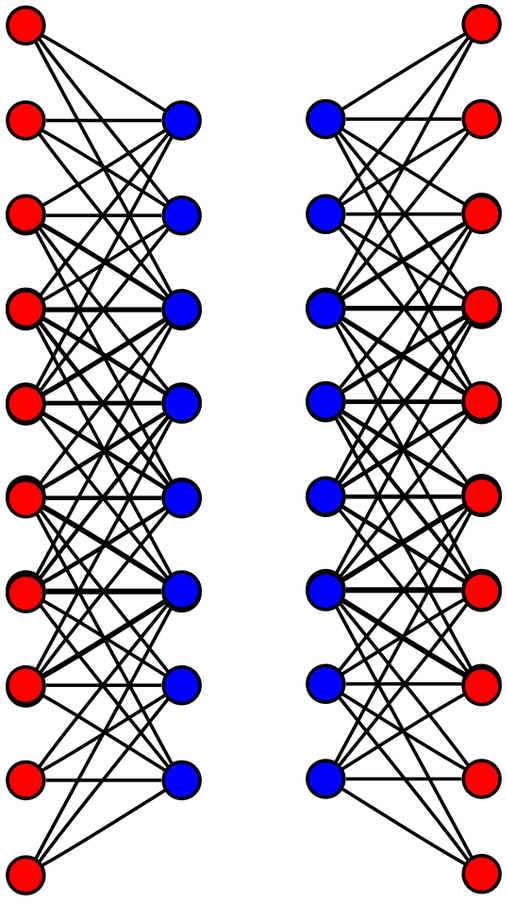
Q > 0.3 - 0.7 means significant community structure

Modularity

Modularity is different than **clustering coefficient**:

Graph composed of two bipartite complete subgraphs:

high Q but low connectivity (C)



Modules

A **module** (or **community**) is a set of nodes with a lot of **internal connections**, but **fewer external connections.**

How to identify modules? Maximise Q

$$Q \propto \sum_{s \in S} [(e_s) - (\text{expected } e_s)]$$

Brute-force approach:

1. Start with 1 node/module
2. Compute distances between nodes
3. Join closest node
4. Compute Q
5. Re-compute distances between a 2n module and each 1n module
6. Join them if Q increases

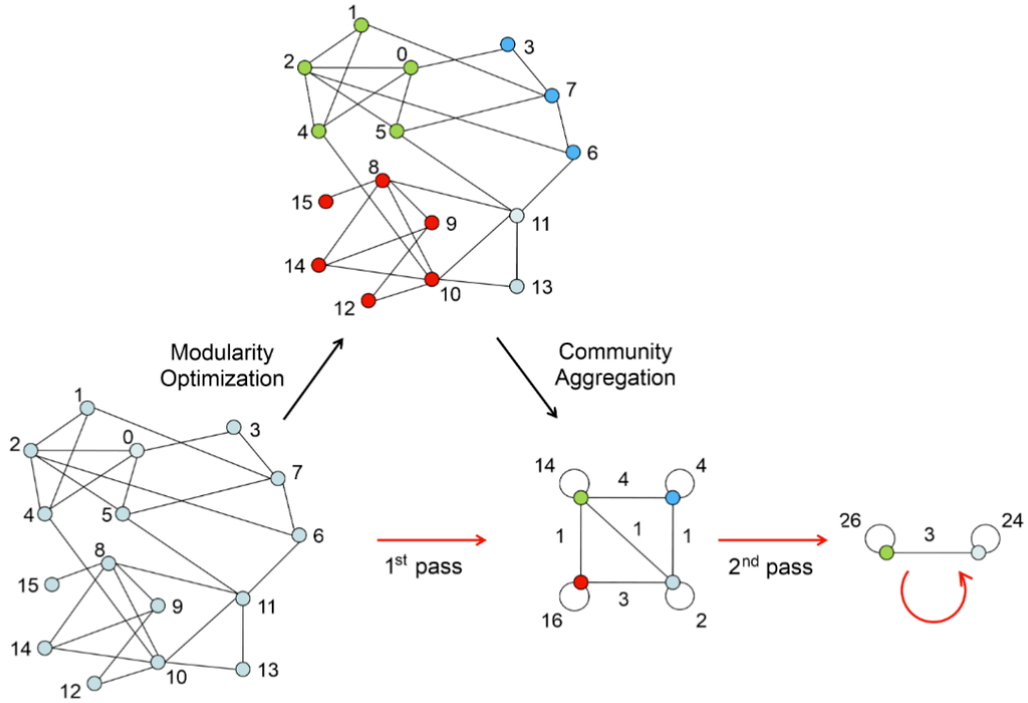
Module detection: Louvain algorithm

Phase 1: greedy modularity optimisation

1. Start with 1n/community
2. Compute Q by moving *i* to the community of *j*
3. If ΔQ>1, node is placed in community
4. Repeat 1-3 until no improvement is found. Ties solved arbitrarily

