Large-scale structures in networks: Hidden communities and latent hierarchies

Daniel Larremore
Assistant Professor
Dept. of Computer Science
& BioFrontiers Institute

May 27, 2019
NetSci

University of Colorado **Boulder**

daniel.larremore@colorado.edu
@danlarremore

PDF of slides available
http://danlarremore.com/**CommunityDetection_and_Ranking_Larremore_2019.pdf**

Goals for this talk:

1. **Why** do we look for large-scale structure? 🤔
2. **How** do we find communities and hierarchies? 🧐
3. **Where** can we read more details? 📚

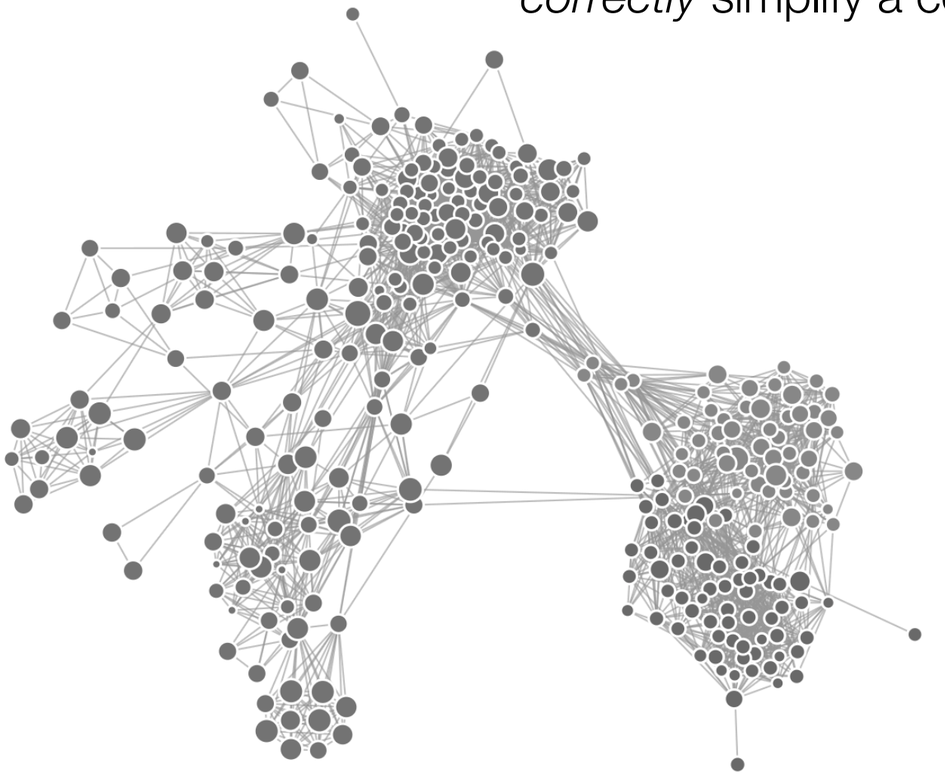
Simplicity is a great virtue but it requires hard work to achieve it and education to appreciate it. And to make matters worse: complexity sells better.

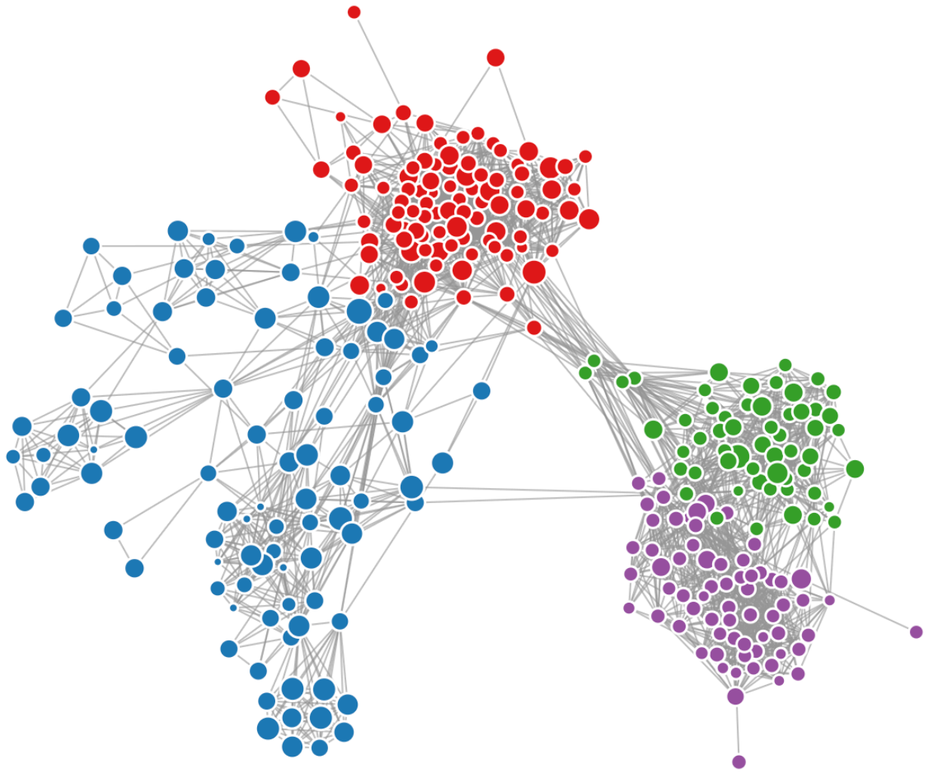
E. W. Dijkstra

We can interpret this in two ways:

The Cynic: Pictures of networks can be *really cool* but our goal is to do good science, not make pretty pictures.

The Scientist: The most beautiful science is when we *correctly* simplify a complex system.





What do we mean by “large-scale structure” ?

Structure is what makes data different from noise.
It’s what makes a network different from a random graph.

Networks are often too large and complex to be adequately summarized by a few scalars, like the number of nodes, the number of edges, or the mean degree.

However, they are also often too large and complex to be analyzed *without* some kind of simplification!

Therefore, understanding what the network *means* requires that we identify key structures.

Searching for large-scale structures in a network reflects a belief that in all the complexity there are patterns that make the network less complicated.

We define these large-scale structures—models, really—to compress complex networks.

I first heard “structure is what makes data different from noise” in a lecture by Aaron Clauset.

Goal: understanding, not a list of parts and dimensions

Finding large-scale structures is the same as anything else:

We want a **simplified model** of something very complicated.

We want to know what the **important pieces** are, and **how they fit together**.

Adapted from a similar slide from Aaron Clauset.

Many uses for models of large-scale structure

Treat the network like a system:

Extrapolation. Make predictions for as-yet unseen nodes (in “space” or time).
Interpolation. Identify missing links.
Generalization. Nodes of this type are like others of the same type.

Treat the network like an artifact:

Mechanisms. How did this network arise? What rules governed its assembly?
Explanations. Coarse-graining or compression.

Treat the network like a means to an end; an intermediate data structure:

Useful division. Need groups so that we can assign treatments in an A/B test.
Simplification. Downstream regression model needs ranks or groups.

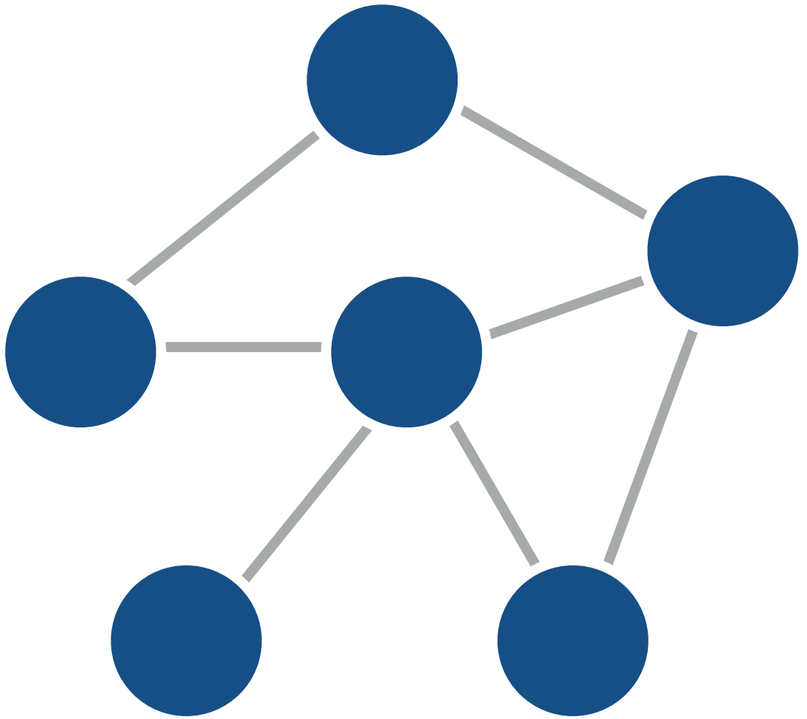
intuition: compare this list with the list you would write for regression

Community structure

Homophily & assortative mixing

like links with *like*

Assortativity coefficient *r* measures extent of homophily.



Newman, Phys. Rev. E 67, 026126 (2003).

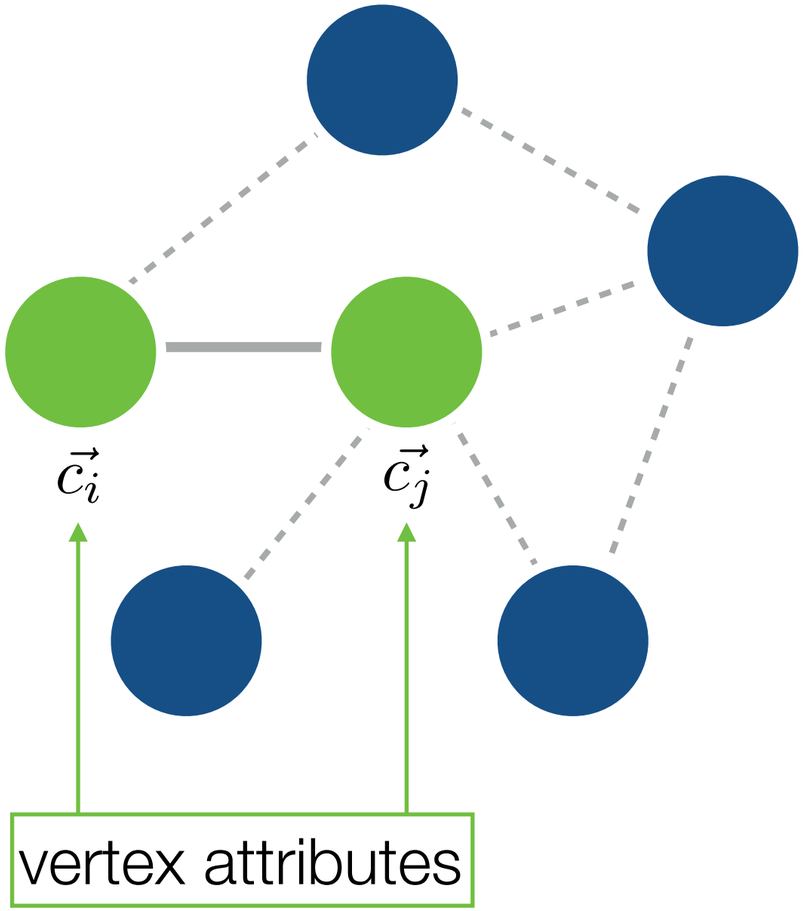
Homophily & assortative mixing

like links with *like*

Assortativity coefficient *r* measures extent of homophily.

Three types:

- scalar attributes
- vertex degrees
- categorical variables



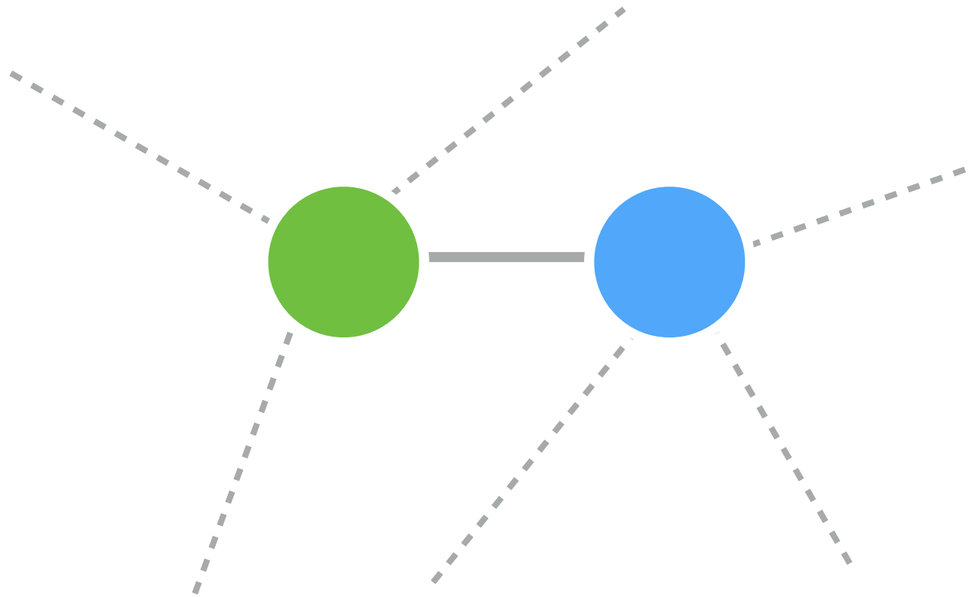
Newman, Phys. Rev. E 67, 026126 (2003).

Homophily & assortative mixing

like links with *like*

We write the correlation of categories across edges this way, and call it *Q.*

Principle: How many more edges are there between nodes with the same label, than we'd expect at random?



$$\sum_{ij} \left(\begin{array}{c}\text{\# actual edges} \\ \text{i}\leftrightarrow\text{j}\end{array} - \begin{array}{c}\text{\# edges if there were} \\ \text{no correlations} \\ \text{i}\leftrightarrow\text{j}\end{array} \right)_{\text{only if i,j have the same label}}$$

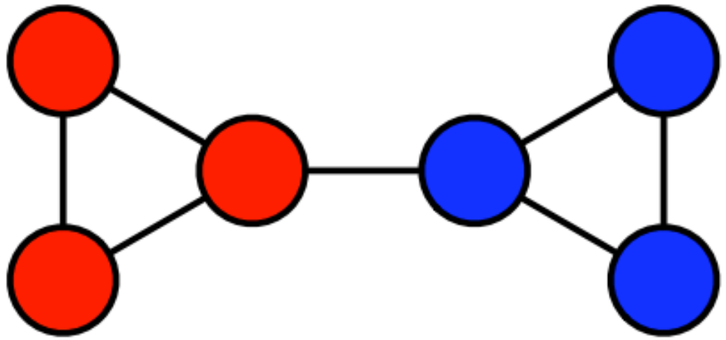
$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta_{b_i, b_j}$$

Practice makes the master

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta_{b_i, b_j}$$

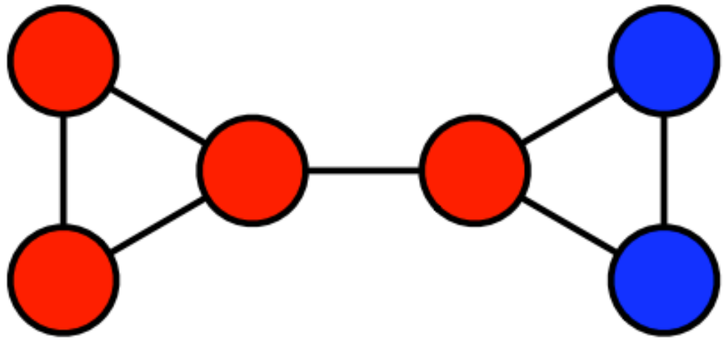
$$Q = \sum_r e_{rr} - a_r^2$$

← Equivalent form:
e_{rs} is the fraction of edges connecting labels r and s



labeling 1	red	blue
red	3/7	1/14
blue	1/14	3/7

$Q_1 = 5/14 = 0.357$



labeling 2	red	blue
red	4/7	2/14
blue	2/14	1/7

$Q = 6/49 = 0.122$

Modularity

Modularity is easily *the* most popular method for community detection. But why?

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta_{b_i, b_j}$$

Why is this more powerful than simply a measure of correlation over node labels?

Community structure in social and biological networks

M **Girvan**, MEJ **Newman** - Proceedings of the national …, 2002 - National Acad Sciences

A number of recent studies have focused on the statistical properties of networked systems such as social networks and the Worldwide Web. Researchers have concentrated particularly on a few properties that seem to be common to many networks: the small-world property, power-law degree distributions, and network transitivity. In this article, we highlight another property that is found in many networks, the property of community structure, in which network nodes are joined together in tightly knit groups, between which there are only …

☆ 99 Cited by 12341  ted articles All 40 versions

Finding and **evaluating** community structure in networks

MEJ **Newman**, M **Girvan** - Physical review E, 2004 - APS

We propose and study a set of algorithms for discovering community structure in networks—natural divisions of network nodes into densely connected subgroups. Our algorithms all share two definitive features: first, they involve iterative removal of edges from the network to split it into communities, the edges removed being identified using any one of a number of possible “betweenness” measures, and second, these measures are, crucially, recalculated after each removal. We also propose a measure for the strength of the community structure …

☆ 99 Cited by 11238  ted articles All 38 versions

Girvan & Newman, 2002. Community structure in social and biological networks. PNAS 99, 2002.

Key: let's reverse our thinking of what Q does

Don't use Q to compute correlation of some given labels.

Instead, **experiment with the labels** and see how you can **maximize Q**!

Now, we have a computer science problem:
how do you search the space of partitions?

(This space is really big!)

How would *you* do it? 🧐

People like modularity. Why?

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta_{b_i, b_j}$$

- Intuitive
- Works for weighted and unweighted networks.
- Corresponds to our social network ideas of what (cohesive) communities are.
  - Automatically choose *k,* the number of groups.
  - Rapid approximate solutions.
  - Follows the usual methods trajectory: idea, demonstration, optimization.
  - Fun customizations:
    - Resolution parameter to “zoom in” and “zoom out.”
    - Find the clusters. Then cluster the clusters. Then cluster those clusters…
    - Directed. Bipartite.

$$Q = \frac{1}{m} \sum_{ij} \left(A_{ij} - \frac{k_i^{\text{out}} k_j^{\text{in}}}{m} \right) \delta_{b_i, b_j}$$

modularity for directed networks

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \gamma \frac{k_i k_j}{2m} \right) \delta_{b_i, b_j}$$

modularity with a resolution parameter

Why aren’t we done here?

