

Bioinformatics 3

V 4 – Weak Indicators and Communities

Mon, Nov 7, 2016

Noisy Data — Clear Statements?

For **yeast**: ~ 6000 proteins → ~18 million potential interactions
rough estimates: ≤ 100000 interactions occur

→ 1 true positive for 200 potential candidates = **0.5%**

→ **decisive** experiment must have **accuracy** << 0.5% false positives

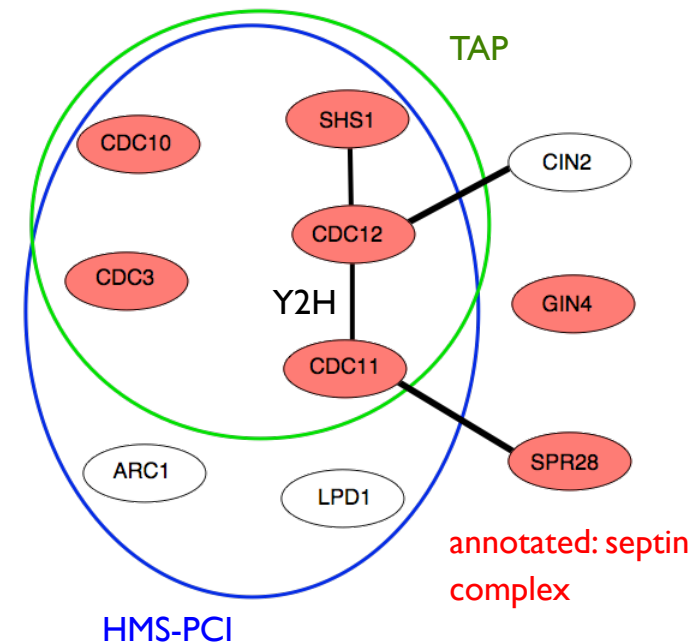
Different experiments detect different interactions

For yeast: 80000 interactions known,
only 2400 found in > 1 experiment

Y2H: → many false positives
(up to 50% errors)

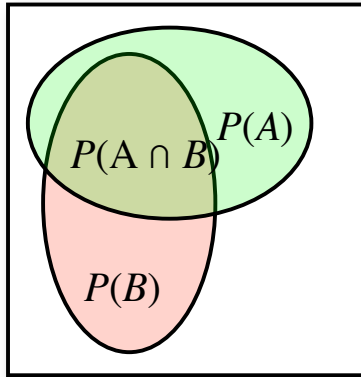
Co-expression: → gives indications at best

Combine weak indicators = ???



Conditional Probabilities

Joint probability for "A and B":



$$P(A \cap B) = P(A|B) P(B) = P(B|A) P(A)$$

Solve for conditional probability for "A when B is true"
→ Bayes' Theorem:

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)} = \frac{P(B|A)}{P(B)} P(A)$$

$P(A)$ = prior probability (marginal prob.) for "A" → no prior knowledge about A

$P(B)$ = prior probability for "B" → normalizing constant

$P(B | A)$ = conditional probability for "B given A"

$P(A | B)$ = posterior probability for "A given B"

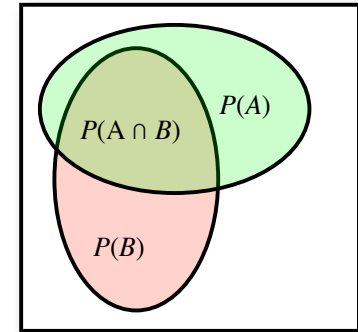
→ Use information about B to improve knowledge about A

What are the Odds?

Express Bayes theorem

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)} = \frac{P(B|A)}{P(B)} P(A)$$

in terms of odds:



- Also consider case "A does not apply": $P(\bar{A}|B) = \frac{P(B|\bar{A})}{P(B)} P(\bar{A})$

- odds for A when we know about B
(we will interpret B as information or features):

$$O(A|B) = \frac{P(A|B)}{P(\bar{A}|B)} = \frac{P(B|A)}{P(B|\bar{A})} \frac{P(A)}{P(\bar{A})} = \Lambda(A|B) O(A)$$

↑
↗
↖

posterior odds for A
likelihood ratio
prior odds for A

$\Lambda(A | B) \rightarrow$ by how much does our knowledge about A improve?

2 types of Bayesian Networks

(1) Naive Bayesian network

→ independent odds

$$O(A|B, C) = \Lambda(A|B) \Lambda(A|C) O(A)$$

(2) Fully connected Bayesian network

→ table of joint odds

	B	!B
C	0.3	0.16
!C	0.4	0.14

$\Leftrightarrow \Lambda(A|B, C)$

Bayesian Analysis of Complexes

A Bayesian Networks Approach for Predicting Protein-Protein Interactions from Genomic Data

Ronald Jansen,^{1*} Haiyuan Yu,¹ Dov Greenbaum,¹ Yuval Kluger,¹
Nevan J. Krogan,⁴ Sambath Chung,^{1,2} Andrew Emili,⁴
Michael Snyder,² Jack F. Greenblatt,⁴ Mark Gerstein^{1,3†}

We have developed an approach using Bayesian networks to predict protein-protein interactions genome-wide in yeast. Our method naturally weights and combines into reliable predictions genomic features only weakly associated with interaction (e.g., messenger RNA coexpression, coessentiality, and colocalization). In addition to de novo predictions, it can integrate often noisy, experimental interaction data sets. We observe that at given levels of sensitivity, our predictions are more accurate than the existing high-throughput experimental data sets. We validate our predictions with TAP (tandem affinity purification) tagging experiments. Our analysis, which gives a comprehensive view of yeast interactions, is available at genecensus.org/intint.

Science 302 (2003) 449

Improving the Odds

Is a given protein pair **AB** a **complex** (from all that we know)?

$$O_{post}(\text{Complex}|f_1, f_2, \dots) = \Lambda(\text{Complex}|f_1, f_2, \dots) O_{prior}(\text{Complex})$$

likelihood ratio:

improvement of the odds when
we know about features f_1, f_2, \dots

Idea: determine from known complexes and
use for prediction of new complexes

prior odds for a
random pair AB to be a
complex

estimate (somehow)

Features used by Jansen et al (2003):

- 4 experimental data sets of complexes
- mRNA co-expression profiles
- biological functions annotated to the proteins (GO, MIPS)
- essentiality for the cell

Gold Standard Sets

To determine $\Lambda(\text{Complex} | f_1, f_2, \dots) = \frac{P(f_1, f_2, \dots | \text{Complex})}{P(f_1, f_2, \dots | \text{no Complex})}$

→ use two data sets with **known** features f_1, f_2, \dots for **training**

Requirements for training data:

- i) independent of the data serving as evidence
- ii) large enough for good statistics
- iii) free of systematic bias

Gold Standard Positive Set (GP):

8250 complexes from the hand-curated MIPS catalog of protein complexes
(MIPS stands for Munich Information Center for Protein Sequences)

Gold Standard Negative Set (GN):

2708746 (non-)complexes formed by proteins from different cellular compartments
(assuming that such protein pairs likely do not interact)

Prior Odds

$$O_{prior}(\text{Complex}) = \frac{P(\text{Complex})}{P(\text{no Complex})} = \frac{P(\text{Complex})}{1 - P(\text{Complex})}$$

Jansen et al:

- estimated ≥ 30000 existing complexes in yeast
- 18 Mio. possible complexes $\rightarrow P(\text{Complex}) \approx 1/600$

$$\rightarrow O_{prior} = 1/600$$

\rightarrow The odds are 600 : 1 against picking a real complex at random

\rightarrow expect 50% good hits ($TP \geq FP$) when $\Lambda \approx 600$ and higher

Note: O_{prior} is mostly an educated guess

Essentiality

Test whether both proteins are essential (E) for the cell or not (N)

→ for protein complexes, EE or NN should occur more often

pos/neg: # of gold standard positives/
negatives with essentiality information

$$L(\text{Ess}) = \frac{P(\text{Ess} | \text{pos})}{P(\text{Ess} | \text{neg})}$$

Essentiality	pos	neg	P(Ess pos)	P(Ess neg)	L(Ess)
EE	1114	81924	5,18E-01	1,43E-01	3,6
NE	624	285487	2,90E-01	4,98E-01	0,6
NN	412	206313	1,92E-01	3,60E-01	0,5
sum	2150	573724	1,00	1,00	

possible values
of the feature

overlap of gold standard
sets with feature values

In the „pos“ case, the essentiality
was only known for 2150 out of
8250 complexes of the gold-
standard.