Phase 2: coarse grained community aggregation

5. Link nodes in a community into single node.
6. Self loops show intra-community associations
7. Inter-community weights kept
8. Repeat phase 1 on new network



Has <u>some known issues</u>:

- Communities may be internally disconnected
- Misses smaller communities

<u>Leiden algorithm</u>

Community characterisation

Clustering coefficient and degree distribution

Enrichment analysis

Assumption: community-associated features show coordinated (directly proportional) changes

Can significantly enriched biological processes serve as “validation”?

- Mutual feature associations may reinforce data characterisations not evident by individual features
- …or need of further network curation based on top biological terms

GSEA calculates overrepresentation by comparison of gene-level statistics against those of the gene-set, considering sample and feature permutation

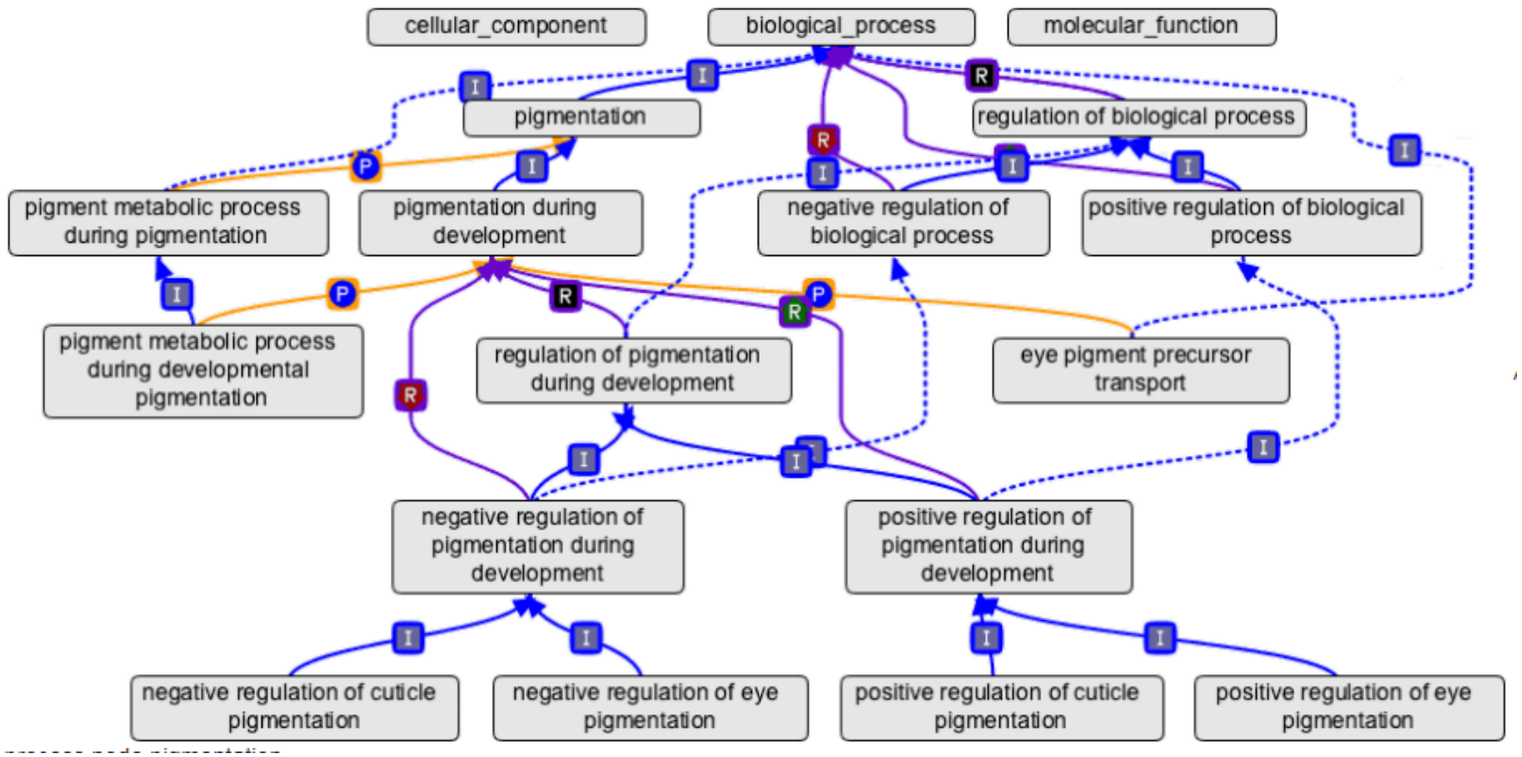
New to Gene Set Enrichment Analysis? See: Mooney 2015

