Biological Network Analysis

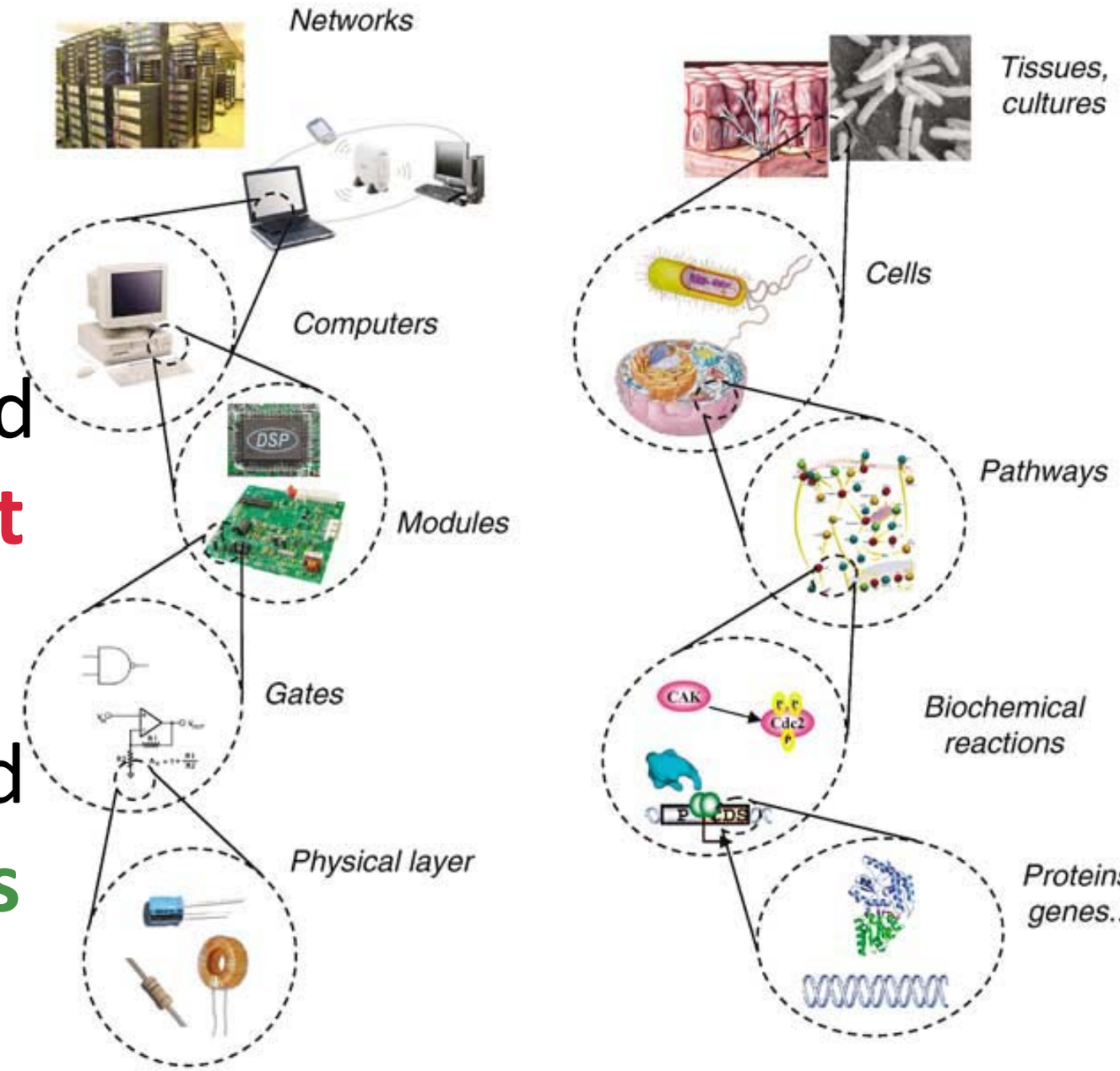
CS224W: Social and Information Network Analysis

Marinka Zitnik, Jure Leskovec, Stanford University

http://cs224w.stanford.edu

Why Biological Networks?

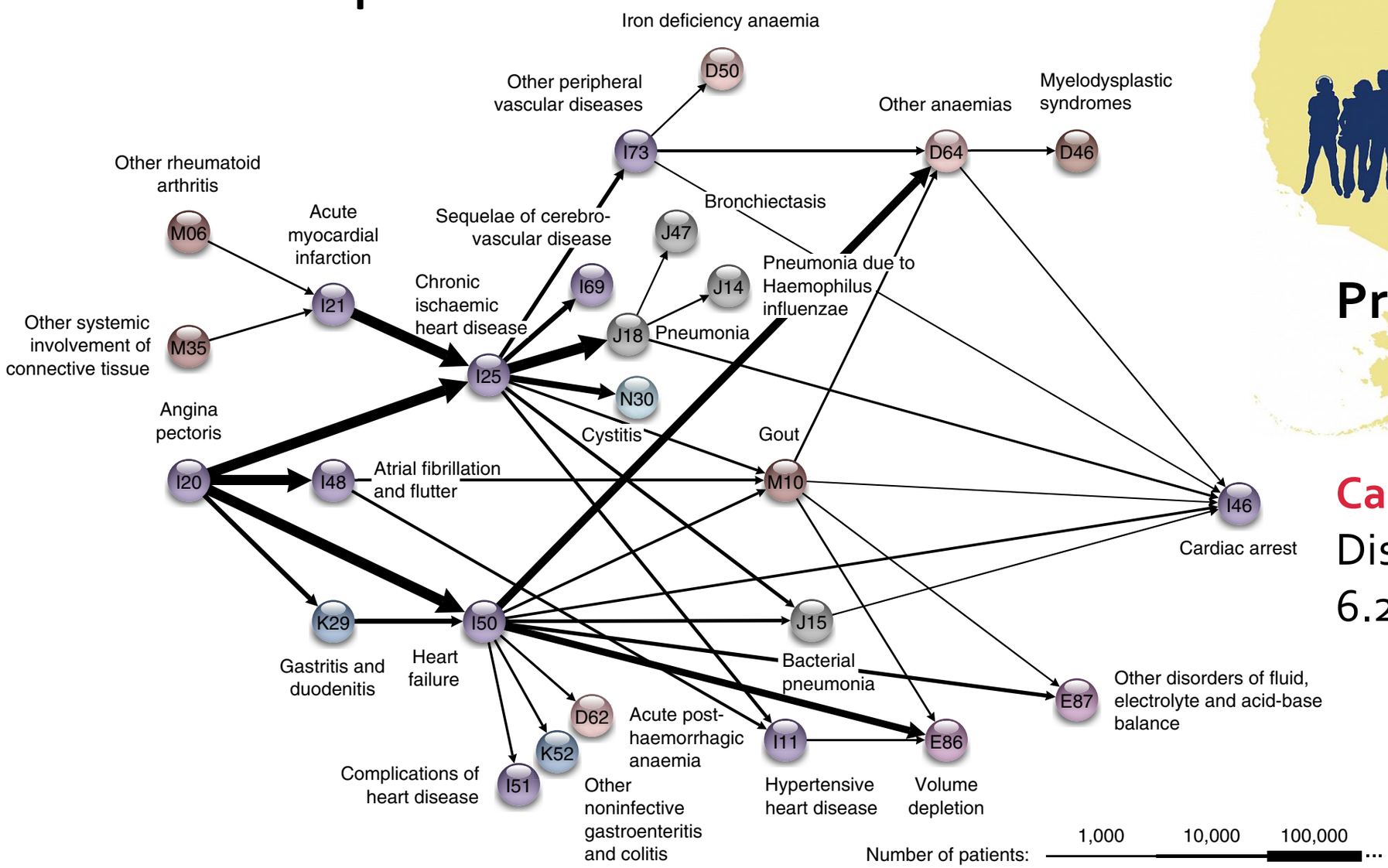
- 20th century biology was largely about **finding and describing components**
- DNA, RNA, proteins and other molecules **do not operate in isolation**
- We want to understand how **biological systems** are **organized**



Network Biology

Jensen *et al.*, Nat Commun 2014
Barabasi *et al.*, Nat Rev Genet 2011

- Network biology provides a **better understanding of life and evolution**
- **Applications in medicine**: disease diagnosis, drug development



Precision medicine

Cardiovascular cluster

Disease trajectory covering 6.2 million patients

An Example of Precision Medicine

- Precision medicine takes biology into personal grounds

Sequence a customer's genome

Link a person with disease and traits

Provide insight into ancestry of the customer

An Example of Precision Medicine

- Precision medicine takes biology into personal grounds

Networks

Sequence a customer's genome

Networks

Link a person with disease and traits

Networks

Provide insight into ancestry of the customer

Plan For Today

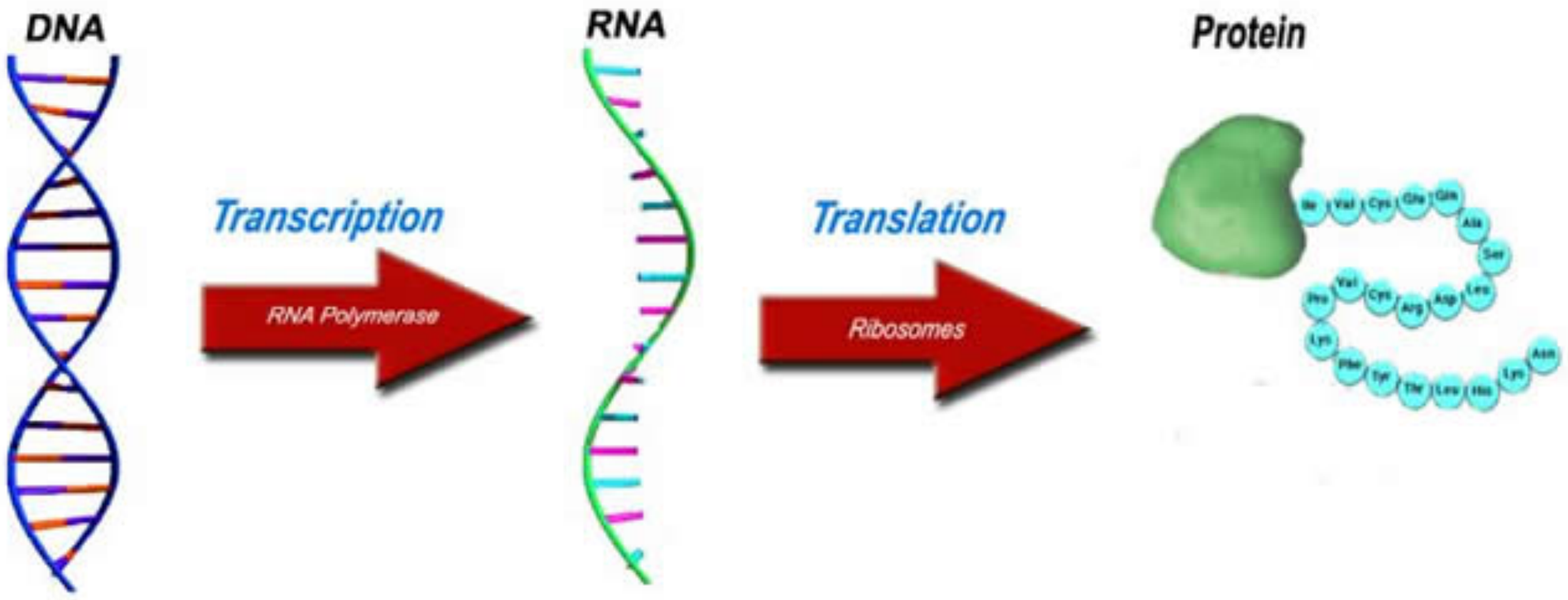
1) Very basic biology
2) Protein-protein interaction networks
3) Finding disease modules in networks
   - It is a community detection task!
4) Predicting biological attributes, such as protein functions
   - Guilt-by-association principle
   - Gene recommender systems

Plan For Today

1) **Very basic biology**
2) Protein-protein interaction networks
3) Finding disease modules in networks
   - It is a community detection task!
4) Predicting biological attributes, such as protein functions
   - Guilt-by-association principle
   - Gene recommender systems

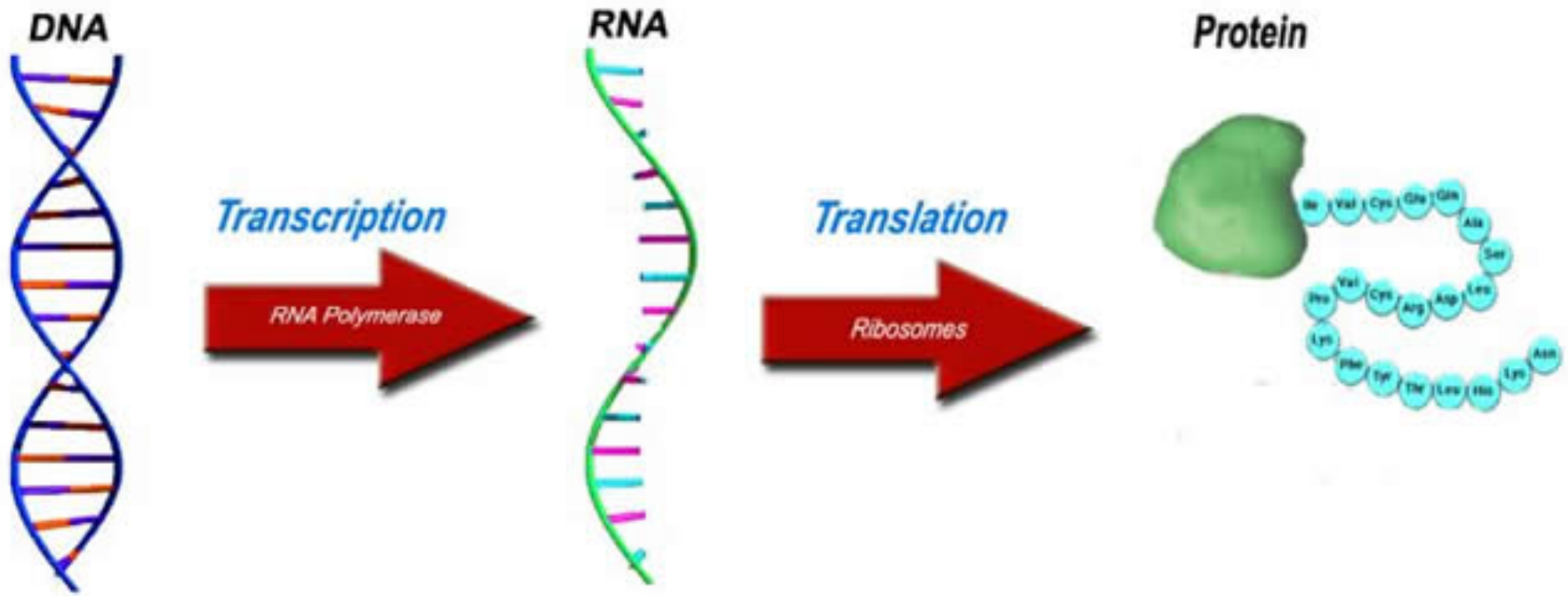
Central Dogma of Biology (1)

- **Gene** is a basic unit of heredity
- Genes are **segments of DNA** that determine properties of an organism as a whole and functions of cells within it
- Genes encode a functional unit called **protein**
- **Central dogma** describes a two-step process, **transcription** and **translation**, by which the information in DNA flows into proteins



Central Dogma of Biology (2)

- **Transcription**: Producing RNA sequence from DNA template in the **nucleus**
- **Translation**: The synthesis of a protein from RNA template in the **cytoplasm**
- Transformation of a gene into a protein is called **expression**

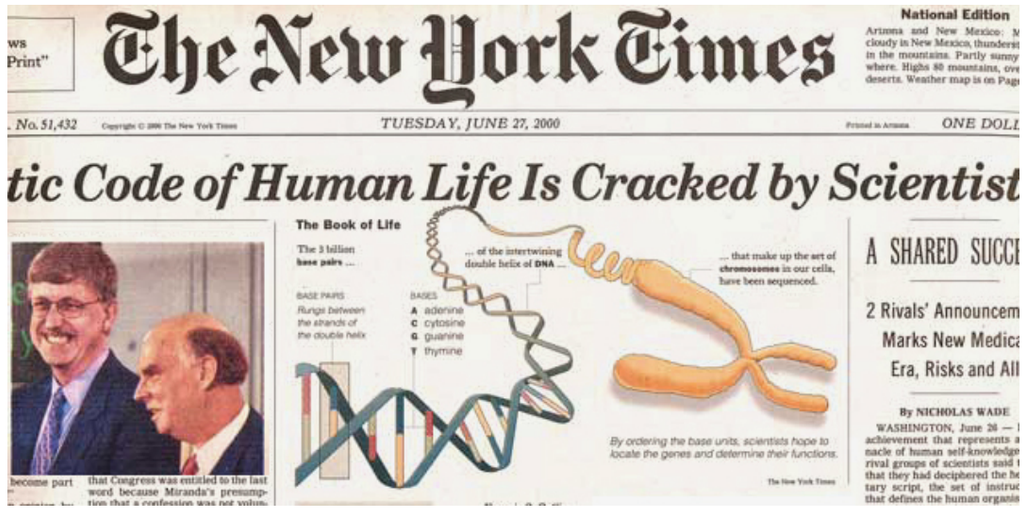


Lander *et al*, Nature 2001

The Human Genome

- **Human Genome Project:**
  1990-2003, $3 billion
- Genome consists of **23 pairs of chromosomes** and has a total of **3.2G** bp
- Average gene length is **8k** bp, there are **~25k genes**
- Only **2-3%** of the human DNA are genes, the rest of the DNA does not encode genes but has important **regulatory** roles

bp = base pair


The New York Times

National Edition

TUESDAY, JUNE 27, 2000

ONE DOLL

tic Code of Human Life Is Cracked by Scientist

The Book of Life

The 3 billion base pairs ...

... of the intertwining double helix of DNA ...

... that make up the set of chromosomes in our cells, have been sequenced.

BASE PAIRS Rungs between the strands of the double helix

BASES A adenine C cytosine G guanine T thymine

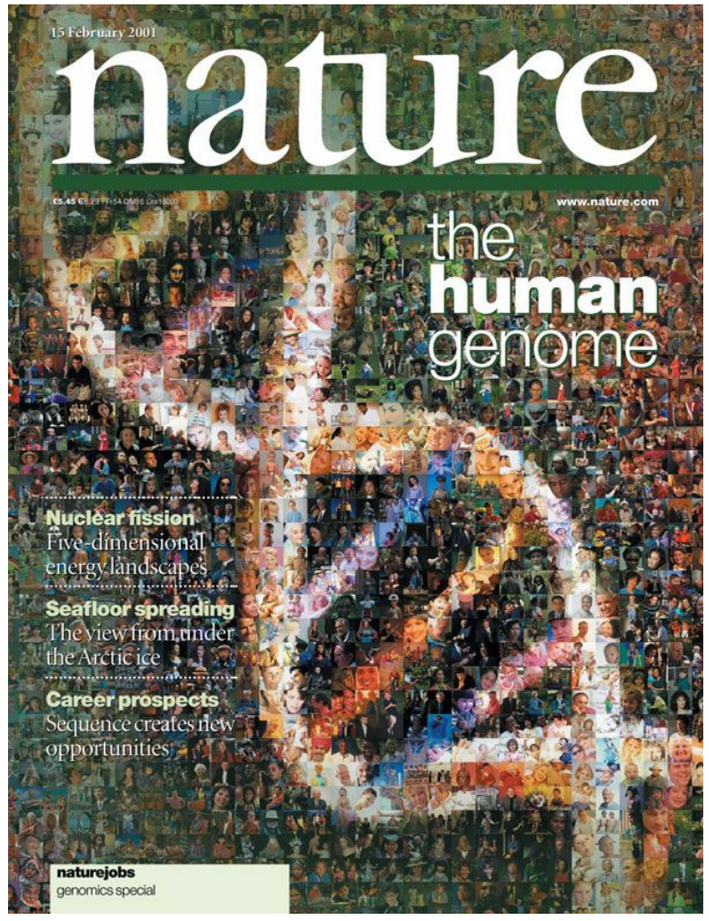
By ordering the base units, scientists hope to locate the genes and determine their functions.