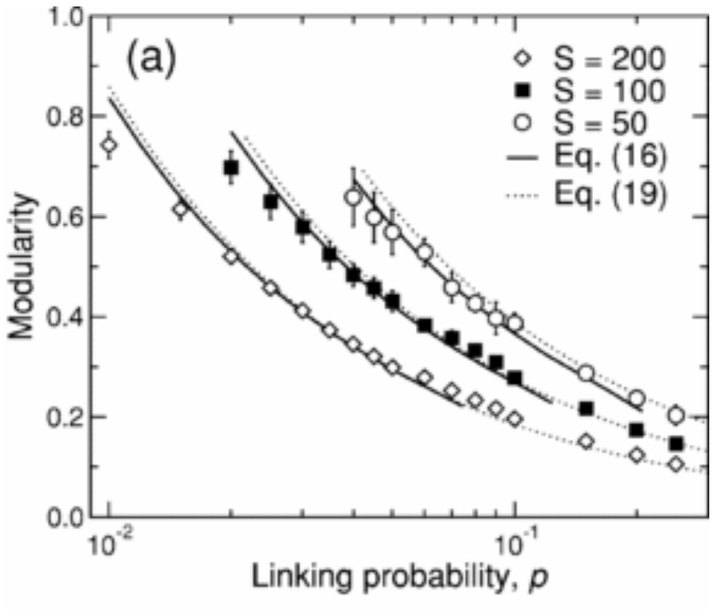
> Physicists like to minimize things because rocks fall.
> — Cris Moore

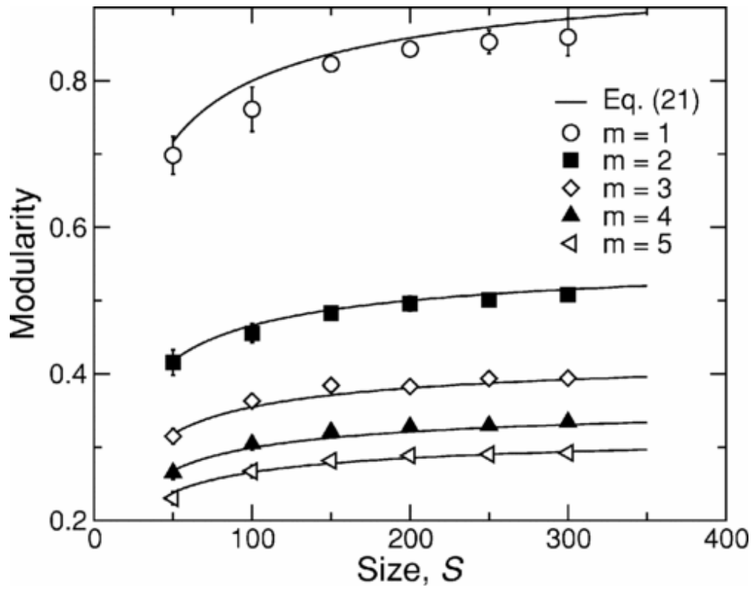
We can always maximize Q to find a partition, but is it meaningful?

Fooled by “structure” in totally random networks

As it turns out, you can find high-modularity partitions in random networks.

Structure is what makes data different from noise.
It’s what makes a network different from a random graph.



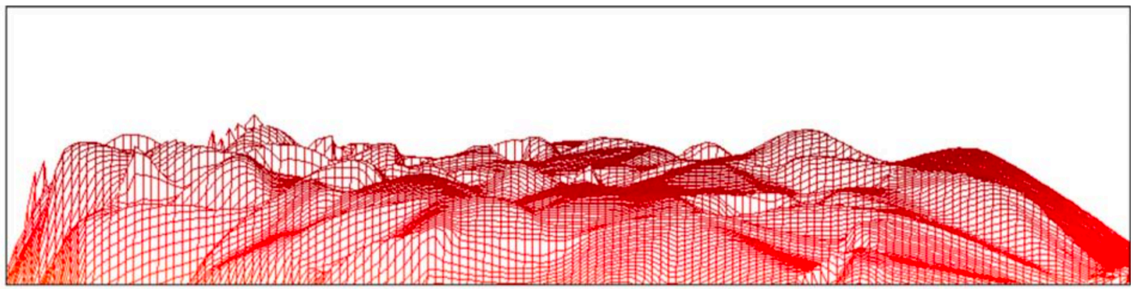


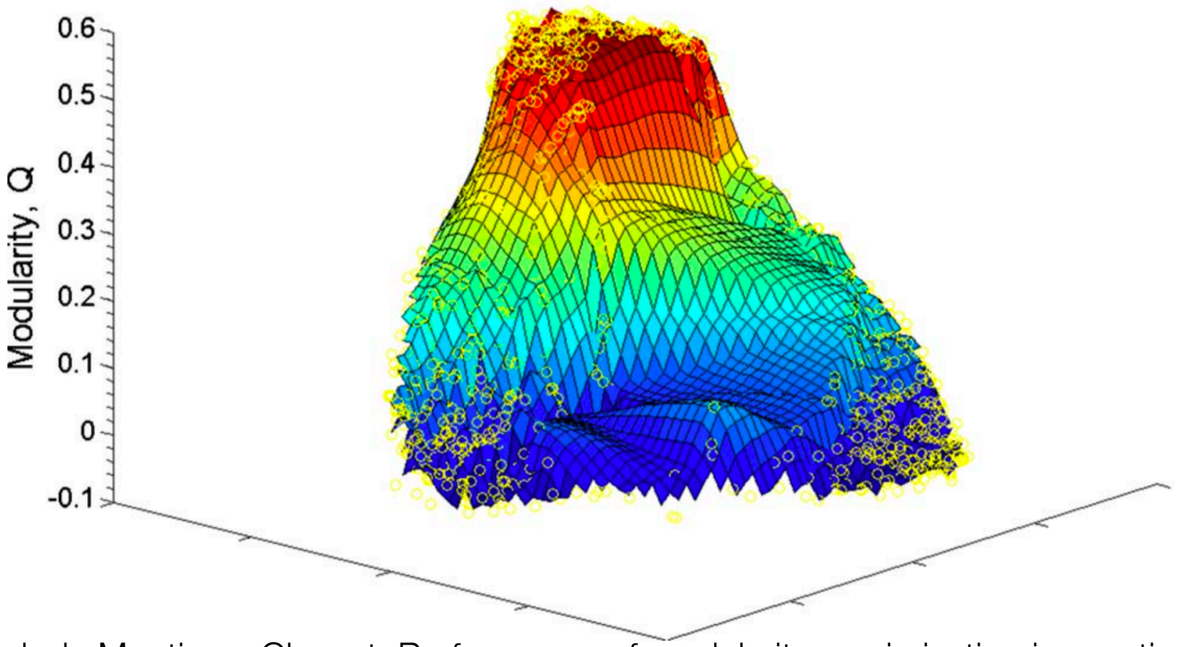
We prefer that our methods fail gracefully, and tell us when they fail. (like R^2)
[alternative perspective: maybe you *want* to find clusters in randomness?]

Guimera, Sales-Pardo, Amaral. "Modularity from fluctuations in random graphs and complex networks." Physical Review E 70.2 (2004): 025101.

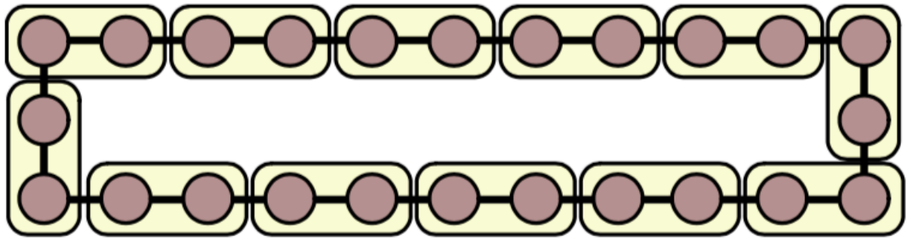
Modularity: degeneracy and strange behavior

Lots of different but nearly-as-good partitions.
The optimization landscape is *degenerate*.





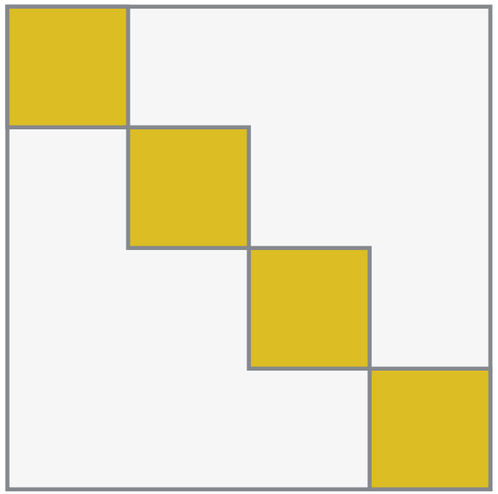
Unintuitive behavior: find the communities in a chain of cliques and you get pairs of cliques…not the cliques themselves!



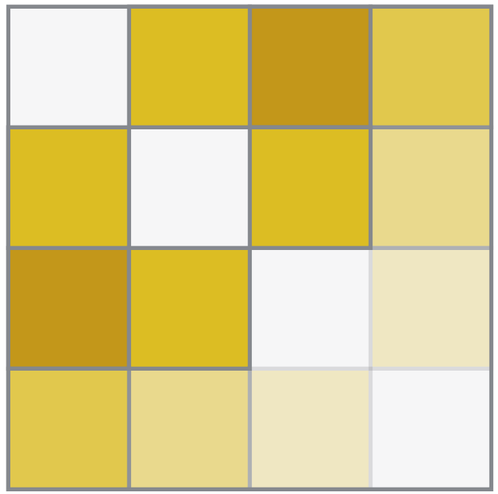
Good, de Montjoye, Clauset. Performance of modularity maximization in practical contexts. Phys. Rev. E 81, 046106 (2010).

Q is restricted to *assortative* community structure

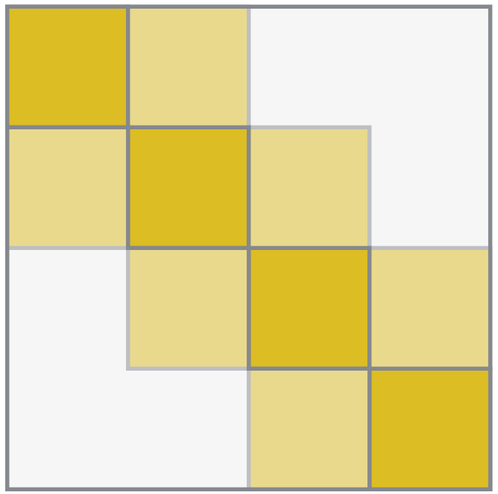
The zoo of possible structures is diverse and interesting!
Build intuition: what do these networks look like?



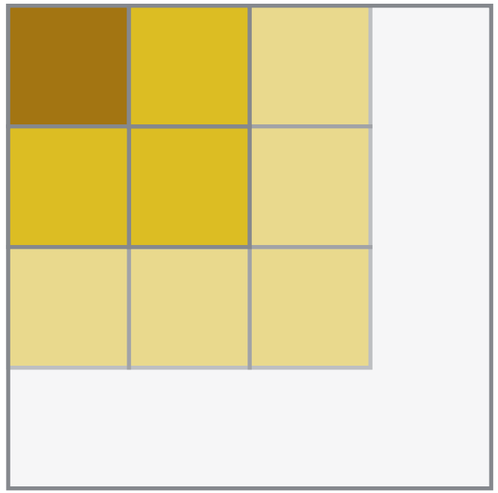
Assortative



Disassortative



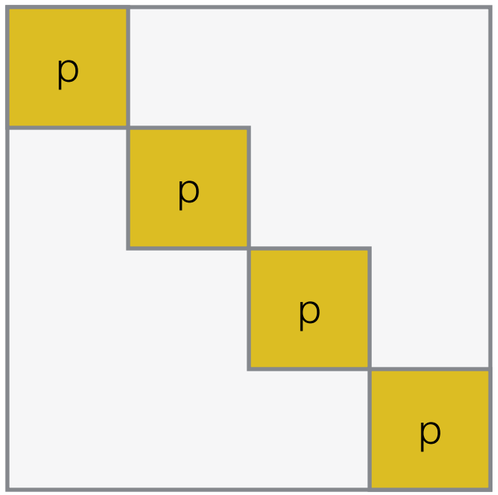
Ordered



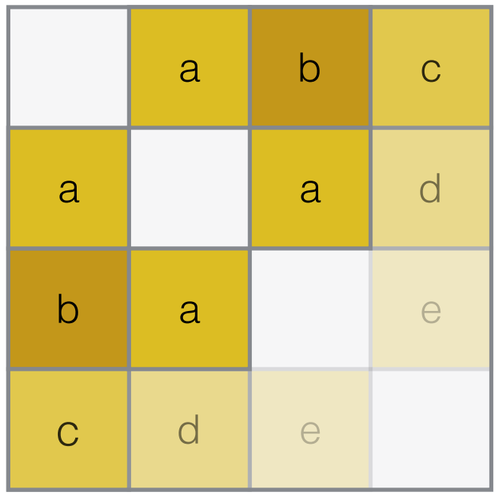
Core-periphery

Beyond assortativity: block models

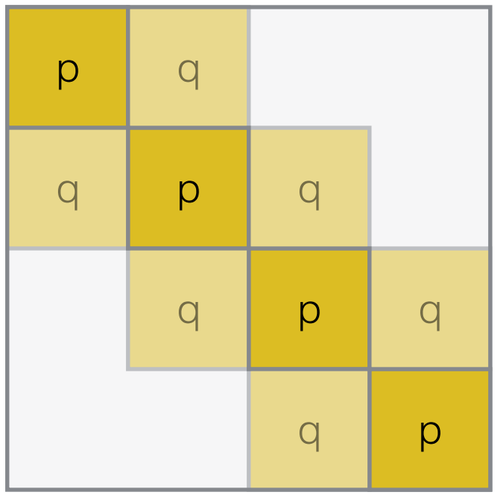
What do these have in common?



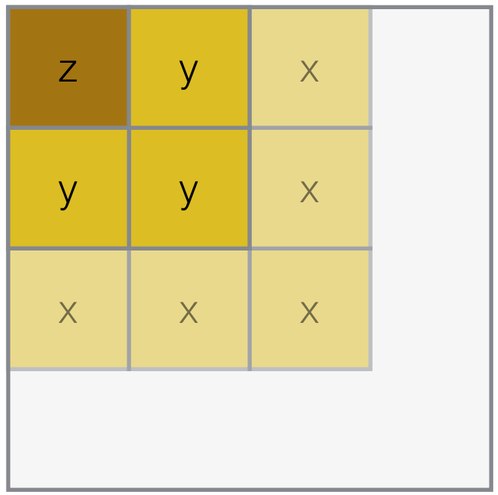
Assortative



Disassortative



Ordered



Core-periphery

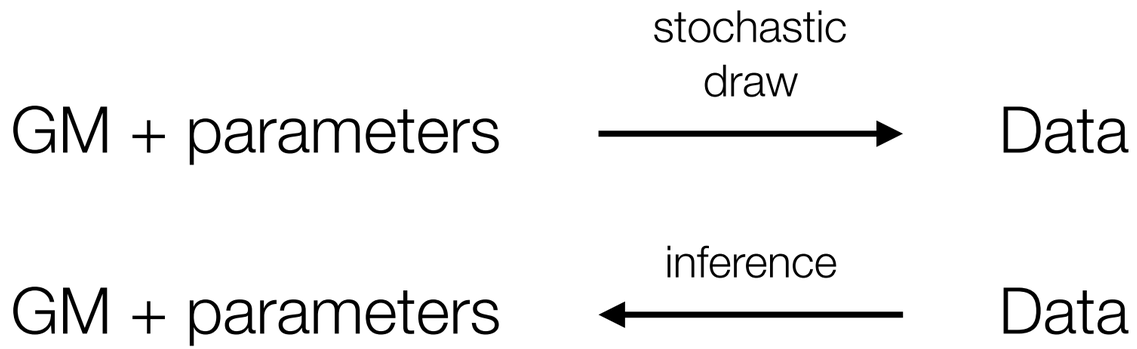
Nodes are in groups with other nodes that *connect to other groups in similar ways.*

Key idea: all nodes in a group are *stochastically equivalent*.

Generative model approach

Generate the structure you wish to infer.

We like generative models because they open the door to inference:



In other words: let's write down a recipe for generating block structure. 👨🏾‍🍳 👩🏻‍🍳

The stochastic block model

GM + parameters —stochastic draw→ Data

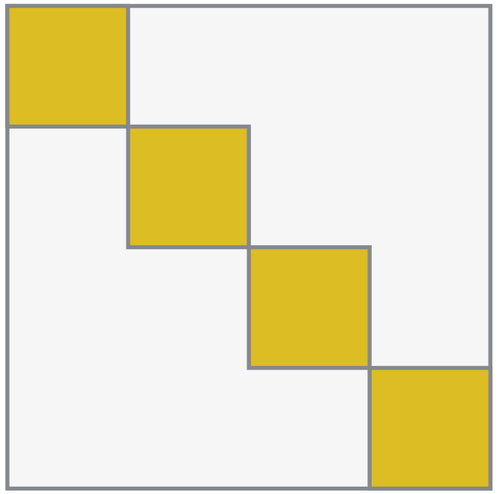
Assign each node to one of *B* blocks.

b_i

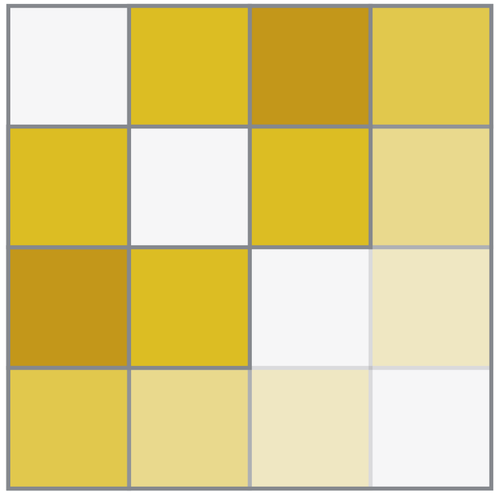
Let the probability that two nodes connect depend *only* on their blocks:

$$Pr(A_{ij}|b_i, b_j) = \omega_{b_i,b_j}$$

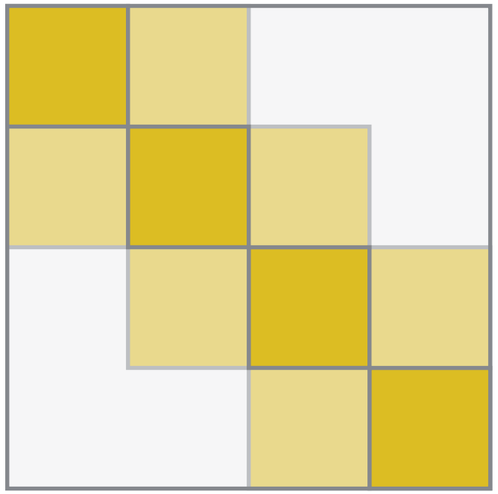
Then we can choose the matrix ω to have whatever structure we want!