probabilities for each
feature value

likelihood
ratios

$$\frac{1114}{2150}$$

$$= 0,518$$

$$\frac{0,19}{0,36}$$

$$= 0,5$$

mRNA Co-Expression

Publicly available expression data from

- the Rosetta compendium
- the yeast cell cycle



Correlation between the data sets

→ use principal component

Expression correlation		# protein pairs	Gold standard overlap		$P(\text{exp} \text{pos})$	$P(\text{exp} \text{neg})$	L
			pos	neg			
Values	0.9	678	16	45	2.10E-03	1.68E-05	124.9
	0.8	4,827	137	563	1.80E-02	2.10E-04	85.5
	0.7	17,626	530	2,117	6.96E-02	7.91E-04	88.0
	0.6	42,815	1,073	5,597	1.41E-01	2.09E-03	67.4
	0.5	96,650	1,089	14,459	1.43E-01	5.40E-03	26.5
	0.4	225,712	993	35,350	1.30E-01	1.32E-02	9.9
	0.3	529,268	1,028	83,483	1.35E-01	3.12E-02	4.3
	0.2	1,200,331	870	183,356	1.14E-01	6.85E-02	1.7
	0.1	2,575,103	739	368,469	9.71E-02	1.38E-01	0.7
	0	9,363,627	894	1,244,477	1.17E-01	4.65E-01	0.3
	-0.1	2,753,735	164	408,562	2.15E-02	1.53E-01	0.1
	-0.2	1,241,907	63	203,663	8.27E-03	7.61E-02	0.1
	-0.3	484,524	13	84,957	1.71E-03	3.18E-02	0.1
	-0.4	160,234	3	28,870	3.94E-04	1.08E-02	0.0
	-0.5	48,852	2	8,091	2.63E-04	3.02E-03	0.1
	-0.6	17,423	-	2,134	0.00E+00	7.98E-04	0.0
	-0.7	7,602	-	807	0.00E+00	3.02E-04	0.0
	-0.8	2,147	-	261	0.00E+00	9.76E-05	0.0
	-0.9	67	-	12	0.00E+00	4.49E-06	0.0
Sum		18,773,128	7,614	2,675,273	1.00E+00	1.00E+00	1.0

-> Co-expression is a much better feature!

Biological Function

Use MIPS function catalog and Gene Ontology function annotations

- determine functional class shared by the two proteins; small values (1-9)

Indicate highest MIPS function or GO BP similarity

- count how many of the 18 Mio potential pairs share this classification

MIPS function similarity		# protein pairs	Gold standard overlap					$P(\text{MIPS} \text{pos})$	$P(\text{MIPS} \text{neg})$	L
			pos	neg	sum(pos)	sum(neg)	sum(pos)/ sum(neg)			
Values	1 -- 9	6,584	171	1,094	171	1,094	0.16	2.12E-02	8.33E-04	25.5
	10 -- 99	25,823	584	4,229	755	5,323	0.14	7.25E-02	3.22E-03	22.5
	100 -- 1000	88,548	688	13,011	1,443	18,334	0.08	8.55E-02	9.91E-03	8.6
	1000 -- 10000	255,096	6,146	47,126	7,589	65,460	0.12	7.63E-01	3.59E-02	21.3
	10000 -- Inf	5,785,754	462	1,248,119	8,051	1,313,579	0.01	5.74E-02	9.50E-01	0.1
Sum		6,161,805	8,051	1,313,579	-	-	-	1.00E+00	1.00E+00	1.0

GO biological process similarity		# protein pairs	Gold standard overlap					$P(\text{GO} \text{pos})$	$P(\text{GO} \text{neg})$	L
			pos	neg	sum(pos)	sum(neg)	sum(pos)/ sum(neg)			
Values	1 -- 9	4,789	88	819	88	819	0.11	1.17E-02	1.27E-03	9.2
	10 -- 99	20,467	555	3,315	643	4,134	0.16	7.38E-02	5.14E-03	14.4
	100 -- 1000	58,738	523	10,232	1,166	14,366	0.08	6.95E-02	1.59E-02	4.4
	1000 -- 10000	152,850	1,003	28,225	2,169	42,591	0.05	1.33E-01	4.38E-02	3.0
	10000 -- Inf	2,909,442	5,351	602,434	7,520	645,025	0.01	7.12E-01	9.34E-01	0.8
Sum		3,146,286	7,520	645,025	-	-	-	1.00E+00	1.00E+00	1.0

-> Co-Functionality is a semi-weak feature!

Experimental Data Sets

In vivo pull-down:	Gavin et al, <i>Nature</i> 415 (2002) 141	31304 pairs
	Ho et al, <i>Nature</i> 415 (2002) 180	25333 pairs
HT-Y2H:	Uetz et al, <i>Nature</i> 403 (2000) 623	981 pairs
	Ito et al, <i>PNAS</i> 98 (2001) 4569	4393 pairs

4 experiments on overlapping PP pairs

→ $2^4 = 16$ categories — table represents fully connected Bayes network

Gavin (g)	Ho (h)	Uetz (u)	Ito (i)	# protein pairs	Gold-standard overlap					$P(g,h,u,i pos)$	$P(g,h,u,i neg)$	L
					pos	neg	sum(pos)	sum(neg)	sum(pos)/ sum(neg)			
1	1	1	0	16	6	0	6	0	-	7.27E-04	0.00E+00	-
1	0	0	1	53	26	2	32	2	16.0	3.15E-03	7.38E-07	4268.3
1	1	1	1	11	9	1	41	3	13.7	1.09E-03	3.69E-07	2955.0
1	0	1	1	22	6	1	47	4	11.8	7.27E-04	3.69E-07	1970.0
1	1	0	1	27	16	3	63	7	9.0	1.94E-03	1.11E-06	1751.1
1	0	1	0	34	12	5	75	12	6.3	1.45E-03	1.85E-06	788.0
1	1	0	0	1920	337	209	412	221	1.9	4.08E-02	7.72E-05	529.4
0	1	1	0	29	5	5	418	227	1.8	6.06E-04	1.85E-06	328.3
0	1	1	1	16	1	1	413	222	1.9	1.21E-04	3.69E-07	328.3
0	1	0	1	39	3	4	421	231	1.8	3.64E-04	1.48E-06	246.2
0	0	1	1	123	6	23	427	254	1.7	7.27E-04	8.49E-06	85.7
1	0	0	0	29221	1331	6224	1758	6478	0.3	1.61E-01	2.30E-03	70.2
0	0	1	0	730	5	112	1763	6590	0.3	6.06E-04	4.13E-05	14.7
0	0	0	1	4102	11	644	1774	7234	0.2	1.33E-03	2.38E-04	5.6
0	1	0	0	23275	87	5563	1861	12797	0.1	1.05E-02	2.05E-03	5.1
0	0	0	0	2702284	6389	2695949	8250	2708746	0.0	7.74E-01	9.95E-01	0.8

Statistical Uncertainties

Gavin (g)	Ho (h)	Uetz (u)	Ito (i)	# protein pairs	Gold		$P(g,h,u,i pos)$	$P(g,h,u,i neg)$	L
					pos	neg			
1	1	1	0	16	6	0	7.27E-04	0.00E+00	-
1	0	0	1	53	26	2	3.15E-03	7.38E-07	4268.3
1	1	1	1	11	9	1	1.09E-03	3.69E-07	2955.0
1	0	1	1	22	6	1	7.27E-04	3.69E-07	1970.0
1	1	0	1	27	16	3	1.94E-03	1.11E-06	1751.1
1	0	1	0	34	12	5	1.45E-03	1.85E-06	788.0

1) $L(IIII) < L(I00I)$

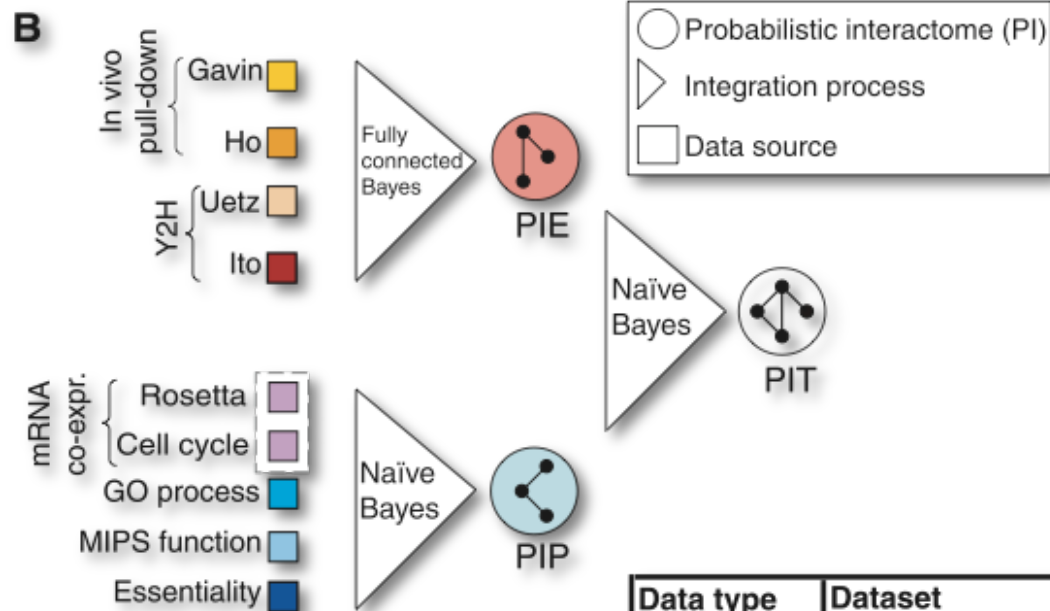
statistical uncertainty: $\Delta N = \sqrt{N + 1}$

Overlap with all experiments is smaller \rightarrow larger uncertainty

2) $L(IIIO) = \text{NaN?}$

Use conservative lower bound \rightarrow assume 1 overlap with GN
 $\rightarrow L(IIIO) \geq 1970$