Enrichment analysis

GO-terms, pathways, subcellular location, TF-targets, disease, drug, other?

Tests for significant overlap between groups

Some biological processes may have no biological meaning in your analysis



http://geneontology.org/

Enrichment analysis

MSigDB

GSEA
Gene Set Enrichment Analysis

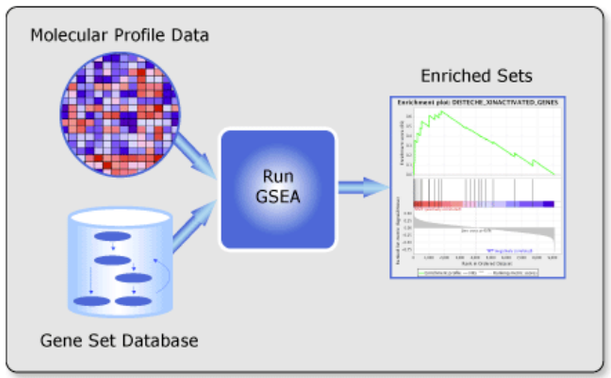
GSEA Home | Downloads | Molecular Signatures Database | Documentation | Contact

Overview

Gene Set Enrichment Analysis (GSEA) is a computational method that determines whether an a priori defined set of genes shows statistically significant, concordant differences between two biological states (e.g. phenotypes).

From this web site, you can:

- **Download** the GSEA software and additional resources to analyze, annotate and interpret enrichment results.
- **Explore the Molecular Signatures Database (MSigDB)**, a collection of annotated gene sets for use with GSEA software.
- **View documentation** describing GSEA and MSigDB.



What's New

20-Aug-2019: MSigDB 7.0 released. This is a major release that includes a complete overhaul of gene symbol annotations, Reactome and GO gene sets, and corrections to miscellaneous errors. See the release notes for more information.

20-Aug-2019: GSEA 4.0.0 released. This release includes support for MSigDB 7.0, plus major internal updates for Java 11 support and performance improvements. See the release notes for more information.

16-Jul-2018: MSigDB 6.2 released. This is a minor release that includes updates to gene set annotations, corrections to miscellaneous errors, and a handful of new gene sets. See the release notes for more information.

Follow @GSEA_MSigDB

License Terms

GSEA and MSigDB are available for use under these license terms.

Please register to download the GSEA software, access our web tools, and view the MSigDB gene sets. After registering, you can log in at any time using your email address. Registration is free. Its only purpose is to help us track usage for reports to our funding agencies.

Contributors

GSEA and MSigDB are maintained by the GSEA team. Our thanks to our many contributors. Funded by: National Cancer Institute, National Institutes of Health, National Institute of General Medical Sciences.

Citing GSEA

To cite your use of the GSEA software, please reference Subramanian, Tamayo, et al. (2005, PNAS 102, 15545-15550) and Mootha, Lindgren, et al. (2003, Nat Genet 34, 267-273).