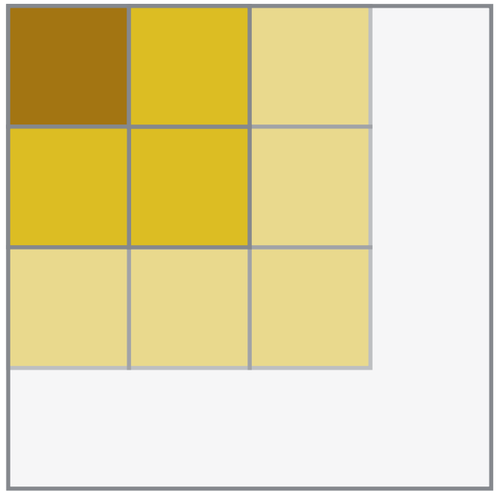
Assortative



Disassortative



Ordered



Core-periphery

SBM inference

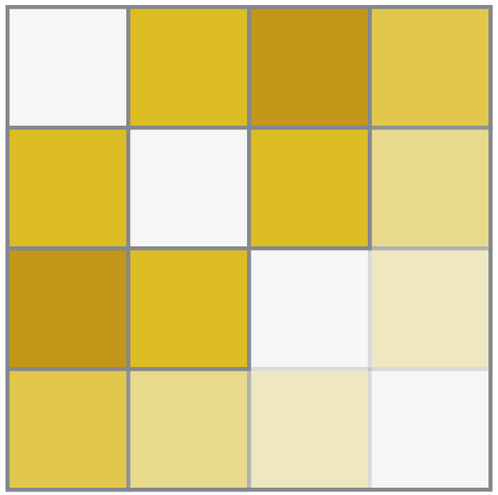
GM + parameters ← (inference) Data

no more math on slides 😭

but the derivations are beautifully described in:

Karrer, Newman. Stochastic blockmodels and community structure in networks. Phys. Rev. E 83, 016107 (2011).

Recommended reading!



example matrix of parameters, *B=4*

Summary:

1. Write down the SBM *likelihood function* for a fixed number of blocks *B*.
2. Maximize the likelihood with respect to matrix parameters.
3. Search over divisions into *B* blocks to find the best blocks.

There's a subtlety here, which I haven't written out, called *degree correction*. In practice, we also take into account the exact degree sequence. This allows us to find community structure while controlling for variability in the nodes' degrees.

The problem with parameterized models…

You have to choose their parameters!

How should we choose *B*, the number of blocks?

Hint: we can't simply maximize the likelihood over all choices of *B*:

Why? If we place each node in its own community, we can get Likelihood=1.

[Actually, this wouldn't model the data at all: it would *memorize* it.]

We need a way to penalize the complexity of the model. Any ideas?

Description length & Occam's razor

The Description Length of a message is:
bits required to send the compressed message + # bits in encoding scheme.

Occam's razor: among all possible explanation for a phenomenon, choose the simplest one. Therefore, choose the model with Minimum Description Length (MDL).

The stochastic block model also has a Description Length:

$$\Sigma = \mathcal{S} + \mathcal{L}$$

description length = entropy of data, given the model (fit SBM) + entropy of model

Consider the original problem:
what happens to this equation when I increase the number of blocks B?

Peixoto. Entropy of stochastic blockmodel ensembles. Phys. Rev. E 85, 056122 (2012).
Peixoto. Parsimonious Module Inference in Large Networks. Phys. Rev. Lett. 110, 148701 (2013).

MDL criterion suggests an algorithm:

Fit the SBM with 1 block and record the Description Length.
Fit the SBM with 2 blocks and record the Description Length.
…
when the Description Length starts to increase, go back one step and stop.*

Bonus: what happens if I *try to trick you* and give you a *random network with no blocks*?

MDL approach will tell you: your network is a random network with *one* block.

*Actually, use something clever, like Golden Ratio / Fibonacci search
Press et al. Numerical Recipes: The Art of Scientific Computing, (Cambridge University Press, Cambridge, England, 2007), 3rd ed.

So how does the search part work?

Markov-chain Monte Carlo:

Wander from one partition to another partition by proposing to take a node from one group and move it to a new group.

If this move *increases* the likelihood score, then keep the move.
If this move *decreases* the likelihood score, then maybe keep it, depending on how bad it is.

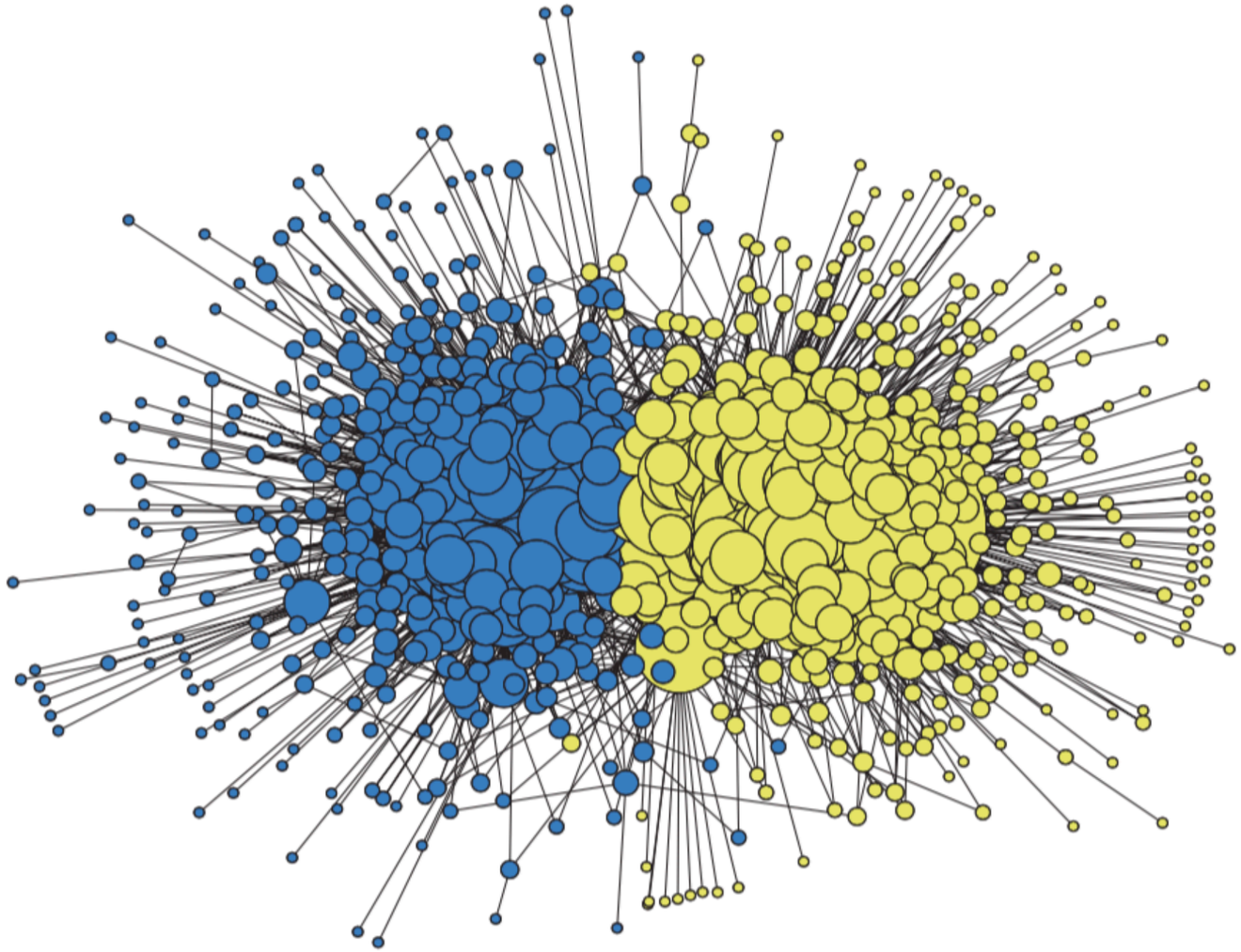
Thorough details in the documentation for graph-tool. https://graph-tool.skewed.de

Does it work?

Adamic & Glance mapped the link structure of USA political blogs in 2004.

Karrer & Newman used this network as a testbed for community detection using the SBM.

What does this say about the process that may be generating (or pruning?) the links in this network?



Division in political blog network. 👨🏽‍💻👩🏻‍💻

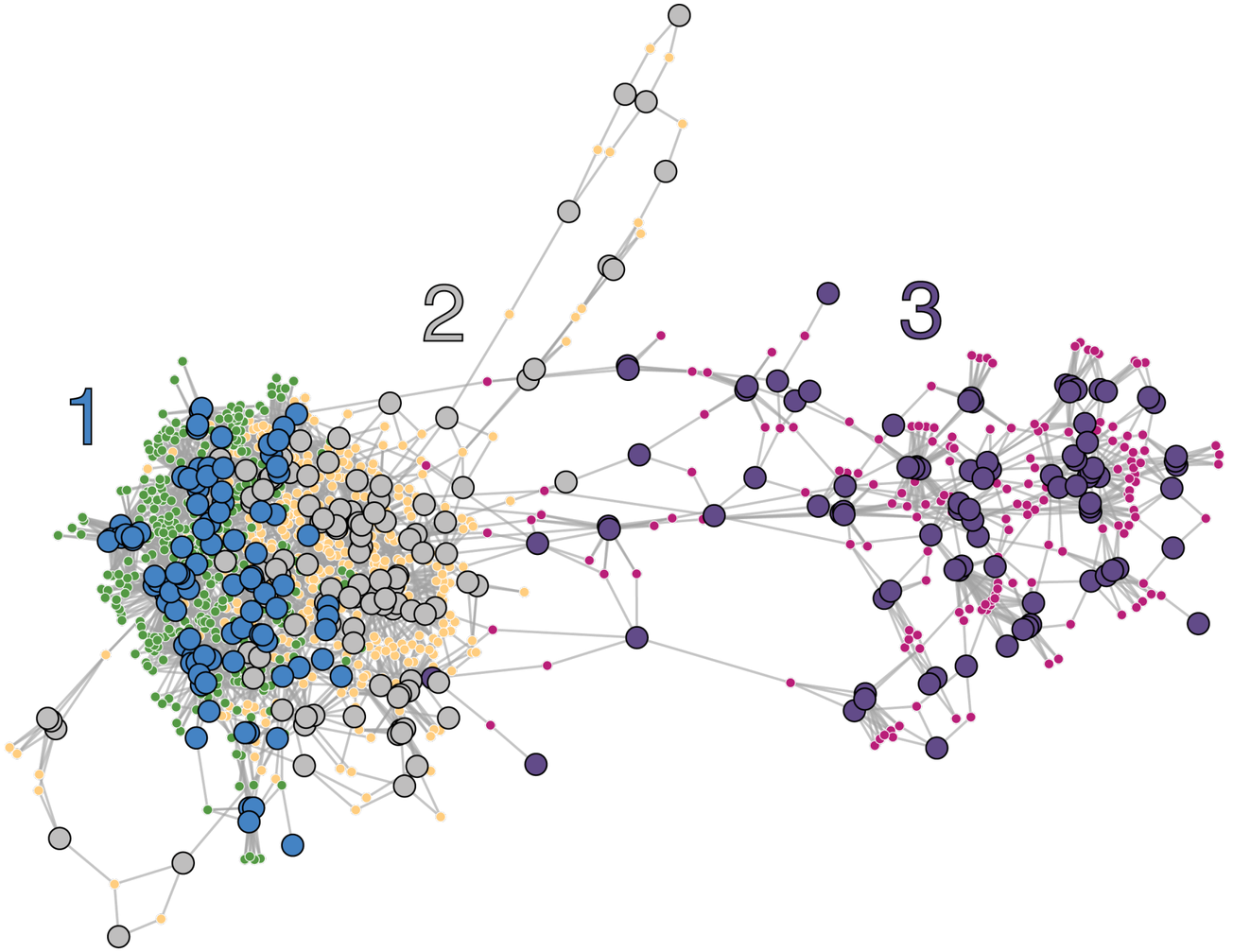
Karrer & Newman PRE 2011. https://arxiv.org/abs/1008.3926
Adamic & Glance KDD 2005. https://dl.acm.org/citation.cfm?id=1134277

Does it work?

In bipartite networks, we *know* the major split in the data already.

Methodologically, we found that exploiting this split improved speed and quality of the partitions we found.

Scientifically, this opened new directions to analyze (and understand) evolutionary constraints on malaria parasites.



Genes & substrings,
malaria immune evasion

Code by Tzu-Chi Yen https://github.com/junipertcy/det_k_bisbm.
Figure http://danlarremore.com/webweb/

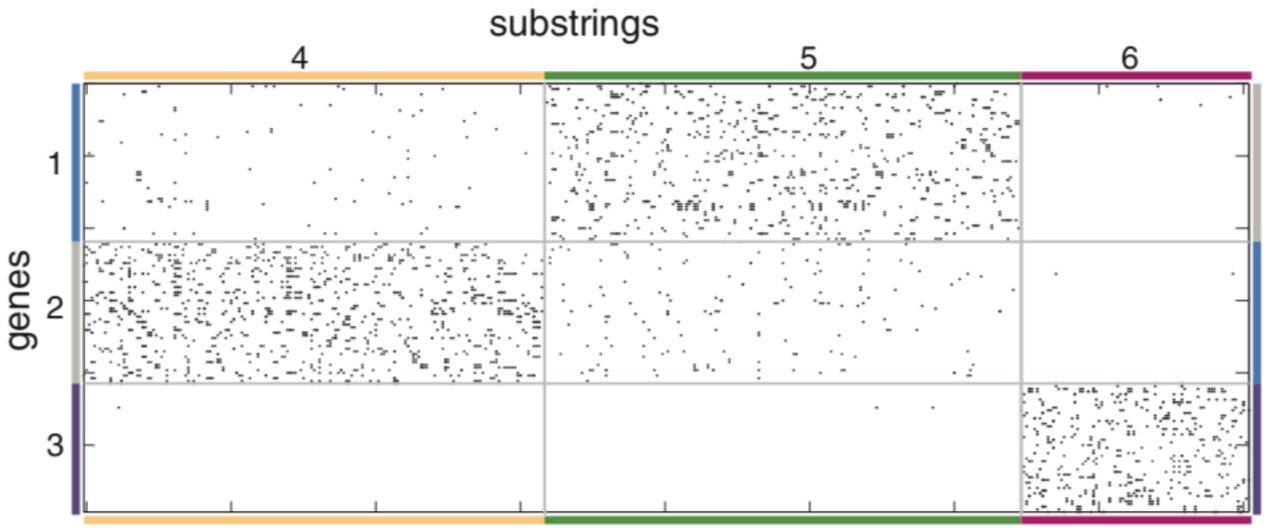
Larremore, Clauset, Buckee, *PLoS Comp Biol*, 2013.
Larremore, Clauset, Jacobs, *Physical Review E*, 2014.

Does it work?

In bipartite networks, we *know* the major split in the data already.

Methodologically, we found that exploiting this split improved speed and quality of the partitions we found.

Scientifically, this opened new directions to analyze (and understand) evolutionary constraints on malaria parasites.



Genes & substrings,
malaria immune evasion

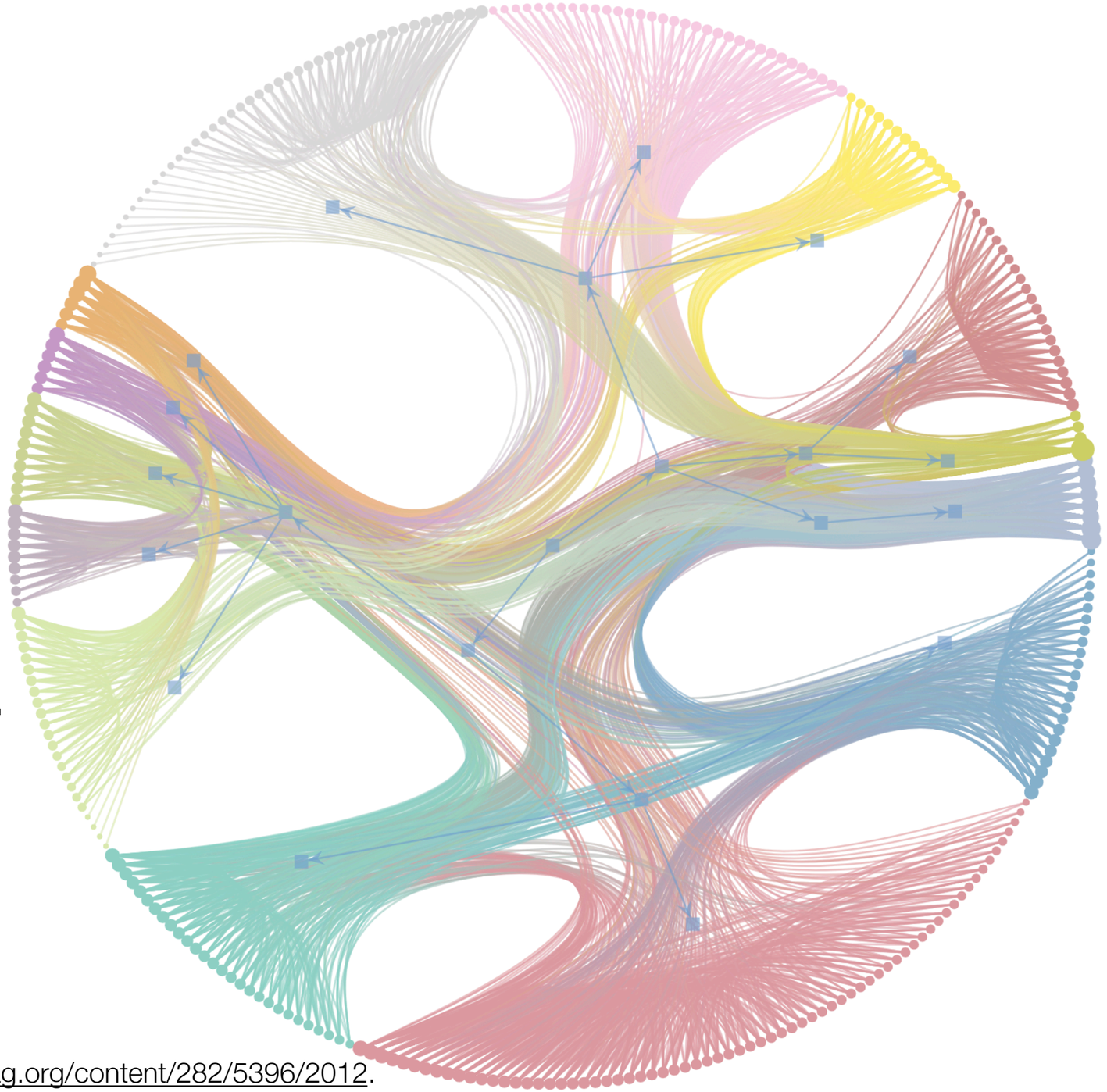
Larremore, Clauset, Buckee, *PLoS Comp Biol*, 2013.
Larremore, Clauset, Jacobs, *Physical Review E*, 2014.