Overview

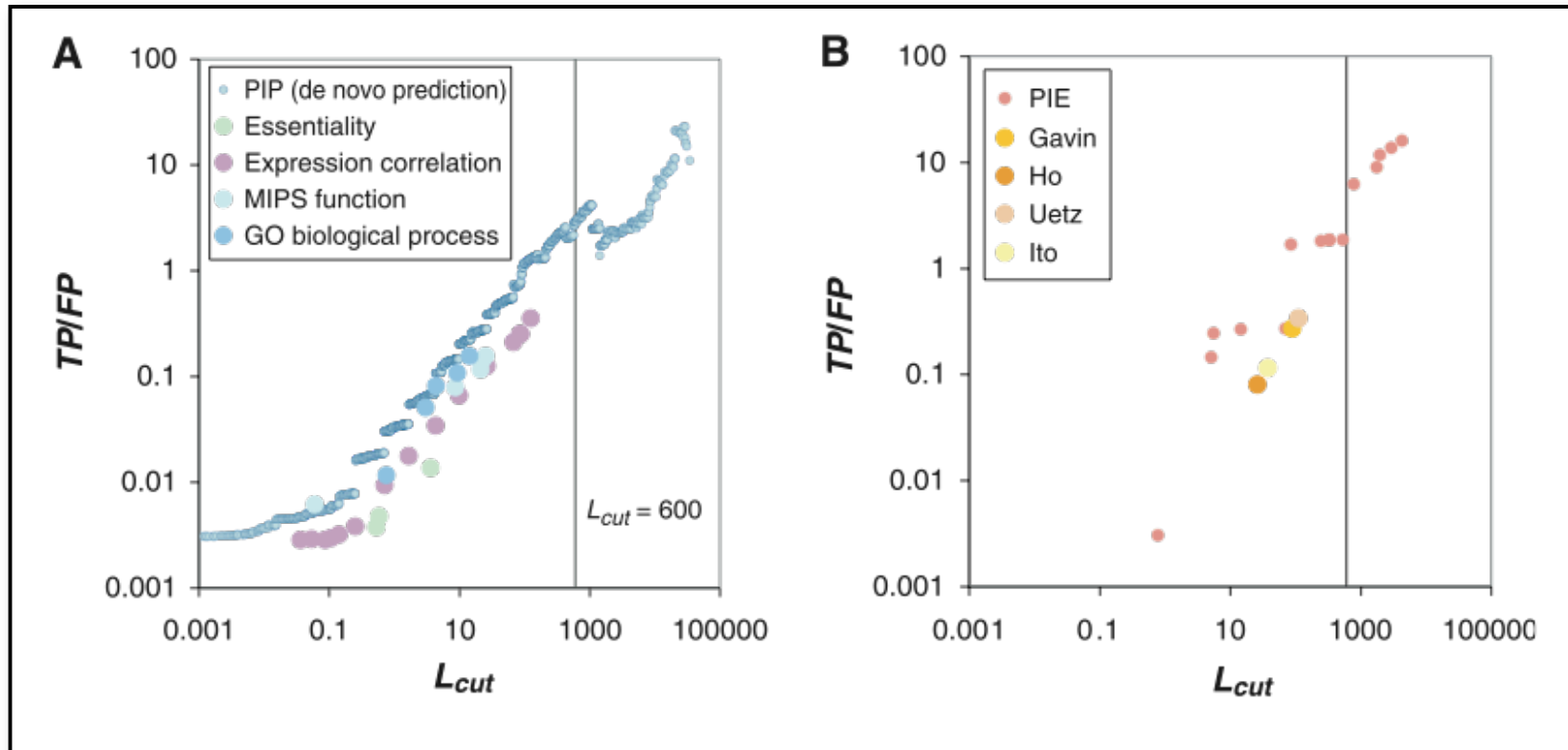


Data type	Dataset			# protein pairs	Used for ...
Experimental interaction data	In-vivo pull-down	Gavin et al.		31,304	Integration of experimental interaction data (PIE)
		Ho et al.		25,333	
	Yeast two-hybrid	Uetz et al.		981	
		Ito et al.		4,393	
Other genomic features	mRNA Expression	Rosetta compendium		19,334,806	De novo prediction (PIP)
		Cell cycle		17,467,005	
	Biological function	GO biological process		3,146,286	
		MIPS function		6,161,805	
		Essentiality		8,130,528	
Gold standards	Positives	Proteins in the same MIPS complex		8,250	Training & testing
	Negatives	Proteins separated by localization		2,708,746	

Performance of complex prediction

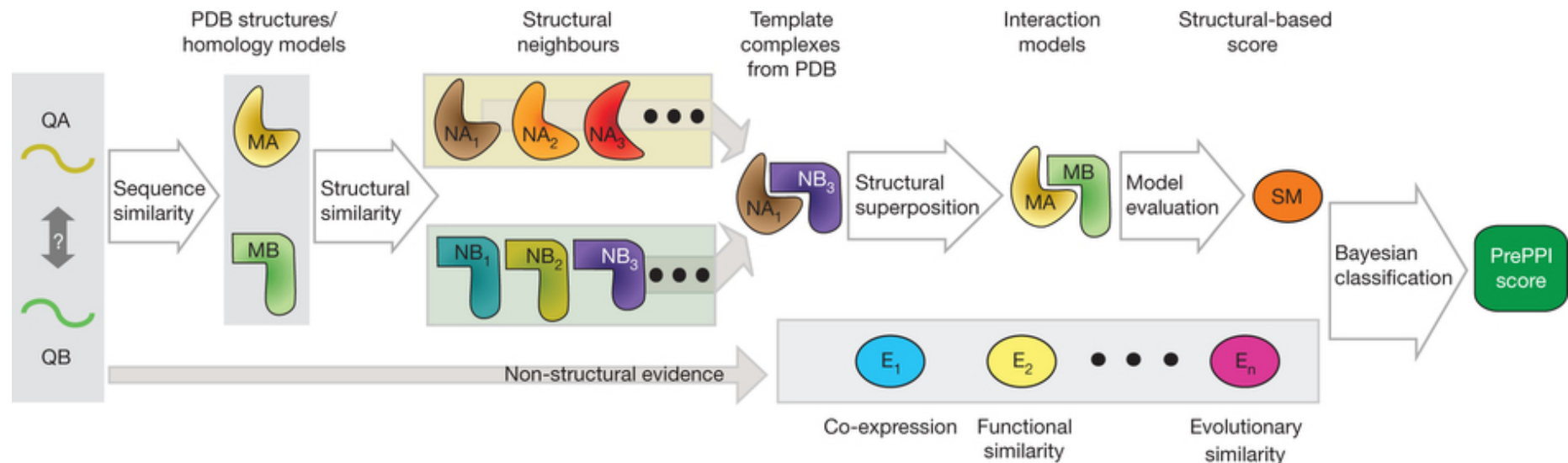
Predictions

Experimental data



None of the individual evidences alone was enough to get a likelihood ratio > 600 , neither predicted nor experimental evidences

Follow-up work: PrePPI (2012)



Given a pair of query proteins that potentially interact (QA, QB), representative structures for the individual subunits (MA, MB) are taken from the PDB, where available, or from homology model databases.

For each subunit we find both close and remote structural neighbours. A 'template' for the interaction exists whenever a PDB or PQS structure contains a pair of interacting chains (for example, NA₁-NB₃) that are structural neighbours of MA and MB, respectively. A model is constructed by superposing the individual subunits, MA and MB, on their corresponding structural neighbours, NA₁ and NB₃.

We assign 5 empirical-structure-based scores to each interaction model and then calculate a likelihood for each model to represent a true interaction by combining these scores using a Bayesian network trained on the HC and the N interaction reference sets.

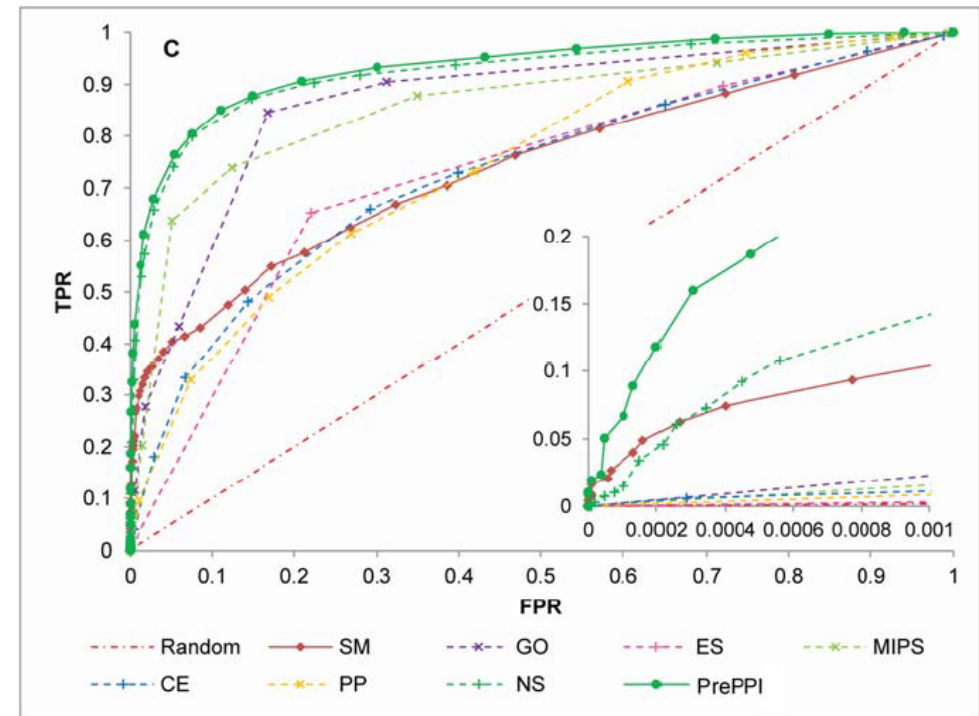
We finally combine the structure-derived score (SM) with non-structural evidence associated with the query proteins (for example, co-expression, functional similarity) using a naive Bayesian classifier.