Enrichr

Enrichr

Login | Register

21,153,478 lists analyzed
307,486 terms
154 libraries

Analyze | What's New? | Libraries | Find a Gene | About | Help

Gene-set Library	Terms	Gene Coverage	Genes per Term
Genes_Associated_with_NIH_Grants	32876	15886	9.0
Cancer_Cell_Line_Encyclopedia	967	15797	176.0
Achilles_fitness_decrease	216	4271	128.0
Achilles_fitness_increase	216	4320	129.0
Aging_Perturbations_from_GEO_down	286	16129	292.0
Aging_Perturbations_from_GEO_up	286	15309	308.0
Allen_Brain_Atlas_down	2192	13877	304.0
Allen_Brain_Atlas_up	2192	13121	305.0
ARCHS4_Cell-lines	125	23601	2395.0
ARCHS4_IDG_Coexp	352	20883	299.0
ARCHS4_Kinases_Coexp	498	19612	299.0
ARCHS4_TFs_Coexp	1724	25983	299.0
ARCHS4_Tissues	108	21809	2316.0
BioCarta_2013	249	1295	18.0
BioCarta_2015	239	1678	21.0
BioCarta_2016	237	1348	19.0
BioPlex_2017	3915	10271	22.0
ChEA_2013	353	47172	1370.0
ChEA_2015	395	48230	1429.0
ChEA_2016	645	49238	1550.0
Chromosome_Location	386	32740	85.0
Chromosome_Location_hg19	36	27360	802.0
CORUM	1658	2741	5.0
Data_Acquisition_Method_Most_Popular_Genes	12	1073	100.0
dbGaP	345	5613	36.0
DepMap_WG_CRISPR_Screens_Broad_CellLines_2019	558	7744	363.0
DepMap_WG_CRISPR_Screens_Sanger_CellLines_2019	325	6204	387.0
Disease_Perturbations_from_GEO_down	839	23939	293.0
Disease_Perturbations_from_GEO_up	839	23561	307.0
Disease_Signatures_from_GEO_down_2014	142	15406	300.0

Enrichment analysis

Important databases with gene-sets:

- MSigDB (gene)
- Enrichr (gene)
- KEGG (metabolite, gene)
- DIANA (miRNA)
- MetaboAnalyst (metabolite)
- DAVID (web)
- Reactome (web)