A SHARED SUCCE

2 Rivals' Announcem Marks New Medica Era, Risks and All

By NICHOLAS WADE

WASHINGTON, June 26 —



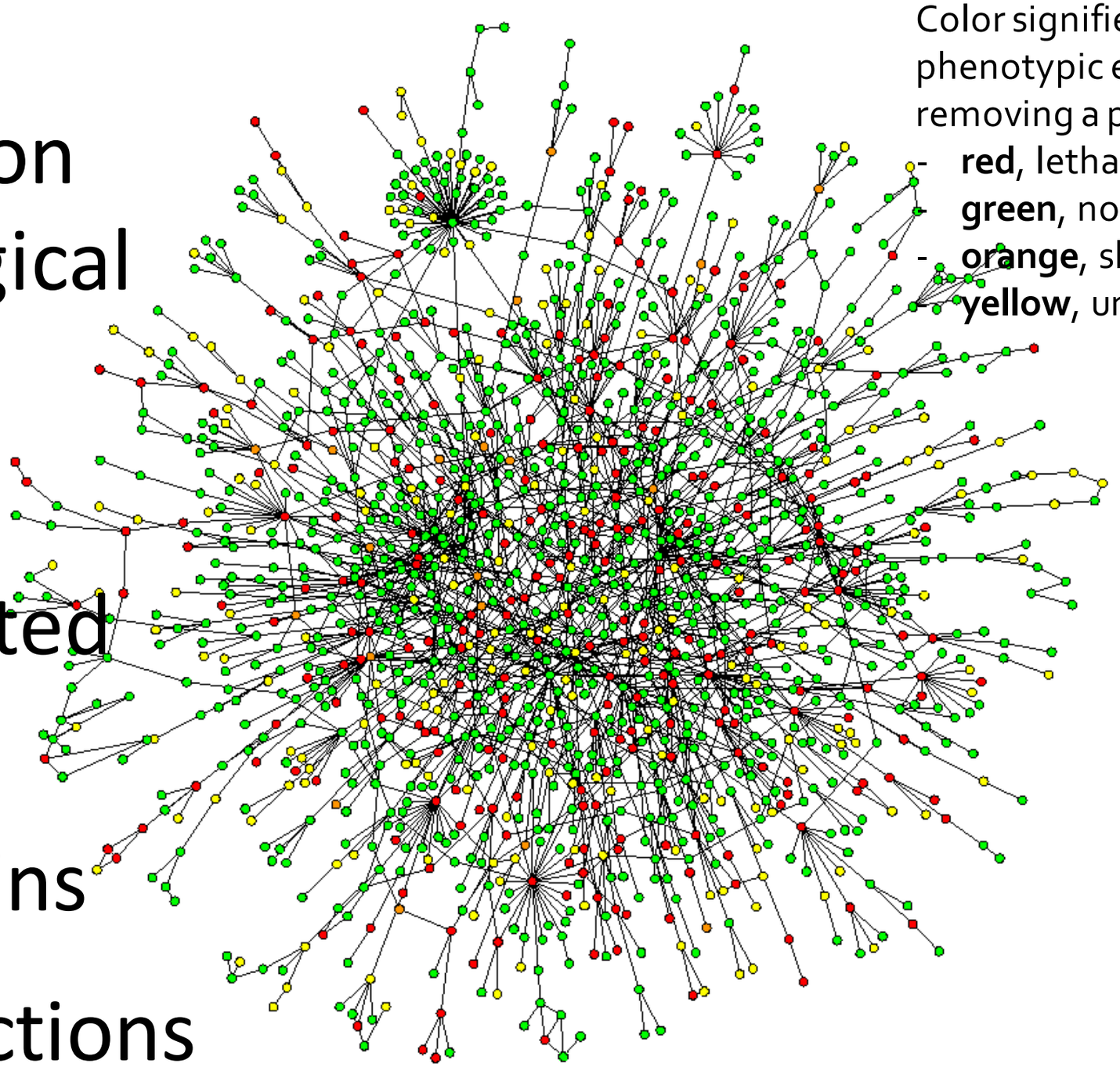
Plan For Today

1) **Very basic biology**
2) Protein-protein interaction networks
3) Finding disease modules in networks
   - It is a community detection task!
4) Predicting biological attributes, such as protein functions
   - Guilt-by-association principle
   - Gene recommender systems

Jeong et al., Nature 2001

Protein Interaction Networks

- A very common type of biological networks
- Undirected, binary/weighted network
- **Nodes:** proteins
- **Edges:** interactions



Color signifies the phenotypic effect of removing a protein
- **red**, lethal
- **green**, non-lethal
- **orange**, slow growth
- **yellow**, unknown

Yeast protein-protein interaction (PPI) network

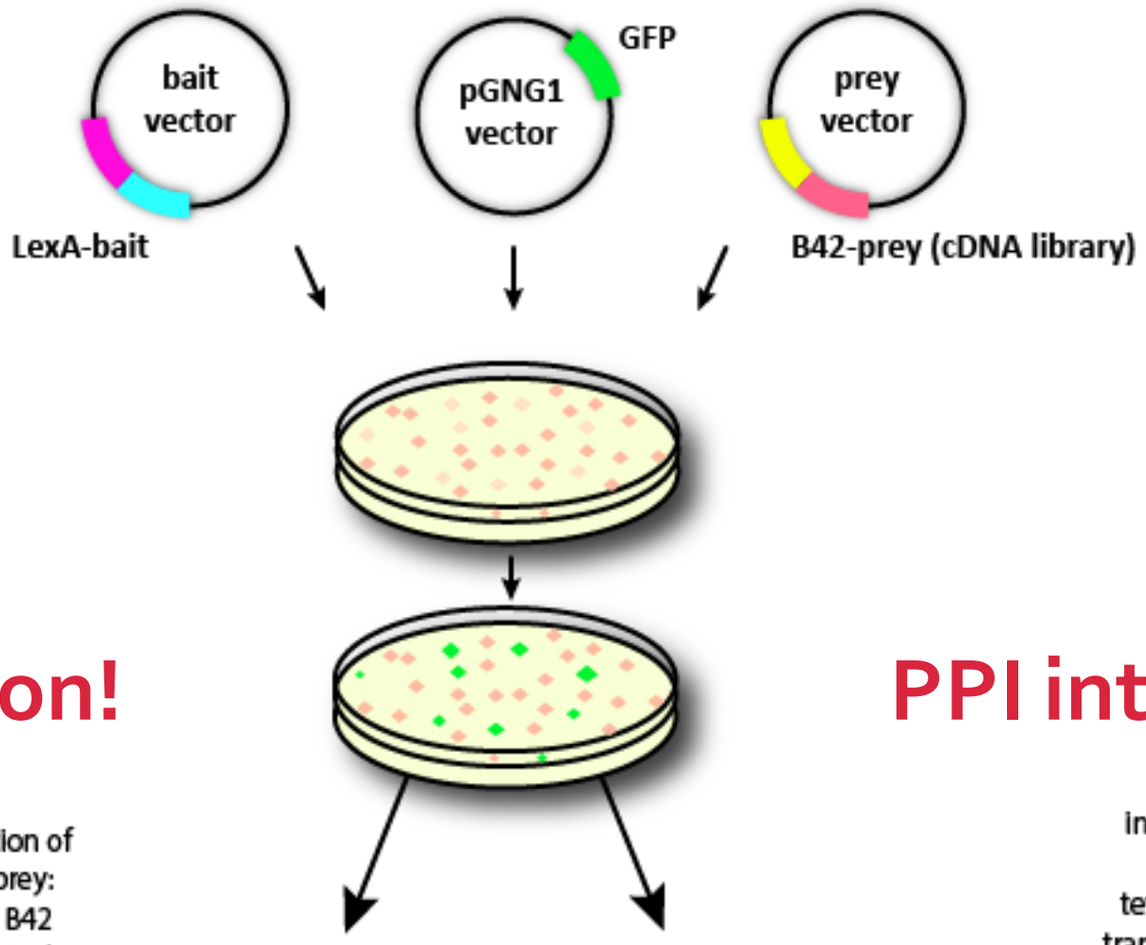
Protein-Protein Interactions

- How do we know that a pair of proteins interact?
- A **complex** containing these two proteins has been crystallized
- **High throughput** screening methods enable **rapid, parallel** acquisition of experimental data
  - **Yeast two-hybrid system**
- Problems with high throughput methods:
  - **False positive** and **false negative** edges
  - Networks are **incomplete** and **noisy**

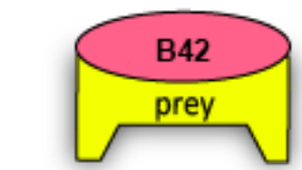
Yeast Two-Hybrid Screening (Y2H)

- Classical screening technology for the study of PPI

Checking for interaction between two proteins, called here *Bait* and *Prey*

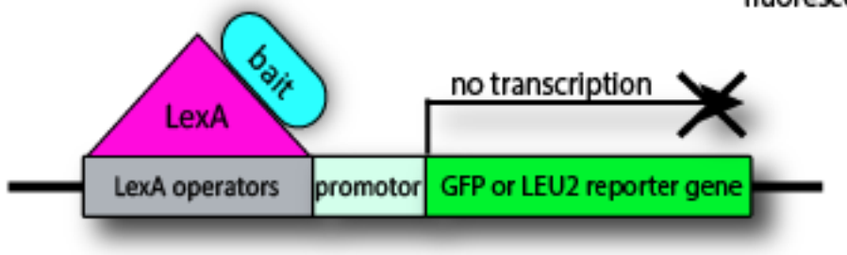


No PPI interaction!



no interaction of bait and prey: LexA and B42 are separated

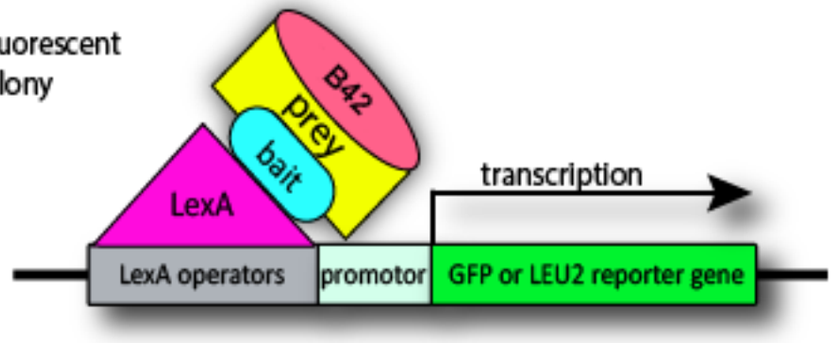
colony is not fluorescent



PPI interaction!

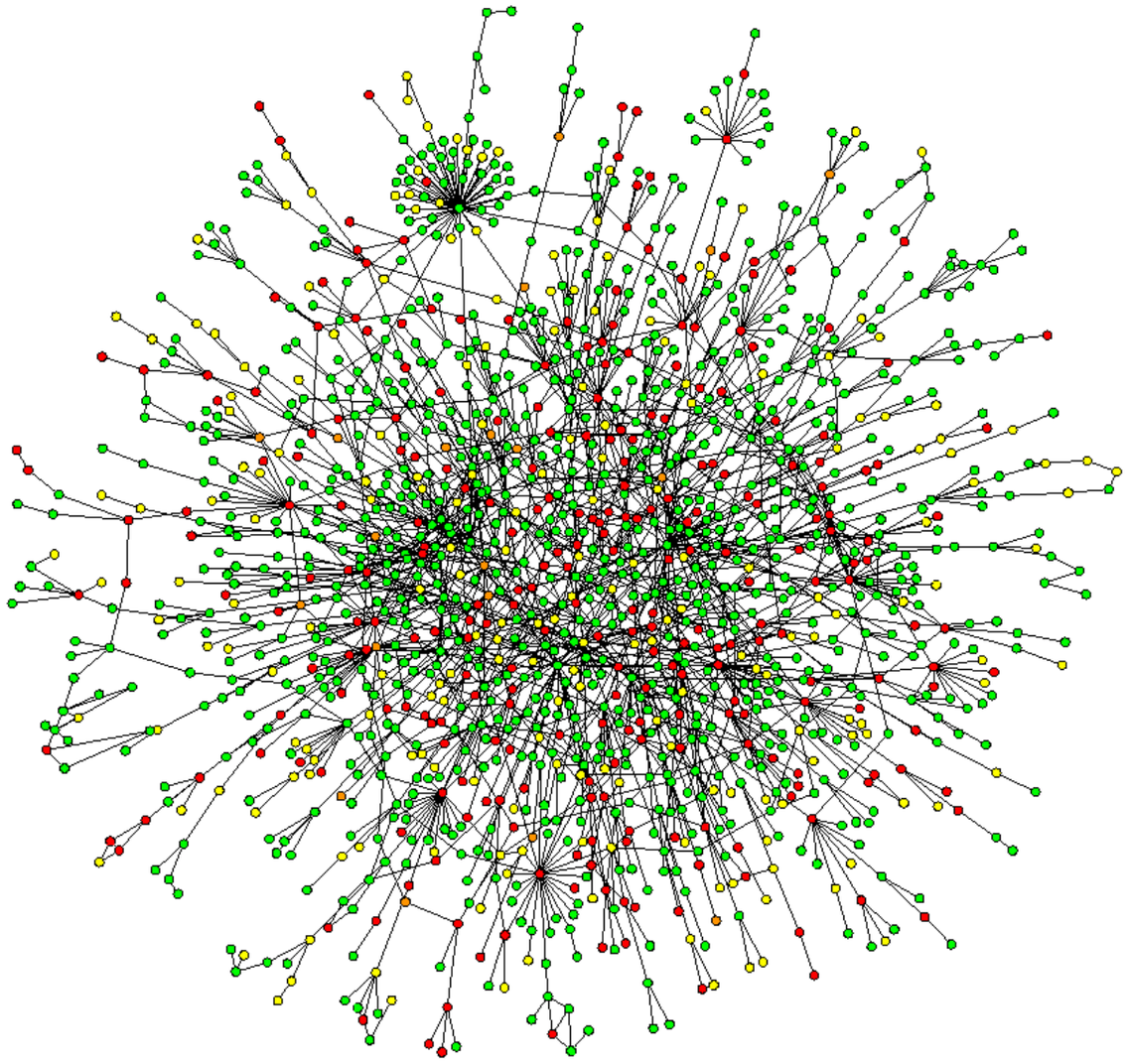
green fluorescent colony

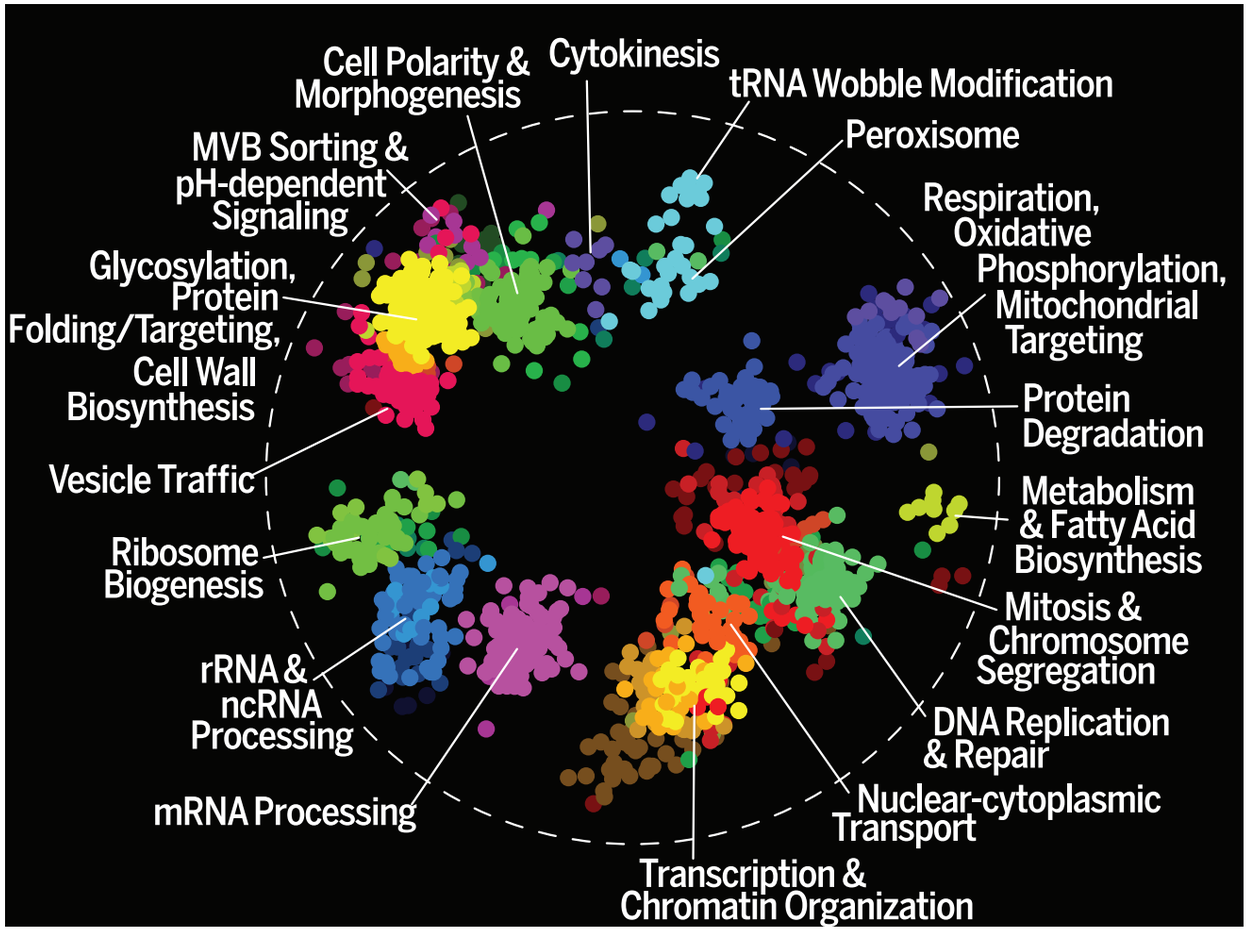
Interaction of bait and prey: LexA and B42 are tethered to one another and transactivate the reporter gene



Jeong et al., Nature 2001
Costanzo et al., Science 2016

Protein Interaction Network



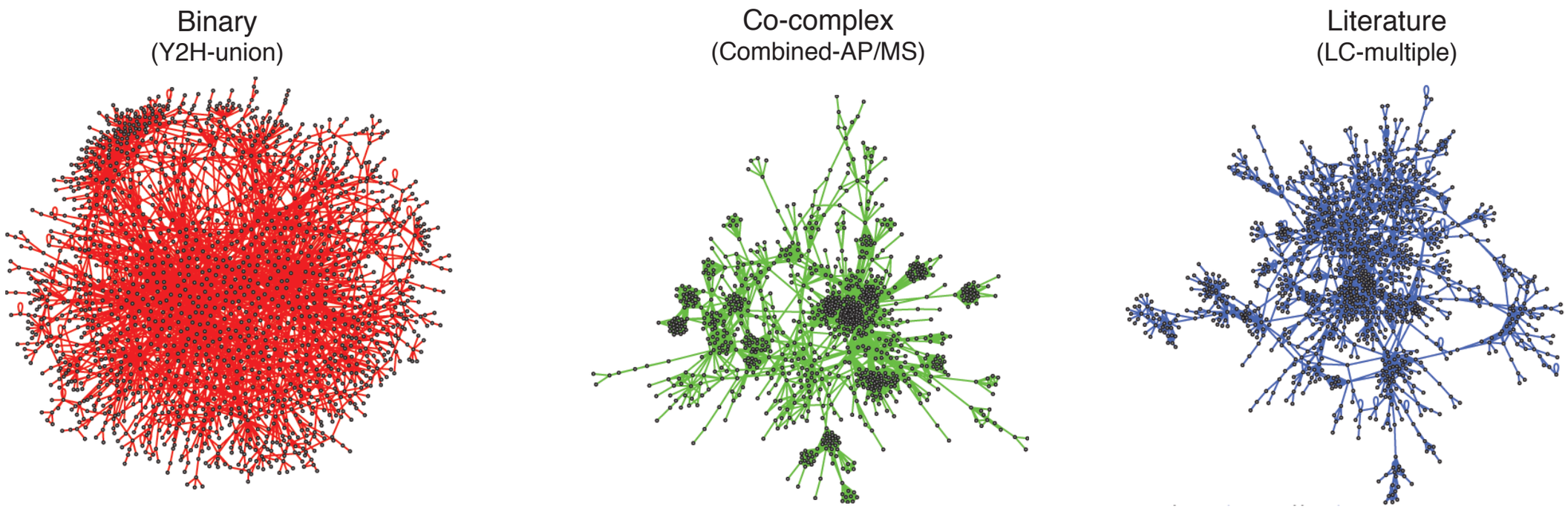


Is there a relation between network structure and biological function and disease?