Does it work?

C. elegans neuronal network.

297 neurons, completely mapped.
The neurons do not fire action potentials, and do not express any voltage-gated ion channels.

Note the different layout…



C. elegans 1st *multicellular* genome.1998. http://science.sciencemag.org/content/282/5396/2012.
Bacteriophage 1st genome: https://www.nature.com/articles/265687a0

A good alternative? cross-validation via link prediction

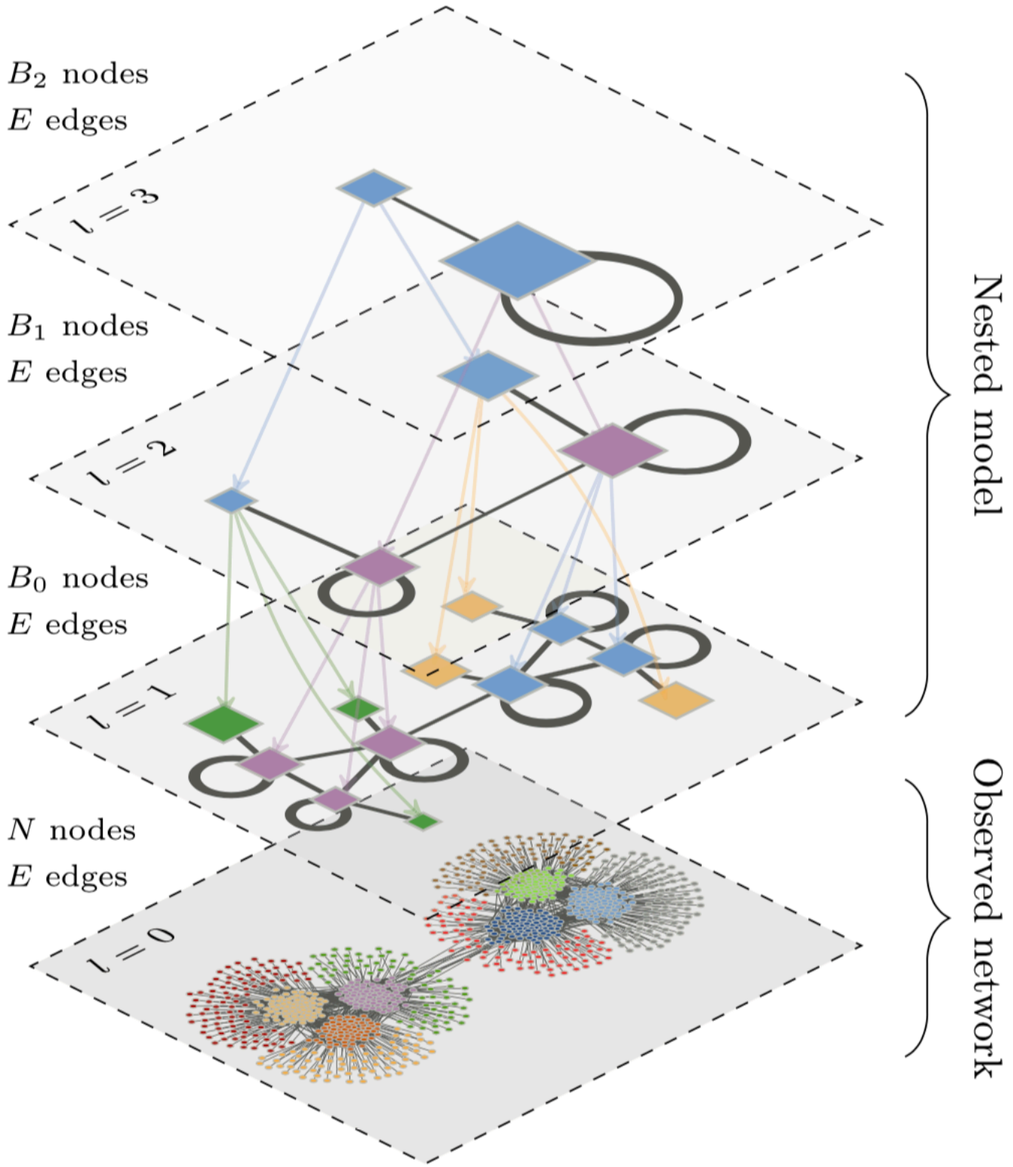
Select *B* by choosing the model that makes the best predictions.

Perform *k*-fold cross validation:

1. Divide the edges of the network into *k* groups, called folds.
2. Hide one of the folds (the “test set”)
3. Fit each SBM to the remaining k-1 folds (the “training set”), varying B.
4. Test the ability of the fitted models to predict the hidden test data.
5. Switch which fold is “test” and which are “training” and repeat.

Choose the B with the highest performance on link prediction over all k folds.

Advanced topic 1: hierarchical communities



Don’t minimize the description length of the data and the model…

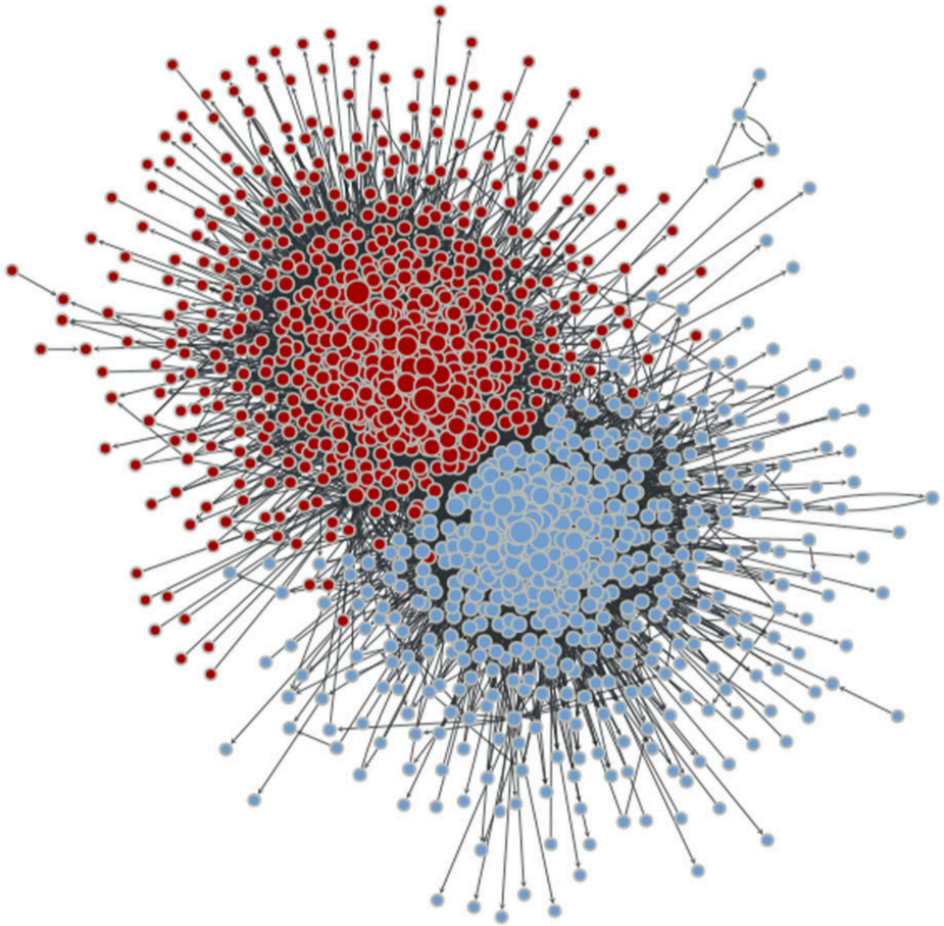
Model the model as well. Why?

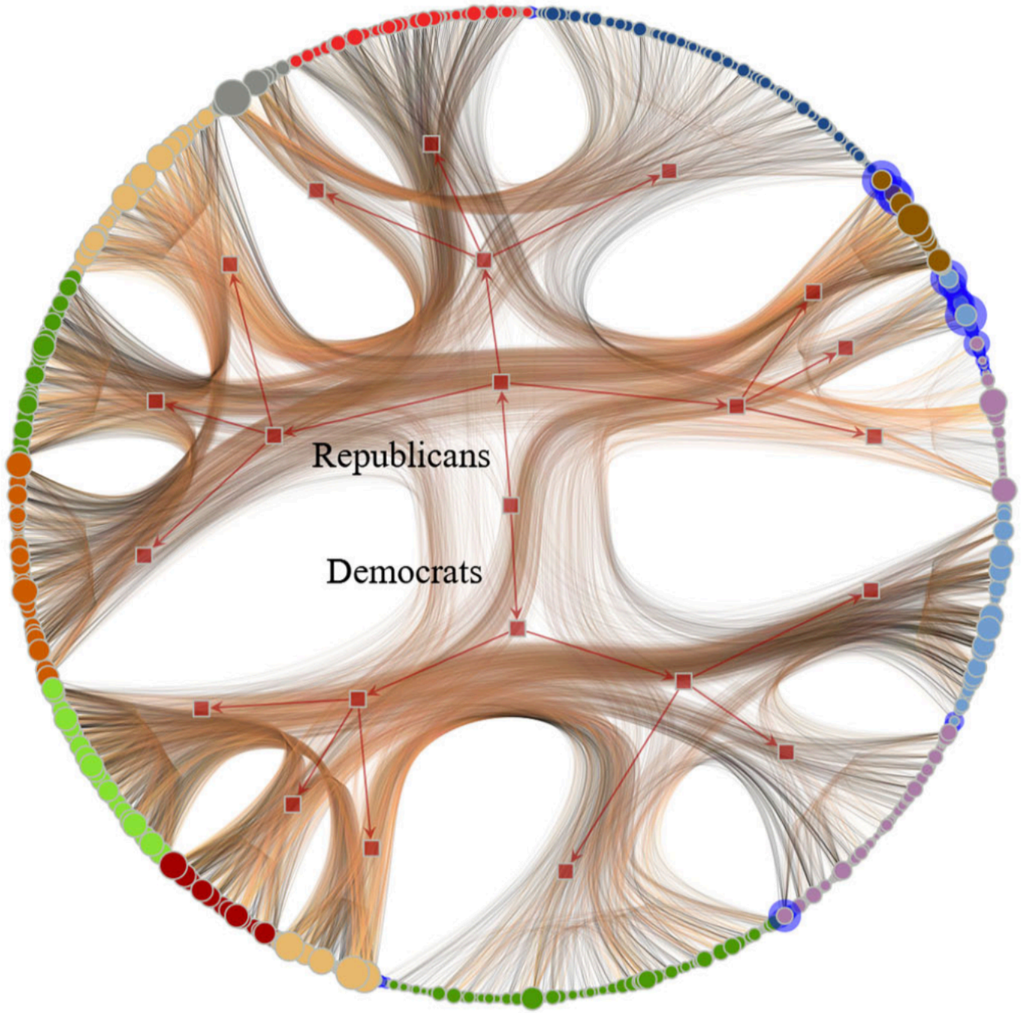
If we compress the model, we can afford a bigger model, but a lower overall cost.

Except now, the description length includes two models. Or three? Or?

Peixoto. Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. Phys. Rev. X 4, 011047 (2014).

Advanced topic 1: hierarchical communities



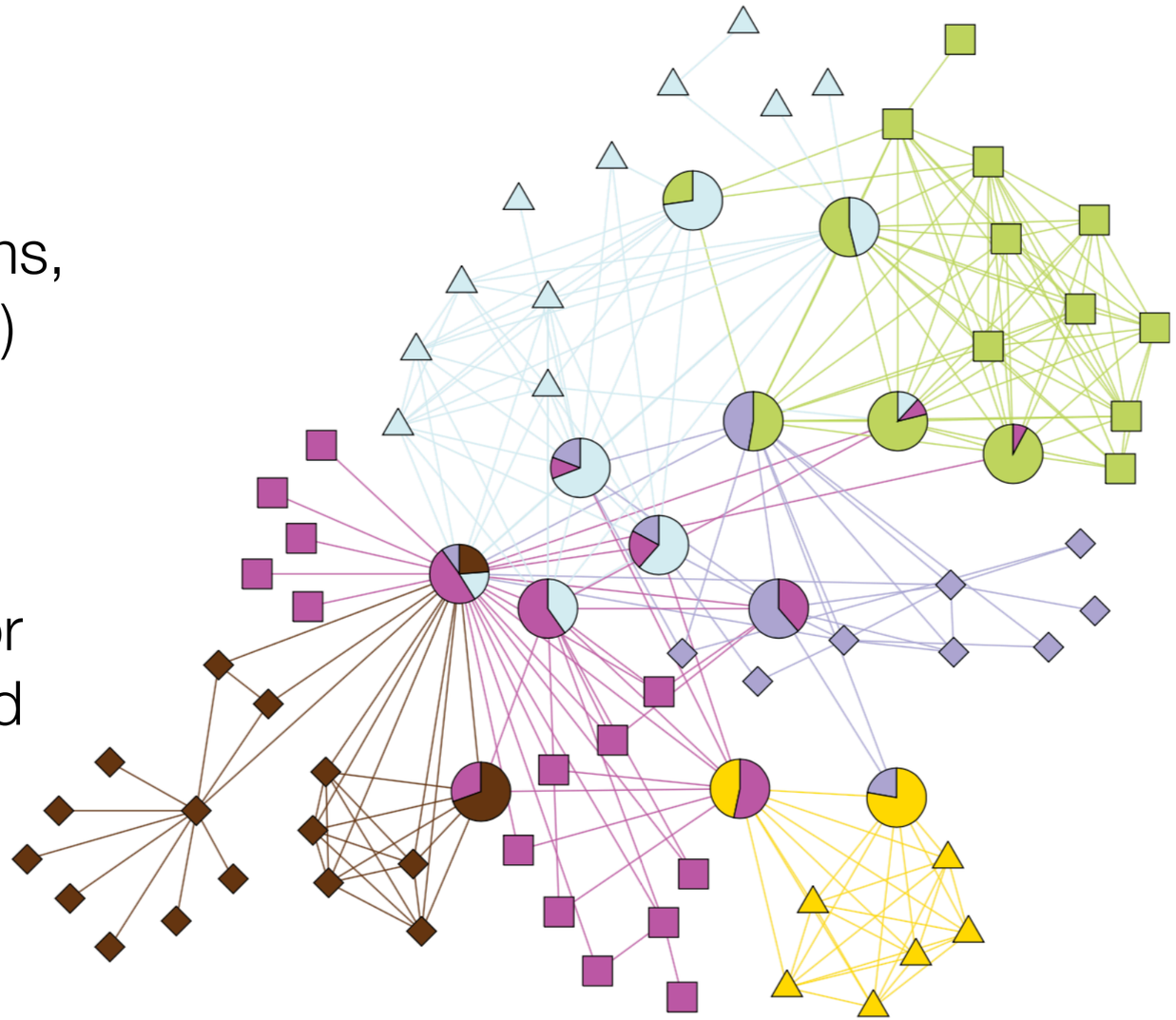


Peixoto. Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. Phys. Rev. X 4, 011047 (2014).

Advanced topic 2: mixed-membership

Nodes are often pulled between communities. (Or in real social systems, individuals belong to multiple groups.)

“Mixed membership” models allow for that, by assigning *links* to groups, and assigning nodes to groups based on their links.