Creating custom sets and joint sets

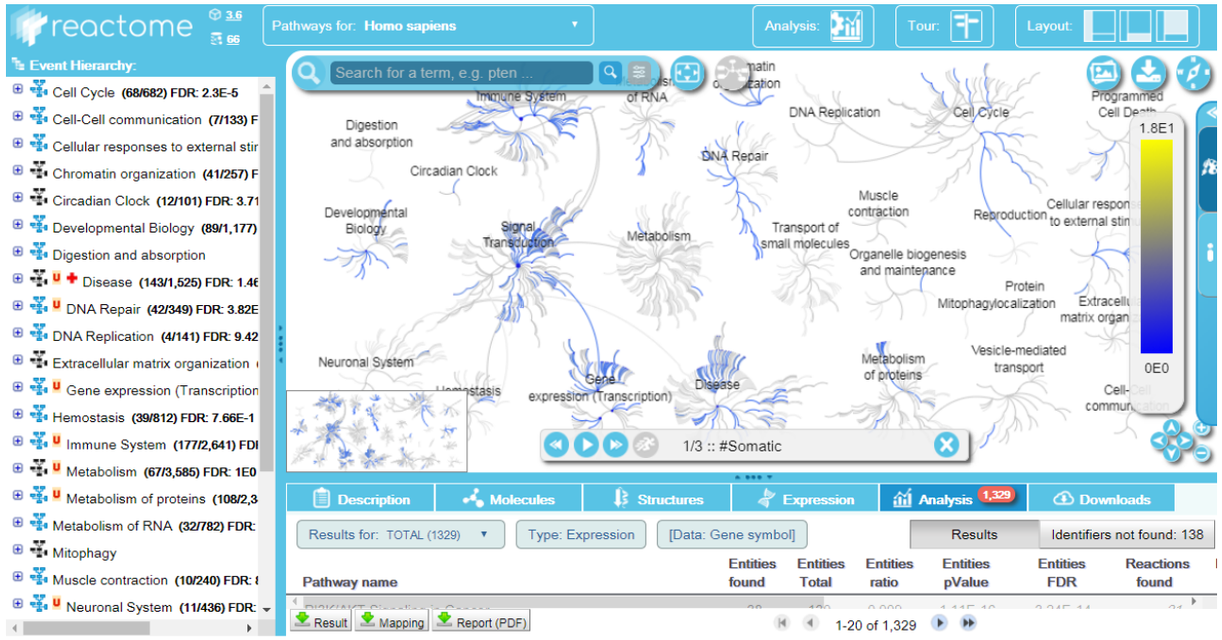
Mapping your data to common IDs

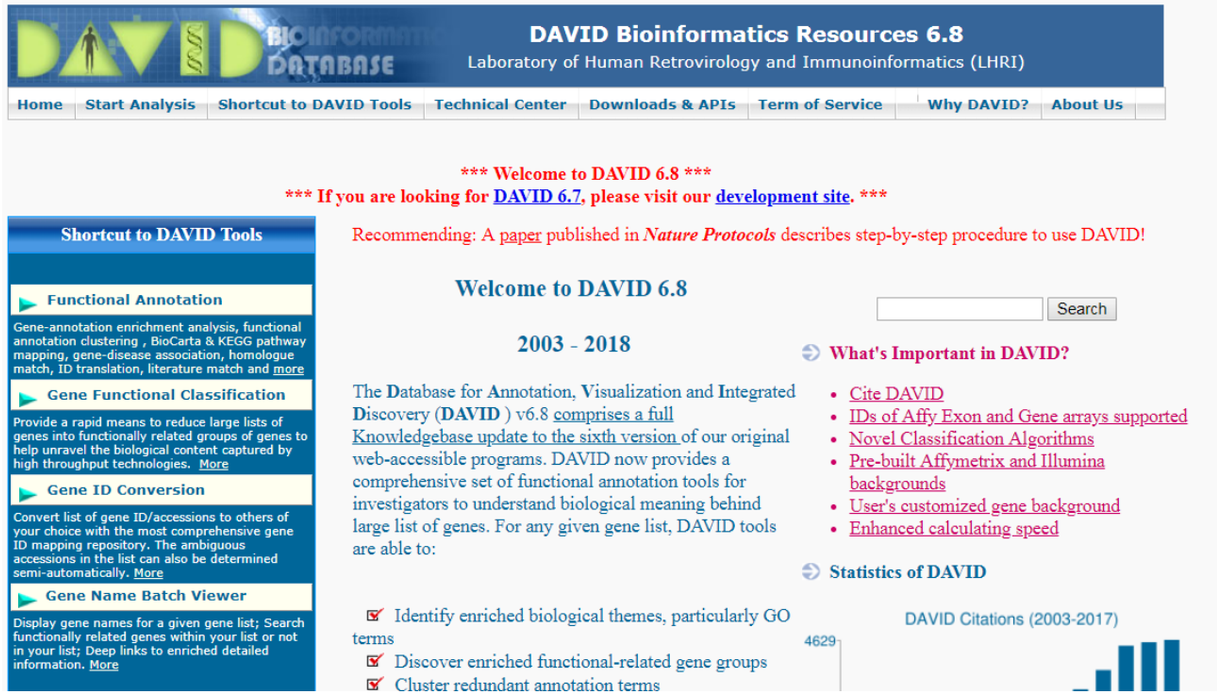
- Easy for genes and proteins: use DAVID, Biomart, or MyGene (in Python or R)
- Hard for other types

Tools for Enrichment analysis

Popular tools for GSEA:

- (most tools above)
- PIANO (highly recommended in R)
- Cytoscape (BINGO plugin)





Considerations for enrichment analysis

What background to consider?

Direction of change

- DAVID, Enrichr, and others do not consider direction
- PIANO takes into account gene-level statistics including directions

Bonus: GSEA in PIANO

distinct-directional: takes direction of change. gene sets with both significantly up- and down-regulated will cancel out.

non-directional: disregards direction and uses absolute values of gene-level statistics

mixed-directional: considers up- and down-regulated subsets separately. Important when

```
[1] -4.0 -3.0  2.0  3.5
```

Gene-level statistics

```
> mean(c(-4, -3, 2.5, 4.5))
[1] 0
```

Distinct-directional

```
> mean(abs(c(-4, -3, 2.5, 4.5)))
[1] 3.5
```

non-directional

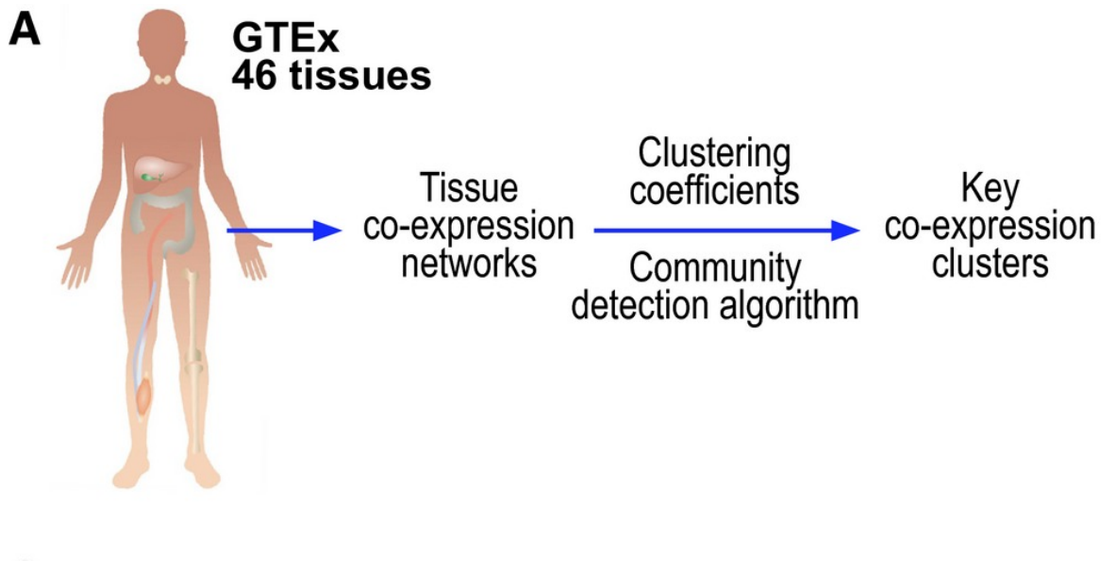
```
> mean(abs(c(-4, -3)))
[1] 3.5
> mean(abs(c(2.5, 4.5)))
[1] 3.5
```