Yu et al., Science 2008

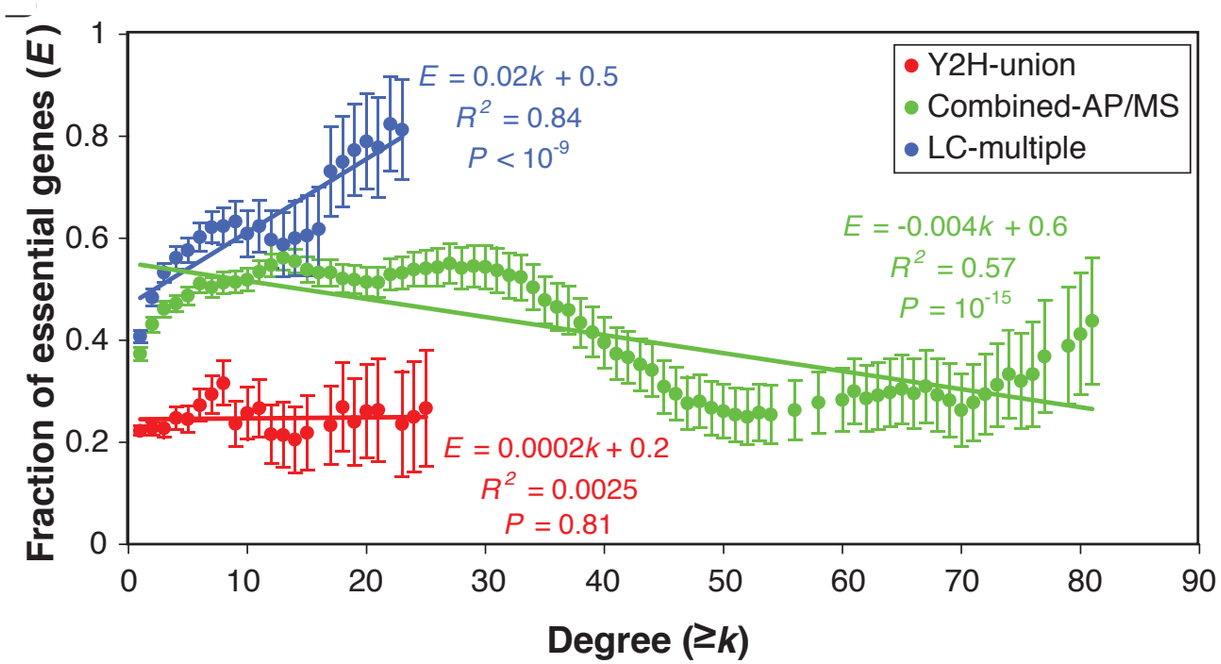
Protein Interaction Networks

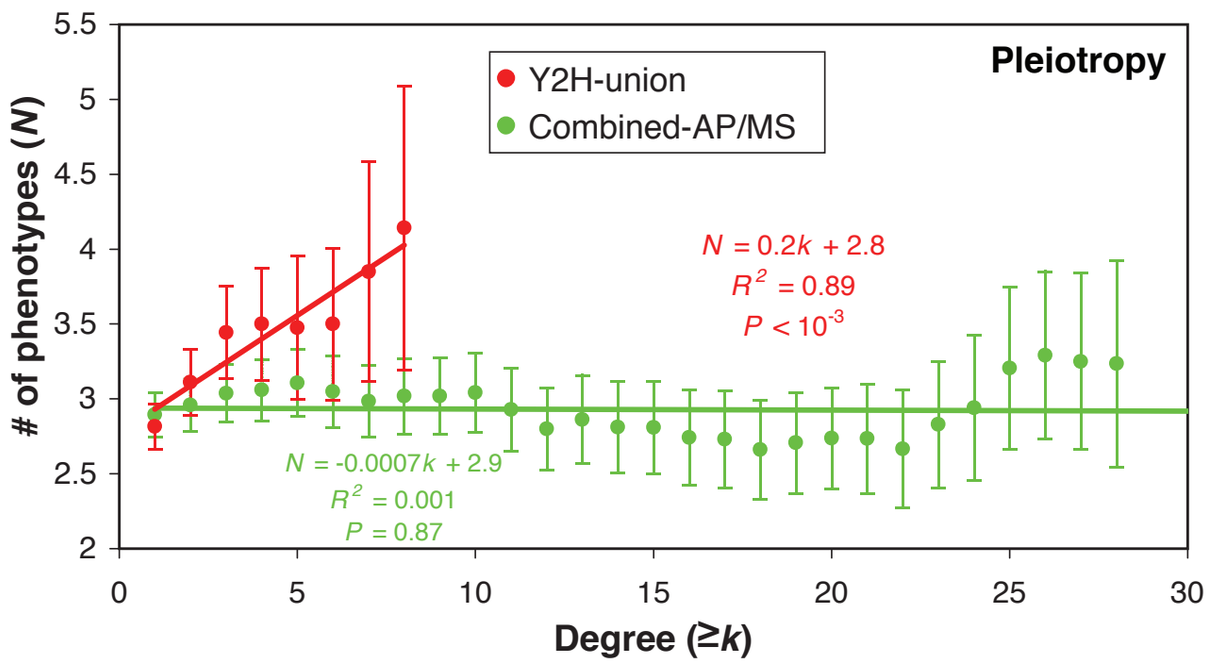
- Data:
  - Three yeast protein-protein interaction (PPI) networks
  - List of **essential** yeast proteins, these proteins form a minimal protein set required for a living cell
  - Mapping of proteins to **phenotypes** (i.e., observable traits, such as diseases) associated with **deletion of each protein**



Hub Proteins

- **Hub proteins:** 20% nodes in the network with the highest degree
- Observations:
  - **Hub proteins** associate with **essential proteins**, confirmed in many but not all networks
  - **Hub proteins** associate with **larger numbers of phenotypes** than non-hub proteins





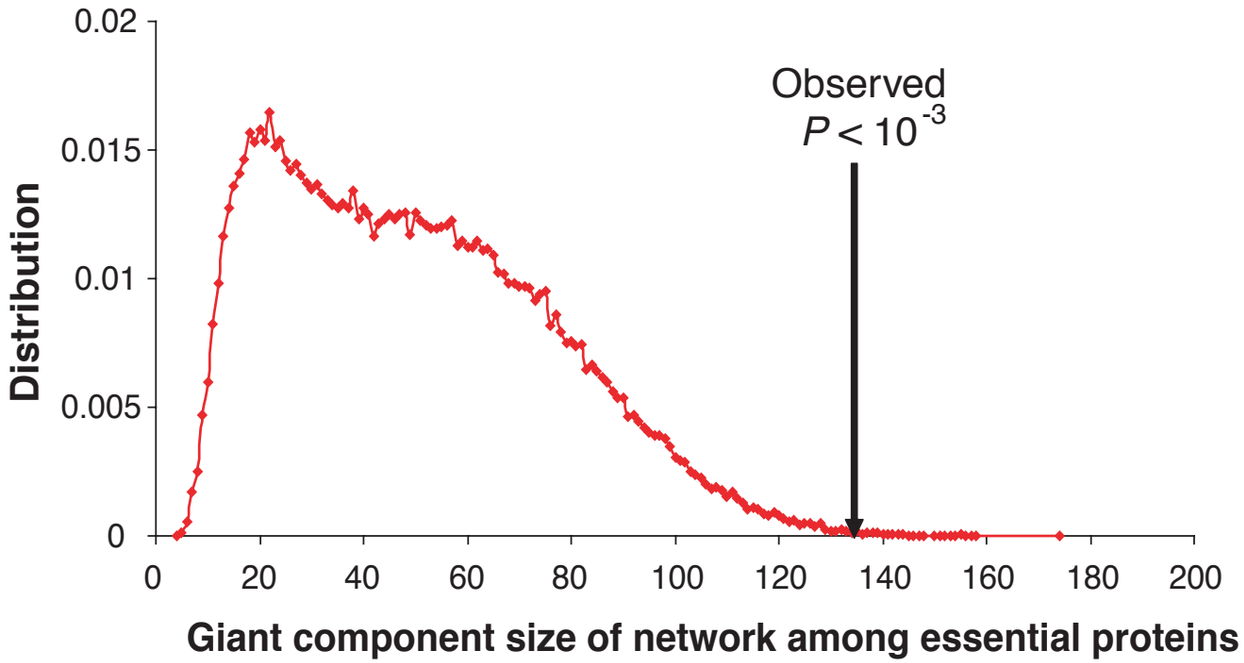
Subnetwork of Essential Proteins

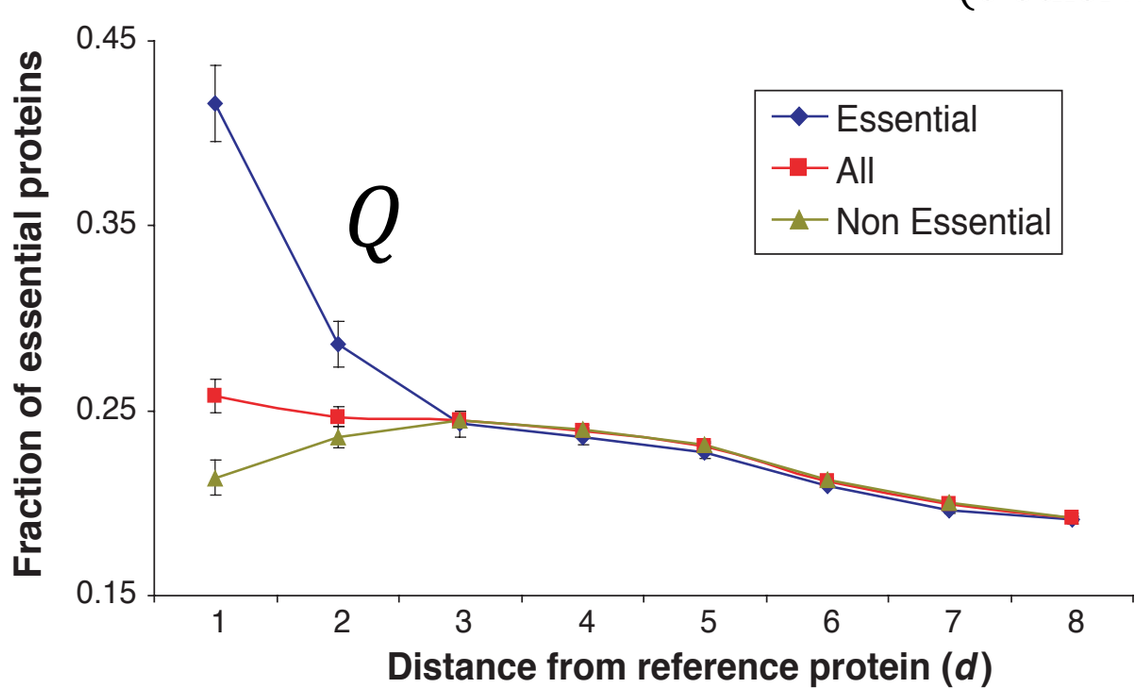
- For a protein p_1, take the **fraction of essential proteins** among all proteins whose distance to protein p_1 is equal to d:

$$Q(p_1, d) = \sum_{p \in S_d(p_1)} \frac{I(p \text{ is essential})}{|S_d(p_1)|}$$

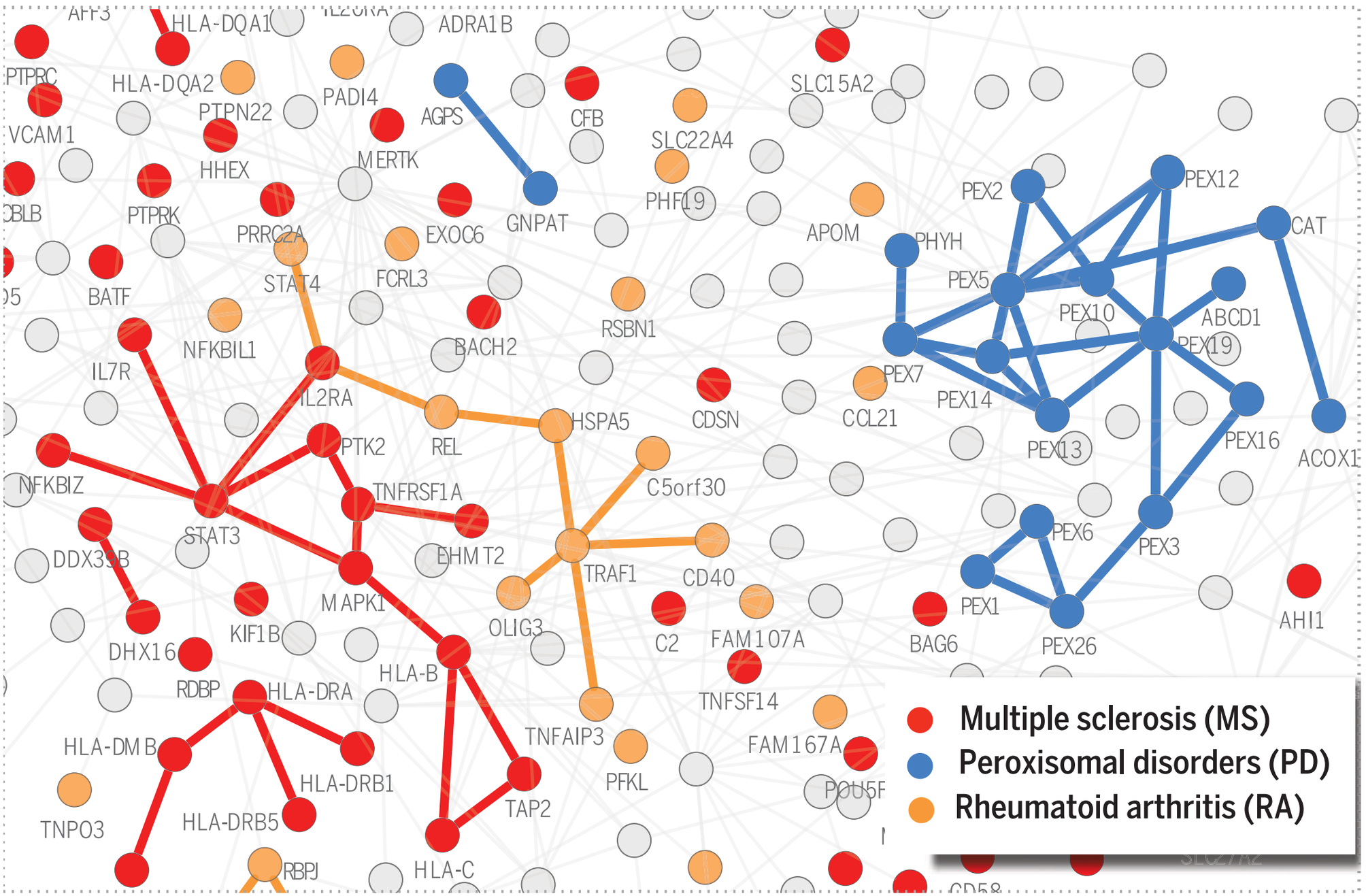
Note:

$$I(X) = \begin{cases} 1 \text{ if X is true} \\ 0 \text{ otherwise} \end{cases}$$





Disease Proteins



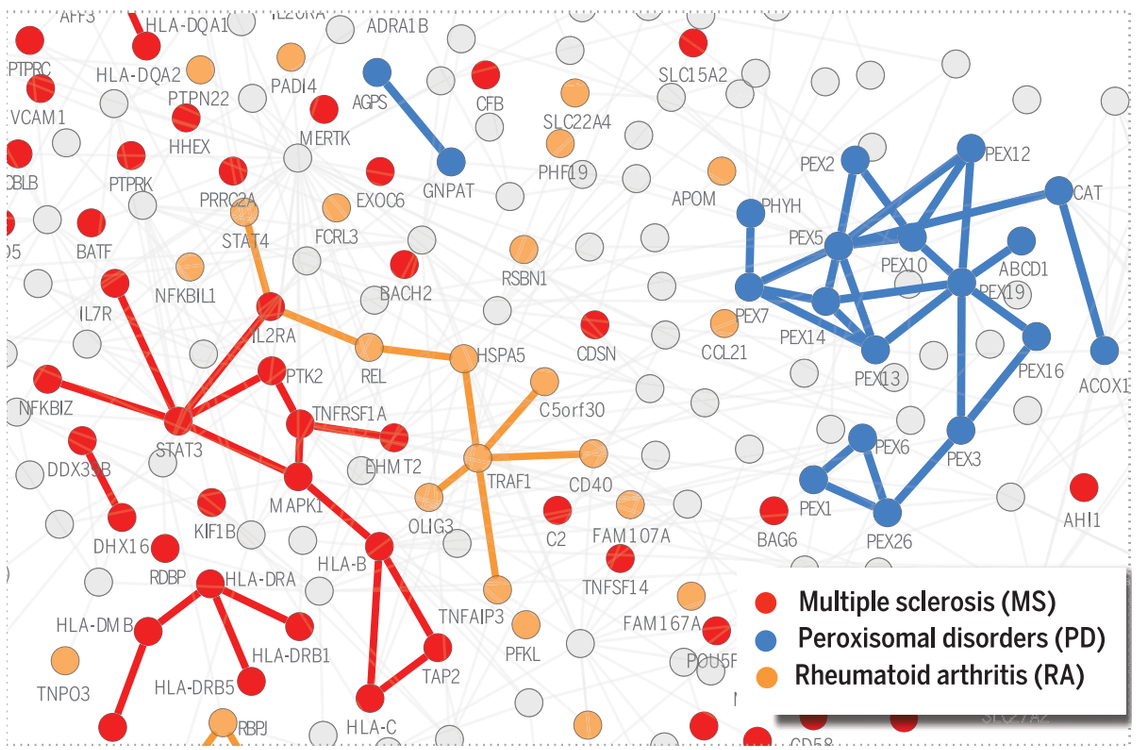
Nodes: proteins

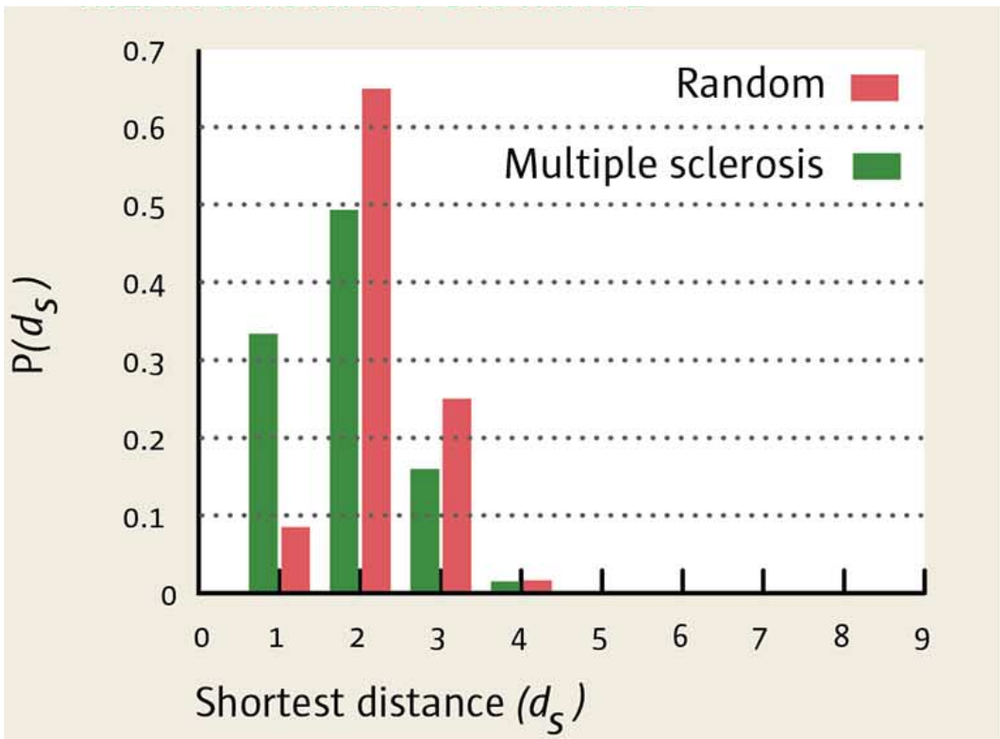
Edges: PPI interactions

Node colors: protein-disease associations

Disease Proteins

- Given **disease proteins**, compute **shortest path distance** d_s of each disease protein to the closest disease protein
- $P(d_s)$ is **shifted towards smaller** d_s compared to the random expectation $P^{\text{rand}}(d_s)$
  - ⇒ Disease proteins **agglomerate** in one network neighborhood