Results of PrePPI

Receiver-operator characteristics (ROC) for predicted yeast complexes.

Examined features:

- structural modeling (SM),
- GO similarity,
- protein essentiality (ES) relationship,
- MIPS similarity,
- co-expression (CE),
- phylogenetic profile (PP) similarity.



Also listed are 2 combinations:

- NS for the integration of all non-structure clues, i.e. GO, ES, MIPS, CE, and PP, and
- PrePPI for all structural and non-structure clues).

This gave 30.000 high-confidence PP interactions for yeast and 300.000 for human.

Summary: Bayesian Analysis

Combination of weak features yields powerful predictions

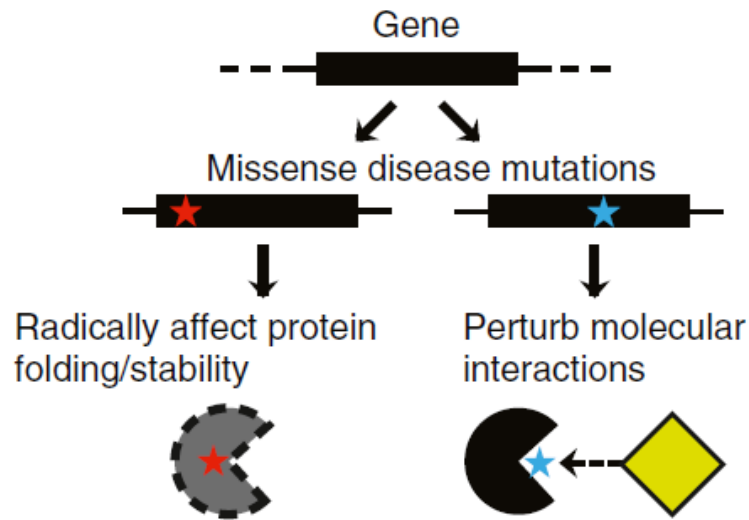
- boosts odds via Bayes' theorem
- Gold standard sets for training the likelihood ratios

Bayes vs. other **machine learning** techniques:

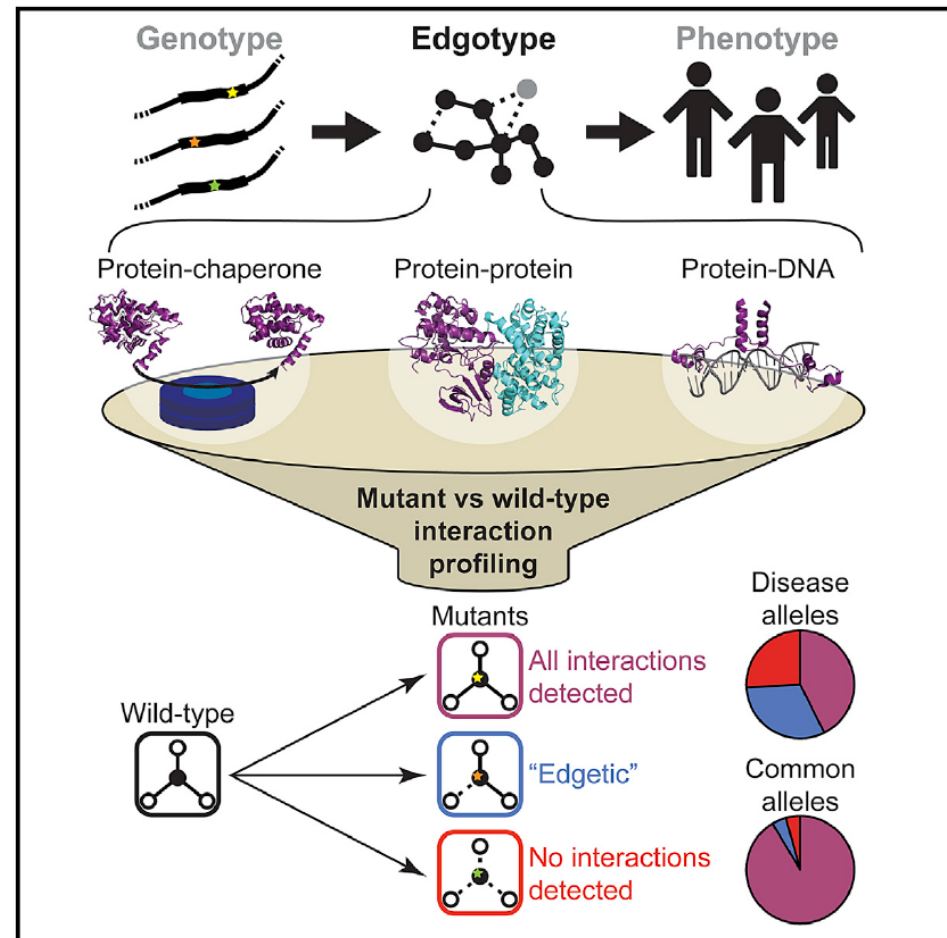
(voting, unions, SVM, neuronal networks, decision trees, ...)

- **arbitrary types** of data can be combined
- weight data according to their **reliability**
- include conditional relations between evidences
- easily accommodates missing data (e.g., zero overlap with GN)
- **transparent** procedure
- predictions easy to **interpret**

Insert: Relation of PPI networks to diseases



In principle, a protein mutant can destabilize proteins (left) or perturb interactions (right)



Can one study this systematically on a genome-level?

Sahni et al., Marc Vidal (2015)
Cell 161, 647–660

Experimental pipeline

Select mutations associated with a wide range of disorders, including cancer susceptibility and heart, respiratory, and neurological diseases.

Out of 16,400 mutations affecting over 1,200 genes for which we have a wild-type (WT) open-reading frame (ORF) clone in our human “ORFeome” collection we selected up to four mutations per gene.

Sahni et al., Marc Vidal (2015)
Cell 161, 647–660

Effect on protein folding

How do disease mutations impact protein folding and disposition?

Measure how well hmORF-encoded proteins and their WT counterparts interact with quality control factors (QCFs) using a quantitative high-throughput LUMIER assay.

We selected the following QCFs based on their broad specificity:

- (1) the cytoplasmic chaperones HSP90 and HSC70,
- (2) their co-chaperones BAG2 and CHIP/STUB1,
- (3) the proteasomal regulatory subunit PSMD2 and
- (4) the ER chaperones GRP78/BIP and GRP94.

Idea: Increased interaction between a QCF and mutant or WT protein, as measured by the LUMIER assay, indicates a mutation-induced perturbation in conformational stability, often associated with compromised or complete loss of function.

Sahni et al., Marc Vidal (2015)
Cell 161, 647–660

Lumier assay



Luminescence-based mammalian interactome mapping (LUMIER).

In a LUMIER assay, a luciferase-tagged 'bait' protein is screened against a series of Flag-tagged 'prey' proteins.