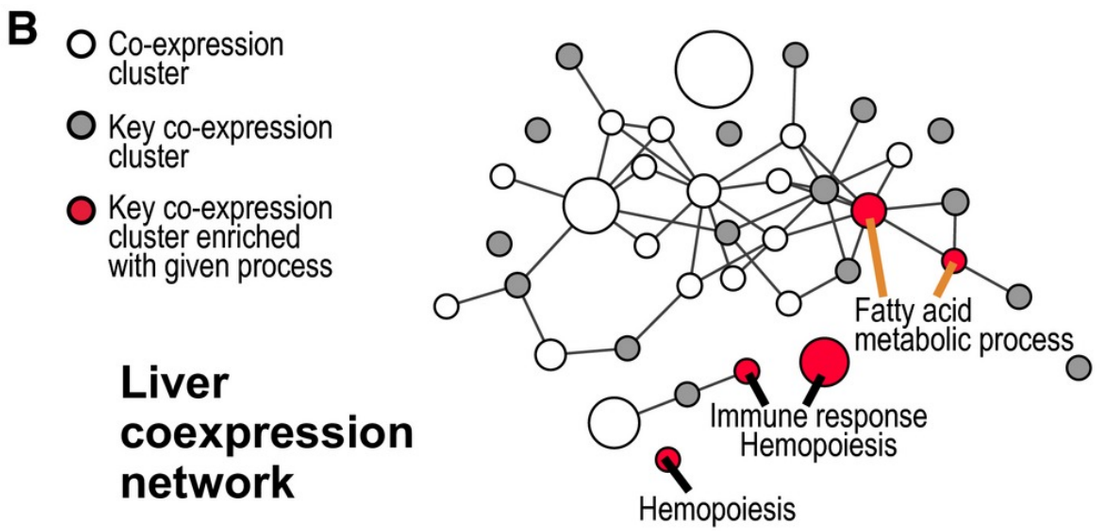
mixed-directional

```
> gs$stats
```

	Name	Value
1	Genes (tot)	243.00000
2	Stat (dist.dir.up)	0.47973
3	p (dist.dir.up)	0.03510
4	p adj (dist.dir.up)	0.40038
5	Stat (dist.dir.dn)	0.52027
6	p (dist.dir.dn)	0.96490
7	p adj (dist.dir.dn)	1.00000
8	Stat (non-dir)	0.15741
9	p (non-dir)	0.00000
10	p adj (non-dir)	0.00000
11	Genes (up)	127.00000
12	Stat (mix.dir.up)	0.15511
13	p (mix.dir.up)	0.00130

