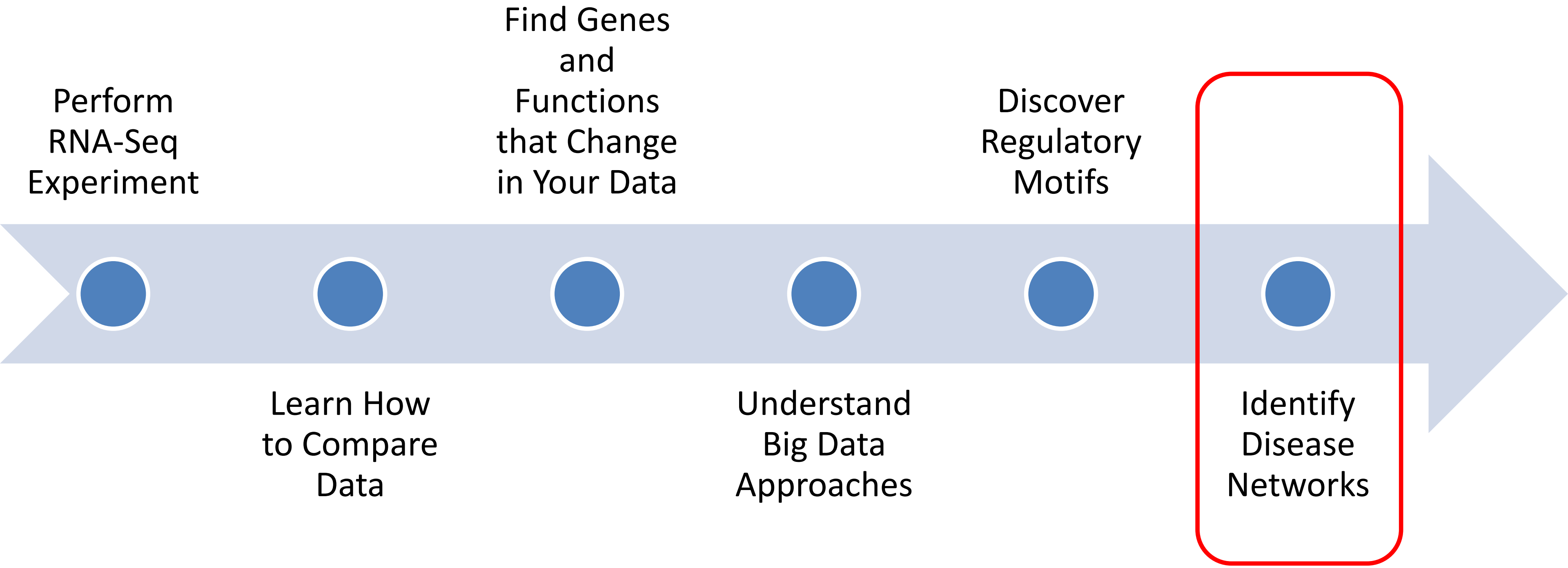


Perform RNA-Seq Experiment

Learn How to Compare Data

Find Genes and Functions that Change in Your Data

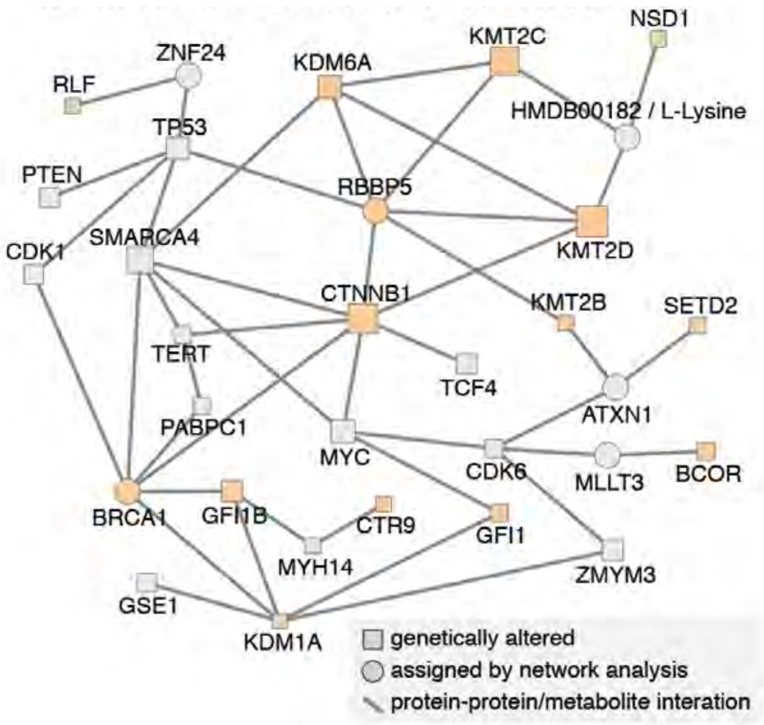
Understand Big Data Approaches

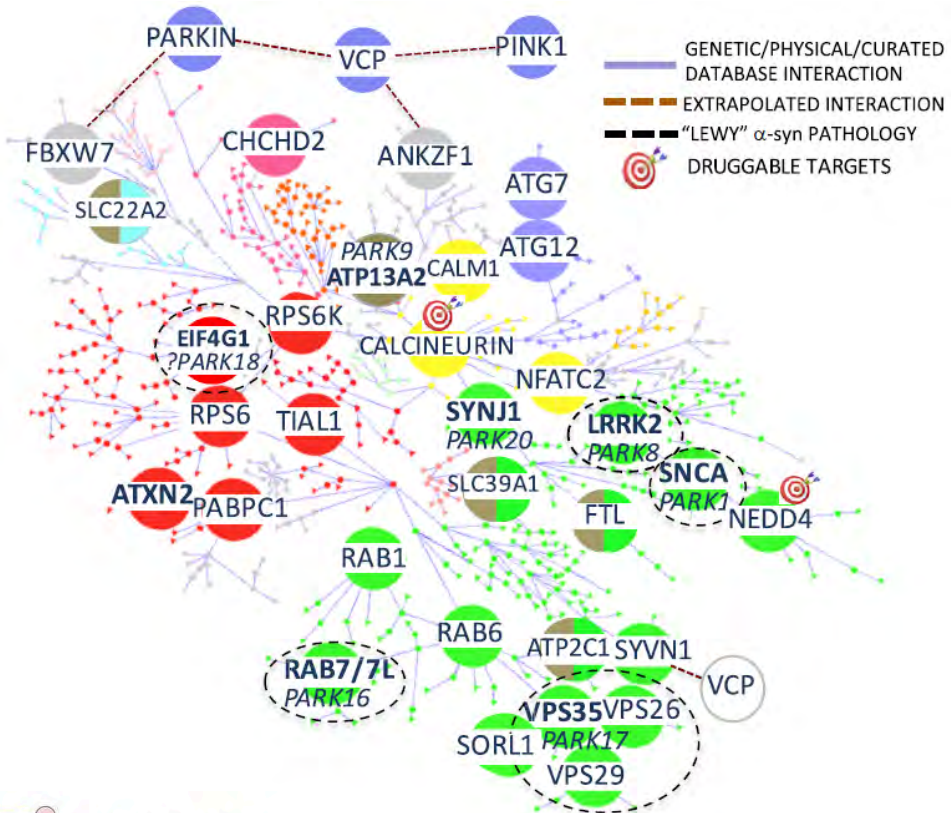
Discover Regulatory Motifs

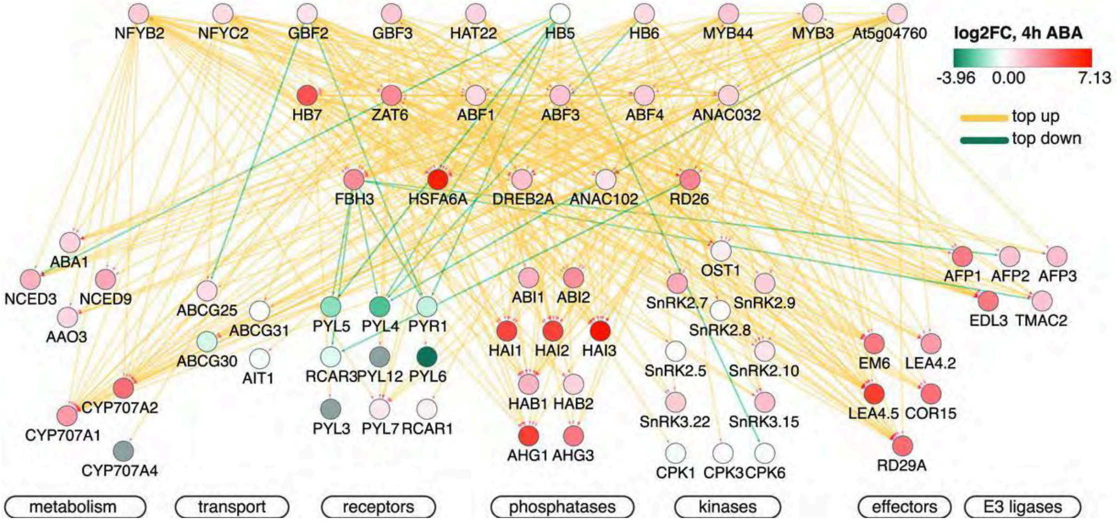
Identify Disease Networks

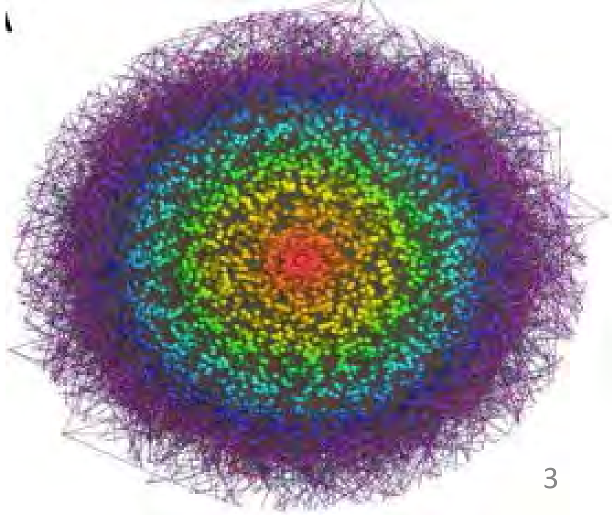
Learning Objectives

- Know how to represent biological data using graph theory
- Know how to describe a graph (network) using an adjacency matrix
- Understand methods for finding network modules
- Understand how networks integrate data


NSD1
KMT2C
ZNF24
KDM6A
RLF
HMDB00182 / L-Lysine
TP53
PTEN
RBBP5
SMARCA4
CDK1
KMT2D
CTNNB1
KMT2B
SETD2
TERT
TCF4
ATXN1
PABPC1
MYC
CDK6
MLLT3
BCOR
BRCA1
GFI1B
CTR9
GFI1
MYH14
ZMYM3
GSE1
KDM1A
genetically altered
assigned by network analysis
protein-protein/metabolite interation


PARKIN
VCP
PINK1
GENETIC/PHYSICAL/CURATED DATABASE INTERACTION
EXTRAPOLATED INTERACTION
"LEWY" α-syn PATHOLOGY
DRUGGABLE TARGETS
FBXW7
CHCHD2
ANKZF1
ATG7
SLC22A2
PARK9
ATP13A2
CALM1
ATG12
RPS6K
EIF4G1
?PARK18
CALCINEURIN
NFATC2
SYNJ1
PARK20
LRRK2
PARK8
RPS6
TIAL1
SNCA
PARK1
SLC39A1
FTL
NEDD4
ATXN2
PABPC1
RAB1
RAB6
ATP2C1
SYVN1
VCP
RAB7/7L
PARK16
VPS35
VPS26
PARK17
SORL1
VPS29

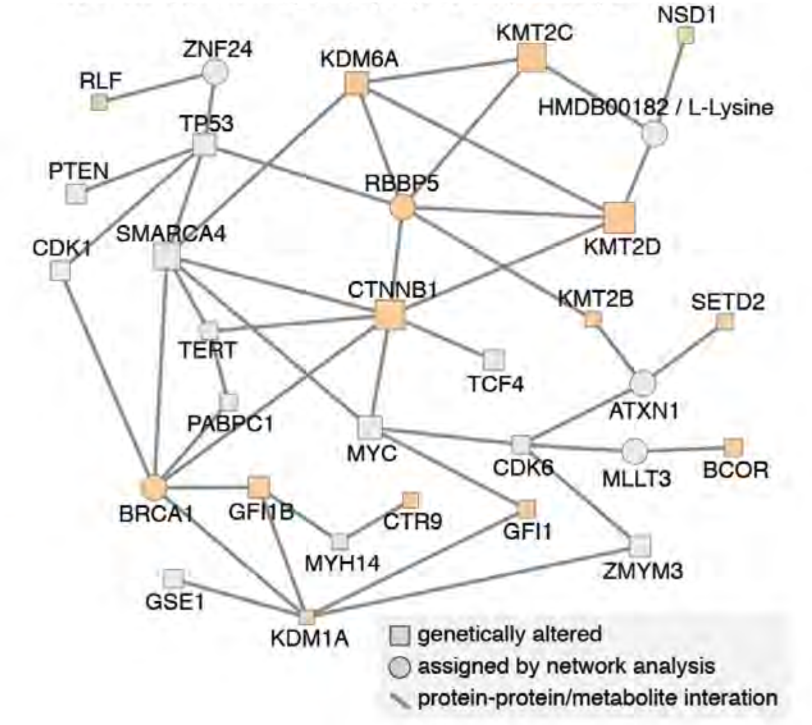

NFYB2
NFYC2
GBF2
GBF3
HAT22
HB5
HB6
MYB44
MYB3
At5g04760
log2FC, 4h ABA
-3.96
0.00
7.13
top up
top down
HB7
ZAT6
ABF1
ABF3
ABF4
ANAC032
FBH3
HSFA6A
DREB2A
ANAC102
RD26
ABA1
NCED3
NCED9
AAO3
ABCG25