Menche et al., Science 2015

Disease Proteins

- **Disease module principle:** Disease proteins **tend to cluster** in one network neighborhood
- **Local interaction principle:** Disease proteins **tend to interact** with each other
- Mutations in interacting proteins tend to lead to **diseases with similar phenotypes (i.e., symptoms)**

Menche et al., Science 2015

Disease Proteins

- **Disease module principle:** Disease proteins **tend to cluster** in one network neighborhood
- **Local interaction principle:** Disease proteins **tend to interact** with each other
- Mutations in interacting proteins tend to lead to **diseases with similar phenotypes (i.e., symptoms)**

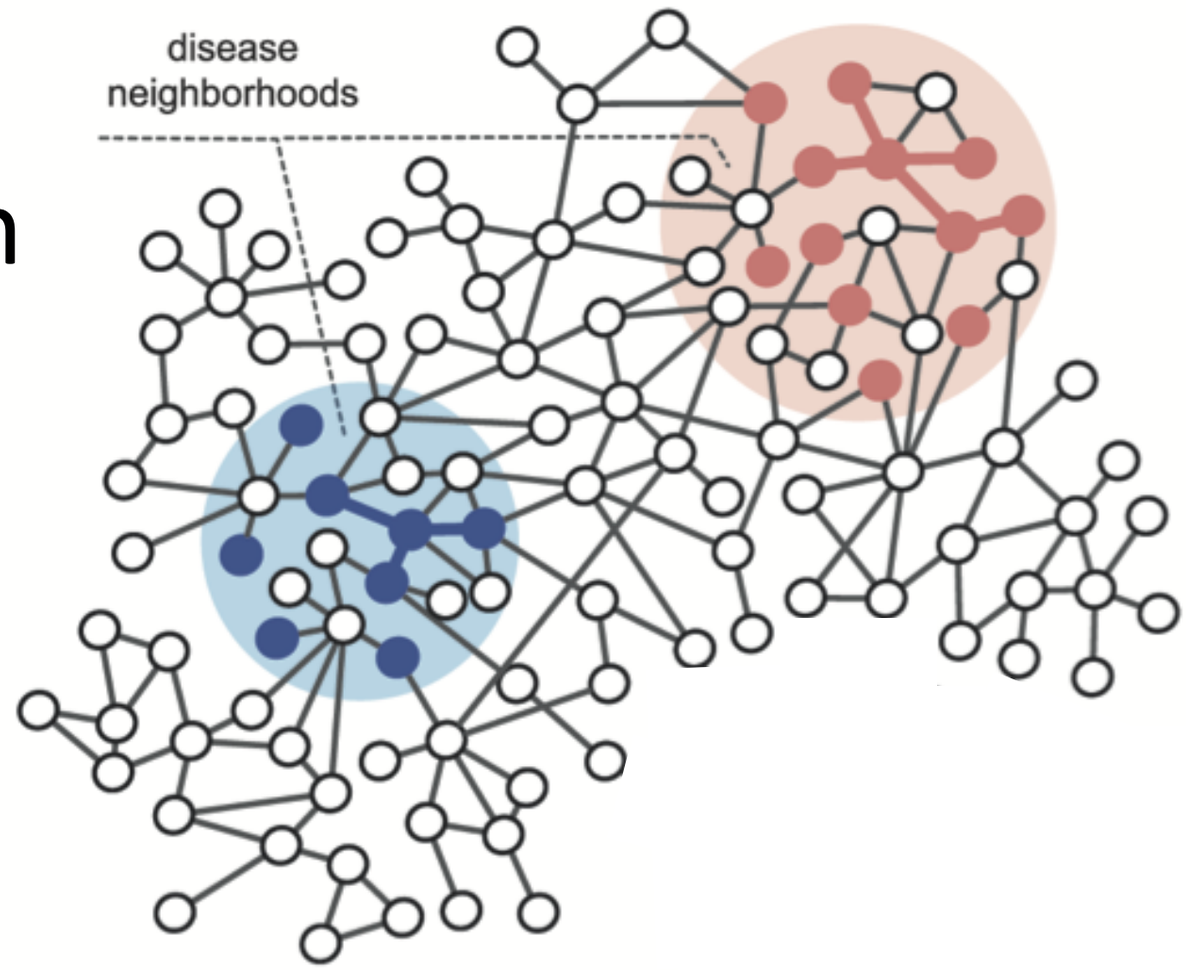
Can we use these principles to detect disease modules in biological networks?

Plan For Today

1) **Very basic biology**
2) **Protein-protein interaction networks**
3) **Finding disease modules in networks**
   - It is a community detection task!
4) Predicting biological attributes, such as protein functions
   - Guilt-by-association principle
   - Gene recommender systems

Disease Modules

- By **disease module principle**, disease proteins are localized in network neighborhoods

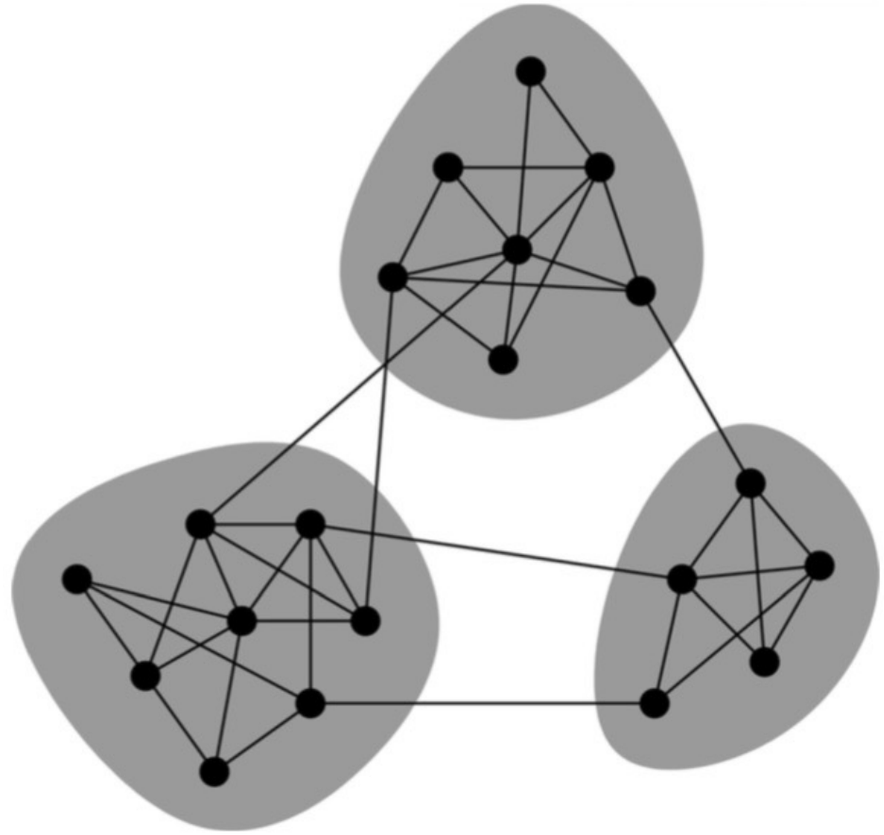


- **Disease module D:**
  - **Set of proteins** involved in disease D
  - **Abnormalities/mutations** in these proteins cause a disease to develop

Disease modules, communities, clusters, groups

Finding Disease Modules

- **Goal: Find disease modules** in a PPI network
- This is a **community detection problem**
- Many community detection methods:
  - **Girvan-Newman** method
  - **Clique percolation** method
  - **Louvain** method
  - **Spectral** clustering
  - **Link** clustering

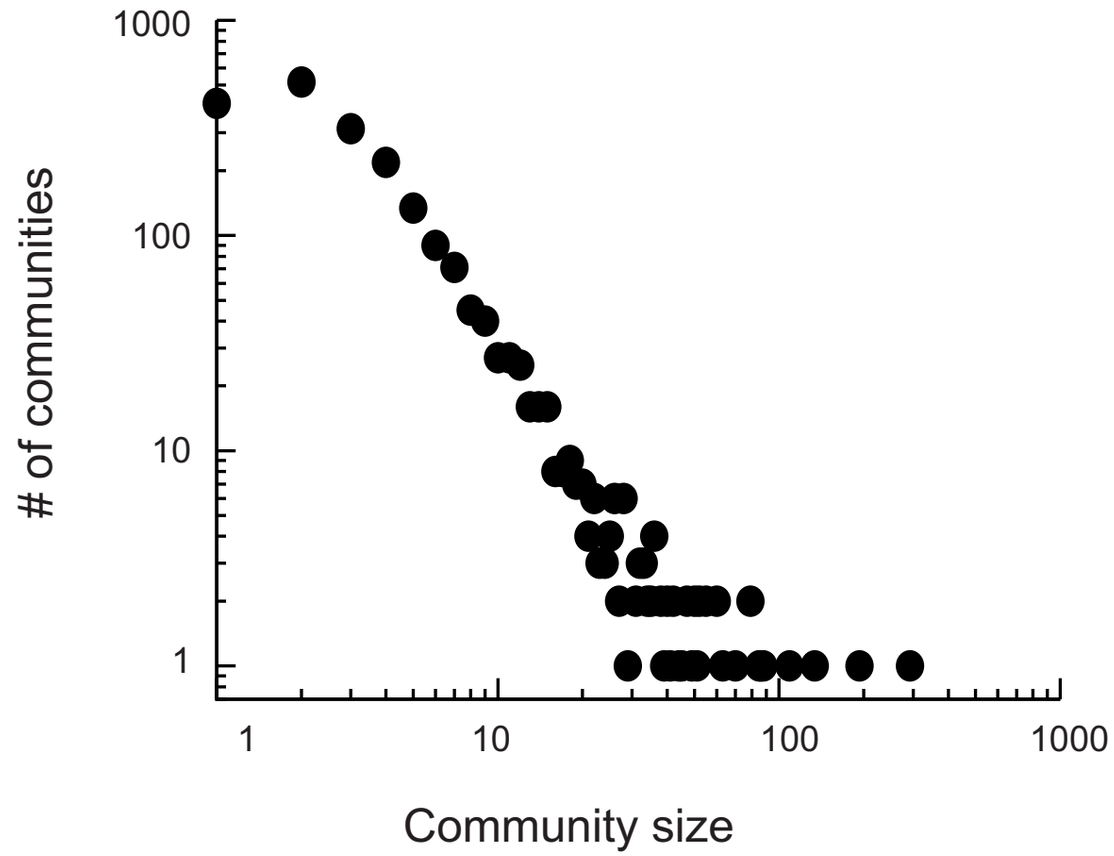


Finding Disease Modules

- Three **basic stages**:
  1. Construct a **PPI network**
  2. Apply a **community detection method**
  3. **Evaluate the quality** of detected communities
- **Questions:**
  - How to evaluate which detected communities are **"good"** disease modules?
  - How to **assign** a detected community to a disease?

Evaluating Detected Communities

- A typical method detects **many communities** in a PPI network
- Some detected communities might have a **biological meaning**, some might represent spurious effects
- **Task:** Evaluate the quality of each detected community



Is there a significant association between proteins in a detected community and a disease?

Evaluating Detected Communities

Is there a significant association between proteins in a detected community and a disease?

- This means:
  - "Are **unusually many** (or: unusually few) proteins in a community actually disease proteins?"
- More precise:
  - "If I picked n proteins at random (with n being the size of a community), **how probable** is it that among these proteins, there are **at least as many disease proteins** as there are in the community?"

Evaluating Detected Communities

- Let $C = \{g_1, g_2, \ldots, g_n\}$ be a **detected community**
- Let $D = \{d_1, d_2, \ldots, d_K\}$ be **disease proteins** involved in disease D
- Let $k = |C \cap D|$ be **the size of the overlap** between C and D

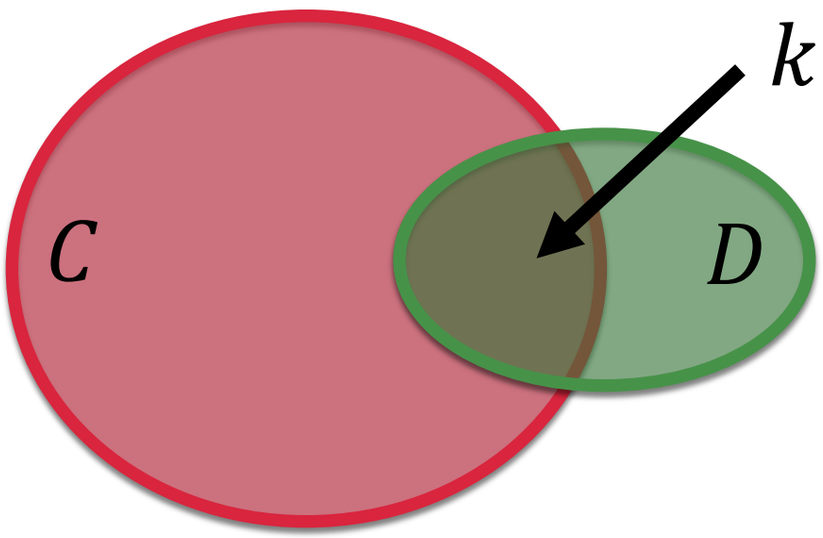
If I picked n proteins at random, how probable is it that among these proteins there are at least k disease proteins?

What is the probability of observing association at least this extreme due to chance?

Hypergeometric Distribution

- Construct a 2 x 2 **contingency table:**

	Associated with disease D	Not associated with disease D	Total
Within community C	k	$n-k$	n
Outside community C	$K-k$	$N-n-K+k$	$N-n$
Total	K	$N-K$	N



$$k = |C \cap D|$$

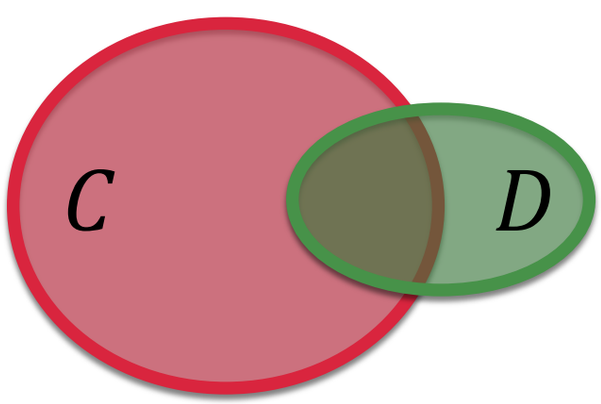
$$D = \{d_1, d_2, \ldots, d_K\}$$

$$C = \{g_1, g_2, \ldots, g_n\}$$

Hypergeometric Distribution

- Probability to get this **contingency table** if there is no association between C and D:

	Associated with disease D	Not associated with disease D	Total
Within community C	k	$n-k$	n
Outside community C	$K-k$	$N-n-K+k$	$N-n$
Total	K	$N-K$	N



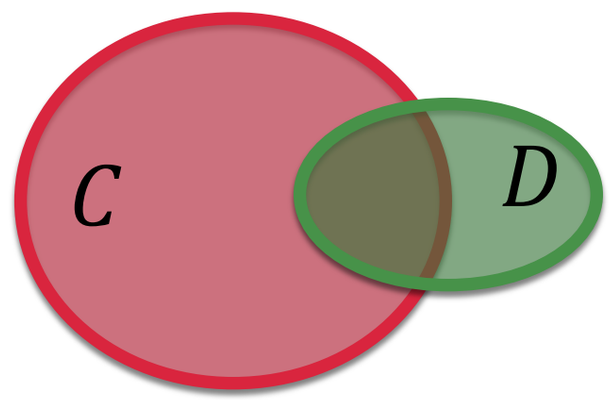
$$P(|C \cap D| = k) = \frac{\binom{K}{k}\binom{N-k}{n-k}}{\binom{N}{n}}$$

This is our null model!