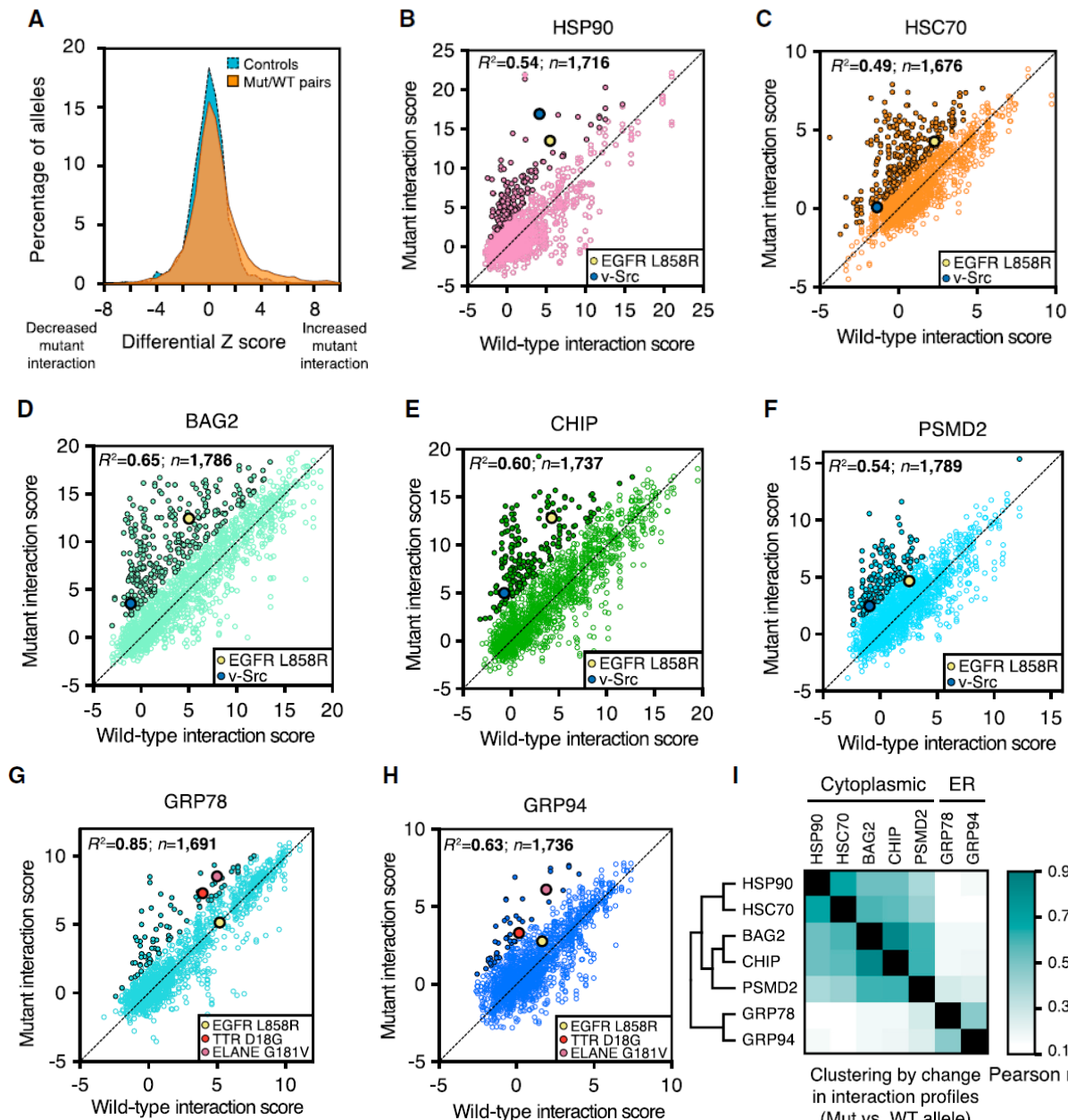
An antibody against Flag is used to affinity-purify the prey, and the prey-associated luminescence reveals the extent of bait interaction

The antibodies (yellow) are immobilized on sepharose beads (black).

An array scanner can be used to quantify the relative extent of interaction for large numbers of assays.

Barrios-Rodiles, M. *et al.* High-throughput mapping of a dynamic signaling network in mammalian cells. *Science* **307**, 1621–1625 (2005).

Interaction with QCFs



The interaction profiles of most mutant proteins correlated with their WT counterparts. However, compared to a background control set, we observed a **significant enrichment** of mutant alleles showing **increased interaction** with QCFs (A–H) but little or no enrichment for decreased interaction (A).

(I) The interaction profiles of mutant proteins with the different cytoplasmic QCFs were highly correlated, distinct from those with ER factors.

These results highlight the coordination and specificity of cellular quality control pathways. 28% of the tested alleles exhibited increased binding to at least 1 of the 7 QCFs tested.

Sahni et al., Marc Vidal (2015)

Cell 161,647–660

V 4 – 24

Y2H: screen native PPIs

Systematic characterization of PPI perturbations associated with disease mutations by Y2H.

2,449 mutant proteins and their 1,072 corresponding WT proteins were screened for interactions with proteins encoded by the 7,200 ORFs in the human ORFeome v1.1.

Mutant and WT proteins were then tested pair-wise against all partners found both in these Y2H screens and in our human interactome map HI-II-14.

Altogether, we obtained interaction profiles for 460 mutant proteins and their 220 WT counterparts and found 521 perturbed interactions out of 1,316 PPIs.

Only two mutations conferred PPI gains, suggesting that gain of interactions may be a rare event in human disease.

Sahni et al., Marc Vidal (2015)
Cell 161, 647–660

Findings

Ca. 60% of disease-associated missense mutations perturb PPIs.

- Of these, half result in complete loss of interactions, generally caused by protein misfolding and impaired expression.
- The other half lead to edgetic perturbations.

Importantly, different mutations in the same gene frequently result in different interaction perturbation profiles.

Sahni et al., Marc Vidal (2015)
Cell 161, 647–660

Connected Regions

Observation: **more interactions inside** a complex than to the outside

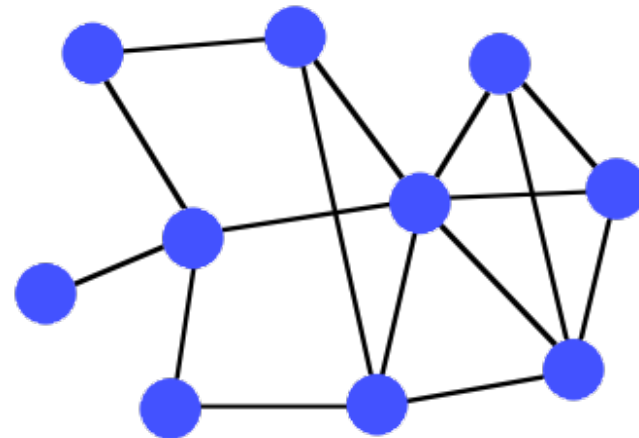
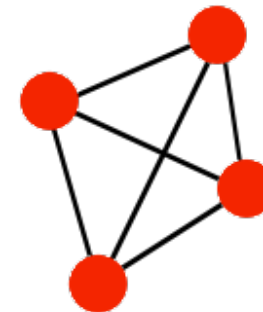
→ how can one identify highly connected regions in a network?

I) Fully connected region: **Clique**

$$\text{clique} := G' = (V', E' = V'^{(2)})$$

Problems with cliques:

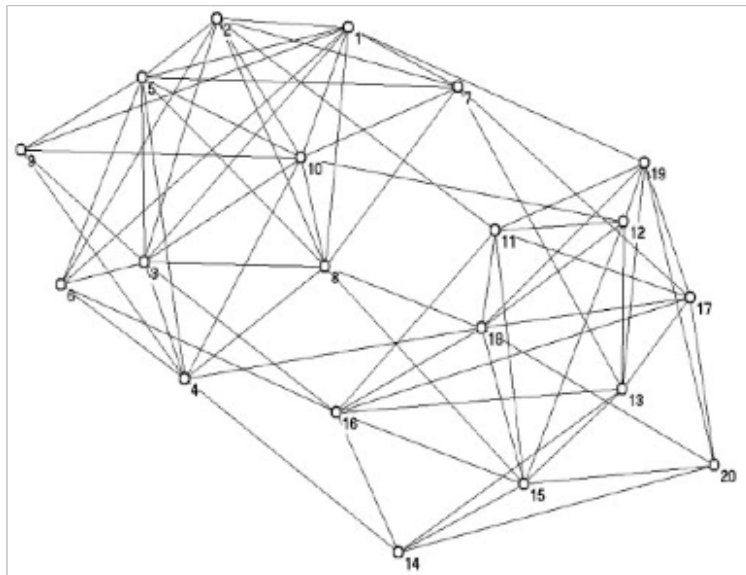
- finding cliques is **NP-hard**
(but "works" $O(N^2)$ for the sparsely connected biological networks)
- **biological** protein complexes are **not** always **fully** connected



Communities

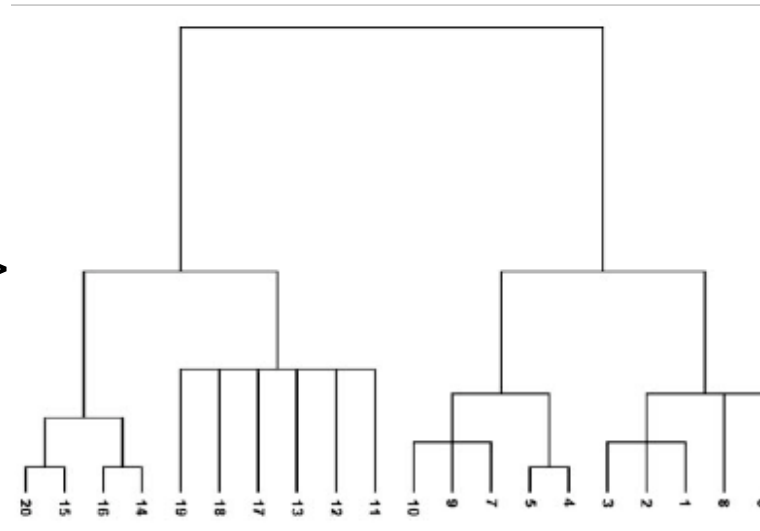
Community := subset of vertices, for which the **internal** connectivity is **denser** than to the outside

Aim: map network onto tree that reflects the community structure



???