ABCG31
ABCG30
AIT1
PYL5
PYL4
PYR1
RCAR3
PYL12
PYL6
PYL3
PYL7
RCAR1
ABI1
ABI2
HAI1
HAI2
HAI3
HAB1
HAB2
AHG1
AHG3
OST1
SnRK2.7
SnRK2.9
SnRK2.8
SnRK2.5
SnRK2.10
SnRK3.22
SnRK3.15
CPK1
CPK3
CPK6
AFP1
AFP2
AFP3
EDL3
TMAC2
EM6
LEA4.2
LEA4.5
COR15
RD29A
CYP707A2
CYP707A1
CYP707A4
metabolism
transport
receptors
phosphatases
kinases
effectors
E3 ligases



Network Models

In Today’s Lecture:

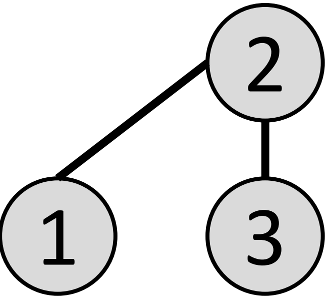
- Structure of network
  - Nodes: molecules
  - Edges: relationships
    - Physical
    - Genetic
    - Statistical



Graph Terminology

- G=(V,E)
- Undirected vs. directed
- Weights – numbers assigned to each edge
- Degree(v) – number of edges incident on v
    - In-degree and out-degree
- Path from a to b is a series of vertices
  <a, v0, …, b>
  where edges exist between sequential vertices.
- Path length = sum of edges weights
  (or number of edges) on path.

Adjacency Matrix



a_{ij}= 1 if there is an edge between *i* and *j*
0 otherwise

Let B=A^N: b_{ij}= m iff there exist exactly m paths of length N between i and j.

$$\begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \\ 3 & 0 & 1 & 0 \end{array} \times \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \\ 3 & 0 & 1 & 0 \end{array} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 1 & 0 & 1 \\ 2 & 0 & 2 & 0 \\ 3 & 1 & 0 & 1 \end{array}$$

Shortest Path Algorithms

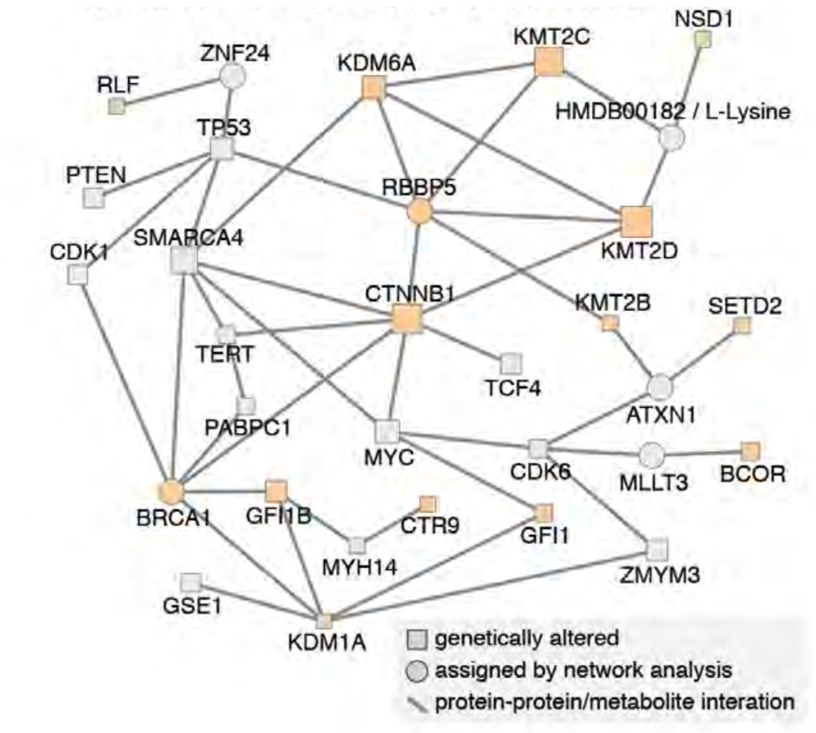
- Efficient Algorithms for
  - single pair (u,v)
  - single source/destination to all other nodes
  - all-pairs