Outcome

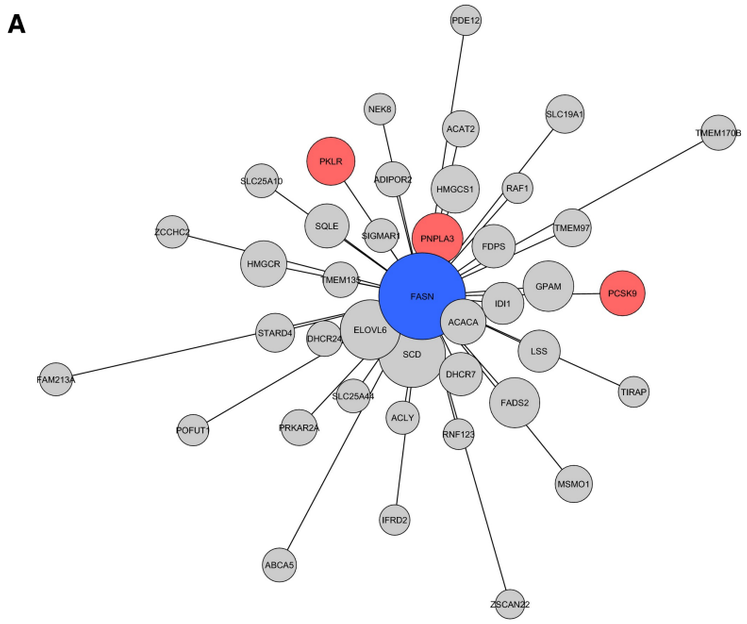


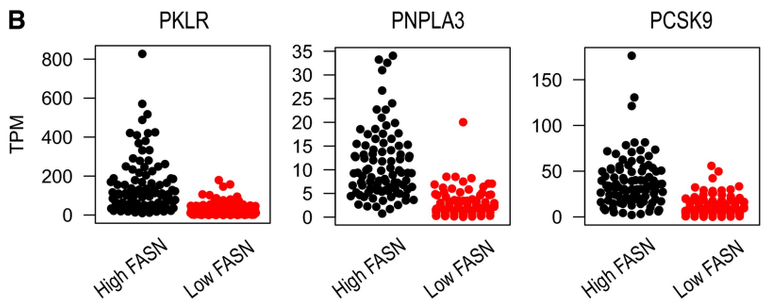


GO BPs from MSigDB

Network analyses identifies key stratifying genes

40-NN of FASN





Visualization

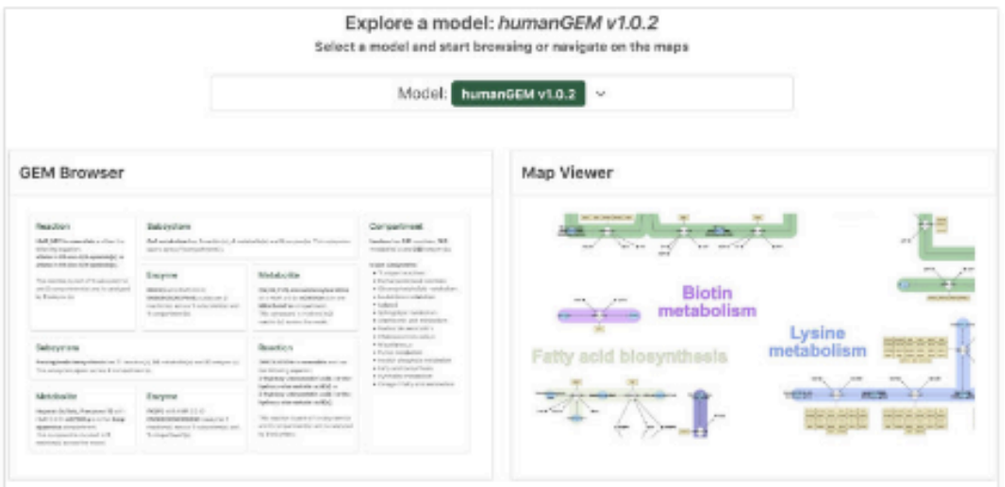
METABOLIC ATLAS

THE ATLAS FOR EXPLORATION OF METABOLISM

Metabolic Atlas integrates open source genome-scale metabolic models (GEMs) of human and yeast for easy browsing and analysis. It also contains many more genome scale metabolic models constructed by our organization.

Detailed biochemical information is provided for individual model components, such as reactions, metabolites, and genes. These components are also associated with standard identifiers, facilitating integration with external databases, such as the Human Protein Atlas.

Article under consideration



reactome

About Content Docs Tools Community Download

Find Reactions, Proteins and Pathways

e.g. O95631, NTN1, signaling by EGFR, glucose Go!

Pathway Browser

Visualize and interact with Reactome biological pathways

Analyze Data

Merges pathway identifier mapping, over-representation, and expression analysis

ReactomeFIViz

Designed to find pathways and network patterns related to cancer and other types of diseases

Documentation

Information to browse the database and use its principal tools for data analysis

USE REACTOME GRAPH DATABASE IN YOUR PROJECT

LEARN MORE

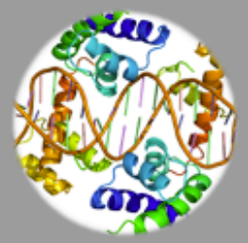
Omics Net - a network analytics platform for multi-omics integration