\Leftrightarrow



Radicchi et al, *PNAS* **101** (2004) 2658:

Hierarchical Clustering

- 1) Assign a weight W_{ij} to each pair of vertices i, j that measures how "closely related" these two vertices are.
- 2) Iteratively add edges between pairs of nodes with decreasing W_{ij}

Measures for W_{ij} :

- 1) Number of **vertex-independent paths** between vertices i and j
(vertex-independent paths between i and j : no shared vertex except i and j)

Menger (1927): the number of vertex-independent paths equals the number of vertices that have to be removed to cut all paths between i and j
→ measure for network robustness

- 2) Number of **edge-independent paths** between i and j

- 3) **Total number of paths** L between i and j

but $L = 0$ or ∞ → weight paths with their length α^L with $\alpha < 1$

Problem: vertices with a single link are separated from the communities

Vertex Betweenness

Freeman (1927): count on how many shortest paths a vertex is visited

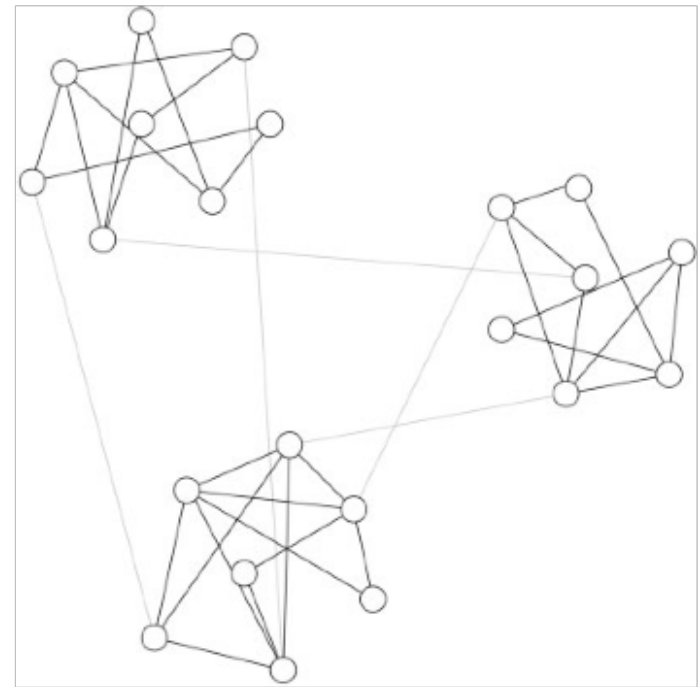
For a graph $G = (V, E)$ with $|V| = n$

Betweenness for vertex ν :

$$C_B(\nu) = \frac{\sum_{s \neq \nu \neq t \in V} \sigma_{st}(\nu)}{(n-1)(n-2)}$$

Alternative: **edge betweenness**

→ to how many shortest paths does
this edge belong



Girvan-Newman Algorithm

Girvan, Newman, *PNAS* **99** (2002) 7821:

For a graph $G = (V, E)$ with $|V| = n$, $|E| = m$

- 1) Calculate **betweenness** for all m edges
- 2) **Remove** edge with highest betweenness
- 3) **Recalculate** betweenness for all affected nodes
- 4) **Repeat** from 2) until no more edge is left (at most m iterations)
- 5) Build up **tree** from V by reinserting vertices in reverse order

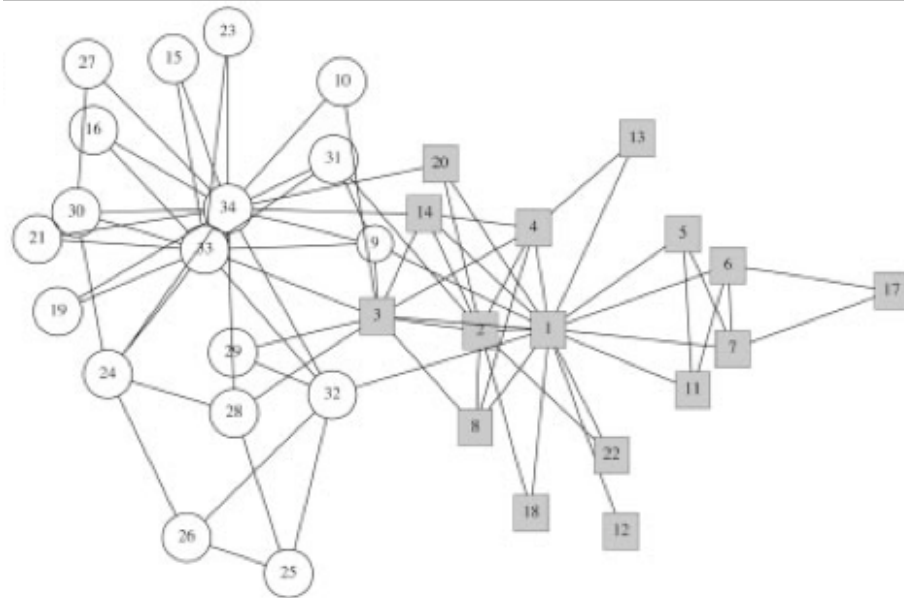
Works well, but **slow**: $O(mn^2) \approx O(n^3)$ for scale-free networks ($|E| = 2 |V|$)

Reason for complexity: compute shortest paths (n^2) for m edges

→ recalculating a **global** property is expensive for larger networks

Zachary's Karate Club

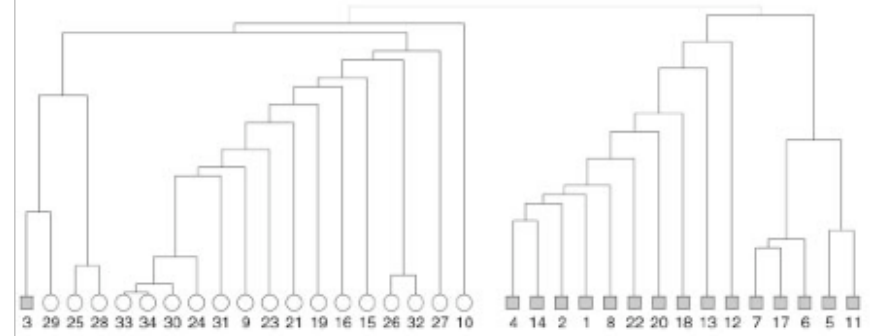
- observed friendship relations of 34 members over two years
- correlate fractions at break-up with calculated communities



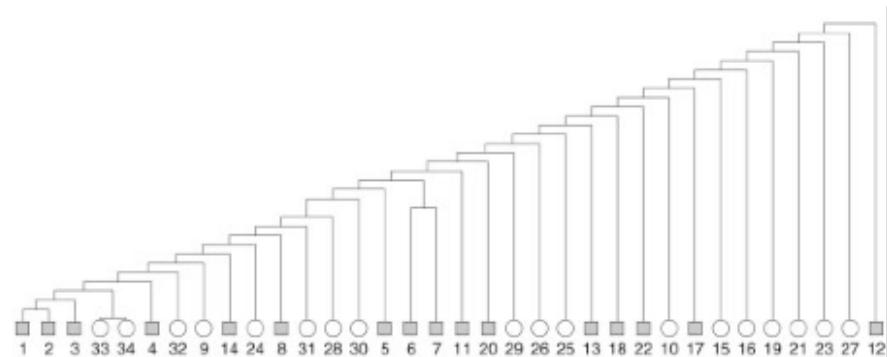
administrator's
fraction

instructor's
fraction

with edge betweenness:

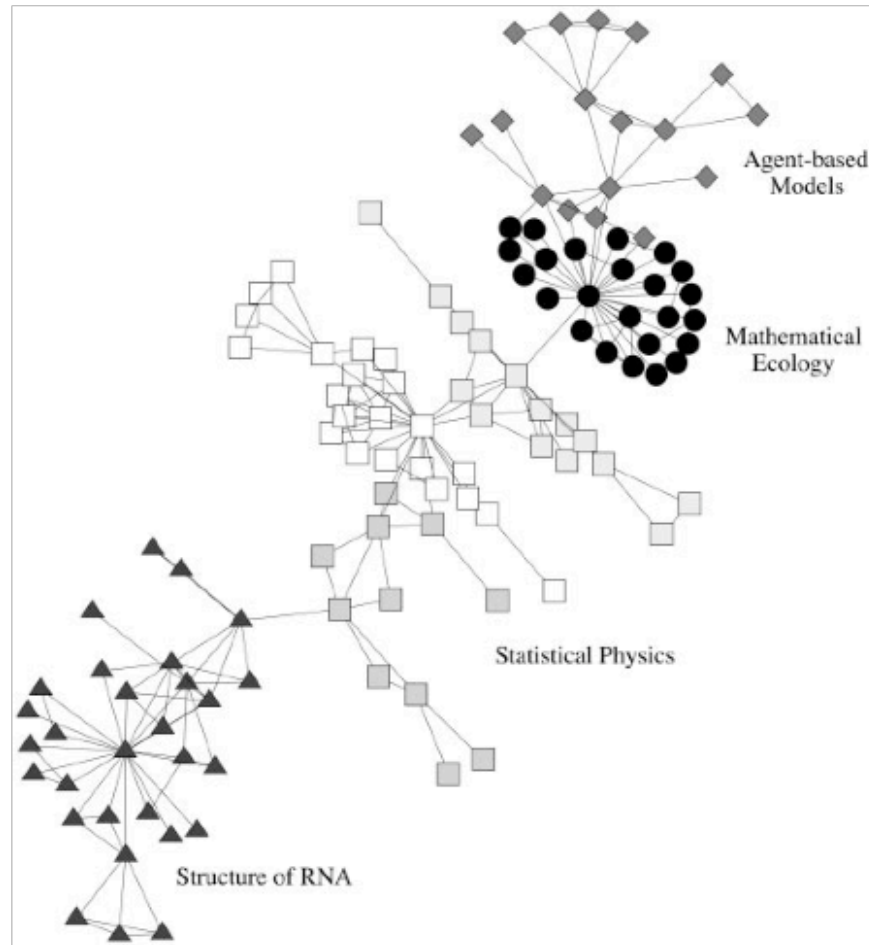


with number of edge-independent paths:



Girvan, Newman, *PNAS* **99** (2002) 7821

Collaboration Network



The largest component of the Santa Fe Institute collaboration network, with the primary divisions detected by the GN algorithm indicated by different vertex shapes.