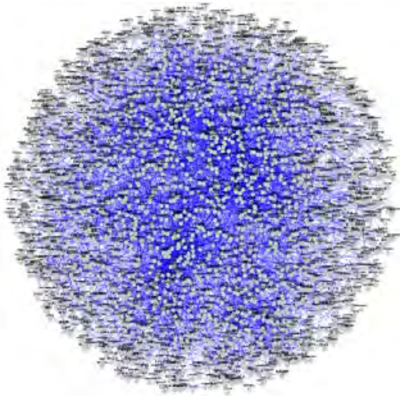
Good place to learn more:
"Introduction to Algorithms"
by Cormen, Leiserson, Rivest, and Stein.

Finding Modules

In Today's Lecture:

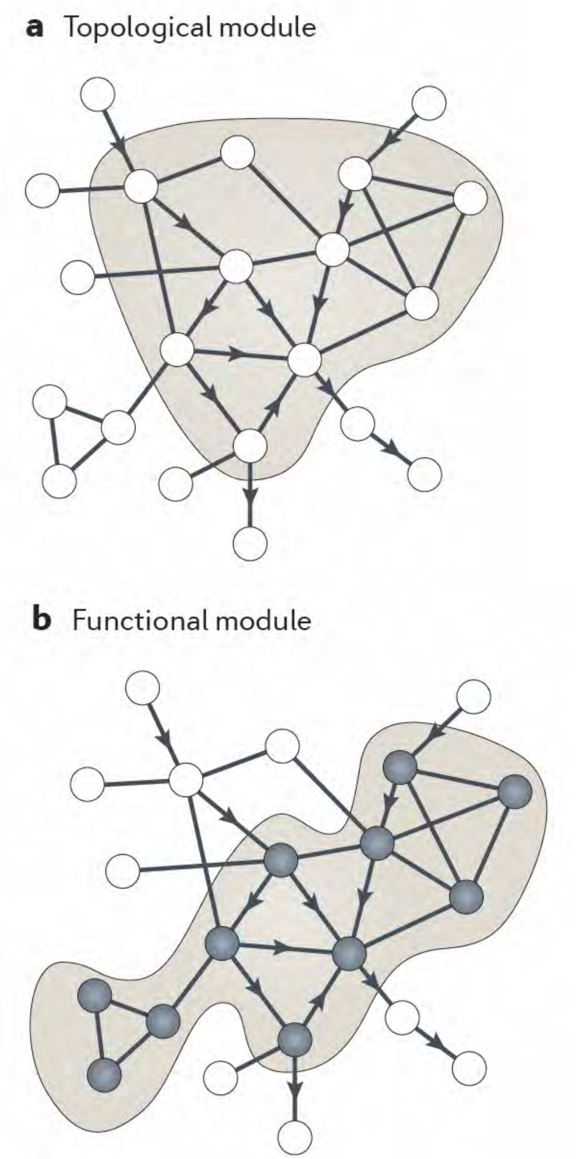
- Use the network to organize and simplify the relationships
  - – Predicting Function of Genes
  - – Identifying Proteins Families, Co-regulated genes
  - – Integrating Data



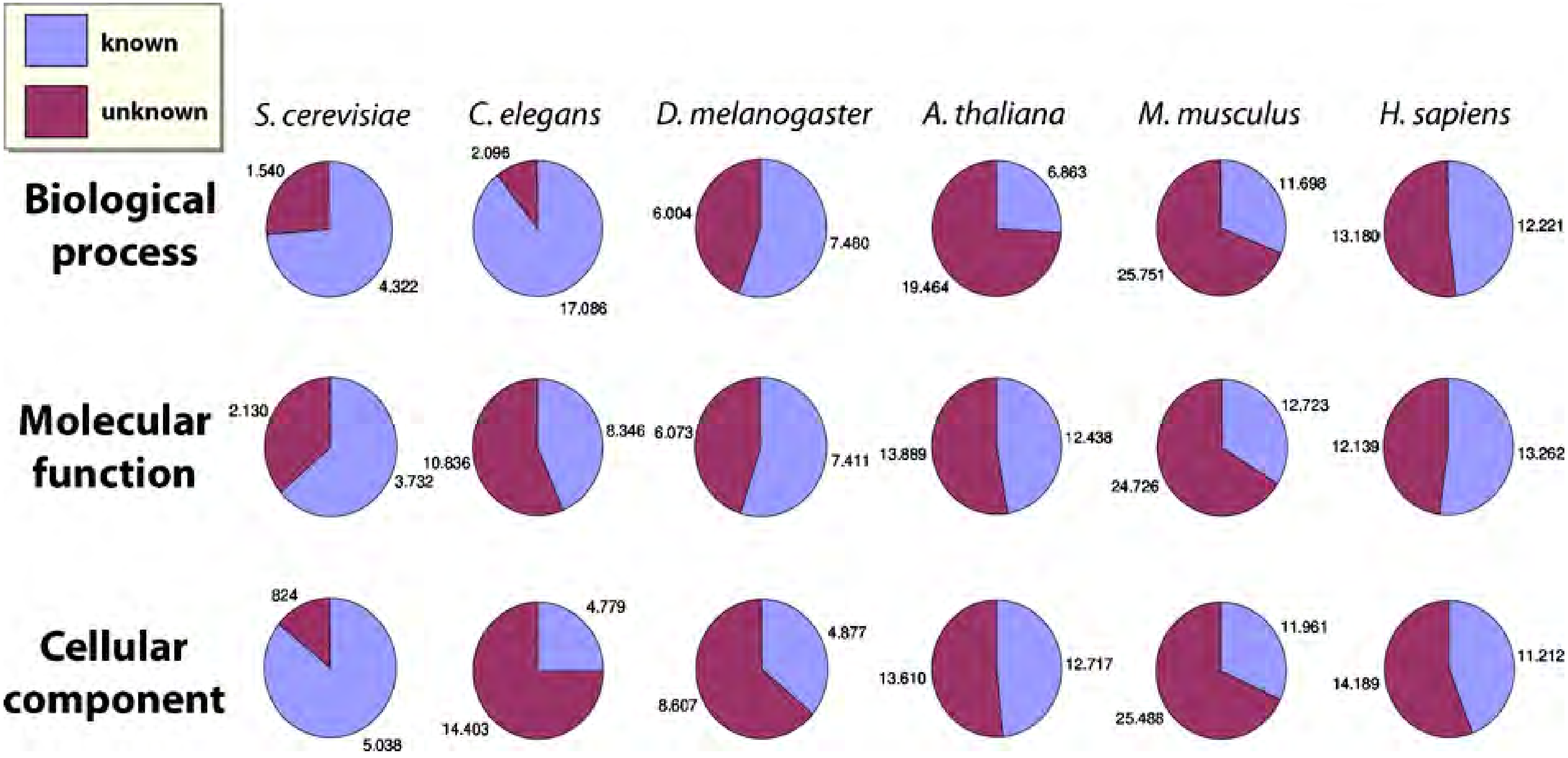


Finding Modules

- Topological module:
  - locally dense
  - more connections among nodes in module than with nodes outside module
- Functional module:
  - high density of functionally related nodes