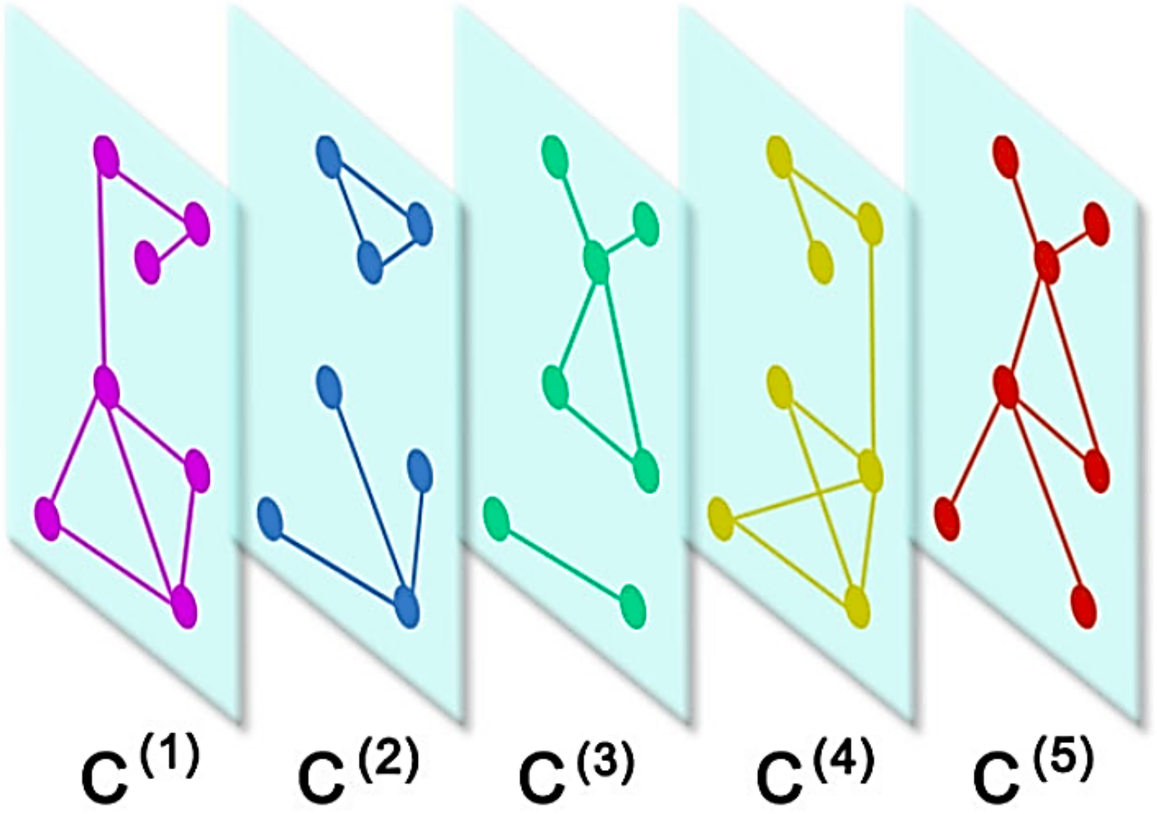
Ball, B., Karrer, B. & Newman, M. E. J. Efficient and principled method for detecting communities in networks. Phys. Rev. E 84, 036103 (2011).

Advanced Topic 3: multilayer networks

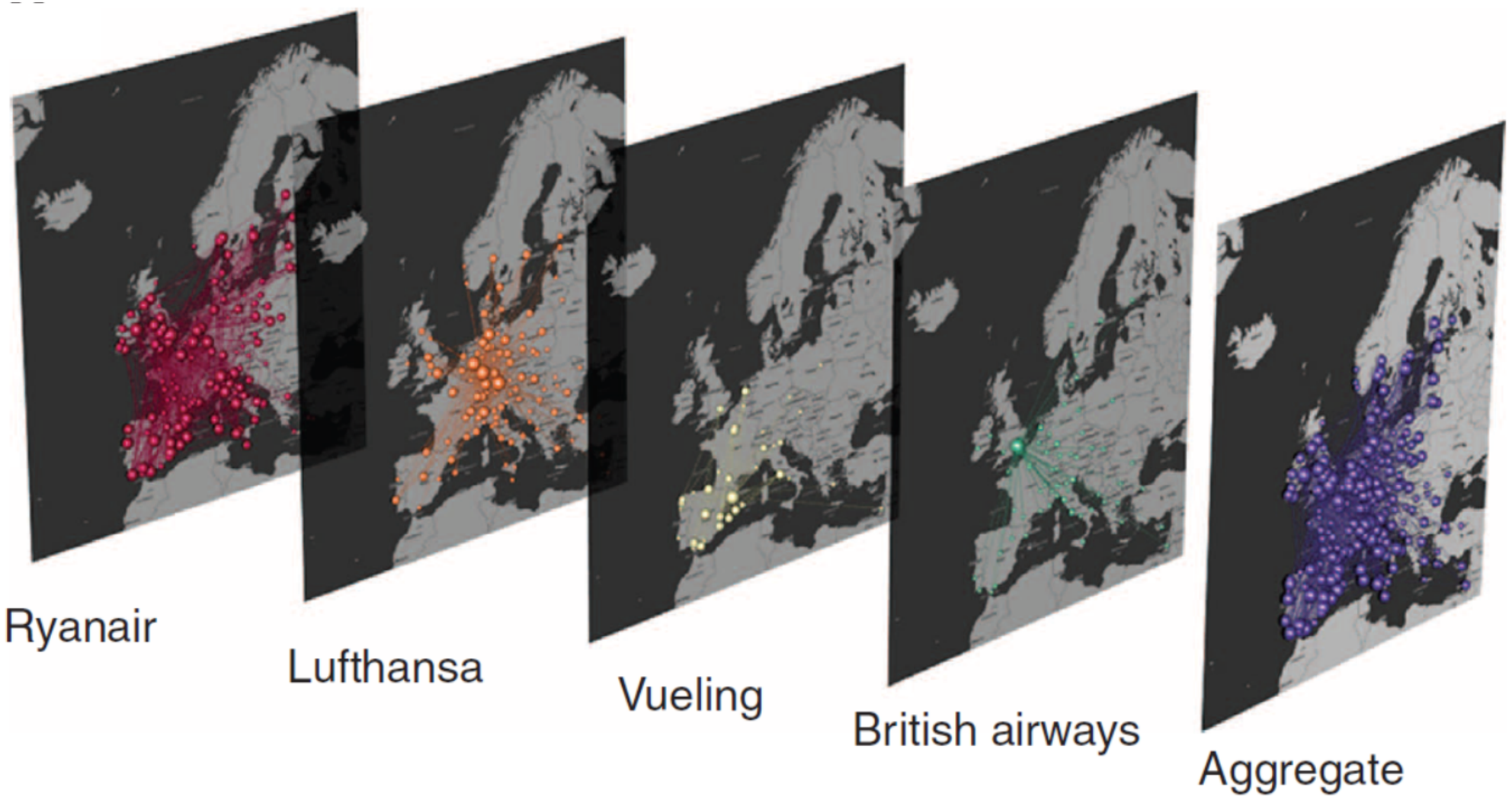
In single-layer networks:
nodes and edges

In multilayer networks:
nodes, edges, and layers

edges: different types of relationships
layers: each layer contains all edges of one type
nodes: same nodes in each layer



Multilayer network: air travel



traditional: booking with airline

disrupted: booking with kayak, expedia, etc

Multilayer network: community structure?

three key approaches:

1. **Non-generative**: modularity maximization; vary inter-layer strength.

   Mucha et al *Science* 2010. http://science.sciencemag.org/content/328/5980/876

2. **Generative**: SBM for each layer, but jointly model layers whenever their structures are sufficiently similar.

   Peixoto, T. P. Phys. Rev. E 92, 042807–15 (2015).

3. **Generative**: SBM for each layer, and model all layers simultaneously with same community structure, but allow relationships between groups to vary.

   De Bacco Power Larremore Moore. Phys. Rev. E 95, 1981–10 (2017).

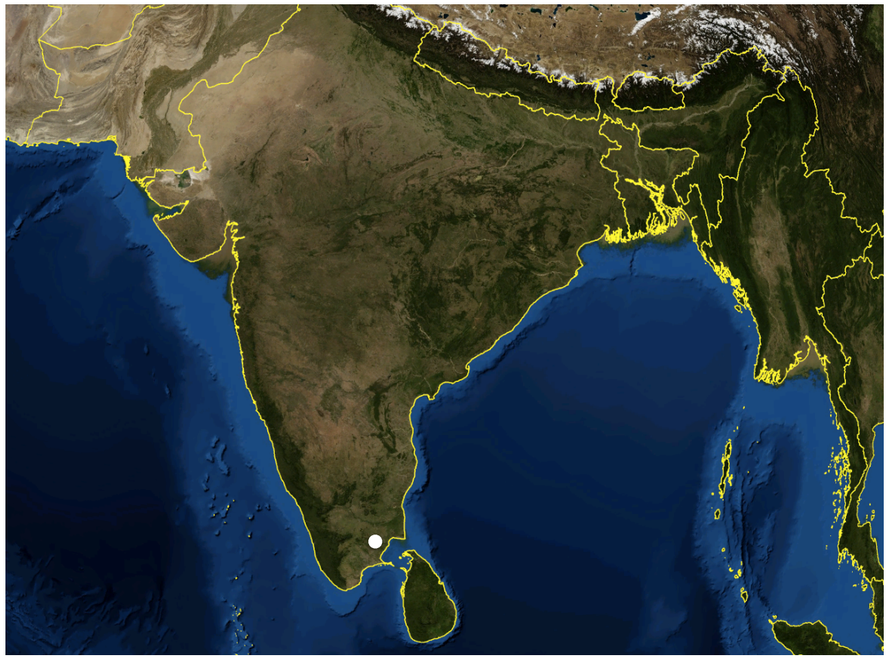
1 is preferred if nodes appear/disappear over time.
2,3 are preferred to solve the *layer interdependence problem*

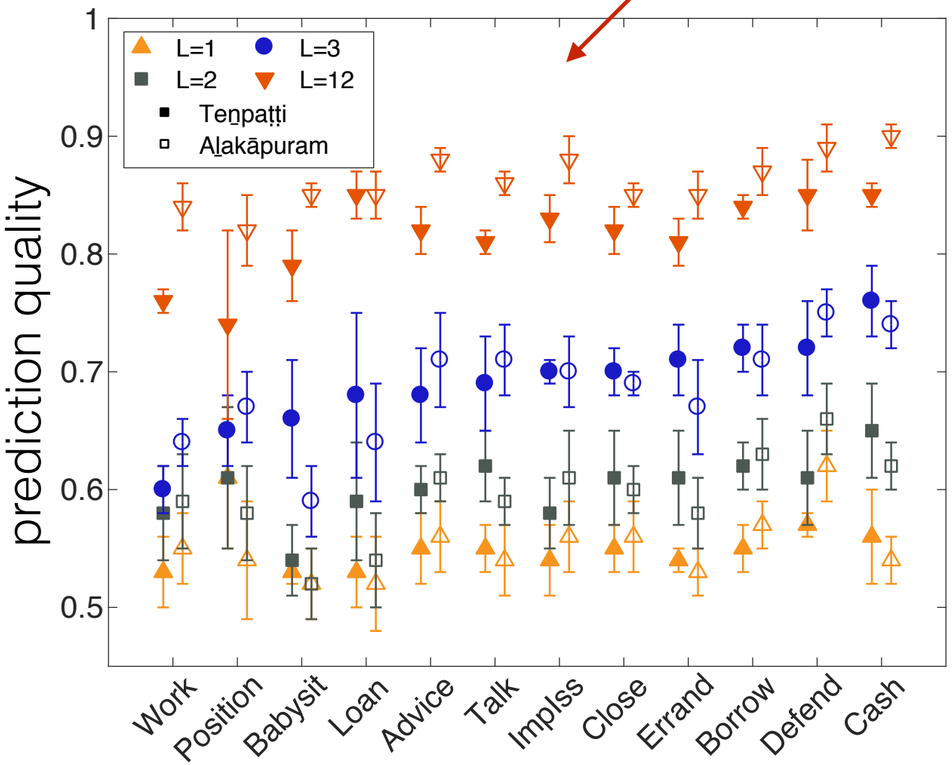
Layer interdependence

more layers = better performance
(layer structure generated by same social mech.)

Are layers structurally similar? Complementary? Neither?

“Learn” a SBM from *m* layers; try to predict links of *m+1*.



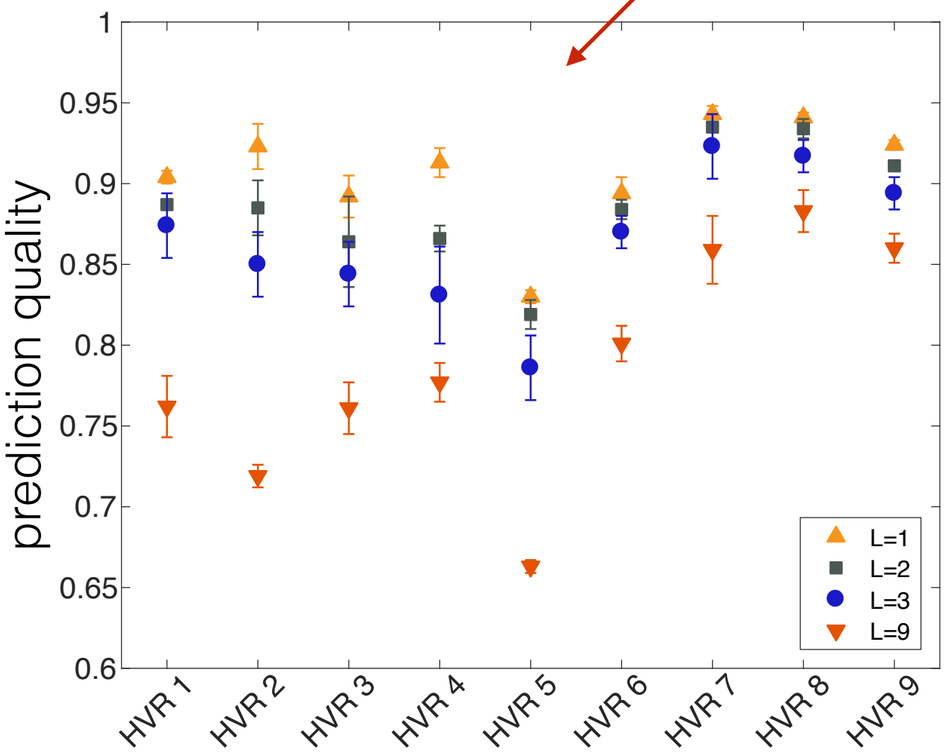


12 layer social support network across 2 villages in South India.

Caterina De Bacco, Eleanor A. Power, Daniel B. Larremore, and Cristopher Moore.
"Community detection, link prediction, and layer interdependence in multilayer networks” Physical Review E 95 042317

Layer interdependence - malaria

more layers = worse performance
(layer structure generated by different biol. mech.)



cannot predict the structure of one region in the immune-evasion genes by using other regions; layers are unrelated!

Caterina De Bacco, Eleanor A. Power, Daniel B. Larremore, and Cristopher Moore.
"Community detection, link prediction, and layer interdependence in multilayer networks" Physical Review E 95 042317

Advanced topic 4: metadata+communities

What are metadata?

How well do metadata explain the network structure? “BESTest”

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

How do metadata relate to network structure? “neoSBM”

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

Can we use metadata to guide community detection? “metadata assisted SBM”

Newman, Clauset. *Nature communications* 7 (2016). https://www.nature.com/ncomms/2016/160616/ncomms11863/full/ncomms11863.html

Can we find patterns in the metadata itself? Apply multilayer SBM

Peixoto. Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. Phys. Rev. X 4, 011047 (2014).

Blockmodel entropy significance test

How well do the metadata explain the network?

randomly assigned metadata
→ model gives no explanation, high H

metadata correlated with communities
→ model gives good explanation, low H

1. Divide the network G into groups according to metadata labels M.

2. Fit the maximum likelihood parameters of an *a posteriori* SBM and compute the entropy $H(G,M)$ of the corresponding ensemble.

3. Compare the entropy of this SBM ensemble to distribution of entropies from SBMs partitioned using shuffled metadata $\underline{M}$.

p-value = $\Pr[H(G,\{\underline{M}\}) \leq H(G,M)]$

https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

Multiple network layers; multiple metadata attributes

Network	Status	Gender	Office	Practice	Law School
Friendship	$< 10^{-6}$	0.034	$< 10^{-6}$	0.033	0.134
Cowork	$< 10^{-3}$	0.094	$< 10^{-6}$	$< 10^{-6}$	0.922
Advice	$< 10^{-6}$	0.010	$< 10^{-6}$	$< 10^{-6}$	0.205

model = SBM

Multiple sets of metadata **significantly explain** multiple networks.
[Should one particular set of metadata be ground truth?]

https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

BESTest accommodates many models of group structure

Network	Model	
	SBM	DCSBM
Malaria 1	0.566	0.066
Malaria 2	0.064	0.126
Malaria 3	0.536	0.415
Malaria 4	0.588	0.570
Malaria 5	0.382	0.097
Malaria 6	0.275	0.817
Malaria 7	0.020	0.437
Malaria 8	0.464	0.143
Malaria 9	0.115	0.104

metadata = parasite origin

A negative result: parasite origin is irrelevant to genetic substring-sharing.

Malaria parasites *do not* have a strong strain structure, with implications for **diversifying selection** among parasites.

https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

DBL, Clauset, A. & Buckee, C. O. A Network Approach to Analyzing Highly Recombinant Malaria Parasite Genes. PLoS Comp Bio 9, e1003268 (2013).

neoSBM

Choose between the SBM partition and the metadata partition.

$$\underset{\text{neoSBM log likelihood}}{\mathcal{L}_{\text{neoSBM}}} = \underset{\text{SBM log likelihood}}{\mathcal{L}_{\text{SBM}}} + \underset{\text{cost}}{f(\theta)}$$

Log likelihood with parameterized prior:

θ is the parameter of a Bernoulli prior on whether the node is free to choose its own community or held fixed at its metadata label.