Home | Overview | FAQs | Tutorials | Gallery | About | Updates

Click an icon below to start

Genes/Proteins

- ID types: Entrez, Ensembl Gene/Transcript, Uniprot and official gene symbol

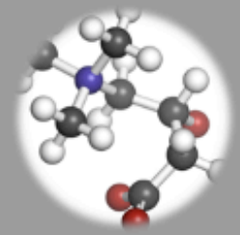


Transcription factors

- ID types: Entrez, Ensembl Gene/Transcript, Uniprot and official gene symbol

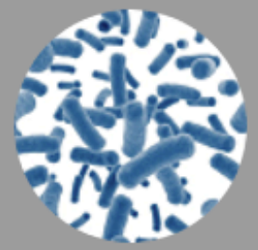
miRNAs

- ID types: miRBase Accession and miRBase ID



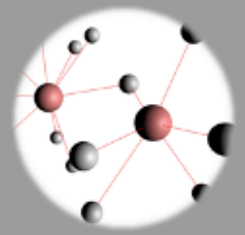
Metabolites

- ID types: KEGG ID, HMDB ID, PubChem and CHEBI ID



Microbiome

- ID types: KEGG Orthology (KO)



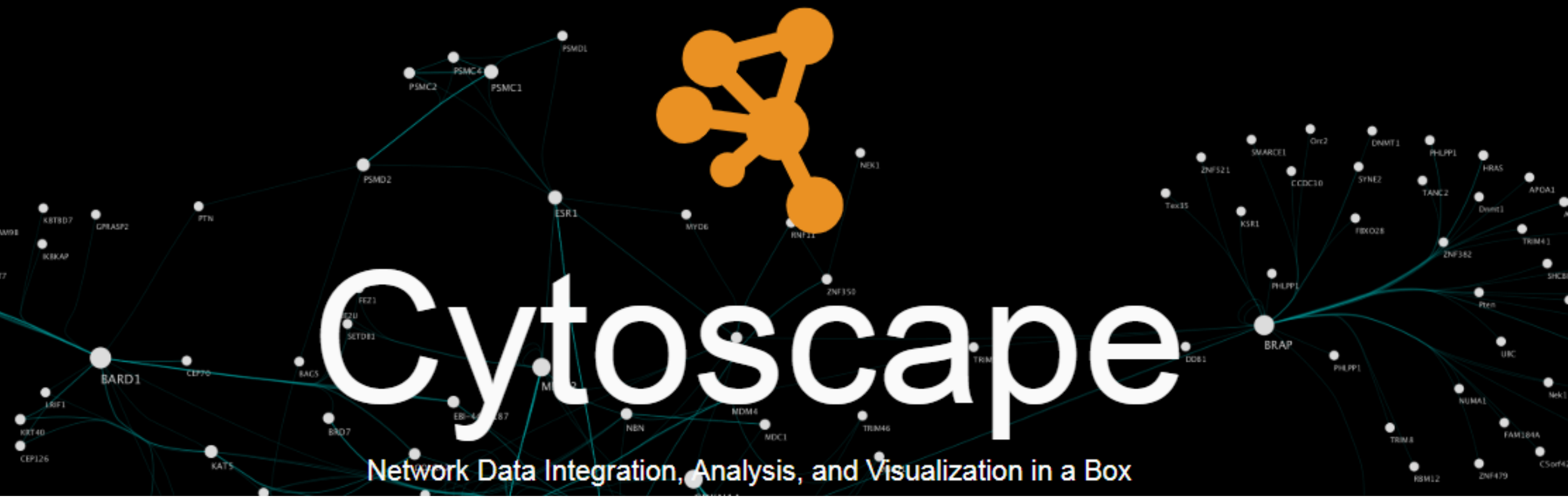
Graph file

- Formats: .sif, .graphml, .json or .txt (edge list)

Proceed

Project Objectives

OmicsNet has been developed to address three needs: 1) systems analysis of a single list of molecules; 2) integrative analysis of multiple lists of different types of molecules; and 3) intuitive web-based 2D/3D visualization. OmicsNet supports four types of molecular interactions - PPI, TF-gene, miRNA-gene and metabolite-protein. The 3D network visualization was implemented based on the innovative WebGL technology. Users can perform various style customization, enrichment analysis, targeted interactor search, and network toplology analysis for hypothesis generation and systems-level insights.



Input: Association network

Tools > Network analyzer: Centrality

Node configuration: Centrality; Community

Edge weight configuration: Correlation

Module detection: ClusterMaker

GSEA: BINGO

Workshop objective:

Build and analyze a coexpression network