Fisher's Exact Test

- **Exact hypergeometric probability** of observing this particular contingency table, assuming the given marginal totals:

$$P(|C \cap D| = k) = \frac{\binom{K}{k}\binom{N-k}{n-k}}{\binom{N}{n}}$$



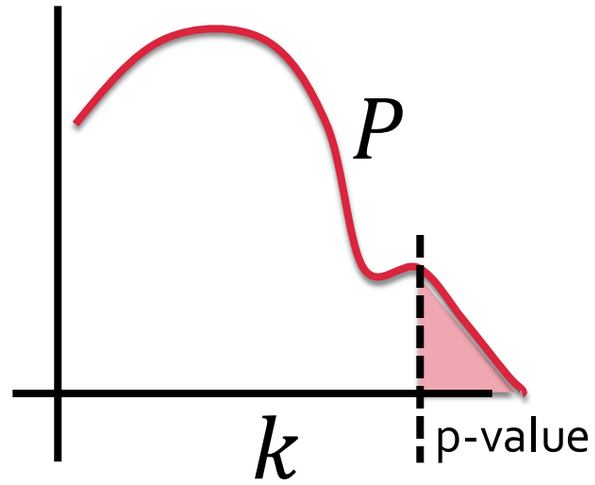
- **Goal:** Probability of observing association between C and D **at least this extreme due to chance**
- Consider **all possible overlaps** between C and D that are **equal or larger than k:**

$$P(|C \cap D| \geq k) = \sum_{r=k}^{\min(K,n)} P(|C \cap D| = r)$$

Fisher's Exact Test

- **One-tailed Fisher's exact test:** Probability of observing the **overlap** **as extreme or more extreme** under the null hypothesis of no association:

$$P(|C \cap D| \geq k) = \sum_{r=k}^{\min(K,n)} P(|C \cap D| = r)$$



Statistical enrichment of community C in disease D : $P(|C \cap D| \geq k)$

Experiment: Data

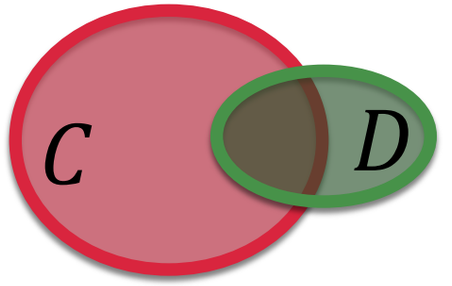
- Data:
  - Human **protein-protein interaction network**
    - 13,460 nodes, 150,000 edges
  - **Human diseases**
    - 70 diseases, each with at least 20 disease proteins
- Community detection methods:
  - Link clustering [Ahn et al., Nature 2010]
  - Louvain method [Blondel et al., TE 2008]
  - Markov clustering method (MCL) [Van Dongen, SIAM 2008]

Disease
adrenal gland diseases
alzheimer disease
Amino acid metabolism inborn errors
amyotrophic lateral sclerosis
anemia aplastic
anemia hemolytic
aneurysm
arrhythmias cardiac
arthritis rheumatoid
asthma
arterial occlusive diseases
arteriosclerosis
basal ganglia diseases
behcet syndrome
bile duct diseases
blood coagulation disorders
blood platelet disorders

Experiment: Setup

- Setup:

1. Use **community detection method** to find communities in the PPI network
2. Use **Fisher's exact test** to determine, for each community-disease pair, if community is **significantly enriched** with disease proteins

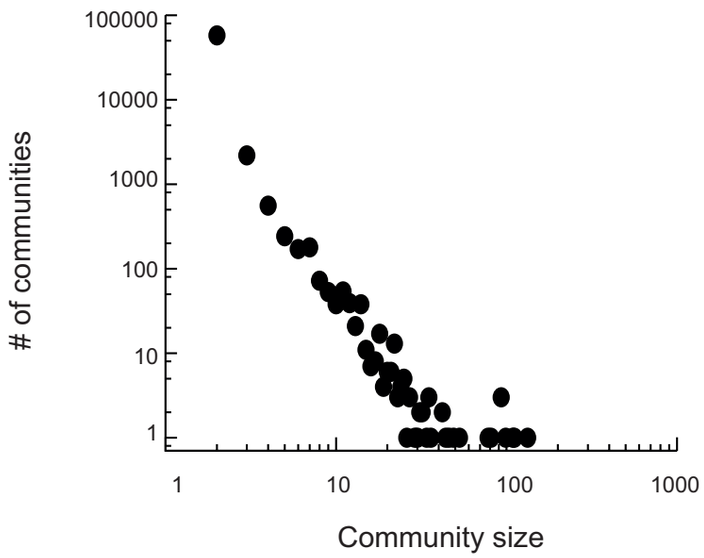


3. Use **Bonferroni correction** to counteract the problem of **multiple statistical comparisons**
   - If testing m hypotheses at a desired significance level $\alpha = 0.05$, then the Bonferroni correction would test each individual hypothesis at $\alpha = 0.05/m$

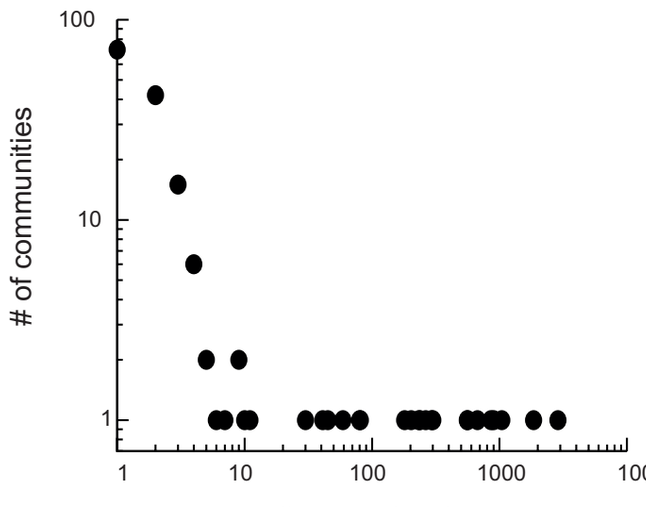
Ghiassian et al., PLoS Comp Bio 2015

Protein Communities

Link clustering



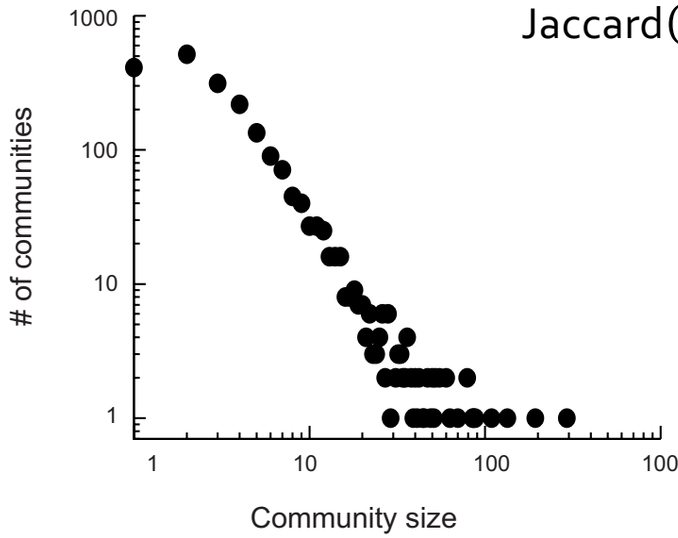
Louvain method



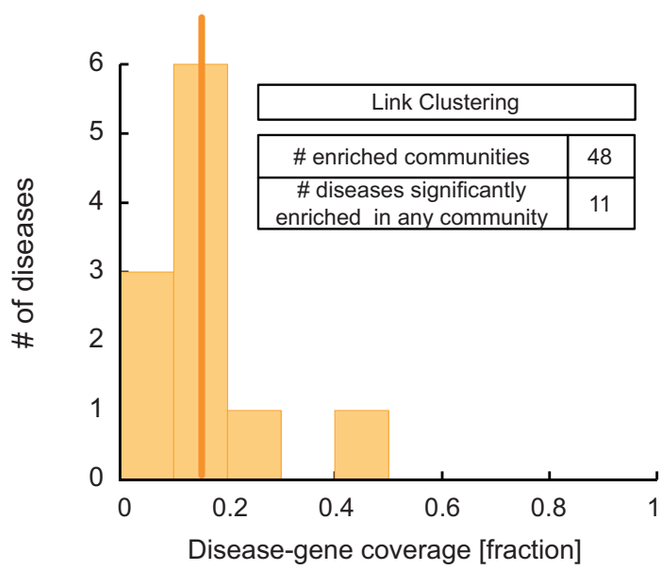
MCL

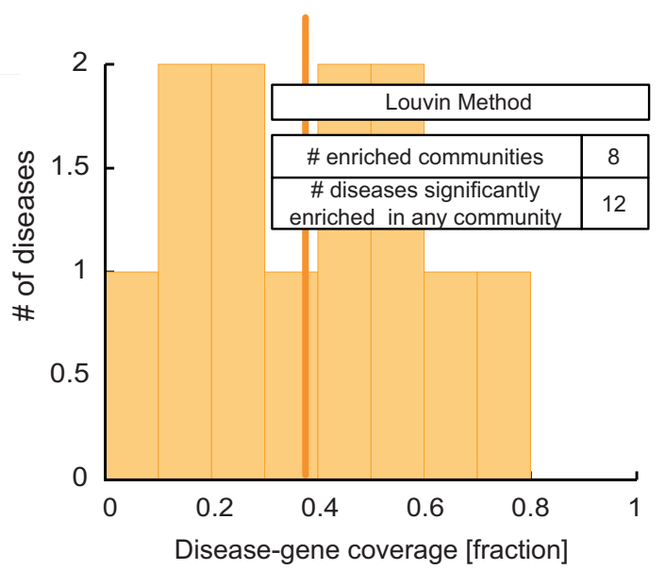
Note:

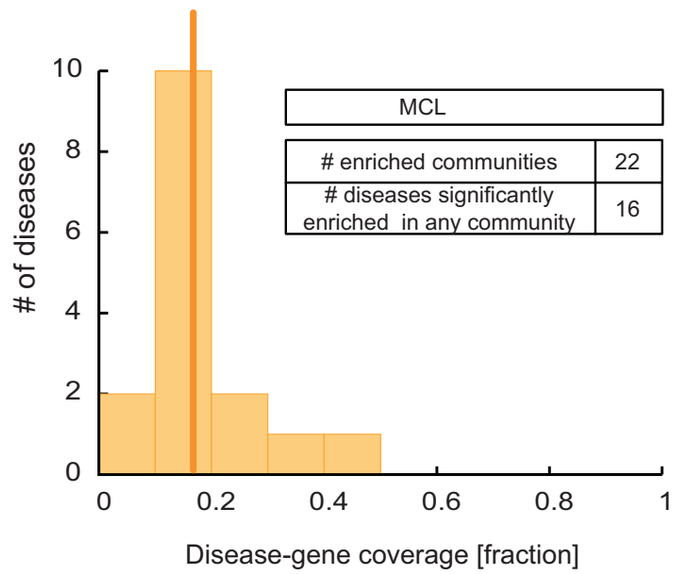
$$\text{Jaccard}(C, D) = \frac{|C \cap D|}{|C \cup D|}$$



- Community-disease pairs with significant overlap versus their Jaccard similarity



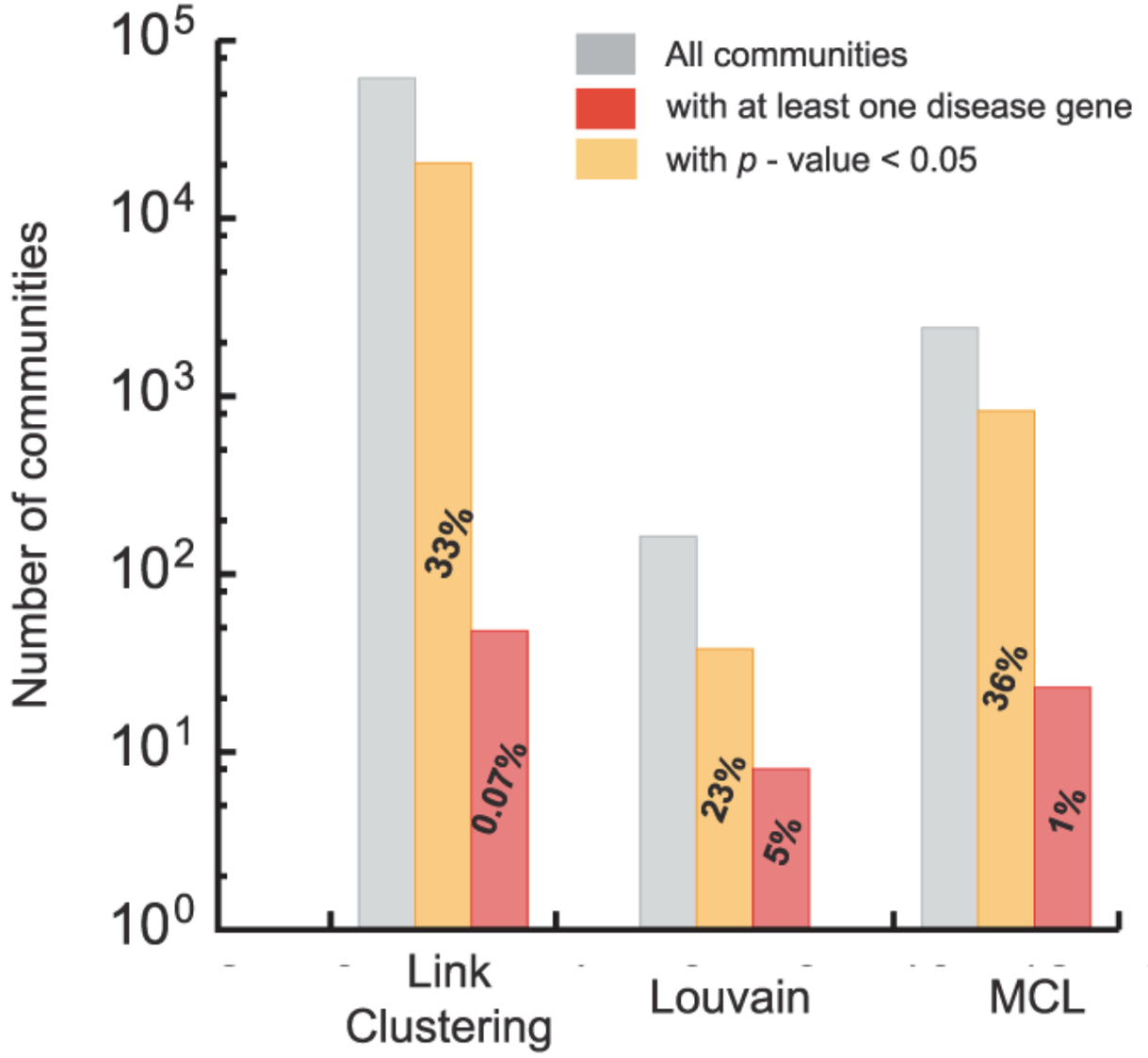




Protein Communities

Ghiassian et al., PLoS Comp Bio 2015

- No detected community coincides with a **full set** of disease proteins
- 36% of MCL communities are **significantly enriched** in at least one disease
- **Proteins in an enriched community** that are not yet associated with a disease are **disease protein candidates**

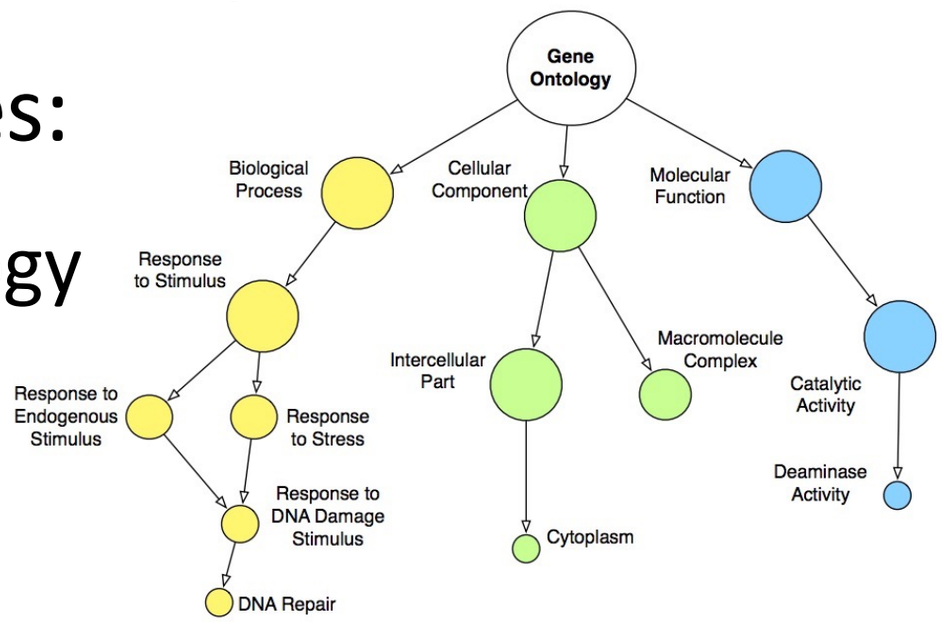


Other Statistical Issues

- Other **tests for enrichment:**
  - Binomial, Chi-squared, Z-test, Kolmogorov-Smirnov, permutation
  - **Gene Set Enrichment Analysis (GSEA)** uses a variation of Kolmogorov-Smirnov statistic to get p-values [http://software.broadinstitute.org/gsea]
- All tests look for **over-enrichment**; some look for **under-enrichment**
- Correction for **multiple hypothesis testing**
- Some diseases may be **subsets** of other diseases

Beyond Disease Associations

- Proteins in detected communities **should have something in common**, e.g., they are:
  - part of the **same biological pathway/cellular component**
  - **co-expressed** under certain conditions
  - putative targets of **the same regulatory factor**
- Use enrichment tests to check whether communities are enriched in biological pathways, components, etc.
- Get data from biomedical databases:
  - **Processes, components:** Gene Ontology
  - **Pathways:** KEGG, Reactome, MSigDB

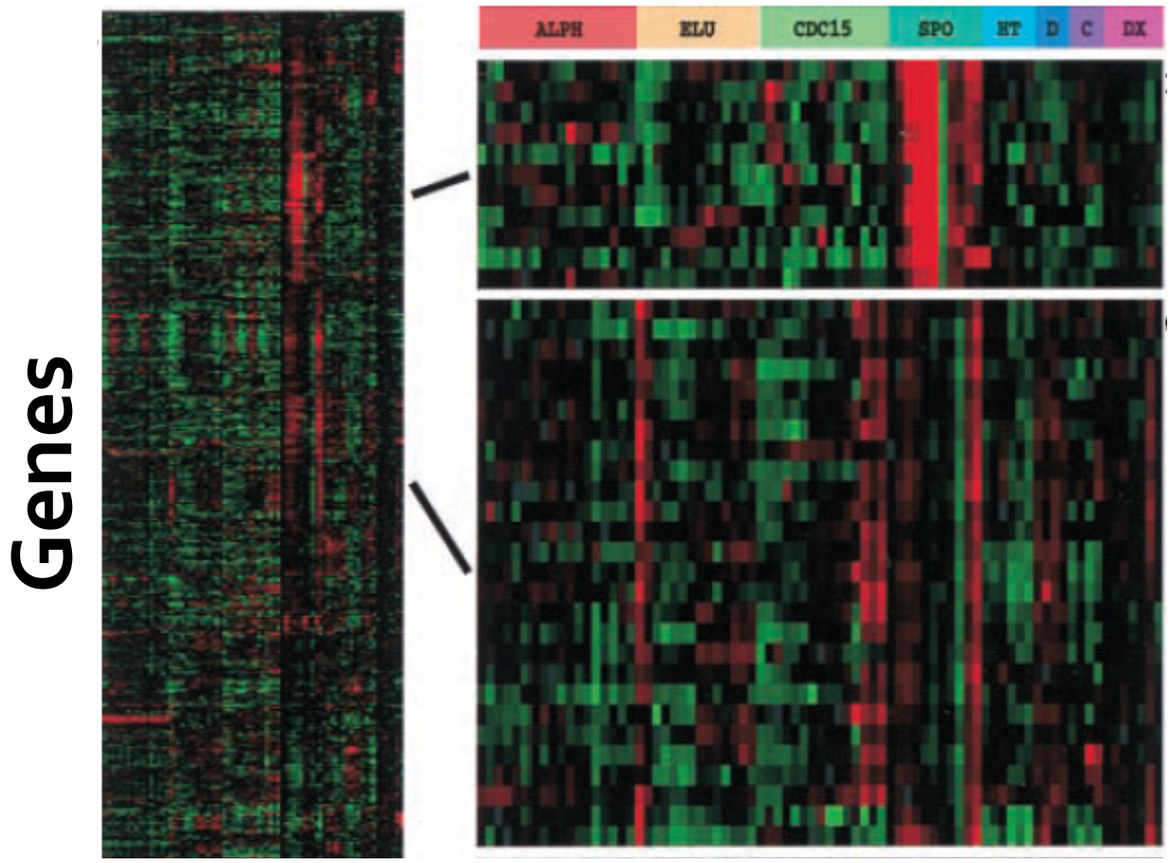


Plan For Today

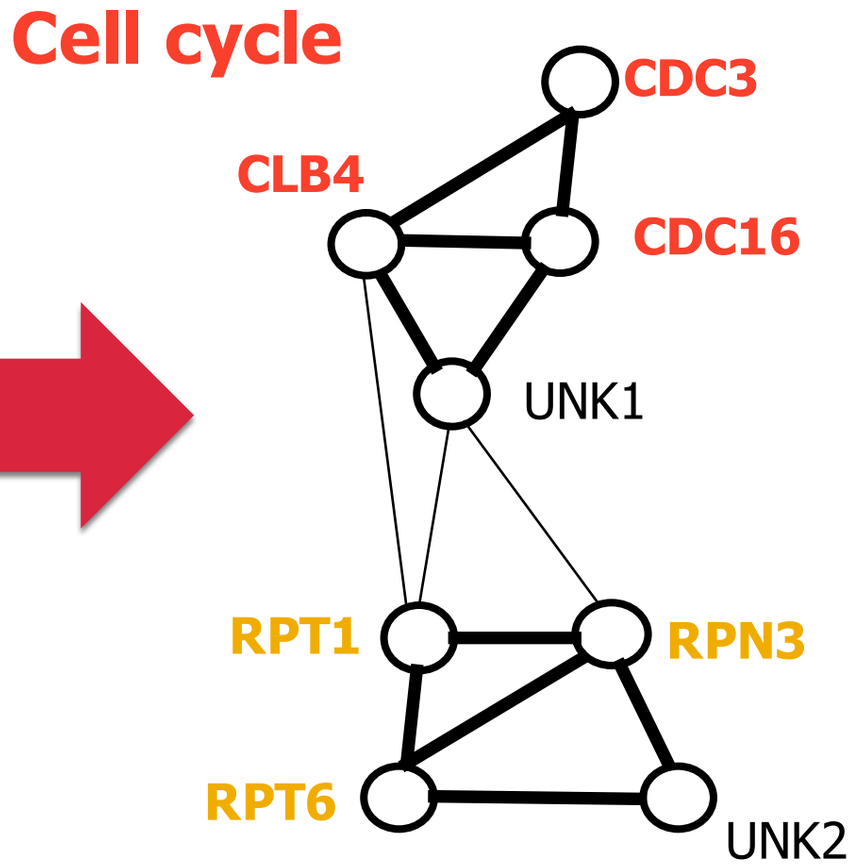
1) **Very basic biology**
2) **Protein-protein interaction networks**
3) **Finding disease modules in networks**
   - It is a community detection task!
4) **Predicting biological attributes, such as protein functions**
   - Guilt-by-association principle
   - Gene recommender systems

Functional Interaction Networks

Conditions



Gene co-expression network



Protein degradation

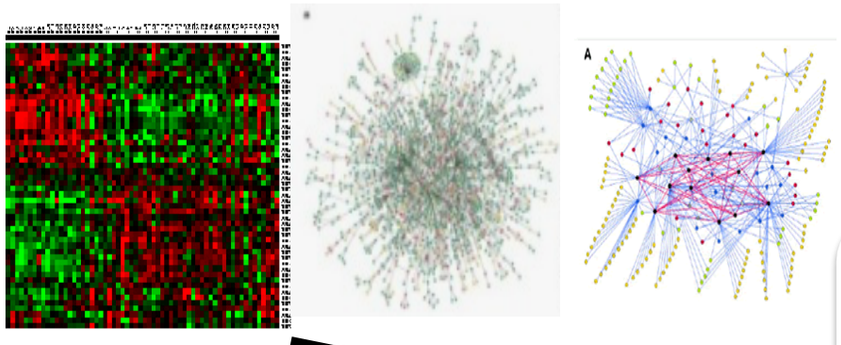
Types of Gene Function Prediction

- **"What does my gene do?"**
  - **Goal:** Determine a gene's function based on who it interacts with – "**guilt-by-association**" **principle**
- **"Give me more genes like these"**
  - E.g., Find more multiple sclerosis genes, find new ciliary genes, find more members of a protein complex

"What Does My Gene Do?"