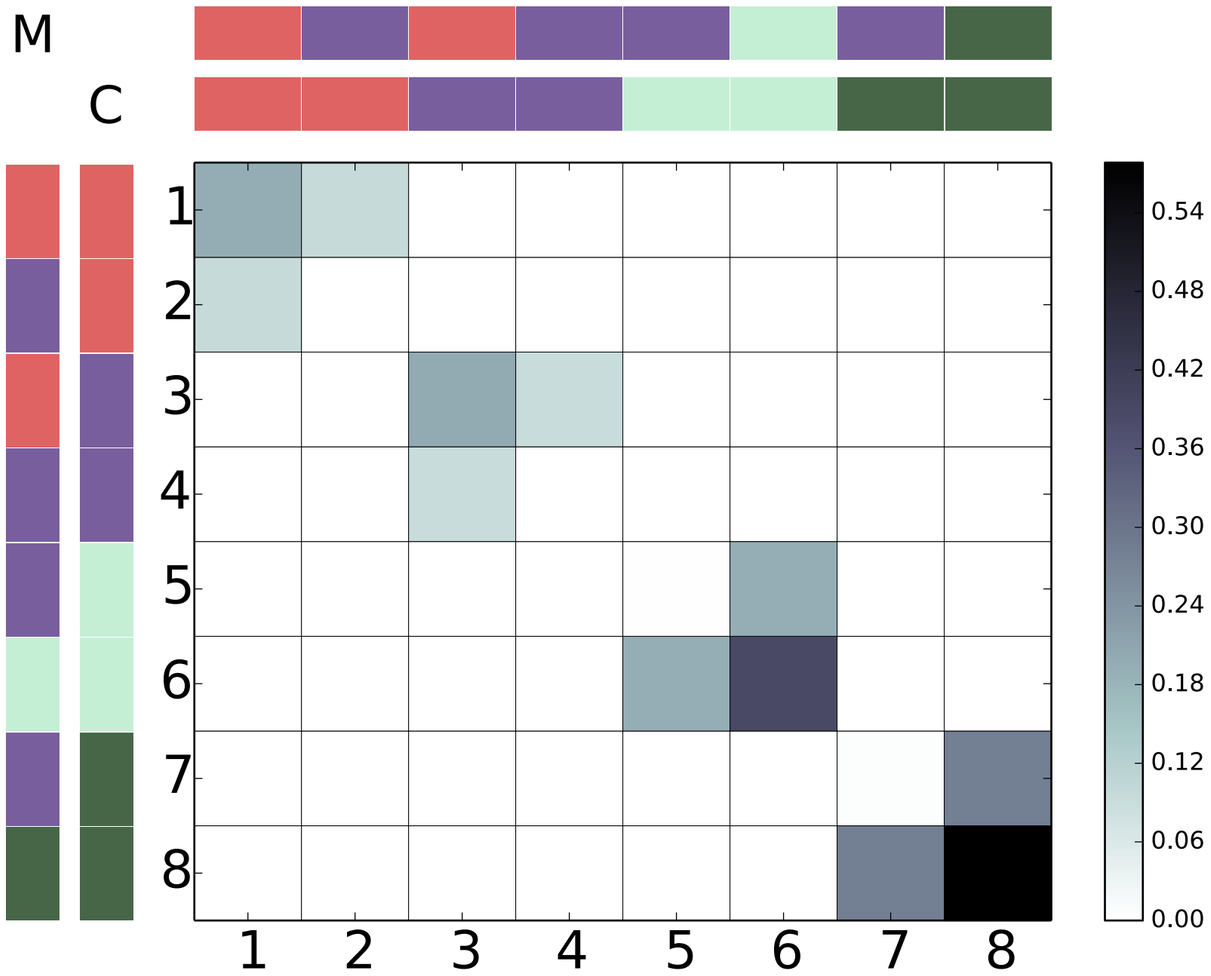
As θ increases, the cost of freeing a node decreases.

Varying θ in the unit interval **explores the space of partitions** between M and C.

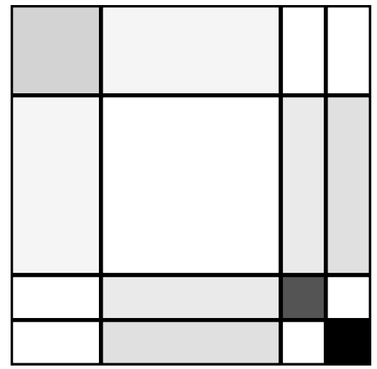
https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

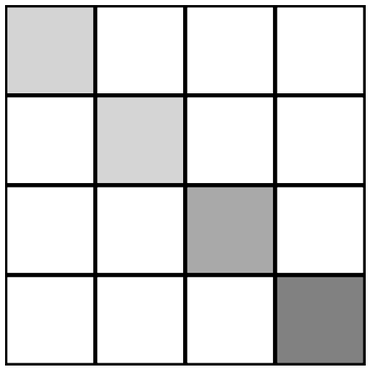
Plant two different *kinds* of structure in a network



SBM with 8 groups and
two interesting 4-group partitions:



i. core-periphery

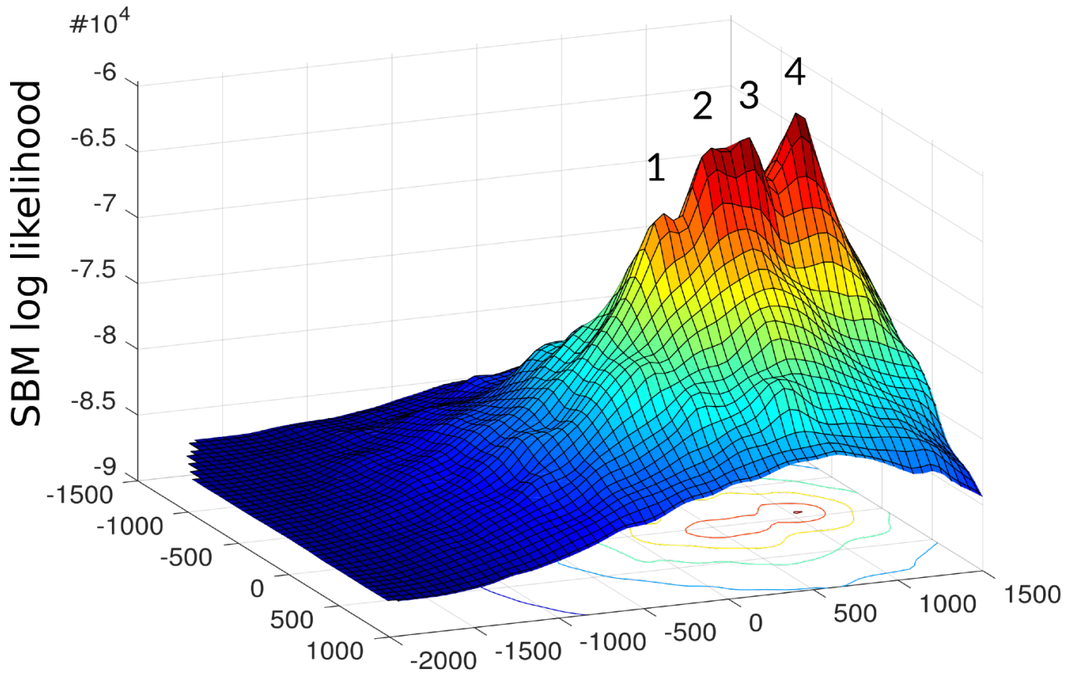


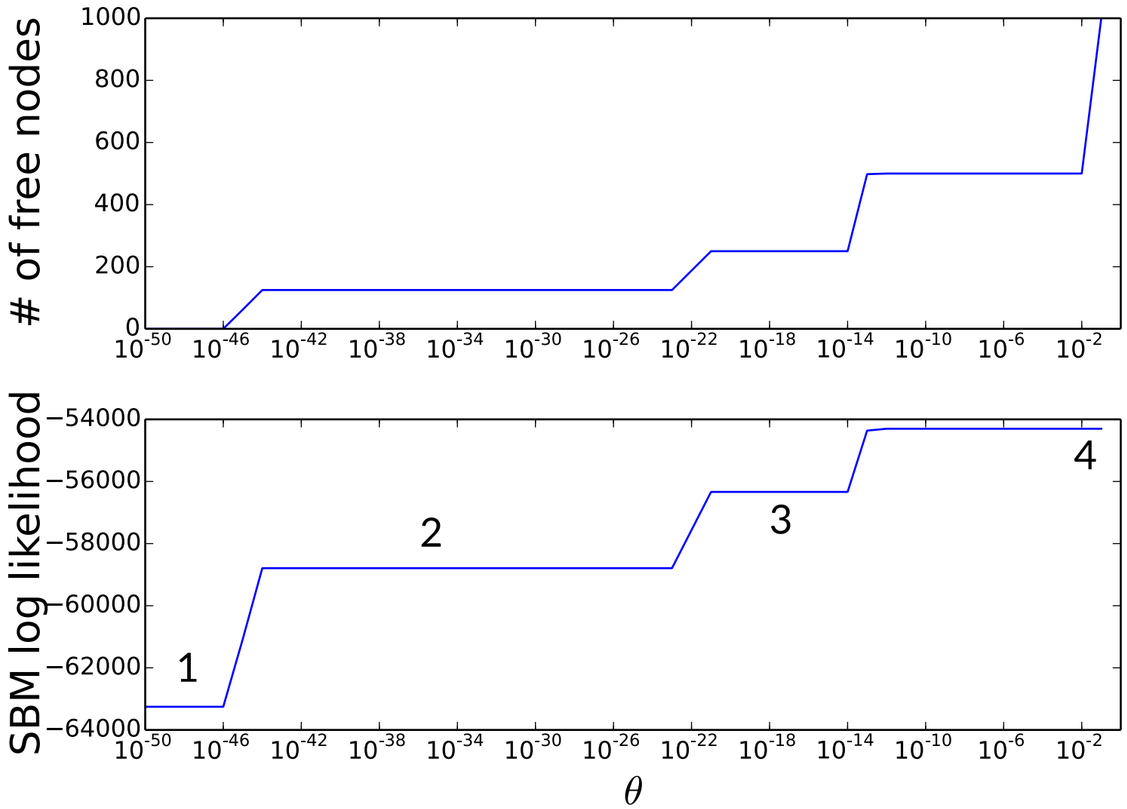
ii. assortative

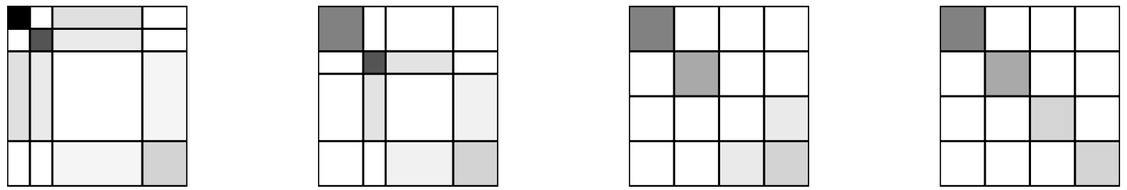
https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

The neoSBM identifies four interesting partitions





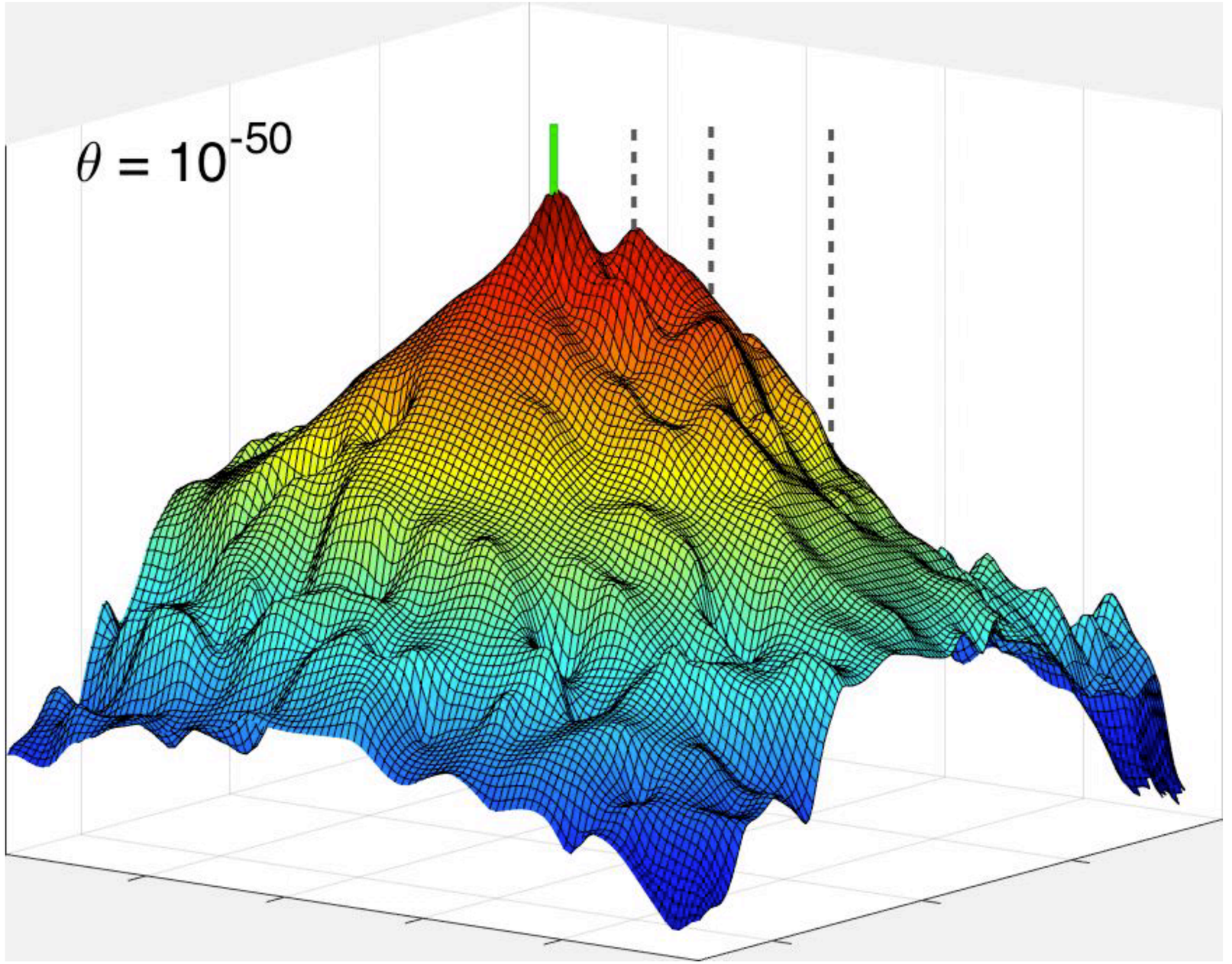


metadata SBM

https://piratepeel.github.io/code.html

Peel*, Larremore*, Clauset. Science Advances 3(5) e1602548. (2017) http://advances.sciencemag.org/content/3/5/e1602548.full

The prior parameter changes the likelihood surface



Other things to know about 1: “The Louvain Method”

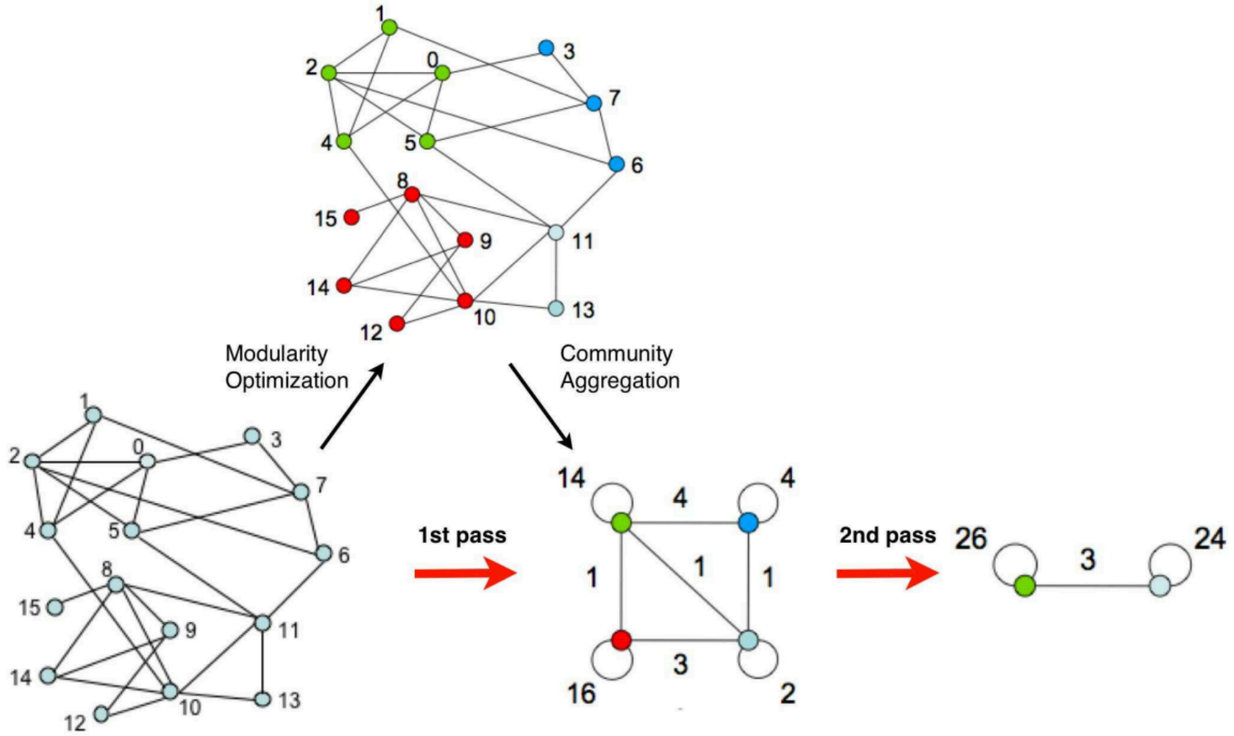
If your network is *really* big. (Millions of nodes, Billions of edges)

Take ClausetNewmanMoore’s approach for greedy Q maximization and find small groups. Run the code again on those groups… And again…

Advantage: fast! big! 🚀

Disadvantage: inherits the assumptions of modularity. (clustering vs modeling)

6K citations. People like it!