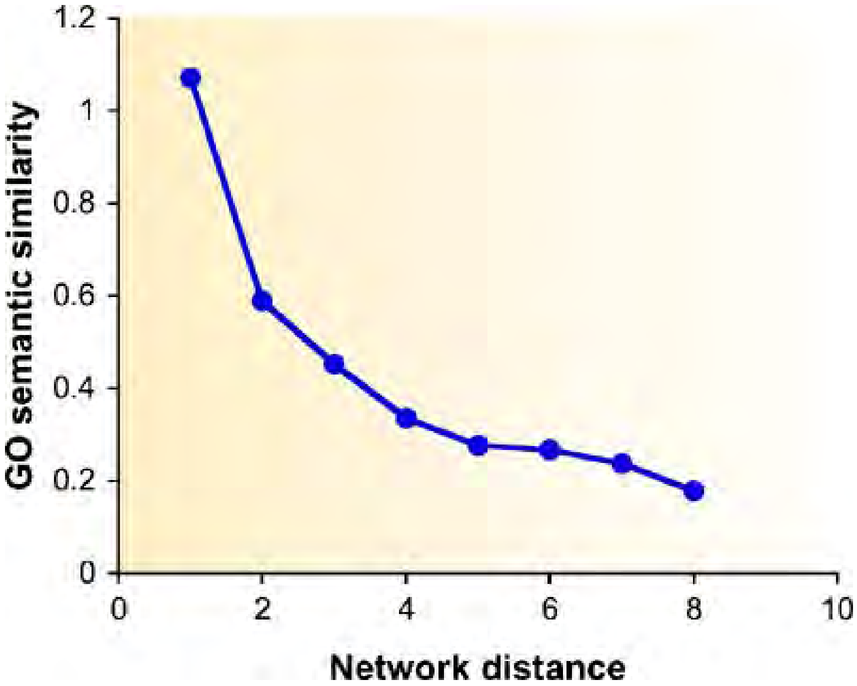
Can we use networks to predict function

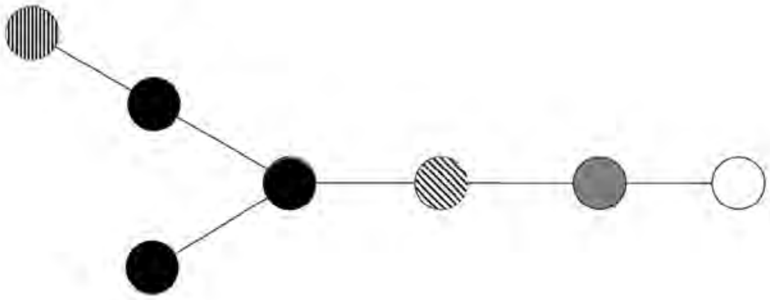


Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129

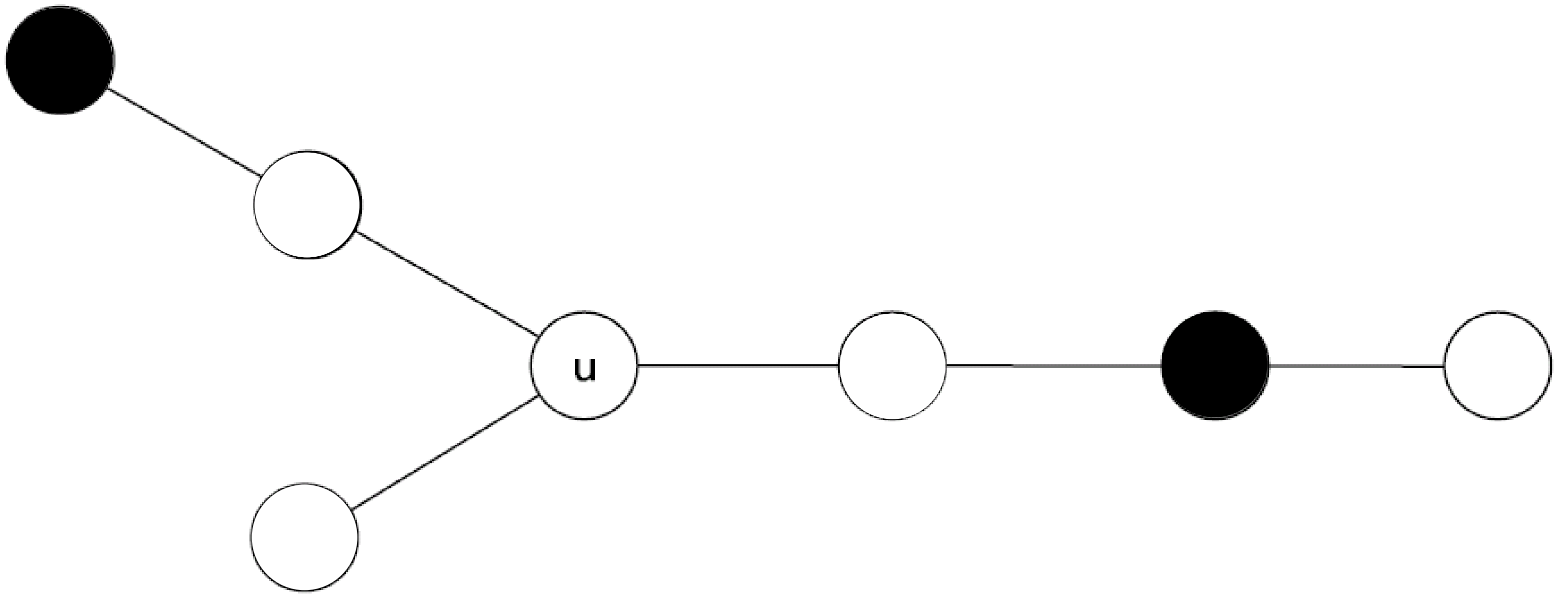
based on the Entrez Gene and the WormBase databases as of September 2006

Can we use networks to predict function?