Input

Network data

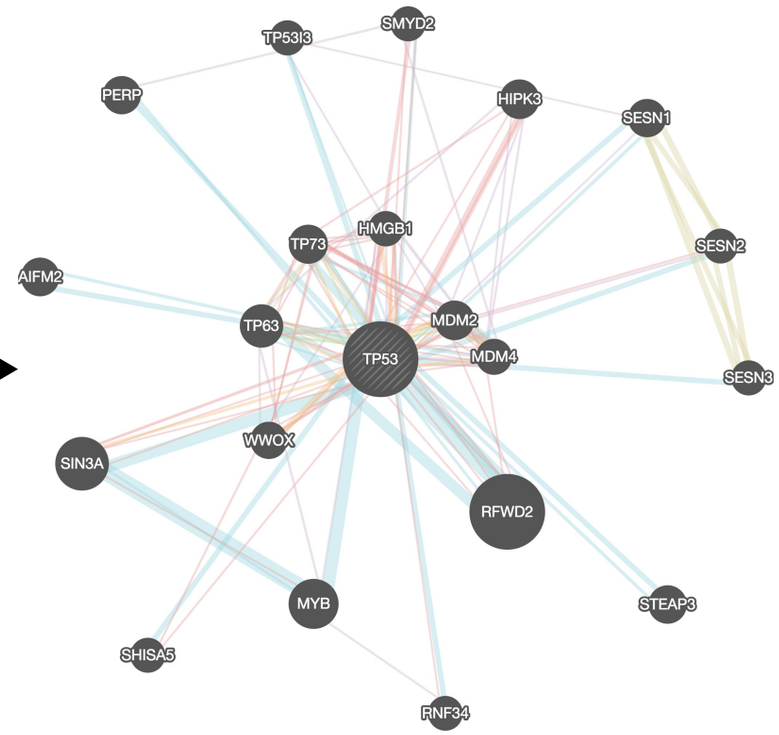


Query gene

TP53

Community detection, then enrichment analysis

Output

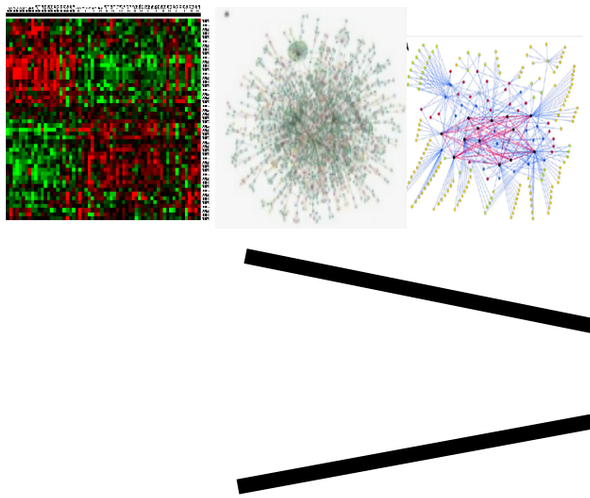


"Guilt-by-association" principles

"Give Me More Genes Like These"

Input

Network data

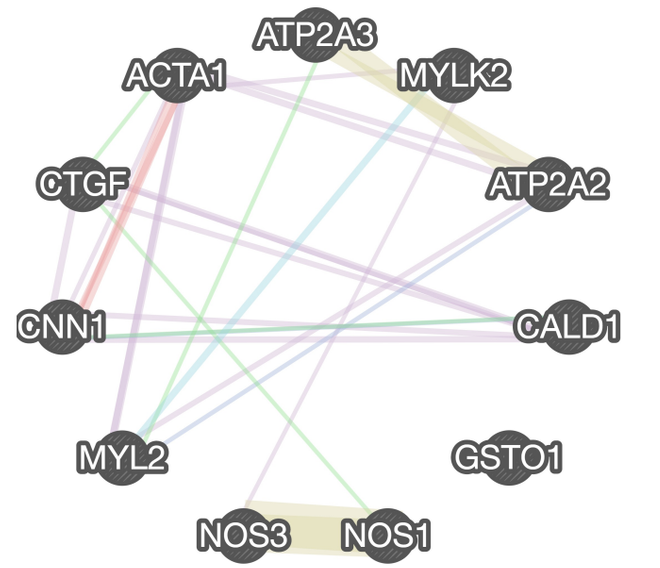


Query list

ATP2A2
ATP2A3
NOS1
CNN1
GSTO1
NOS3
CNN2
MYLK2
CALD1
ACTA1
MYL2

Gene recommender system

Output



Networks

- Co-expression
- Shared protein domains
- Physical interactions
- Pathway
- Co-localization
- Genetic interactions

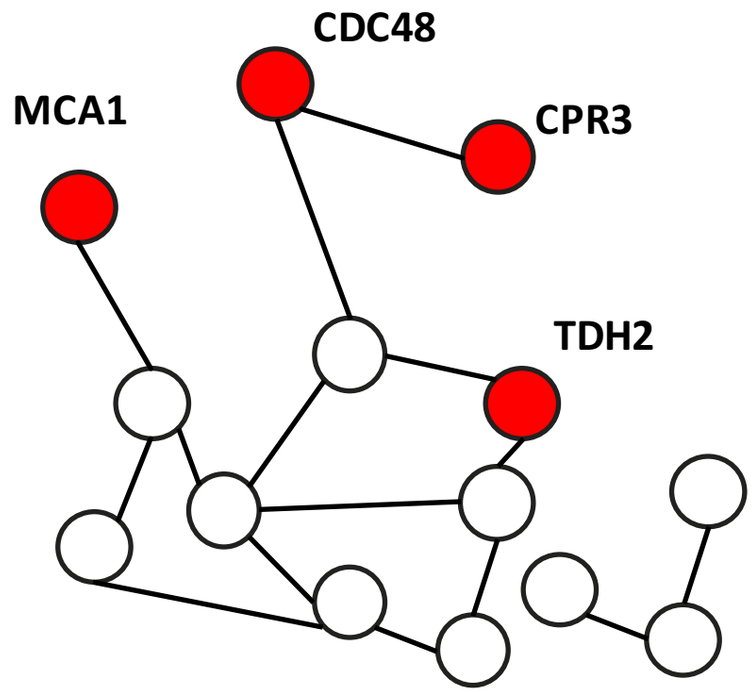
Functions

- muscle system process
- muscle contraction
- regulation of system process
- regulation of muscle system process
- heart contraction

Finding "Guilty Associates"

- Predict gene functions by **guilt-by-association:**

Query list: "positive examples"



Red: Genes involved in a gene function/biological process
White: Unlabeled genes

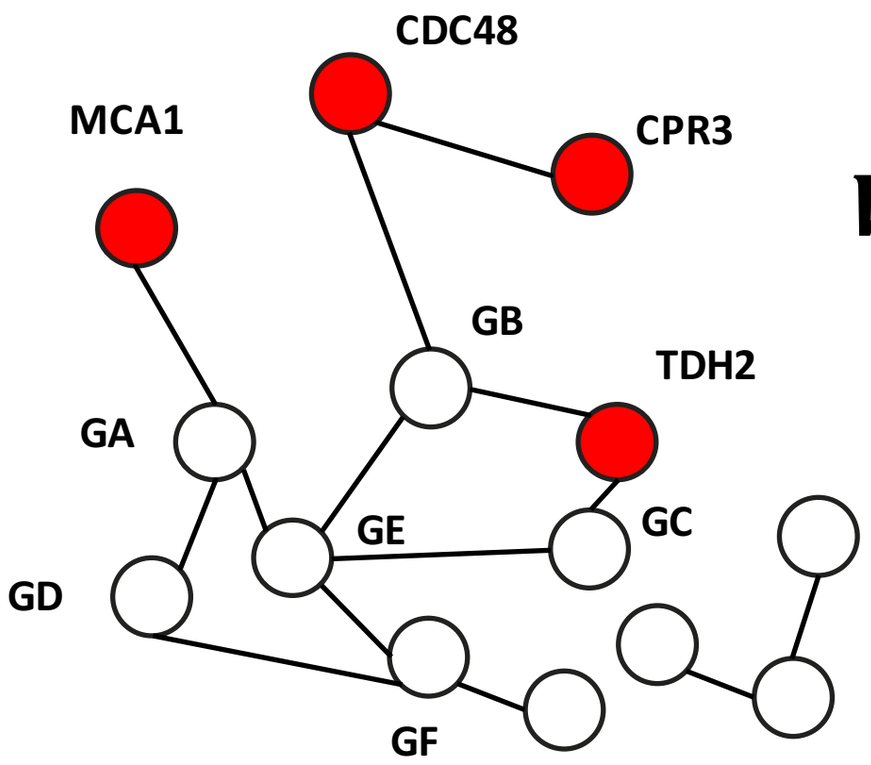
- **Question:** Which of the **unlabeled nodes** are likely involved in this gene function/biological process?
- **Two main approaches:**
  - Direct/Indirect **neighbor** scoring
  - **Label propagation**

"Guilty Associates" Problem

- Let $\boldsymbol{W}$ be a $n \times n$ (weighted) adjacency matrix over n genes in a genome
- Let $\boldsymbol{y} = \{-1, 0, 1\}^n$ be a vector of **labels**:
  - 1: **positive** gene, known to be involved in a gene function/biological process
  - -1: **negative** gene
  - 0: **unlabeled** gene
- **Goal:** Predict which of the unlabeled genes are likely positive

"Guilty Associates" Problem

- **Goal: Predict** which of the unlabeled genes are **likely positive**
- Learn a vector of discriminant scores $\boldsymbol{f}$, where $\boldsymbol{f}_i$ is **the likelihood** that node i is positive
- **Example:**



$$\boldsymbol{y} = [1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$\boldsymbol{W} = \text{(weighted) adjacency matrix}$$

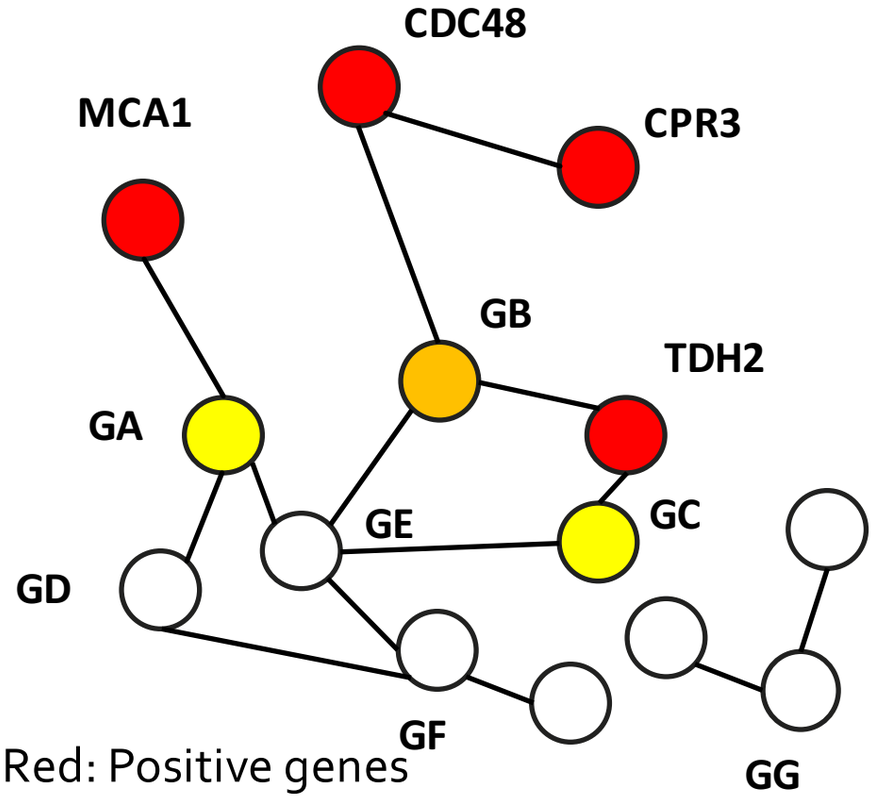
$$\boldsymbol{f} = ?$$

Direct Neighbor Scoring

- **Approach #1:** Node score $\boldsymbol{f}_i$ is the weighted sum of the labels of i's direct neighbors:

$$\boldsymbol{f}_i = \sum_{j=1}^{n} \boldsymbol{W}_{ij} \boldsymbol{y}_j$$

- **Example:**



Red: Positive genes
White: $\boldsymbol{f}_i = 0$

$$\boldsymbol{f}_{\mathrm{GA}} = \boldsymbol{W}_{\mathrm{GA,MCA1}} \cdot \boldsymbol{y}_{\mathrm{MCA1}}$$

$$\boldsymbol{f}_{\mathrm{GB}} = \boldsymbol{W}_{\mathrm{GB,CDC48}} \cdot \boldsymbol{y}_{\mathrm{CDC48}} + \boldsymbol{W}_{\mathrm{GB,TDH2}} \cdot \boldsymbol{y}_{\mathrm{CDC48}}$$

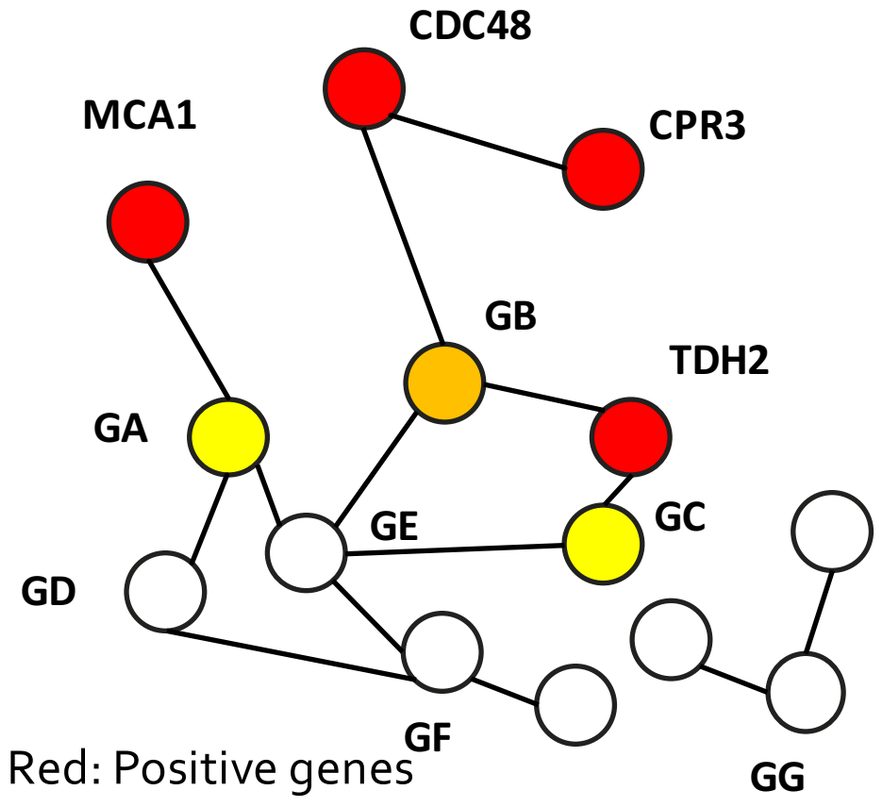
$$\boldsymbol{f}_{\mathrm{GC}} = \boldsymbol{W}_{\mathrm{GC,TDH2}} \cdot \boldsymbol{y}_{\mathrm{TDH2}}$$

Direct Neighbor Scoring

- **Approach #1:** Node score $\boldsymbol{f}_i$ is a weighted sum of the labels of i's direct neighbors:

$$\boldsymbol{f}_i = \sum_{j=1}^{n} \boldsymbol{W}_{ij} \boldsymbol{y}_j$$

- **Example:**



Red: Positive genes
White: $\boldsymbol{f}_i = 0$

$$\boldsymbol{f}_{\mathrm{GA}} = \boldsymbol{W}_{\mathrm{GA,MCA1}} \cdot \boldsymbol{y}_{\mathrm{MCA1}}$$

$$\boldsymbol{f}_{\mathrm{GB}} = \boldsymbol{W}_{\mathrm{GB,CDC48}} \cdot \boldsymbol{y}_{\mathrm{CDC48}} + \boldsymbol{W}_{\mathrm{GB,TDH2}} \cdot \boldsymbol{y}_{\mathrm{CDC48}}$$

$$\boldsymbol{f}_{\mathrm{GC}} = \boldsymbol{W}_{\mathrm{GC,TDH2}} \cdot \boldsymbol{y}_{\mathrm{TDH2}}$$

- **One half** of GC's neighbors are positives
- **One third** of GA's neighbors are positives
- **But:** $\boldsymbol{f}_{\mathrm{GC}} = \boldsymbol{f}_{\mathrm{GA}}$ (if $\boldsymbol{W}$ is binary)

Direct Neighbor Scoring

- **Approach #2:** Normalize matrix $\boldsymbol{W}$ using the weighted node degrees:

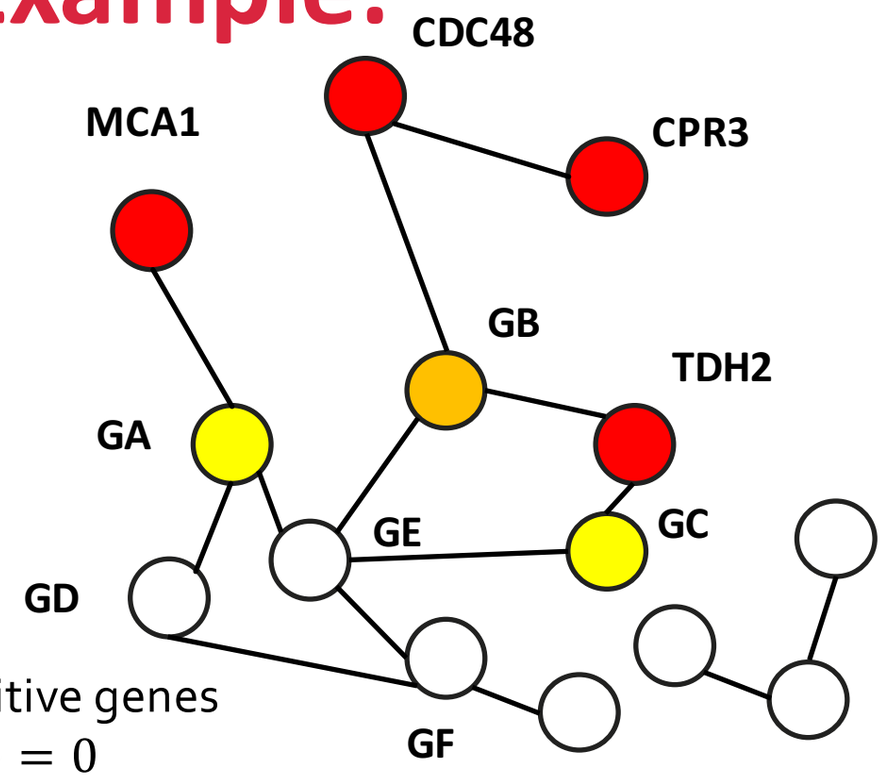
$$\boldsymbol{f}_i = \frac{1}{\boldsymbol{d}_i}\sum_{j=1}^{n} \boldsymbol{W}_{ij}\boldsymbol{y}_j, \qquad \boldsymbol{d}_i = \sum_j \boldsymbol{W}_{ij}$$

Matrix notation:

$$\boldsymbol{f}_i = \boldsymbol{D}^{-1}\boldsymbol{W}\boldsymbol{y}$$

$$\boldsymbol{D} = diag(\boldsymbol{d})$$

- **Example:**



Red: Positive genes
White: $\boldsymbol{f}_i = 0$

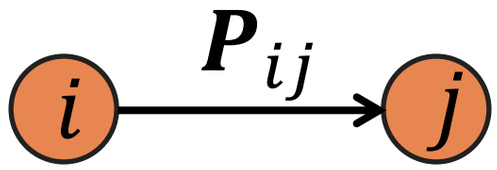
$$\boldsymbol{f}_{\mathrm{GA}} = \frac{1}{3}\boldsymbol{W}_{\mathrm{GA,MCA1}} \cdot \boldsymbol{y}_{\mathrm{MCA1}}$$

$$\boldsymbol{f}_{\mathrm{GB}} = \frac{1}{3}(\boldsymbol{W}_{\mathrm{GB,CDC48}} \cdot \boldsymbol{y}_{\mathrm{CDC48}} + \boldsymbol{W}_{\mathrm{GB,TDH2}} \cdot \boldsymbol{y}_{\mathrm{TDH2}})$$

$$\boldsymbol{f}_{\mathrm{GC}} = \frac{1}{2}\boldsymbol{W}_{\mathrm{GC,TDH2}} \cdot \boldsymbol{y}_{\mathrm{TDH2}}$$

Towards Indirect Neighbor Scoring

- Matrix $\boldsymbol{P} = \boldsymbol{D}^{-1}\boldsymbol{W}$ is known as **Markov transition matrix**
  - $\boldsymbol{D}$ is a diagonal matrix with diagonal elements $\boldsymbol{d}_i$
  - $\boldsymbol{P}$ is **a row stochastic matrix**, $\sum_j \boldsymbol{P}_{ij} = 1$
- Row i is a probability distribution over **random walks** starting at node i

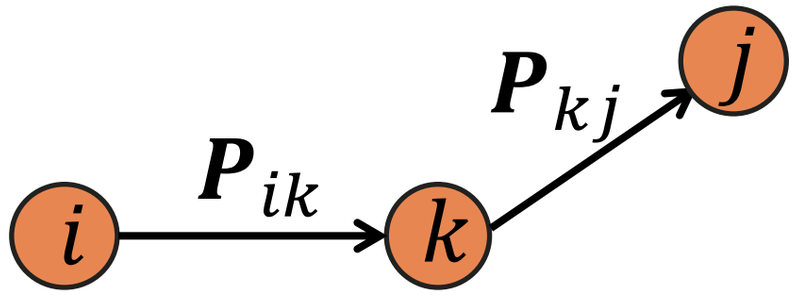


- $\boldsymbol{P}_{ij}$ is probability of a **random walker following a link from node i to node j**

Random Walk Interpretation

- **Random walk** interpretation extends a direct neighbor approach to include **indirect neighbors**
- **Idea:** Extend the formula $\boldsymbol{f} = \boldsymbol{D}^{-1}\boldsymbol{W}\boldsymbol{y} = \boldsymbol{P}\boldsymbol{y}$ to include **second-degree neighbors**
- Probability of a random walk of length **two** between node i and node j is:

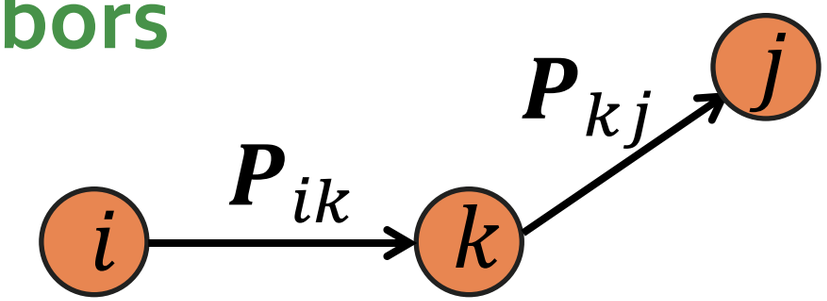
$$[\boldsymbol{P}^2]_{ij} = \sum_{k=1}^{n} \boldsymbol{P}_{ik}\boldsymbol{P}_{kj}$$



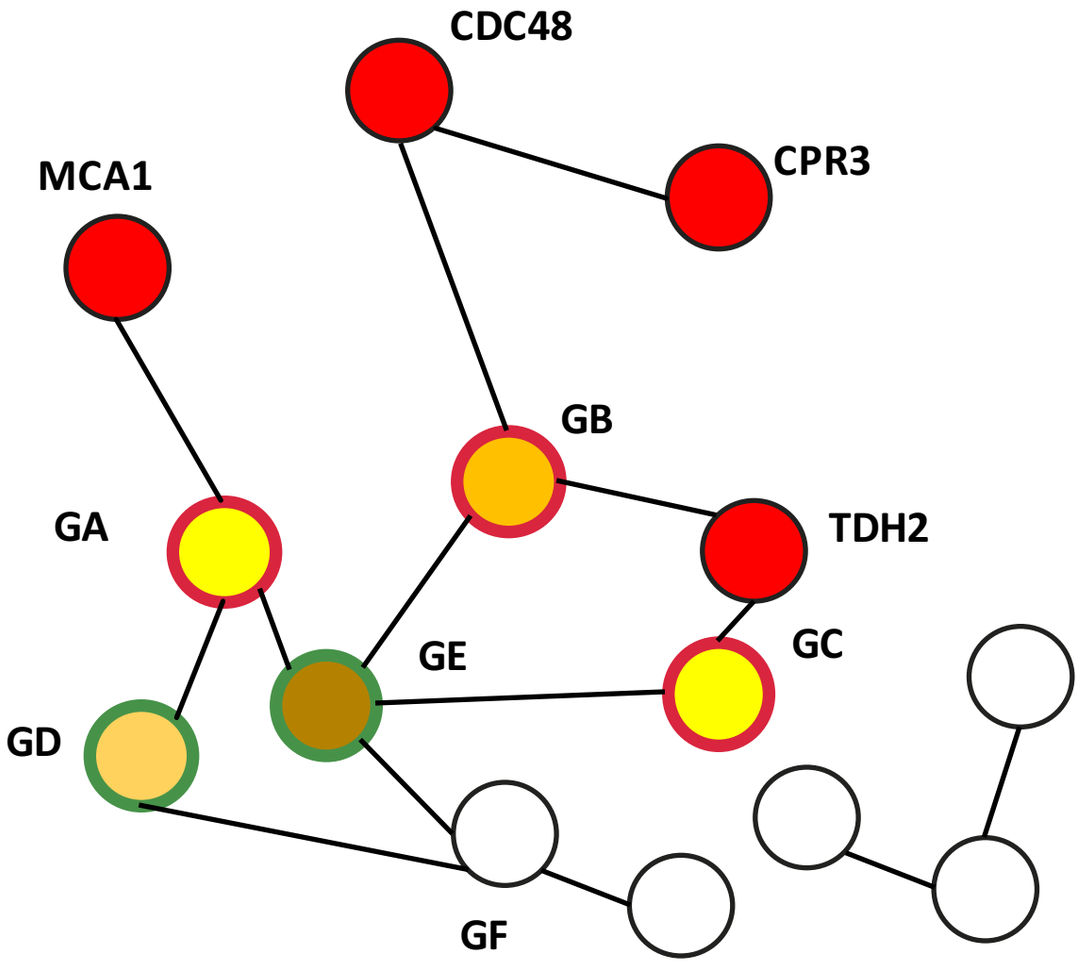
Indirect Neighbor Scoring

- **Approach #3:** Consider **second-degree neighbors** when calculating node score $\boldsymbol{f}_i$ as:

$$\boldsymbol{f}_i = \underbrace{\sum_{j=1}^{n} \boldsymbol{P}_{ij}\boldsymbol{y}_j}_{\text{Direct neighbors}} + \underbrace{\sum_{j=1}^{n} [\boldsymbol{P}^2]_{ij}\,\boldsymbol{y}_j}_{\text{Second-degree neighbors}}$$



Example of Indirect Neighbor Scoring



$$\boldsymbol{P} = \boldsymbol{D}^{-1}\boldsymbol{W}$$

$$\boldsymbol{f}_i = \underbrace{\sum_{j=1}^{n} \boldsymbol{P}_{ij}\boldsymbol{y}_j}_{\text{Direct neighbors}} + \underbrace{\sum_{j=1}^{n} [\boldsymbol{P}^2]_{ij}\,\boldsymbol{y}_j}_{\text{Second-degree neighbors}}$$

$$\boldsymbol{f}_{\mathrm{GA}} = \boldsymbol{P}_{\mathrm{GA,MCA1}} \cdot \boldsymbol{y}_{\mathrm{MCA1}}$$

$$\boldsymbol{f}_{\mathrm{GE}} = \boldsymbol{P}^2_{\mathrm{GE,MCA1}} \cdot \boldsymbol{y}_{\mathrm{MCA1}} + \boldsymbol{P}^2_{\mathrm{GE,TDH2}} \cdot \boldsymbol{y}_{\mathrm{TDH2}} + \boldsymbol{P}^2_{\mathrm{GE,CDC48}} \cdot \boldsymbol{y}_{\mathrm{CDC48}}$$

- Direct neighbor of a positive gene
- Second-order neighbor of a positive gene

Red: Positive genes

White: $\boldsymbol{f}_i = 0$

$[\boldsymbol{P}^2]_{ij} > 0$ if there is a walk of length 2 between i and j

Random Walk Interpretation

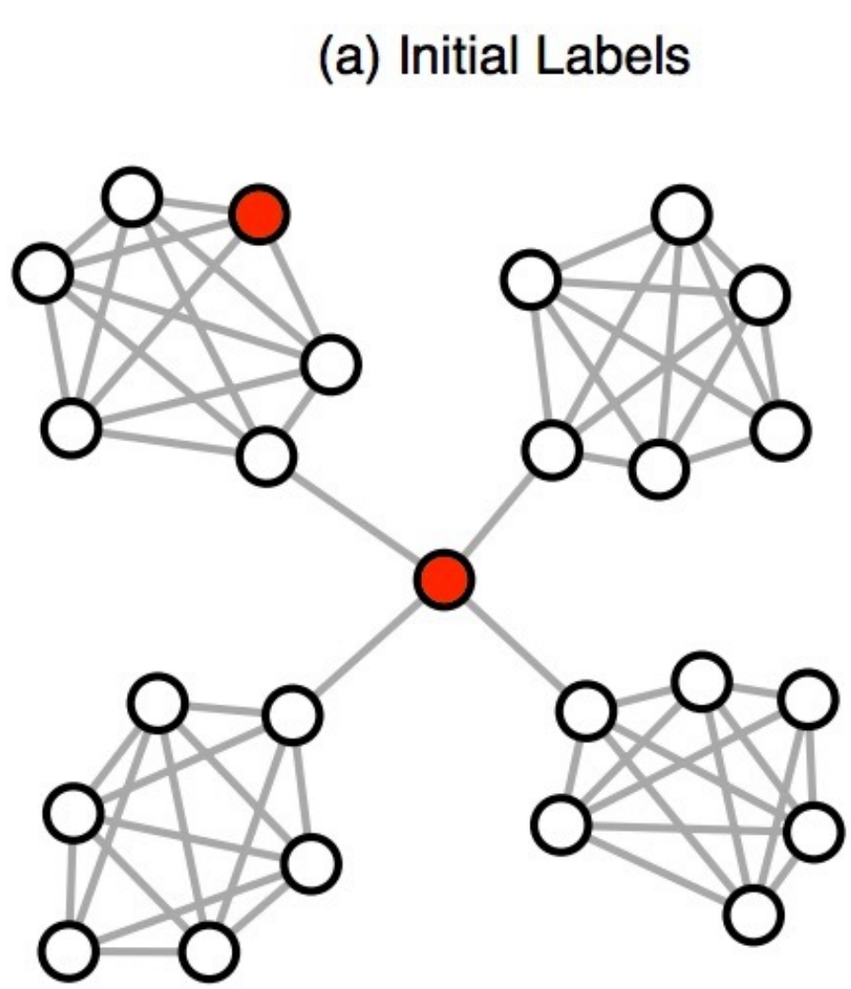
- **Approach #3** can be **extended** to include other nodes at **a distance of length r** (usually $r < 4$)
- Increasing r beyond 2 often results in degradation of prediction performance [Chua et al., Bioinformatics 2006, Myers et al., Genome Biology 2005]
- **Note:** Probability of a random walk from i to j in r **steps** is given by $[\boldsymbol{P}^r]_{ij}$
- **Next:** Use **random walks** to derive **label propagation**

Label Propagation Approach

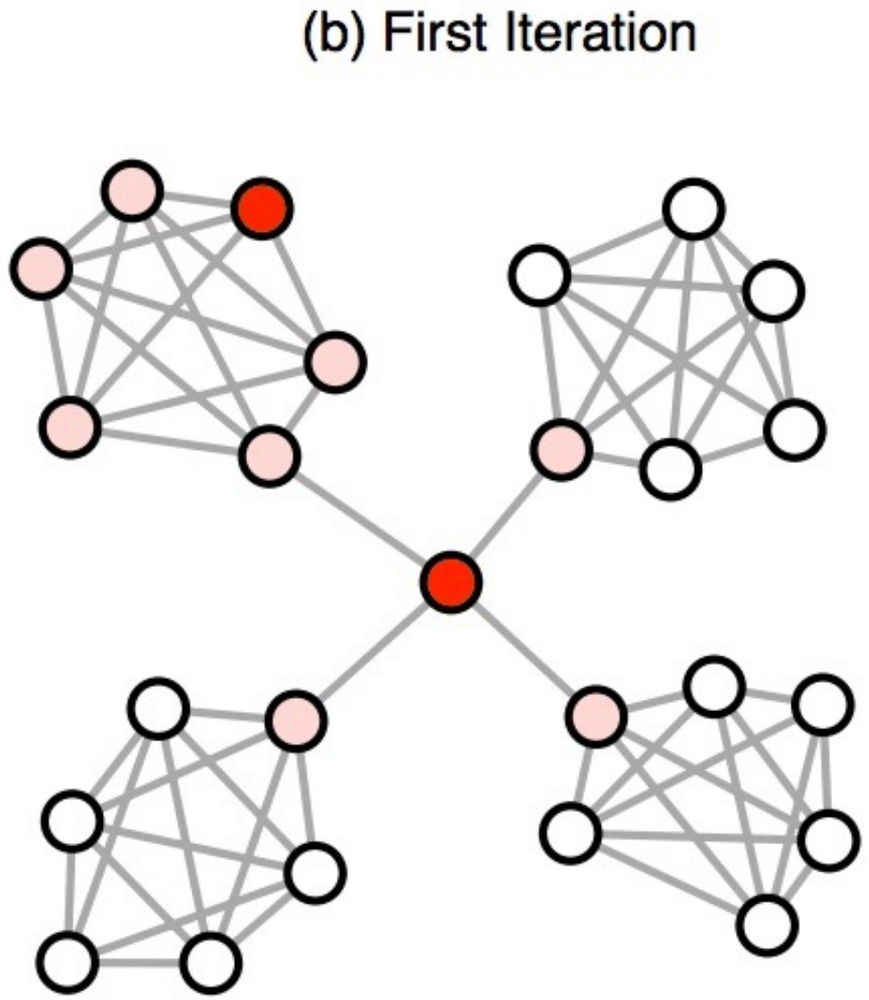
- Label propagation **generalizes** local neighborhood-based approaches by **considering random walks of all lengths** between nodes
- The algorithm can be derived as:
  1. Iterative diffusion process [Zhou et al., NIPS 2004]
  2. Solution to a specific convex optimization task [Zhou et al., NIPS 2004, Zhu et al., ICML 2003]
  3. Maximum a posteriori (MAP) estimation in Gaussian Markov Random Fields [Rue and Held, Chapman & Hall, 2005]
- **Next:** Derivation using an **iterative formulation**

Label Propagation Approach

Intuition: Diffuse labels through edges of the network



Red: positive nodes
White: unlabeled nodes



Red: positive nodes
Pink: $f_i > 0$
White: $f_i = 0$

Diffusion Process: Idea

- The **diffusion process** is defined as an **iterative process** [Zhou et al., NIPS 2004]
- **Diffusion of labels through edges:**
  - Start with initial label information, $\boldsymbol{f}_i^{(0)} = \boldsymbol{y}_i$
  - In each iteration, each node receives **label information from its neighbors**, and also **retains its initial label**
  - λ specifies **relative amount** of label information from its neighbors and its initial label
  - Finally, the label of each unlabeled node is set to be the label of which it has **received most information**

Diffusion Process: Formally

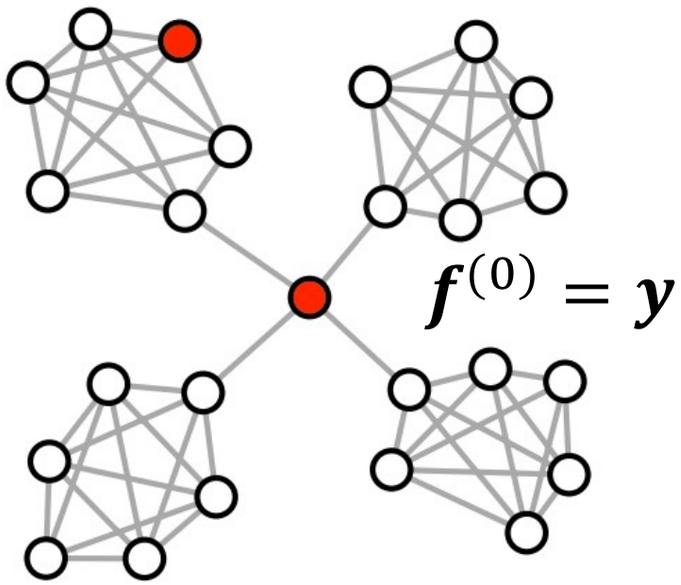
- The **diffusion process** is defined as the following **iteration** [Zhou et al., NIPS 2004]
- At iteration $r = 0$, define $\boldsymbol{f}_i^{(0)} \leftarrow \boldsymbol{y_i}$
- At iteration $r + 1$, the score of node i is the **weighted average** of the scores of i's neighbors in iteration r, and i's initial label:

$$\boldsymbol{f}_i^{(r+1)} \leftarrow (1-\lambda)\boldsymbol{y}_i + \lambda \sum_{j=1}^{n} \boldsymbol{W}_{ij}\boldsymbol{f}_j^{(r)}$$