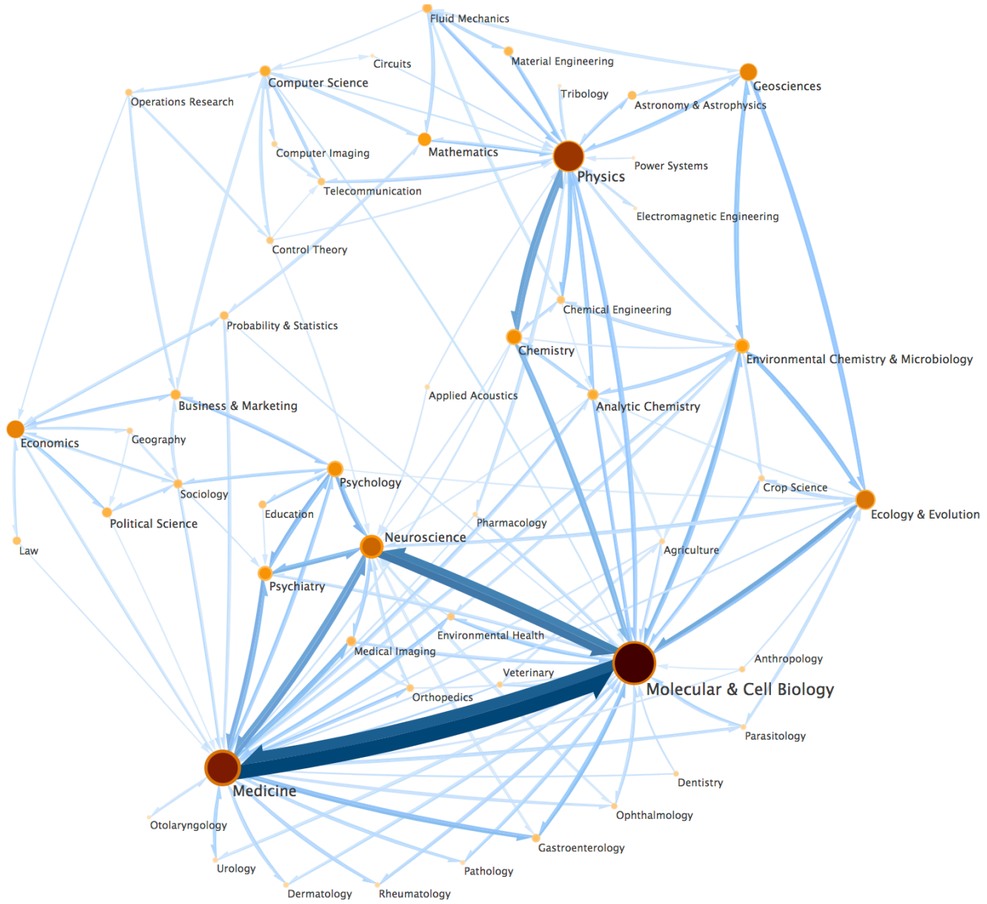
https://en.wikipedia.org/wiki/Louvain_Modularity https://arxiv.org/pdf/0803.0476.pdf

Other things to know about 2: InfoMap

Imagine a random walker on a network.

A description of her walk can be compressed if the network has regions in which the random walker tends to stay for a long time.

Minimizing the “map equation” over all possible network partitions is the same as finding the best codebook.



http://www.mapequation.org/apps/MapDemo.html 🤩

http://www.mapequation.org/code.html

Outlook for community detection

Simply put, we have amazingly powerful tools that did not exist 15 years ago.

Many are principled, statistically rigorous, and we learn more all the time. Those that aren't statistically rigorous are really, really fast.

There is no multiple regression for networks.

"Controlling for C, how important is X in predicting Y?"

Tradeoffs between general and bespoke methods are still being explored.

Outside of SBM, Modularity, Louvain, Infomap, it's a wild west.

Methodologists are keen to be challenged by new problem types.

New scientific questions inspire new methods.

Rankings and linear hierarchies

Many uses for the same techniques. cf regression

Treat the network like a system:

Extrapolation. Make predictions for as-yet unseen nodes (in “space” or time).
Interpolation. Identify missing links.
Generalization. Nodes of this type are like others of the same type.

Treat the network like an artifact:

Mechanisms. How did this network arise? What rules governed its assembly?
Explanations. Coarse-graining or compression.

Treat the network like a means to an end; an intermediate data structure:

Useful division. Need ranking so that we can assign experimental treatments.
Simplification. Downstream regression model needs ranks or groups.

The idea of rankings—pervasive!

Assumptions:

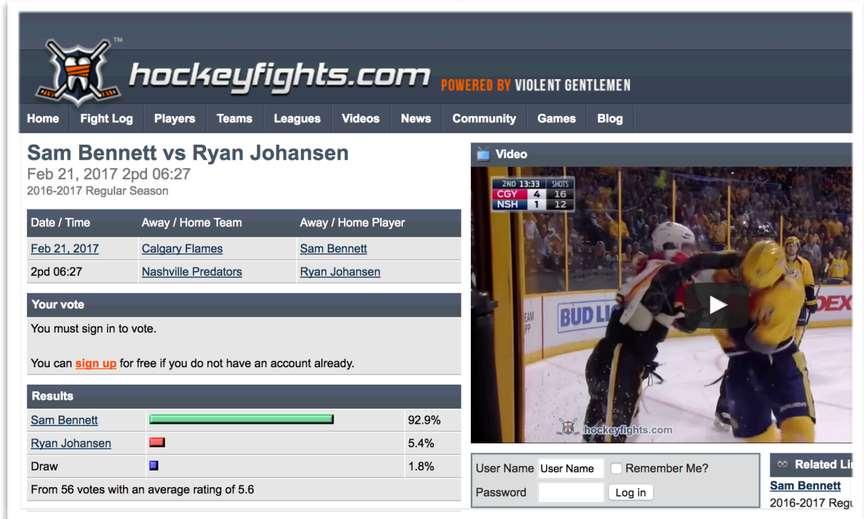
1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.
3. Competitions are pair-wise. (Lee Sedol vs. AlphaGo; Astros vs. Dodgers)

In other words: outcomes are generated by a stochastic process, which is some function of the positions of the competitors.

Systems of dominance

social

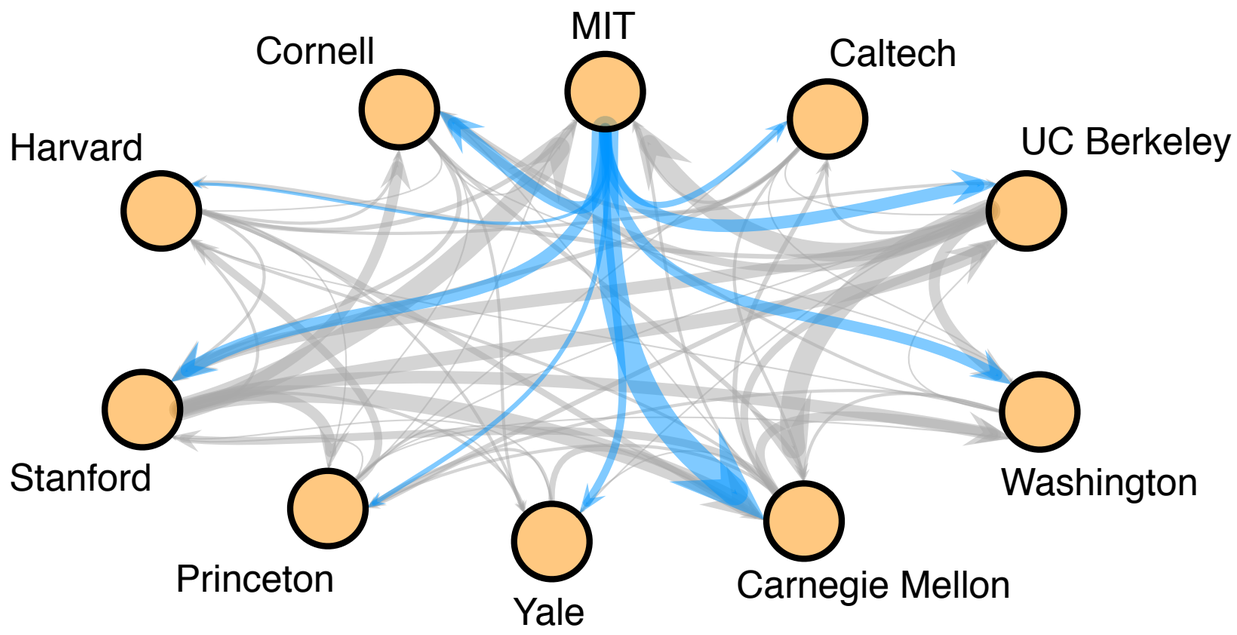
physical

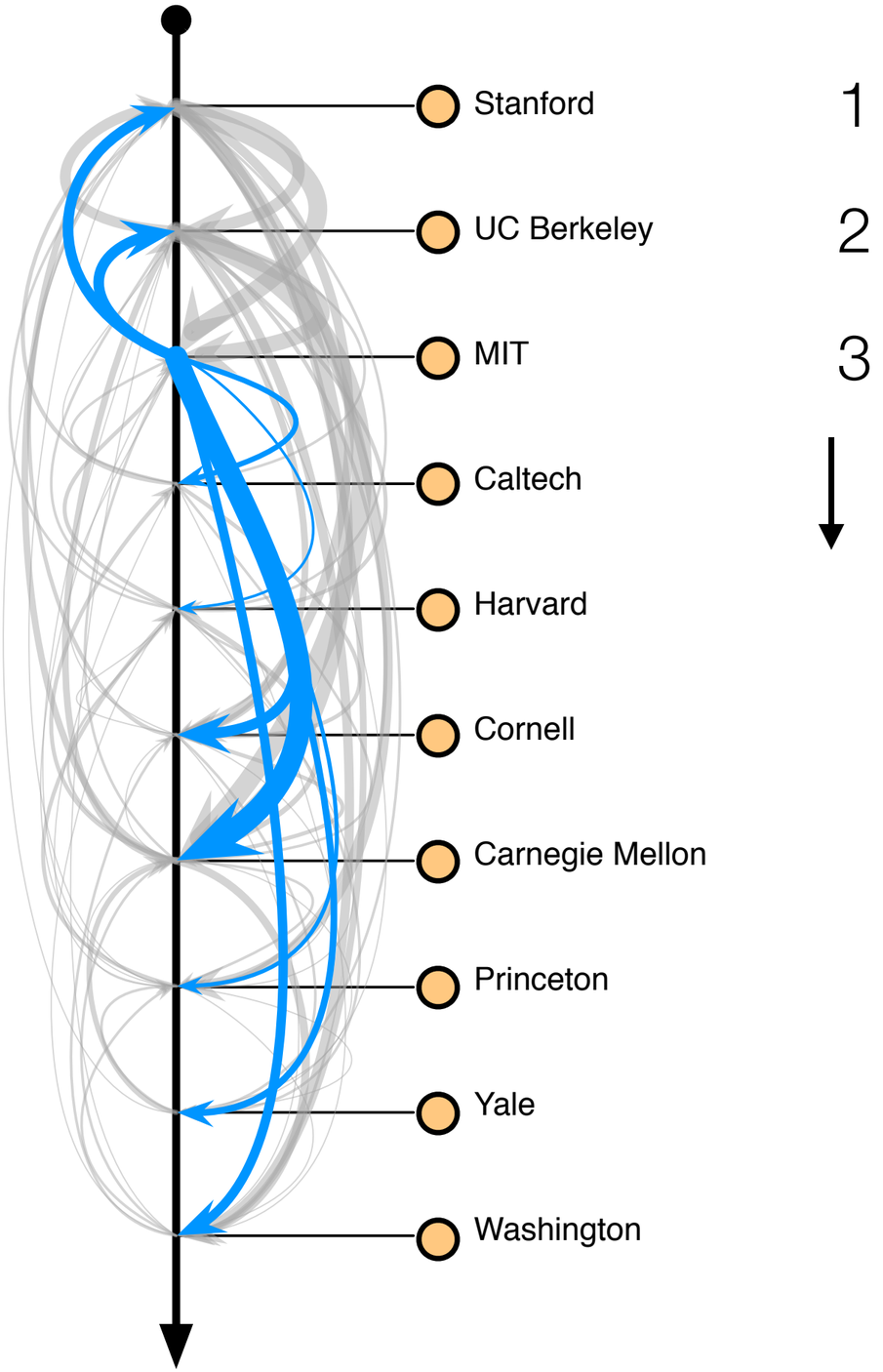


mental

financial

Systems of endorsement



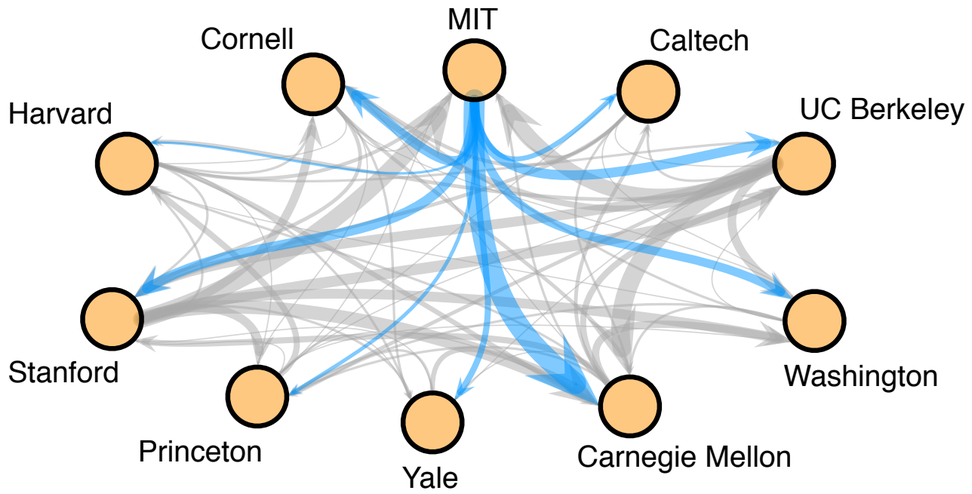


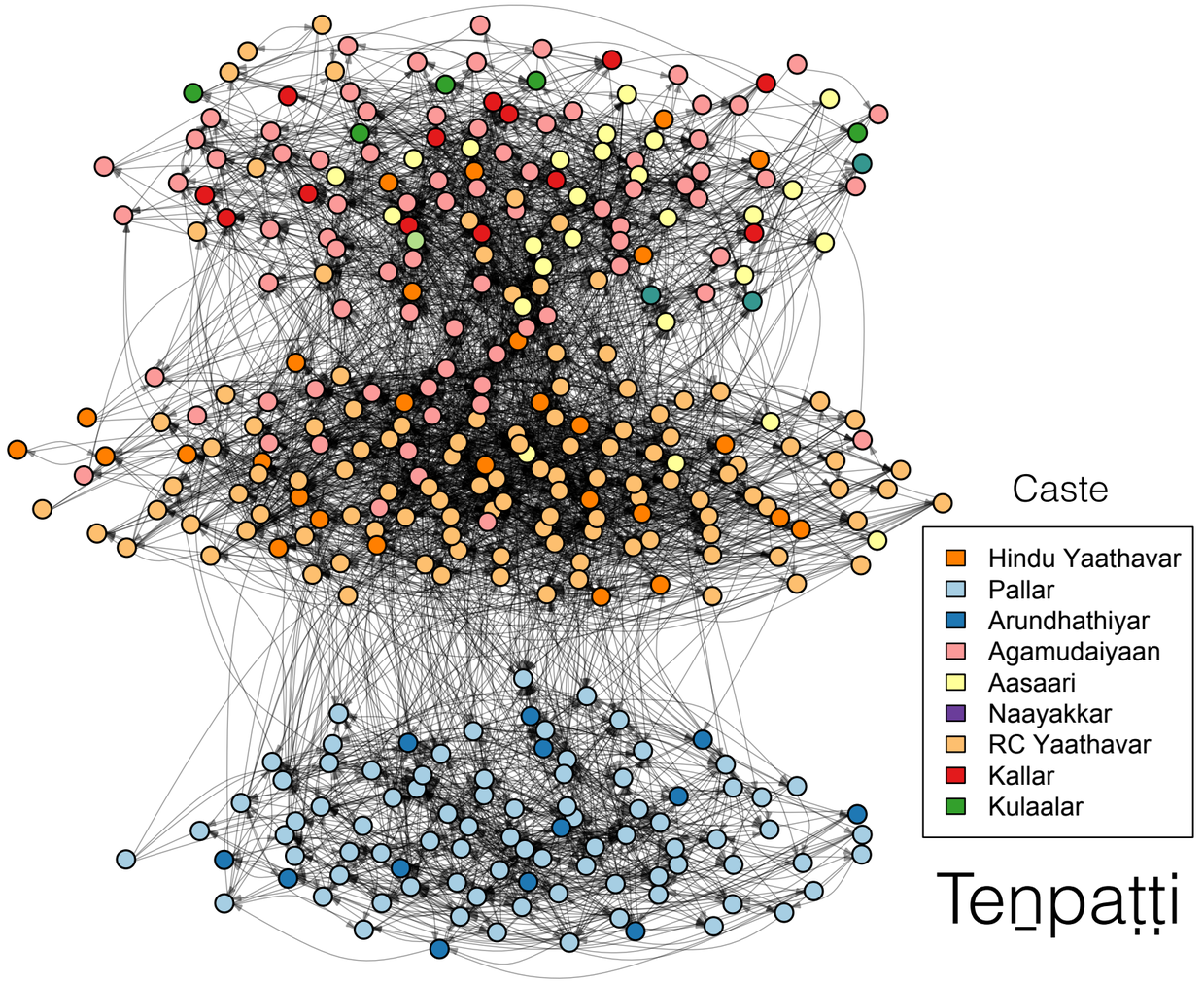
Assumptions:

1. Endorsers have some intrinsic quality.
2. Interactions can reveal differences in qualities.
3. Endorsements are pair-wise.

Clauset, Arbesman, Larremore. *Science Advances* 1, e1400005 (2015).

Systems of endorsement





Latent position can be revealed by **dominance** or **endorsement** interactions.

Figure: Larremore, Hébert-Dufresne, Power. *Draft.*

Data: Power. *Nature Human Behaviour* 1, 0057 (2017)

The setup: suppose we have a *directed* network.

Its adjacency matrix is A .

$A_{ij} = A_{i \to j}$ means i beat j or i was endorsed by j

The problem: Rank the nodes.

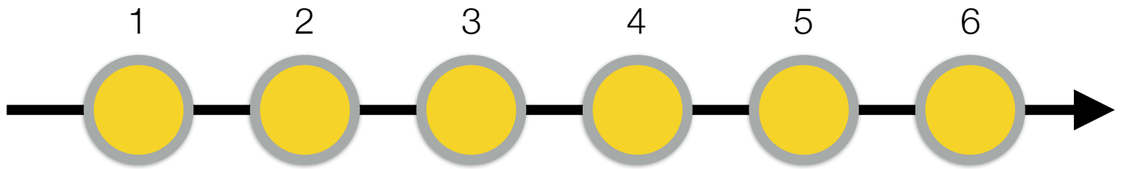
Alternative view: there might be **no network here.** In some cases we're just seeing a network in pairwise comparison data because **networks are a convenient data structure.**

Alternative problem: Which items should be compared next in order to most/best resolve our estimate of the ranks? (sequential tournament design)

Embeddings vs Orderings

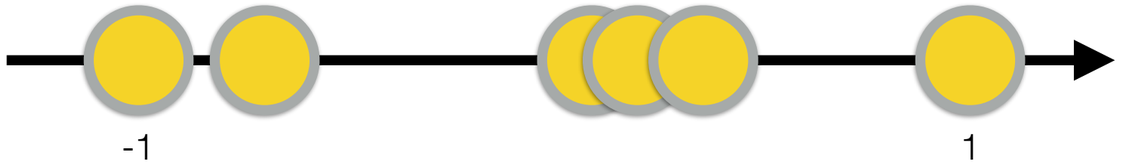
Ordering place the nodes in order:

1, 2, 3, …



Embedding assigns a position to each node:

1, 1.2, 7, 20, 21, 21.2, …



Which one should I use?