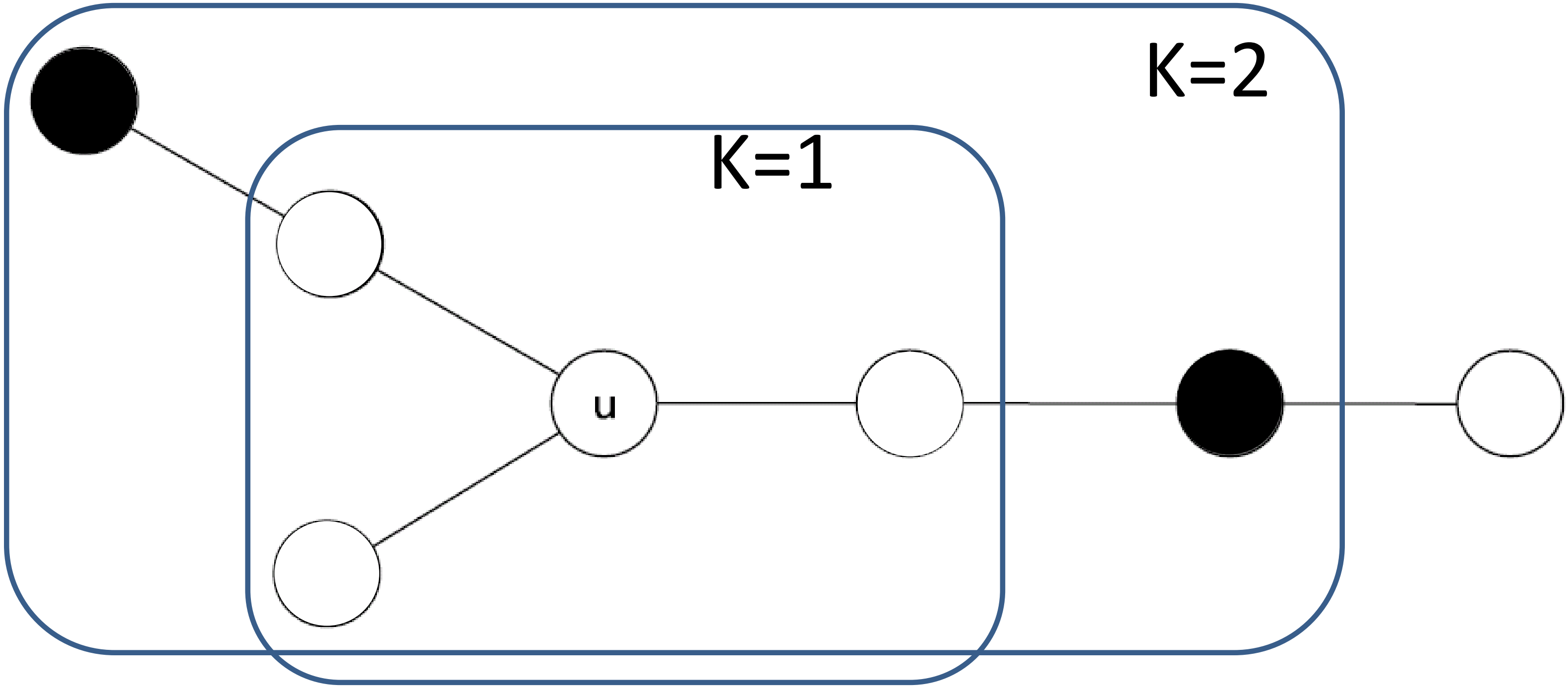




Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129

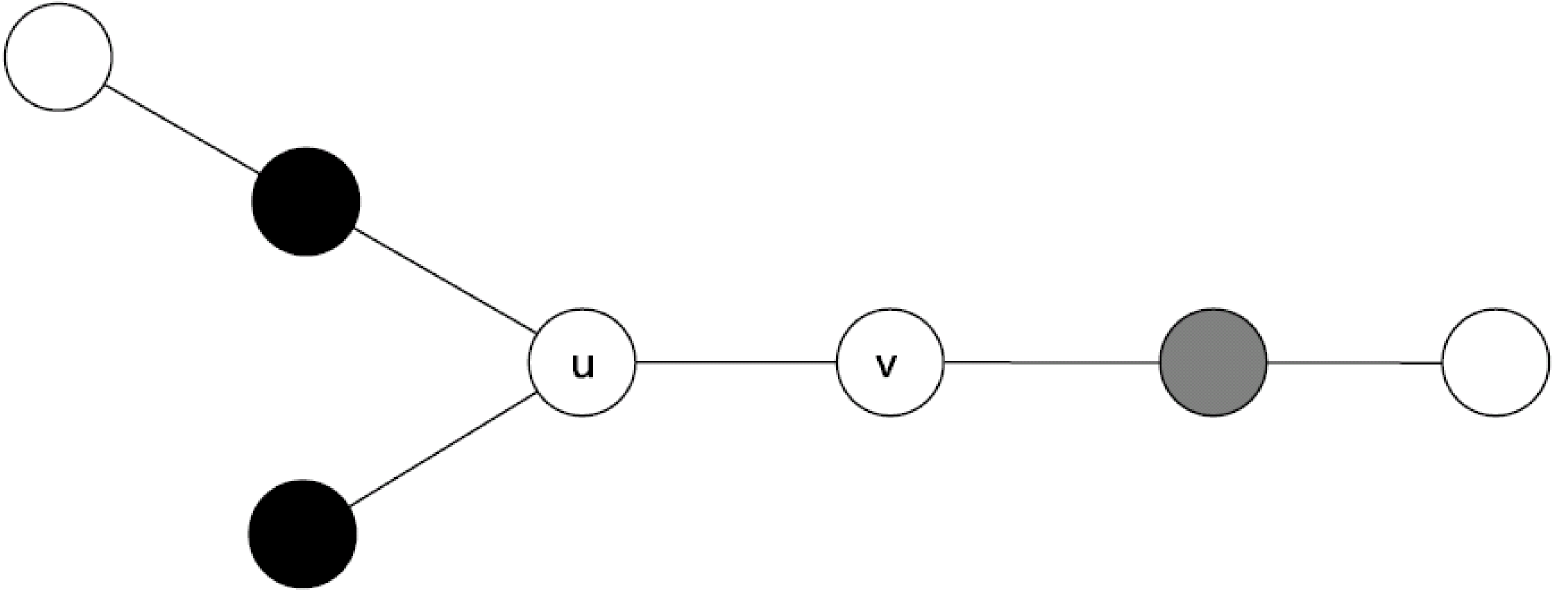


Goal:
Systematically deduce the annotation of unknown nodes *u* from the known (filled) nodes



“Direct” method for gene annotation

- K-nearest neighbors
  - – assume that a node has the same function as its neighbors

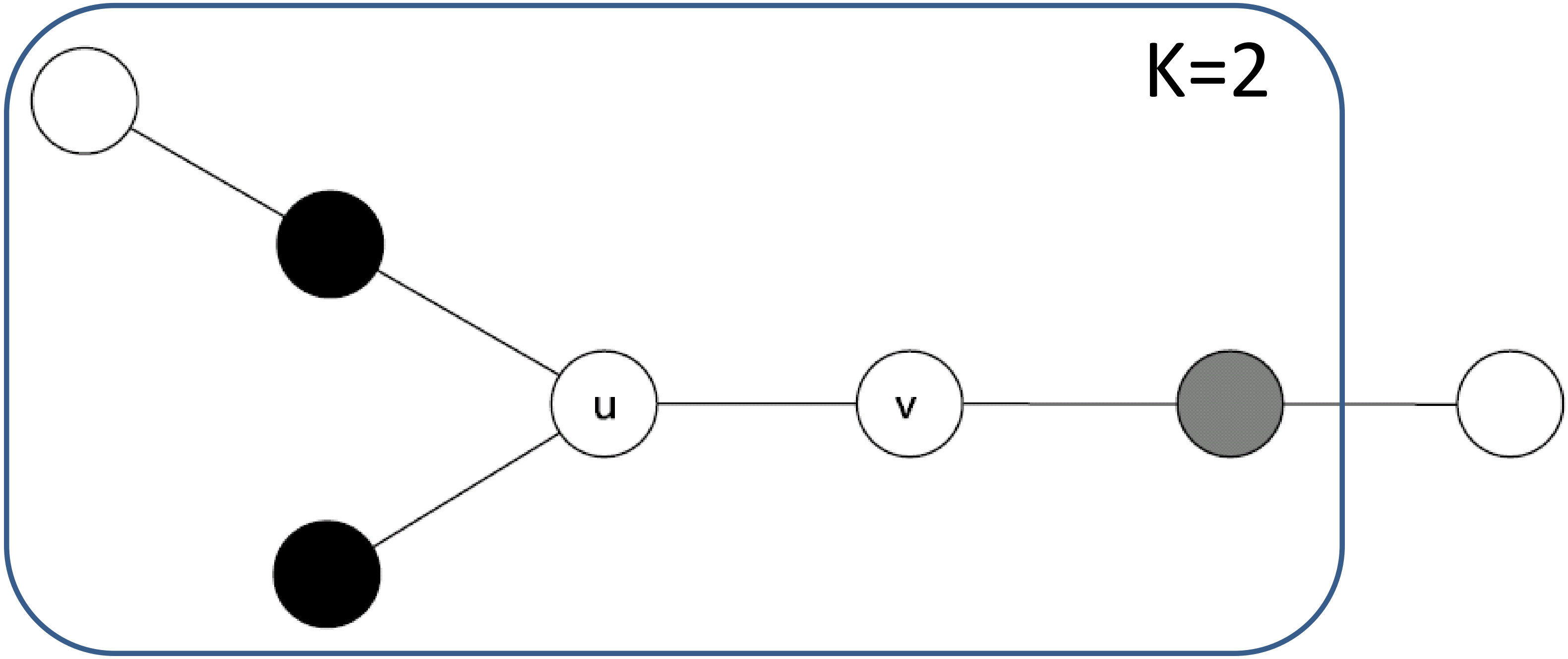


Advantages of kNN approach:

very easy to compute

Disadvantages:

how do you choose the best annotation?



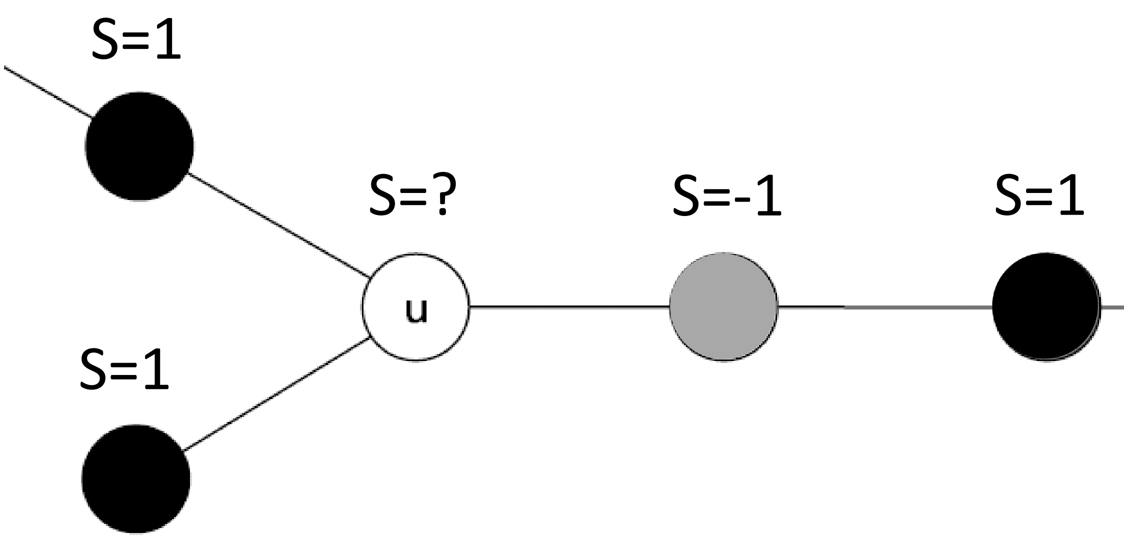
Should *u* and *v* have the same annotation?

A two-nearest neighbor approach would say yes.

But *u* seems more likely to be black and *v* more likely to be grey.