Edge: two authors have co-authored a joint paper.

Girvan, Newman, *PNAS* **99** (2002) 7821

Determining Communities Faster

Radicchi et al, *PNAS* **101** (2004) 2658:

Determine edge weights via **edge-clustering coefficient**

→ local measure

→ much faster, esp. for large networks

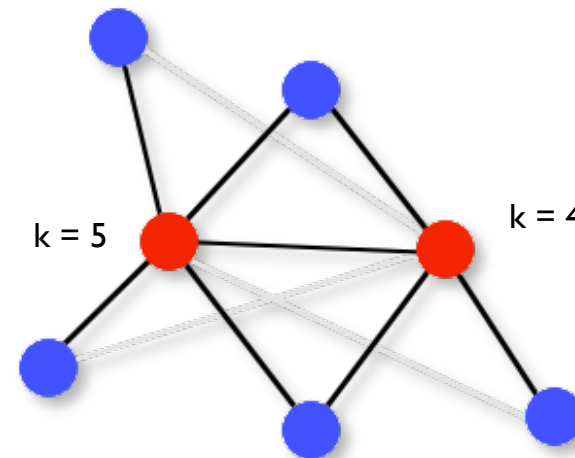
Modified edge-clustering coefficient:

→ fraction of potential triangles
with edge between i and j

$$C_{i,j}^{(3)} = \frac{z_{i,j}^{(3)} + 1}{\min[(k_i - 1), (k_j - 1)]}$$

Here, $z_{i,j}^{(3)}$ is the number of triangles,
 k_i and k_j are the degrees of nodes i and j .

Note: "+ 1" to remove degeneracy for $z_{i,j}^{(3)} = 0$

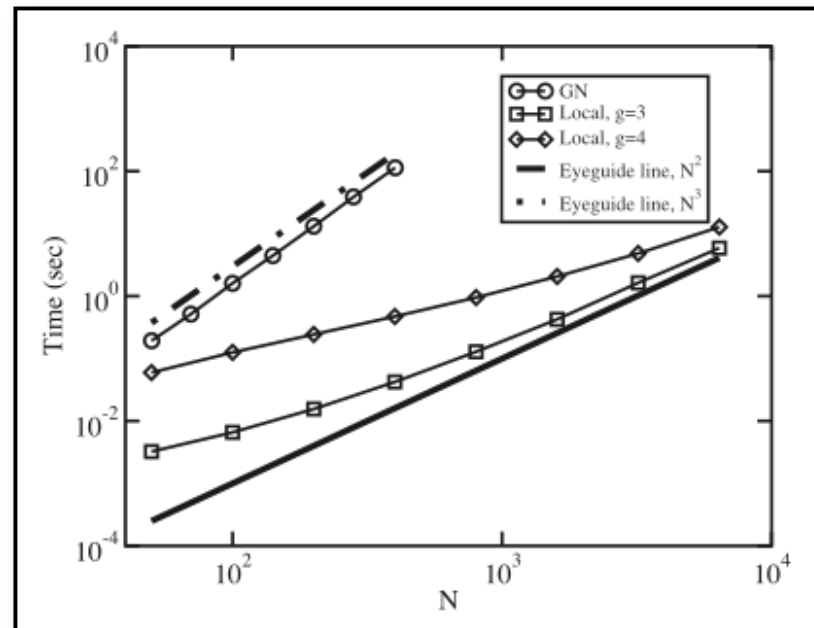


$$C^{(3)} = (2+1) / 3 = 1$$

Performance

Instead of triangles: **cycles** of higher order g
→ continuous transition to a global measure

$$C_{i,j}^{(g)} = \frac{z_{i,j}^{(g)} + 1}{s_{i,j}^{(g)}}$$

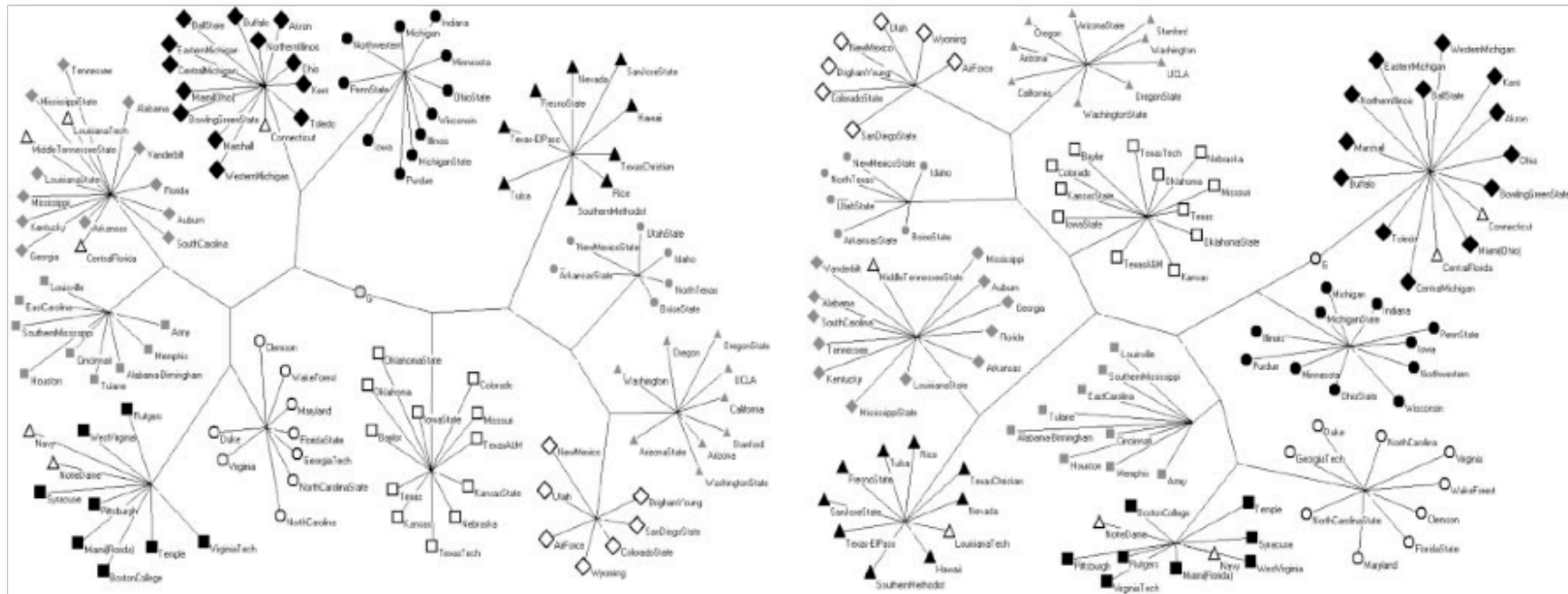


Radicchi *et al*-algorithm: $O(N^2)$ for large networks

Radicchi et al, *PNAS* **101** (2004) 2658:

Comparison of algorithms

Data set: football teams from US colleges; different symbols = different conferences, teams played ca. 7 intraconference games and 4 inter-conference games in 2000 season.



Girven-Newman algorithm

Radicchi with $g = 4$

→ very similar communities

Comparison of modularity maximization methods

A large number of approaches have been developed to maximize modularity for divisions into any number of communities of any sizes.

Author	Ref.	Label	Order
Eckmann & Moses	[13]	EM	$O(m\langle k^2 \rangle)$
Zhou & Lipowsky	[14]	ZL	$O(n^3)$
Latapy & Pons	[15]	LP	$O(n^3)$
Newman	[24]	NF	$O(n \log^2 n)$
Newman & Girvan	[25]	NG	$O(m^2 n)$
Girvan & Newman	[32]	GN	$O(n^2 m)$
Guimerà et al.	[27, 43]	SA	parameter dependent
Duch & Arenas	[31]	DA	$O(n^2 \log n)$
Fortunato et al.	[33]	FLM	$O(n^4)$
Radicchi et al.	[34]	RCCLP	$O(n^2)$
Donetti & Muñoz	[35, 36]	DM/DMN	$O(n^3)$
Bagrow & Boltt	[37]	BB	$O(n^3)$
Capocci et al.	[38]	CSCC	$O(n^2)$
Wu & Huberman	[39]	WH	$O(n + m)$
Palla et al.	[40]	PK	$O(\exp(n))$
Reichardt & Bornholdt	[41]	RB	parameter dependent

Table 1. Table summarising how the computational cost of different approaches scales with number of nodes n , number of links m and average degree $\langle k \rangle$ [42]. The labels shown here are used in Figures 2 and 3.

Danon, Duch, Diaz-Guilera, Arenas, J. Stat. Mech. P09008 (2005)

Comparison of modularity maximization methods

One way to test the **sensitivity** of these methods is to see how well a particular method performs when it is applied to ad hoc networks with a well known, fixed community structure.

Such networks are typically generated with $n = 128$ nodes, split into 4 communities containing 32 nodes each.