$0 < \lambda < 1$ is a model parameter

Example of Label Propagation

(a) Initial Labels



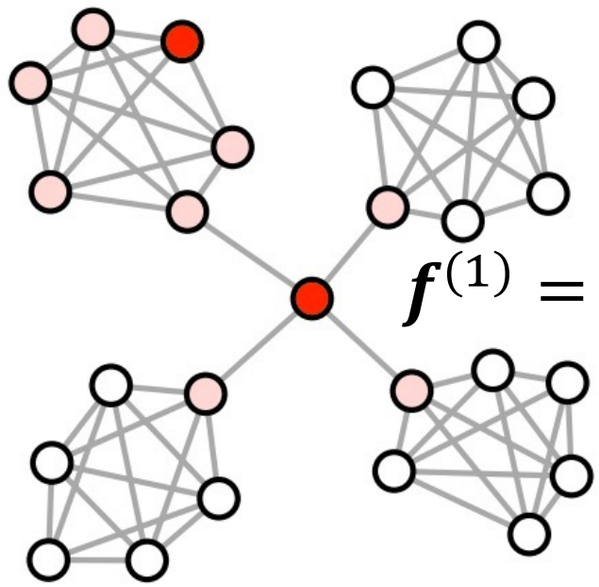
$$\boldsymbol{f}^{(0)} = \boldsymbol{y}$$

Score

high

low

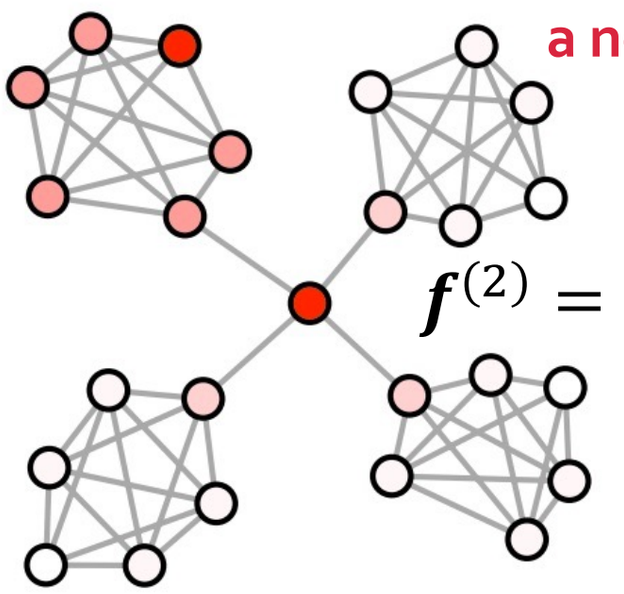
(b) First Iteration



$$\boldsymbol{f}^{(1)} = \lambda \boldsymbol{W}\boldsymbol{y} + (1-\lambda)\boldsymbol{y}$$

All nodes reachable with a walk of length 2 are assigned a non-zero value

(c) Second Iteration



$$\boldsymbol{f}^{(2)} = \lambda \boldsymbol{W}\boldsymbol{f}^{(1)} + (1-\lambda)\boldsymbol{y}$$

Convergence Condition

- If all **eigenvalues** of $\boldsymbol{W}$ are in range $[-1, 1]$, then the sequence $\boldsymbol{f}^{(r)}$ converges to:

$$\boldsymbol{f} = (1-\lambda)\sum_{r=0}^{\infty}(\lambda \boldsymbol{W})^r \boldsymbol{y}$$

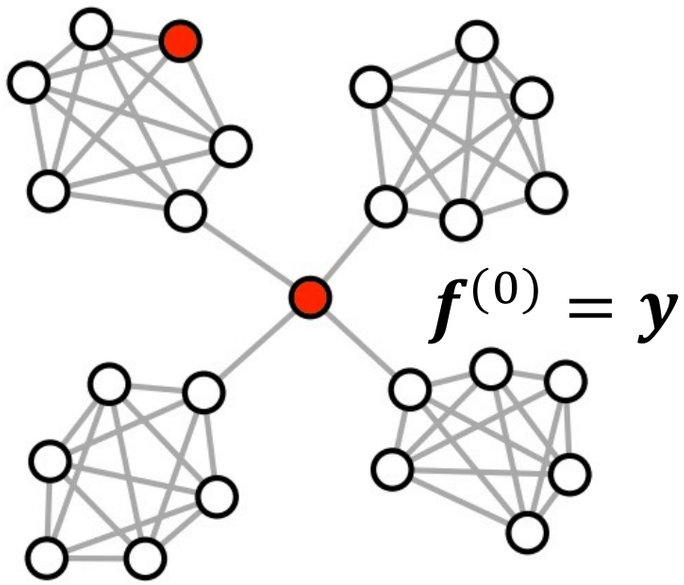
  - $[\boldsymbol{W}^r]_{ij} > 0$ if a walk of length r between i and j
  - Weight λ^r decreases with increasing distance

- $\Rightarrow$ Discriminant scores $\boldsymbol{f}$ are **weighted sum of walks of all lengths between the nodes**

- $\Rightarrow$ **High score $\boldsymbol{f}_i$** is assigned to i if i is connected to positively labeled nodes with **many short walks**

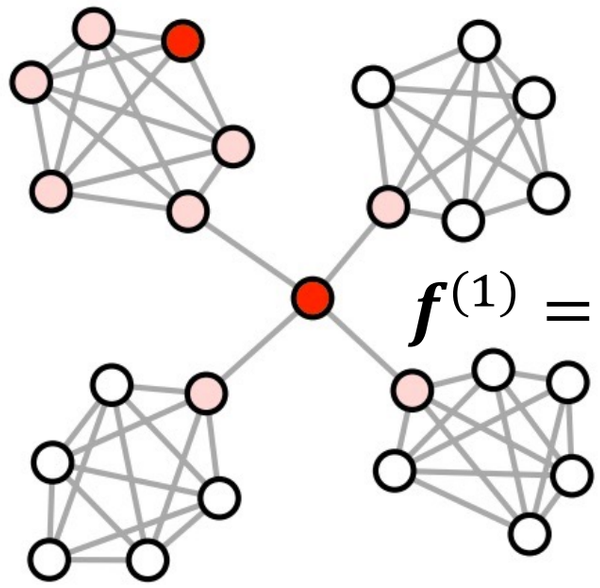
Example of Label Propagation

(a) Initial Labels



$$\boldsymbol{f}^{(0)} = \boldsymbol{y}$$

(b) First Iteration



Score

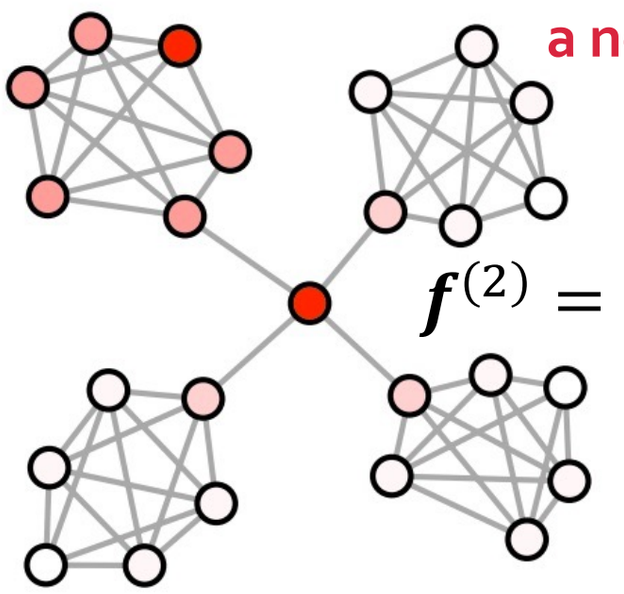
high

low

$$\boldsymbol{f}^{(1)} = \lambda \boldsymbol{W}\boldsymbol{y} + (1-\lambda)\boldsymbol{y}$$

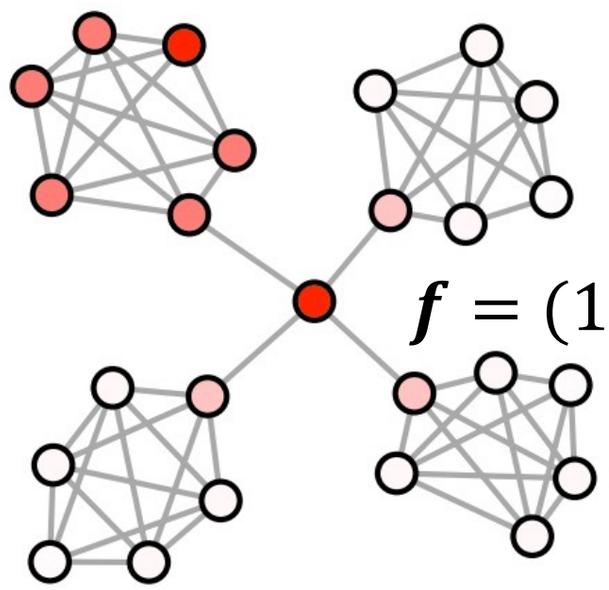
All nodes reachable with a walk of length 2 are assigned a non-zero value

(c) Second Iteration



$$\boldsymbol{f}^{(2)} = \lambda \boldsymbol{W}\boldsymbol{f}^{(1)} + (1-\lambda)\boldsymbol{y}$$

(d) Final Scores



$$\boldsymbol{f} = (1-\lambda)\sum_{r=0}^{\infty}(\lambda \boldsymbol{W})^r \boldsymbol{y}$$

Normalizing Matrix W

- **Recall:** The infinite sum converges only if all eigenvalues of $\boldsymbol{W}$ are in range $[-1, 1]$
- To satisfy this condition, normalize $\boldsymbol{W}$ before diffusion:
  - **Symmetric** normalization:

$$\boldsymbol{S} = \boldsymbol{D}^{-1/2}\boldsymbol{W}\boldsymbol{D}^{-1/2}$$

Note:
$\boldsymbol{D} = diag(\boldsymbol{d})$

  - **Asymmetric** normalization:

$$\boldsymbol{P} = \boldsymbol{D}^{-1}\boldsymbol{W}$$

- **Note:** Avoid **self-reinforcement** by setting diagonal elements of $\boldsymbol{W}$ to 0
- **Note:** Label information is spread **symmetrically** since $\boldsymbol{S}$ is a symmetric matrix

Exact Solution of Label Propagation

- Given that $\rho(W) \leq 1$, use **Taylor expansion** to compute the **exact solution for label propagation:**

$$\boldsymbol{f} = (1-\lambda)\sum_{r=0}^{\infty}(\lambda \boldsymbol{S})^r \boldsymbol{y}$$

$$\Downarrow$$

$$\boldsymbol{f} = (1-\lambda)(\boldsymbol{I} - \lambda \boldsymbol{S})^{-1}\boldsymbol{y}$$

Taylor expansion:

$$(I - \boldsymbol{A})^{-1} = \sum_{r=0}^{\infty} \boldsymbol{A}^r$$

- **Note:** The diffusion result $\boldsymbol{f}$ **does not depend on the initial value** $\boldsymbol{f}^{(0)}$

"Guilty Associates": Recap

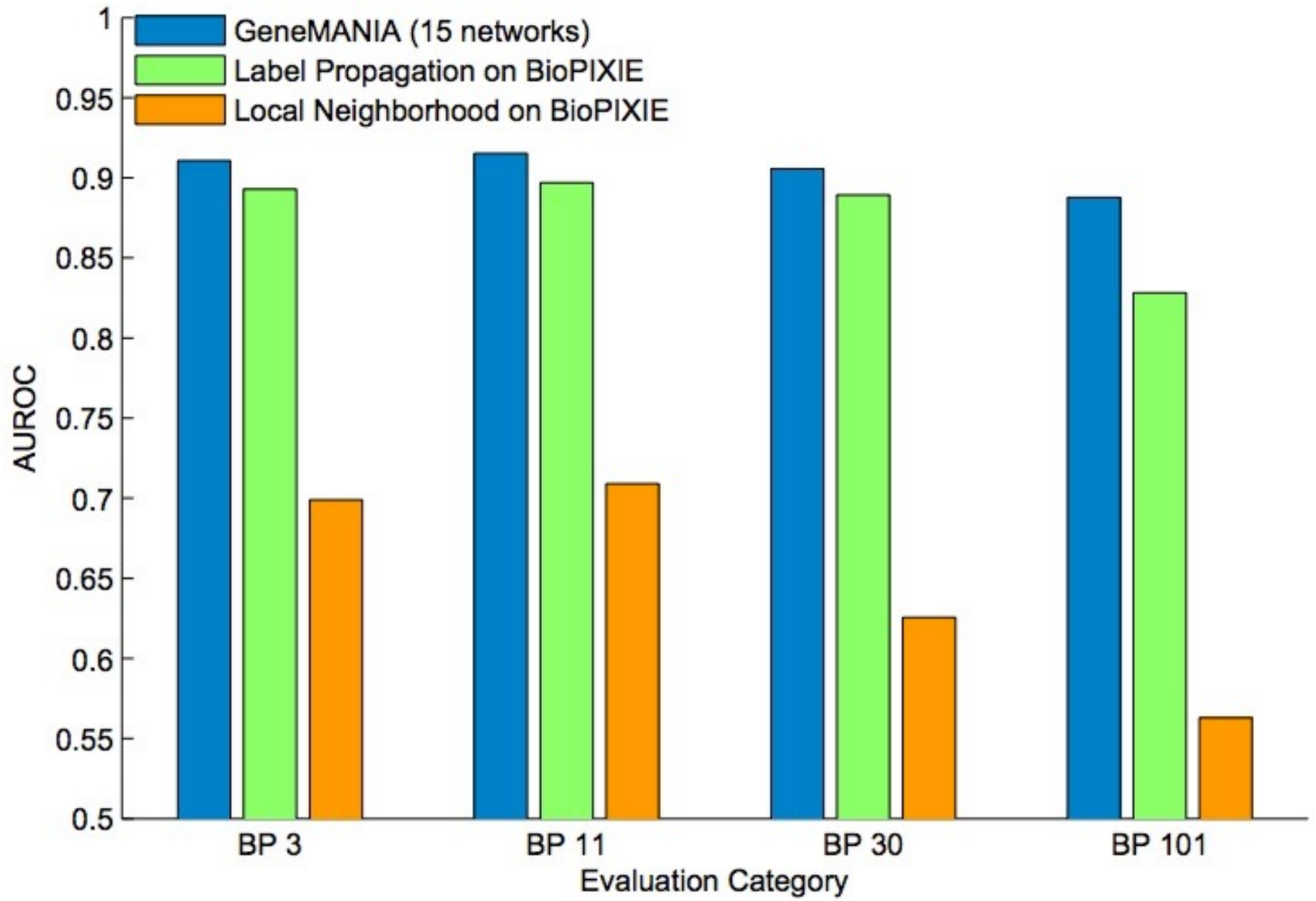
- **Direct neighbor scoring** depends on:
  - Strength of links to query genes
  - **# of query gene neighbors**
  - Example algorithm: **BioPIXIE** [Marcotte et al., Nature 1999, Myers et al., Genome Biology 2005]
- **Label propagation scoring** depends on:
  - Iteratively propagating "direct neighbor score" allowing indirect links to impact scores
  - **Whether or not a gene is in a connected component of genes with query genes**
  - Example algorithm: **GeneMANIA** [Mostafavi et al., Genome Biology 2008]

Example Biological Application

- Gene function prediction is a **multi-label node classification task**
- Every node (gene) is assigned one or more labels (cellular functions)
- **Setup:**
  1. For each gene function we use a **guilt-by-association based approach** to learn a discriminative score $\boldsymbol{f}_i$ for each node i
  2. During the training phase, we observe only a certain fraction of genes and all their functions
  3. The task is then to predict functions for the remaining genes
- Determine the optimal value of λ parameter using cross-validation

Gene Function Prediction: Results

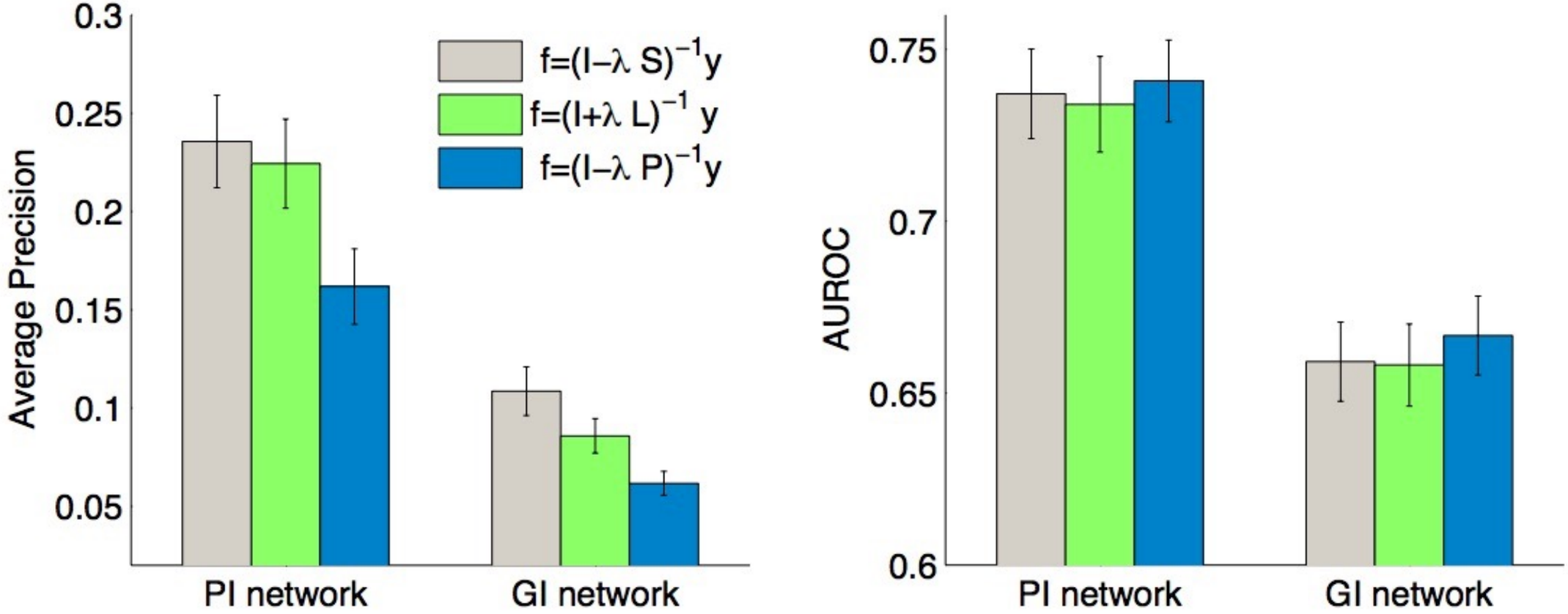
Label propagation-based approaches outperform local neighborhood-based approaches



Mostafavi et al., Genome Biology 2008

Gene Function Prediction: Results

Comparison of label propagation with three normalization methods on the protein-interaction (**PI**) and genetic-interaction (**GI**) networks



GeneMANIA Tool

Query list:

- MRE11A
- RAD51
- MLH1
- MSH2
- DMC1
- RAD51AP1
- RAD50
- MSH6
- XRCC3
- PCNA
- XRCC2