An algorithm for annotation

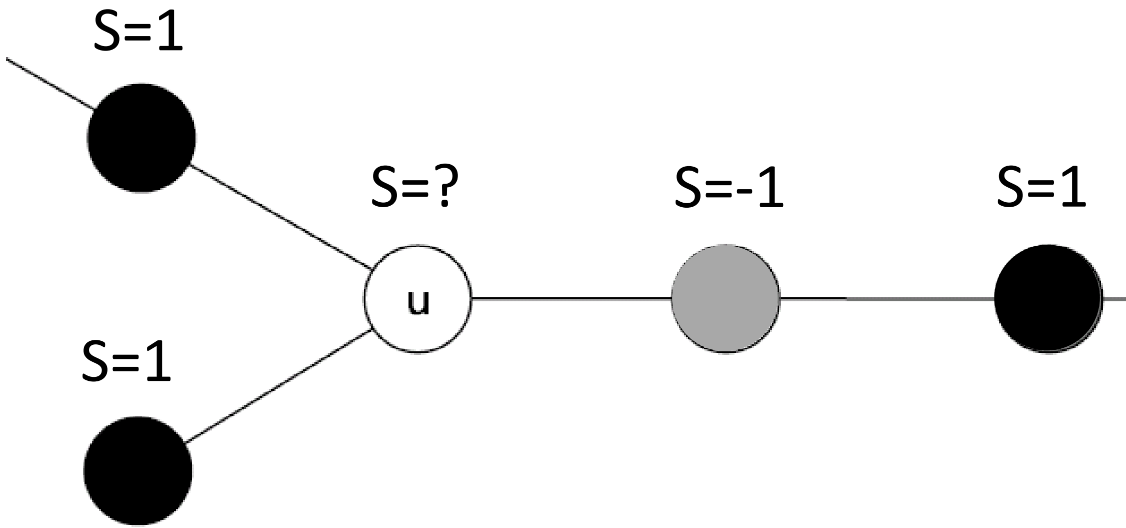
- Motivation: maximize agreement in annotation among connected nodes



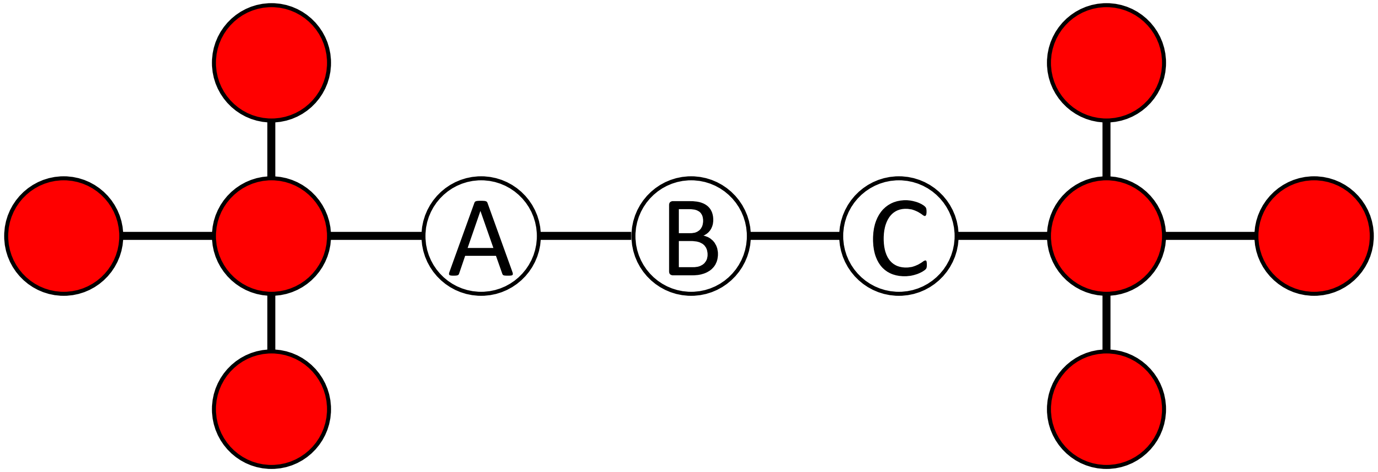
Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129

An algorithm for annotation

- For each annotation:
  - S_v=**+1** if v has the annotation, **-1** otherwise
  - Procedure: for each unassigned node u, set S_u to maximize $\Sigma S_u S_v$ for all edges (u,v)
  - iterate until convergence



Network-based prediction of protein function
Roded Sharan, Igor Ulitsky & Ron Shamir
doi:10.1038/msb4100129

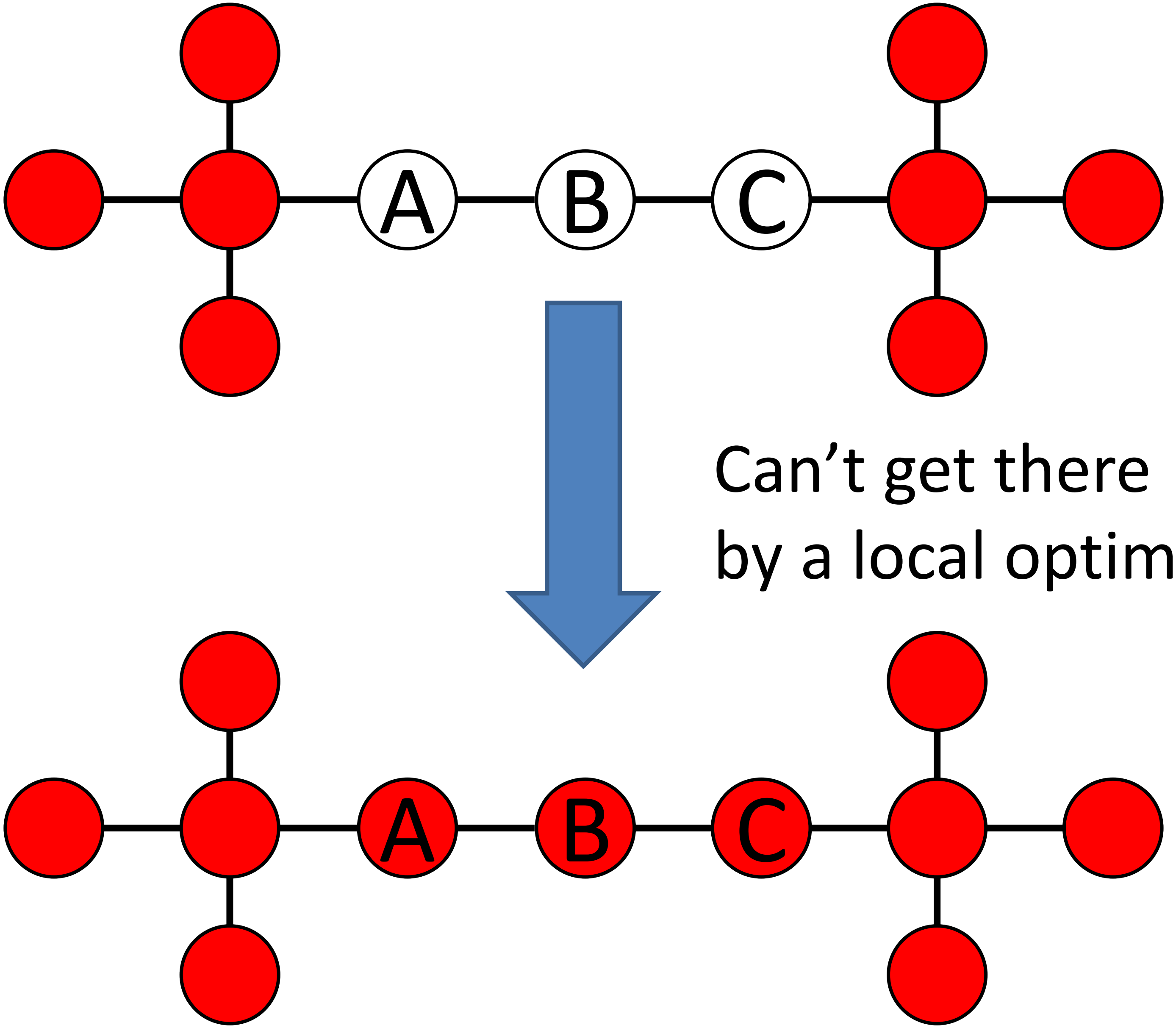


Local search may not find some good solutions.

$\Sigma S_u S_v$ does not improve if I only change A or C. Changing only B makes the score worse.