Pairs of nodes belonging to the same community are linked with probability p_{in} whereas pairs belonging to different communities are joined with probability p_{out} .

The value of p_{out} is taken so that the average number of links that a node has to members of any other community, z_{out} , can be controlled.

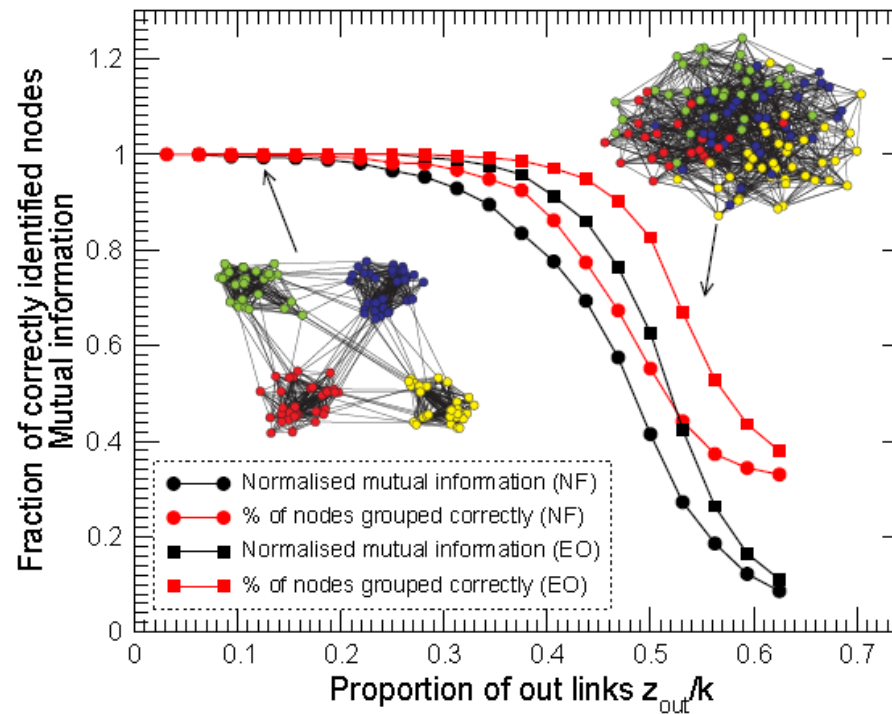
While p_{out} (and therefore z_{out}) is varied freely, the value of p_{in} is chosen to keep the total average node degree, k constant, and is set to 16.

Danon, Duch, Diaz-Guilera, Arenas, J. Stat. Mech. P09008 (2005)

Comparison of modularity maximization methods

As z_{out} increases, the communities become more and more diffuse and harder to identify, (see figure).

Since the “real” community structure is well known in this case, it is possible to measure the number of nodes correctly classified by the method of community identification.



Danon, Duch, Diaz-Guilera, Arenas, J. Stat. Mech. P09008 (2005)

Comparison of modularity maximization methods

One of the most successful approaches is **simulated annealing**.

The process begins with any initial partition of the nodes into communities.

At each step, a node is chosen at random and moved to a different community, also chosen at random.

If the change improves the modularity it is always accepted, otherwise it is accepted with a probability $\exp(\Delta Q/kT)$.

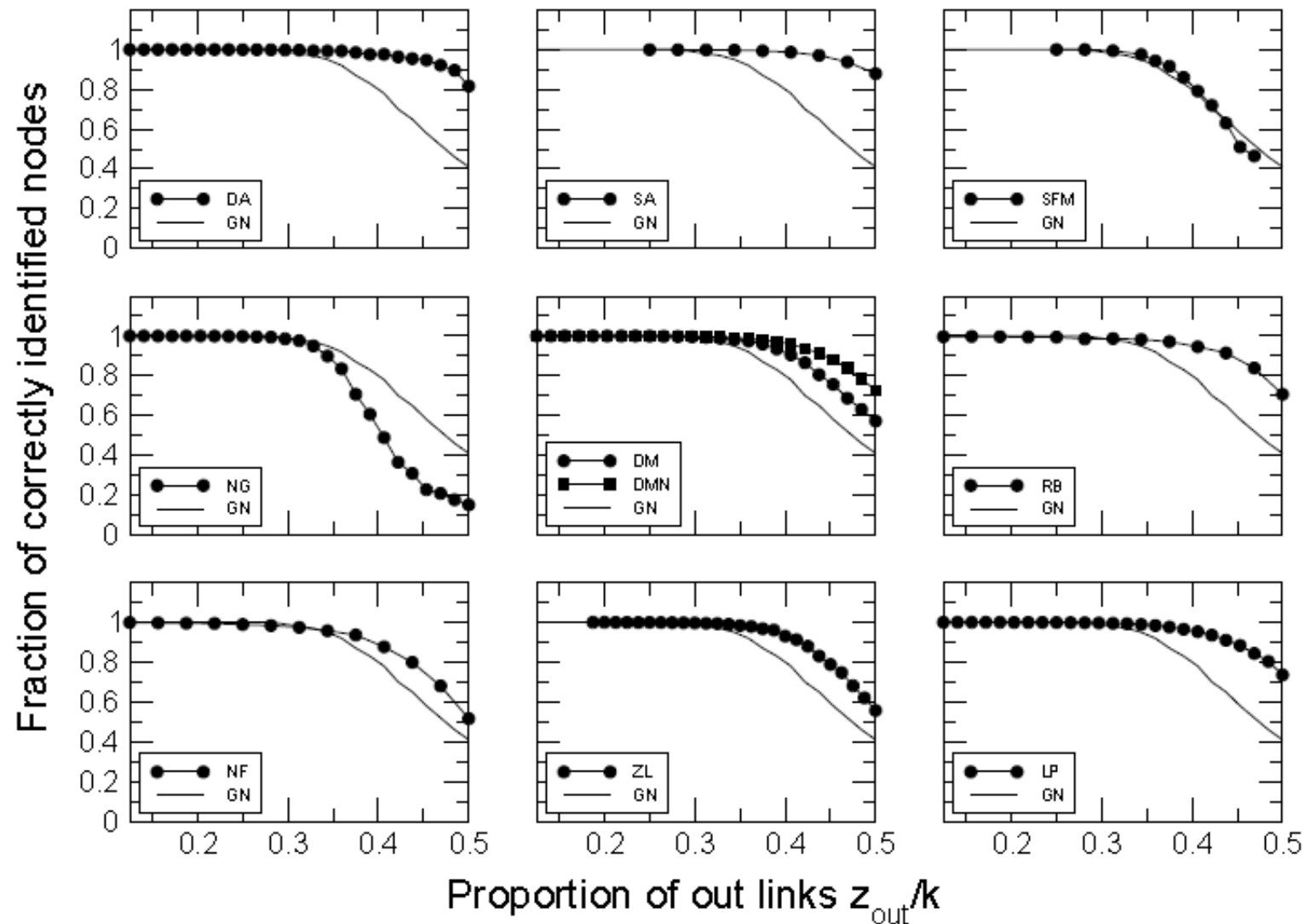
The simulation will start at high temperature T and is then slowly cooled down.

Several improvements have been tested.

Firstly, the algorithm is stopped periodically, or quenched, and ΔQ is calculated for moving each node to every community that is not its own.

Finally, the move corresponding to the largest value of ΔQ is accepted.

Comparison of modularity maximization methods



GN:
Girven-Newman
algorithm (used as
standard here).

Most modern
algorithms work
better than GN.

Danon, Duch, Diaz-Guilera, Arenas, J. Stat. Mech. P09008 (2005)

Strong Communities

"Community := subgraph with more interactions inside than to the outside"

A subgraph V is a **community** in a...

...**strong** sense when:

$$k_i^{in}(V) > k_i^{out}(V) \quad \forall i \in V$$

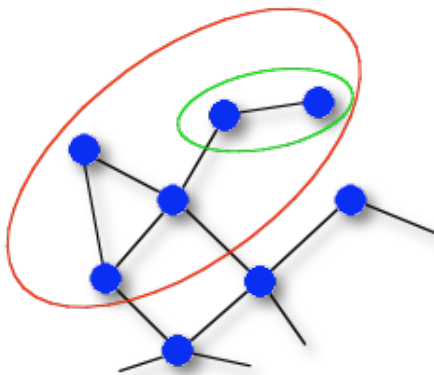
→ Check every node individually

...**weak** sense when:

$$\sum_{i \in V} k_i^{in}(V) > \sum_{i \in V} k_i^{out}(V)$$

→ allow for borderline nodes

Radicchi et al, *PNAS* **101** (2004) 2658



- $\sum k_{in} = 2, \sum k_{out} = 1$
 $\{k_{in}, k_{out}\} = \{1, 1\}, \{1, 0\}$

→ community in a weak sense

- $\sum k_{in} = 10, \sum k_{out} = 2$
 $\{k_{in}, k_{out}\} = \{2, 1\}, \{2, 0\}, \{3, 1\}, \{2, 0\}, \{1, 0\}$

→ community in a strong and weak sense

Summary

What you learned **today**:

- how to combine a set of **noisy evidences** into a **powerful** prediction tool
→ Bayes analysis
- how to find **communities** in a network efficiently
→ betweenness, edge-cluster-coefficient

Next lecture: Fri, Nov 11, 2016

- Modular decomposition
- Robustness