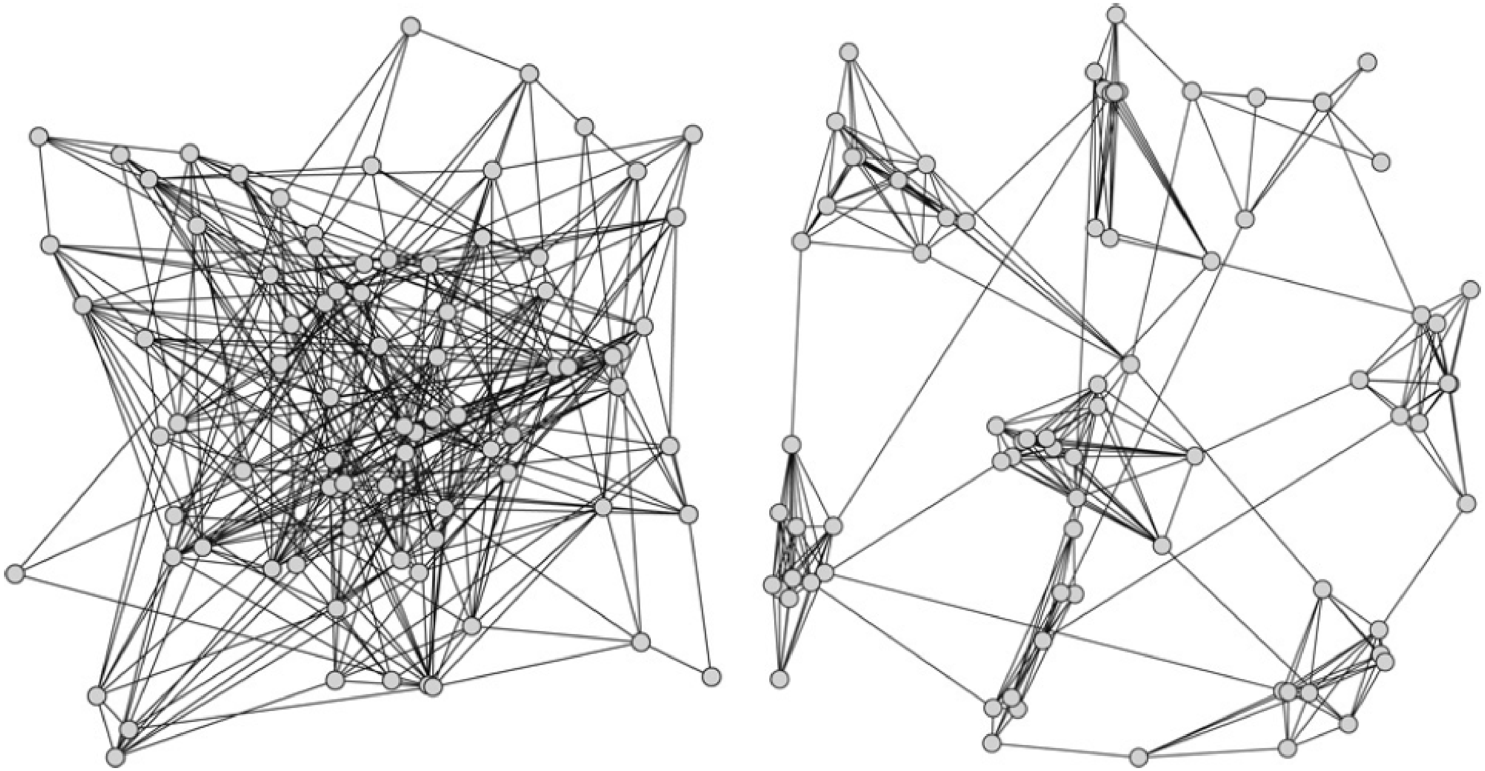
S_v=1 if v has the annotation, -1 otherwise

Goal: maximize $\Sigma S_u S_v$ for all edges (u,v)



Can’t get there
by a local optimization

Clustering Graphs

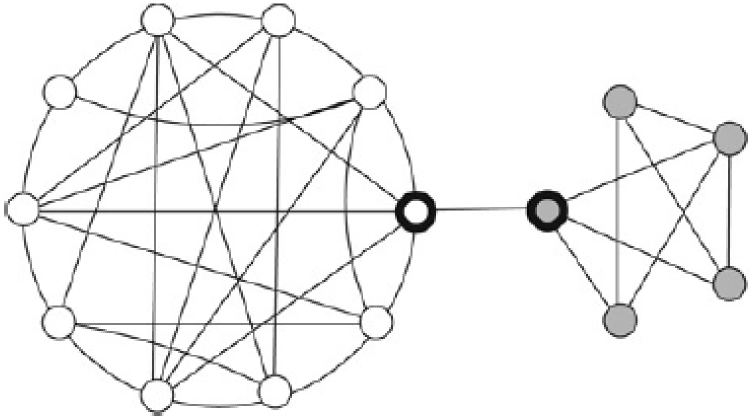


Goal: divide the graph into subgraphs each of which has lots of internal connections and few connections to the rest of the graph

Schaeffer Computer Science Review (2007) http://dx.doi.org/10.1016/j.cosrev.2007.05.001

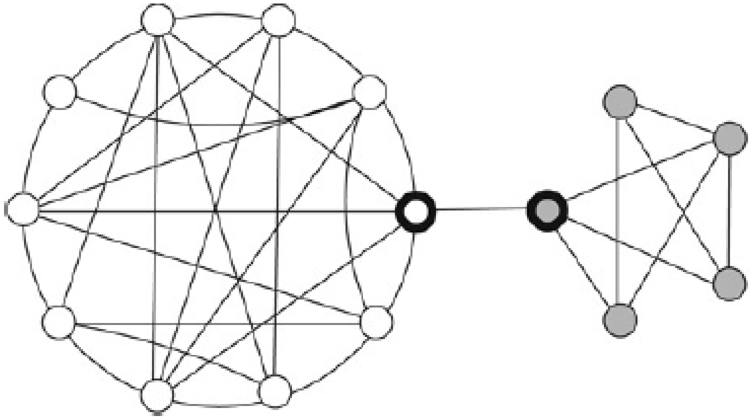
Betweeness clustering

- Edge betweeness = number (or summed weight) of shortest paths between all pairs of vertices that pass through the edge.
    - Take a weighted average if there are >1 shortest paths for the same pair of nodes.



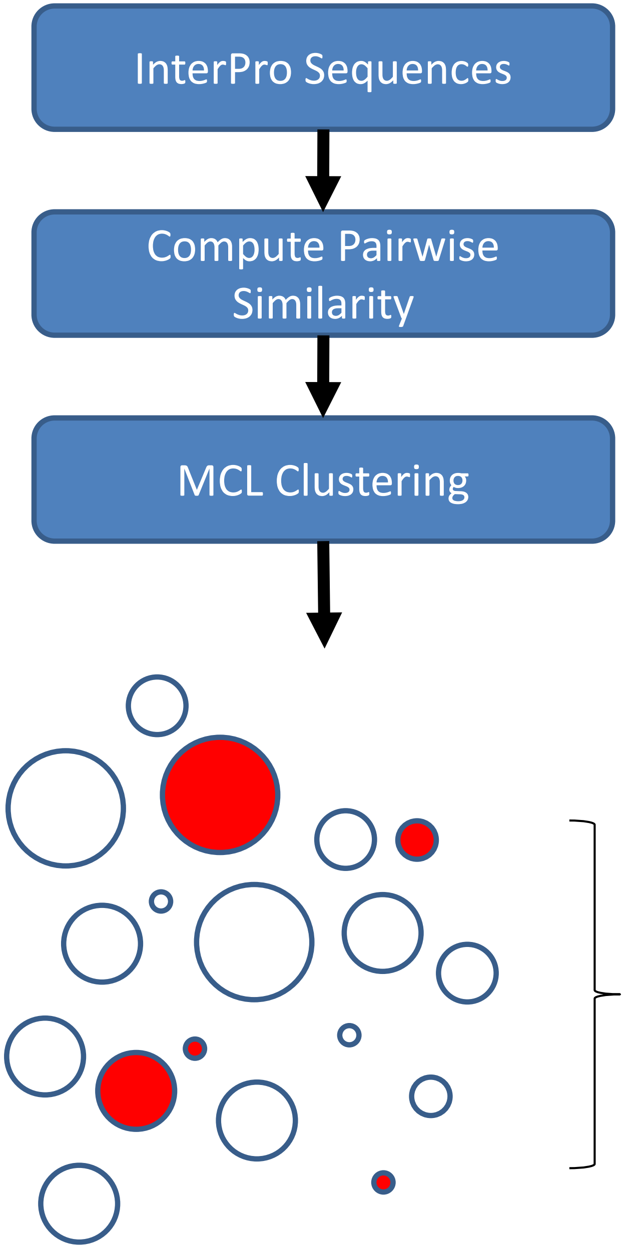
Betweeness clustering

- Repeat until max(betweeness) < threshold:
  - – Compute betweeness
  - – Remove edge with highest betweeness



Example

- Identifying protein families
- BLAST will identify proteins with shared domains, but these might not be very similar otherwise (eg: SH2, SH3 domains)



InterPro Sequences

Compute Pairwise Similarity

MCL Clustering

InterPro ID	No. of families	Domain description
IPR001064	141	Crystallin
IPR000504	110	RNA-binding region RNP-1 (RNA recognition motif)
IPR003006	107	Immunoglobulin and major histocompatibility complex domain
IPR000531	97	TonB-dependent receptor protein
IPR003015	96	Myc-type, helix–loop–helix dimerisation domain
IPR001680	76	G-protein β WD-40 repeats
IPR000561	73	EGF-like domain
IPR000169	72	Eukaryotic thiol (cysteine) proteases active sites
IPR001777	42	Fibronectin type III domain

Distinct clusters identified by MCL can still share a common domain

Enright, A. J. et al. Nucl. Acids Res. 2002 30:1575-1584; doi:10.1093/nar/30.7.1575

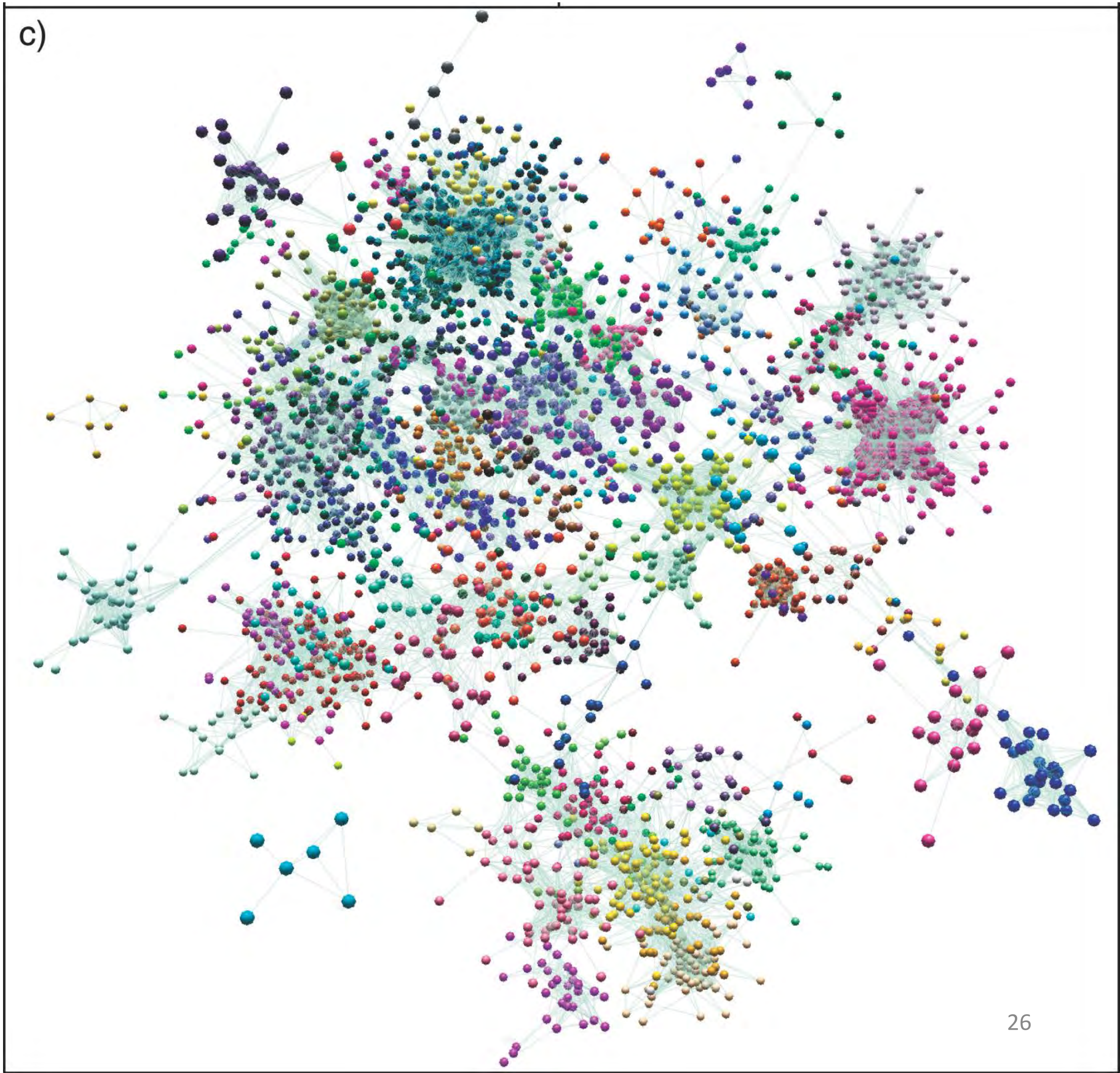
Nucleic Acids Research

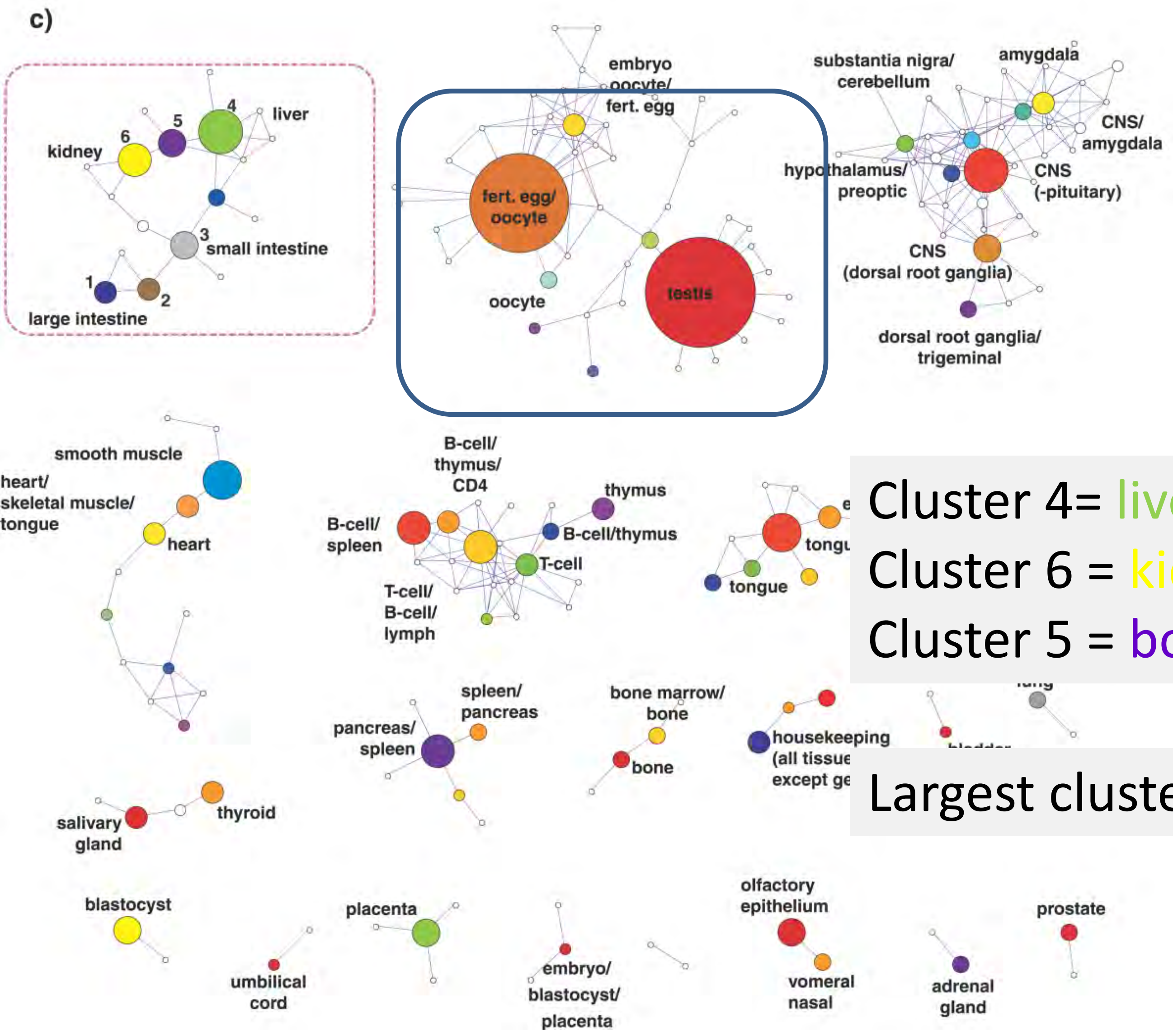
Example

- Clustering expression data for 61 mouse tissues
- Nodes = genes
- Edges = Pearson correlation coefficient > threshold
- Network gives an overview of connections not obvious from hierarchical clustering

Nodes=genes
Edges=pearson correlation of expression in mouse tissues
Clustered by MCL

Freeman, *et al.*(2007) PLoS Comput Biol 3(10): e206. doi:10.1371/journal.pcbi.0030206





Cluster 4= liver specific

Cluster 6 = kidney specific

Cluster 5 = both liver and kidney

Largest clusters are gamete-specific