> Depends on the use case.

> Is it possible for two nodes to occupy the same rank or position? If so, an embedding is more appropriate. Also better when meaning of 1-rank Δ varies.

> Consider that you can always go from an embedding to an ordering, if you have a rule for breaking ties.

Win-Loss is not satisfactory: schedule matters

Beating the grandmaster counts for more than beating a novice.

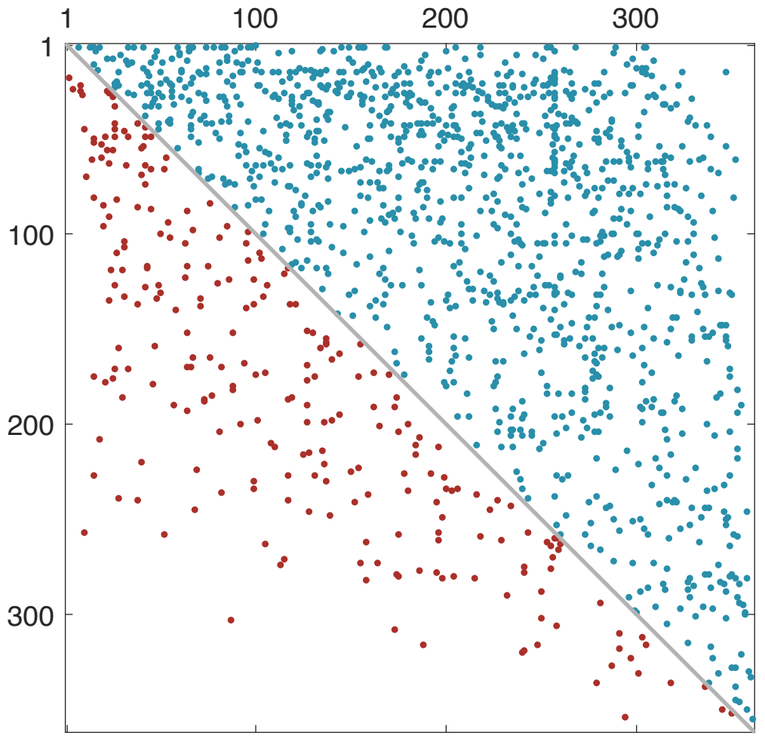
Win and loss tallies don't take this "schedule difficulty" into account. Put differently, win-loss records **leave information on the table.**

One way to make use of this information:

i* beats *j implies $s_i > s_j$

Therefore if we have a whole list of outcomes, we can try to find a total ordering that breaks as few of these implications as possible.

$A_{ij} =$ number of times that *i* beat *j*.



minimum violation ranking: sort *A*.

Win-Loss is not satisfactory: schedule matters

How do we find an ordering that minimizes the number of violations (or upsets) ?

Recipe (MCMC):

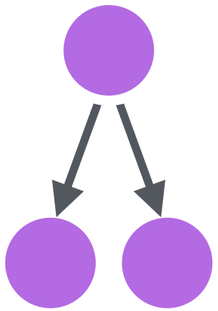
1. Order the nodes randomly.
2. Compute the number of violations. In expectation, this should be 50% of edges.

3. Pick two nodes at random and propose to swap their positions.
4. Compute the number of violations in this scenario.

5. If #violations decreases or stays the same, keep the swap. Otherwise, reject.
6. Repeat until….?

Notes:

* The number of violations is non-increasing over time.
* There may be no unique minimum. Consider this scenario:

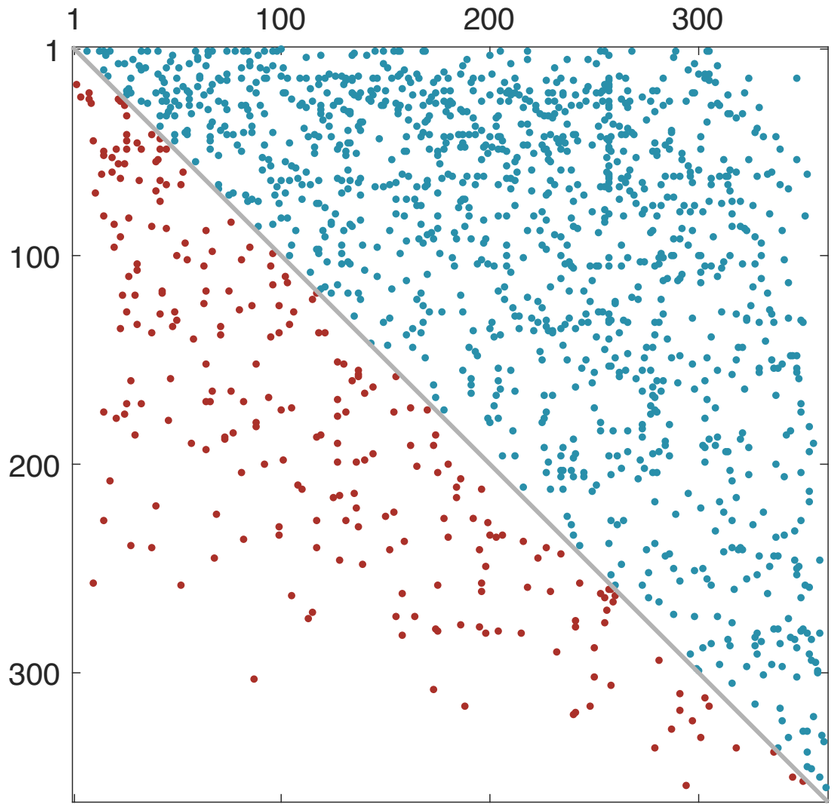


Embeddings & Orderings 0: MVR & Agony

* There is no guarantee of a unique minimizing ranking s.
* Space of ordinal rankings has n! elements, requiring slow search algorithms (e.g. MCMC).
* Ordinal. No ties. No interpretability of rank differences.

What if you allowed for **ties** and then ran Minimum Violation Ranking (MVR)? What would happen?

MVR: uniform cost (1 per edge).
Agony: generic cost function.
for example, difference in ranks.



minimum violation ranking: sort *A.*

What are other premises on which we can base a ranking model?

Gupte et al. Finding hierarchy in directed online social networks WWW (2011)

Embeddings and Orderings 1: Discrete choice models

Louis Leon Thurstone and Thelma Thurstone

Embeddings and Orderings 1: Discrete choice models

Instead of rating everything from 1 to 10, try *paired comparisons*.

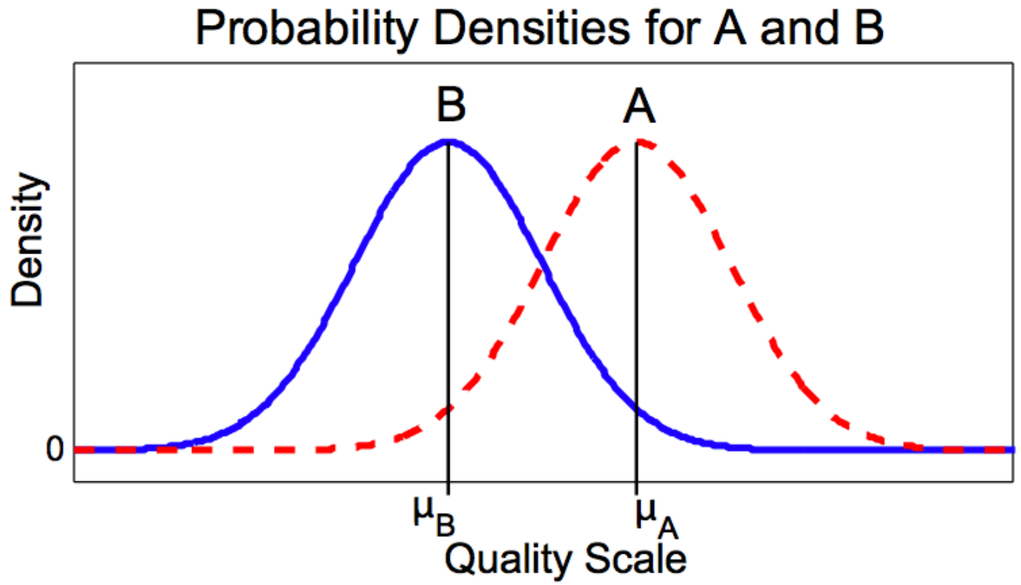
Do you prefer *i* or *j* ?

Why? Consider: My 3 is not your 3. What is 1 and what is 10?

https://xkcd.com/883/

Embeddings and Orderings 1: Discrete choice models

Thurstone: items have quality distributions. When a person judges whether A is better than B they draw from A's distribution and from B's distribution and see which is higher.



Thurstone modeled these as Gaussians.

$$P(A > B) = P(A - B > 0)$$

Difference of Gaussians is Gaussian.

$$\hat{\mu}_{AB} = \Phi^{-1}\left(\frac{C_{A\to B}}{C_{A\to B} + C_{B\to A}}\right)$$

Where $\Phi^{-1}(x)$ is the inverse CDF of standard normal, a.k.a. the *probit*.

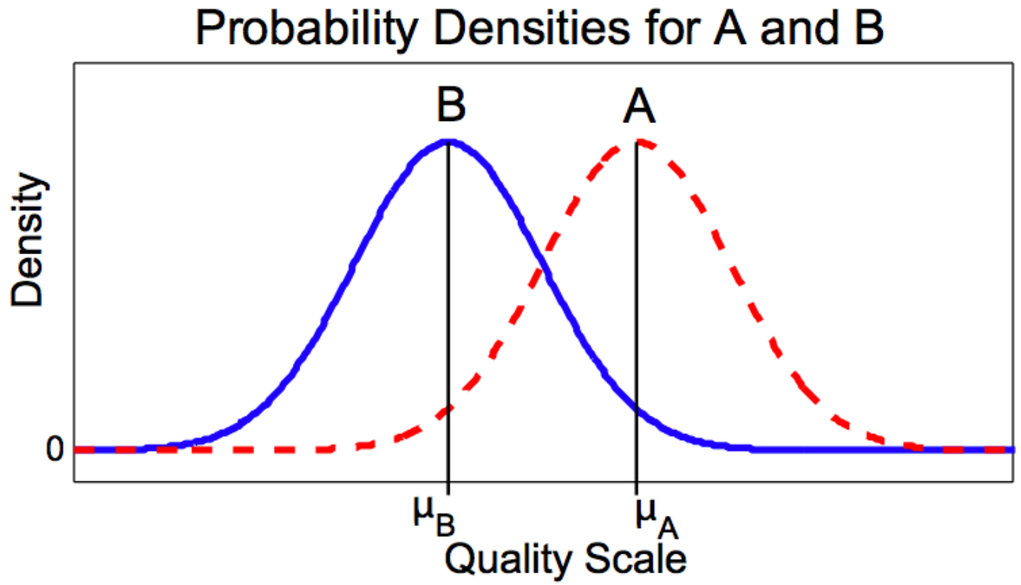
Powerful idea: lots of pairwise comparisons = estimates of all the qualities! An embedding!

Key: pairwise comparisons = directed network. i preferred to j $=$ $i \to j$

Finding the qualities of items from pairwise comparisons = Finding embedding of nodes.

Embeddings and Orderings 1: Discrete choice models

Bradley-Terry & Luce: items have quality distributions. When a person judges whether A is better than B they draw from A's and from B's distribution and see which is higher.



BTL

$$P(A > B) = \frac{\pi_A}{\pi_A + \pi_B}$$

Or usually:

$$P(A > B) = \frac{e^{\mu_A/s}}{e^{\mu_A/s} + e^{\mu_B/s}} = \frac{1}{1 + e^{-(\mu_A - \mu_B)/s}}$$

Same idea; different distribution. (*logit* instead of *probit*; *Gumbel* instead of *Gaussian*)

Powerful idea: lots of pairwise comparisons = estimates of all the qualities! An embedding!

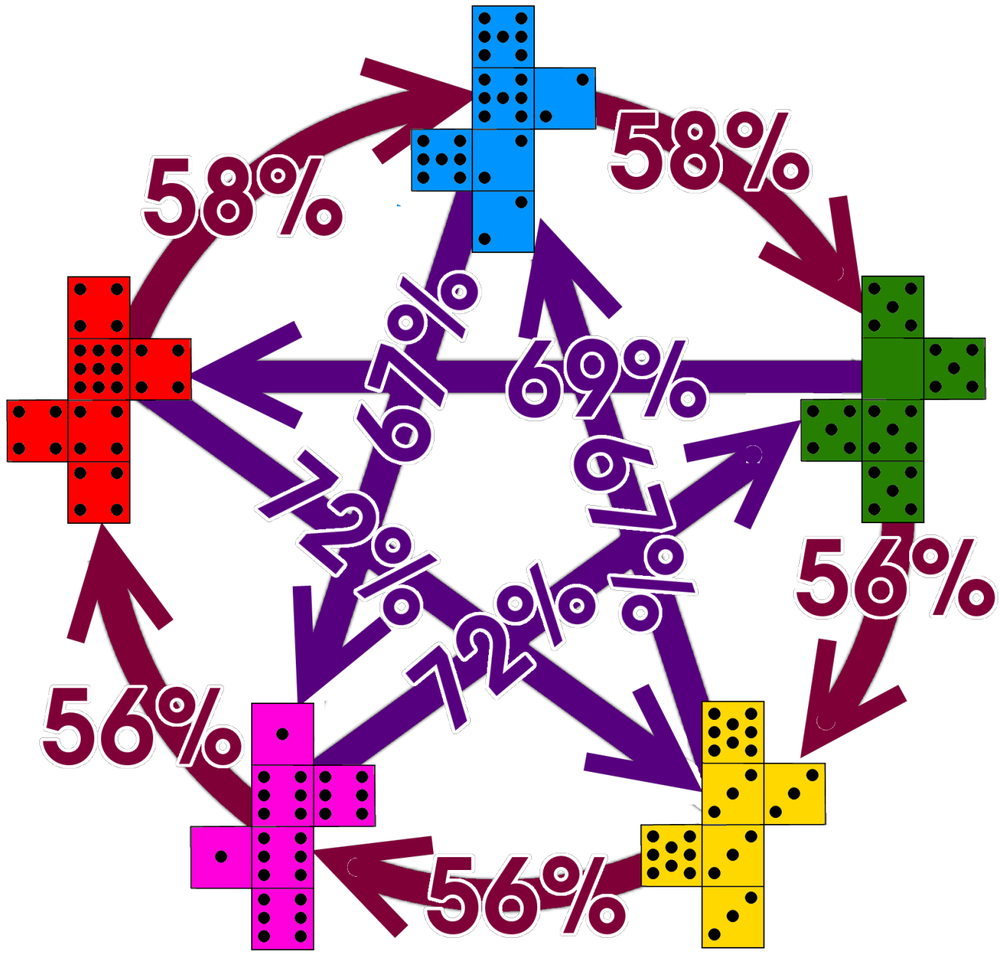
BTL avoids non-transitivities (aka rock-paper-scissors)

Introducing: **non-transitive dice**!

- 3 (or more) dice {A,B,C}
- faces chosen so that they have the property:
  - A>B more than half the time.
  - B>C more than half the time.
  - C>A more than half the time (?!)

https://en.wikipedia.org/wiki/Nontransitive_dice

A great gift for your favorite nerd's desk!
Go to the makerspace and laserbeam your own!



Bradley-Terry-Luce

These methods embed items or players in a 1D space.

- Provably avoids non-transitive properties
- Great when lots of data per interaction.

Pairwise ranking is really nice for ordering large sets of preferences too, and this model specifically models the probability that the preference will be for *i* over *j*.

Iterative algorithms exist. Needs a little regularization so the winningest winners don't fly off to infinity.

$$P(i \to j) = \frac{\gamma_i}{\gamma_i + \gamma_j}$$

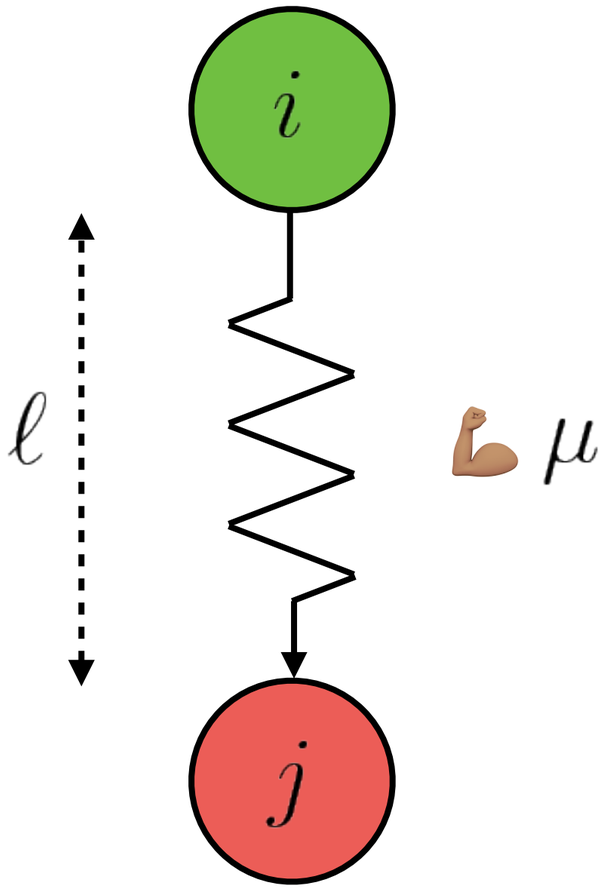
Embeddings and Orderings 1: Discrete choice models

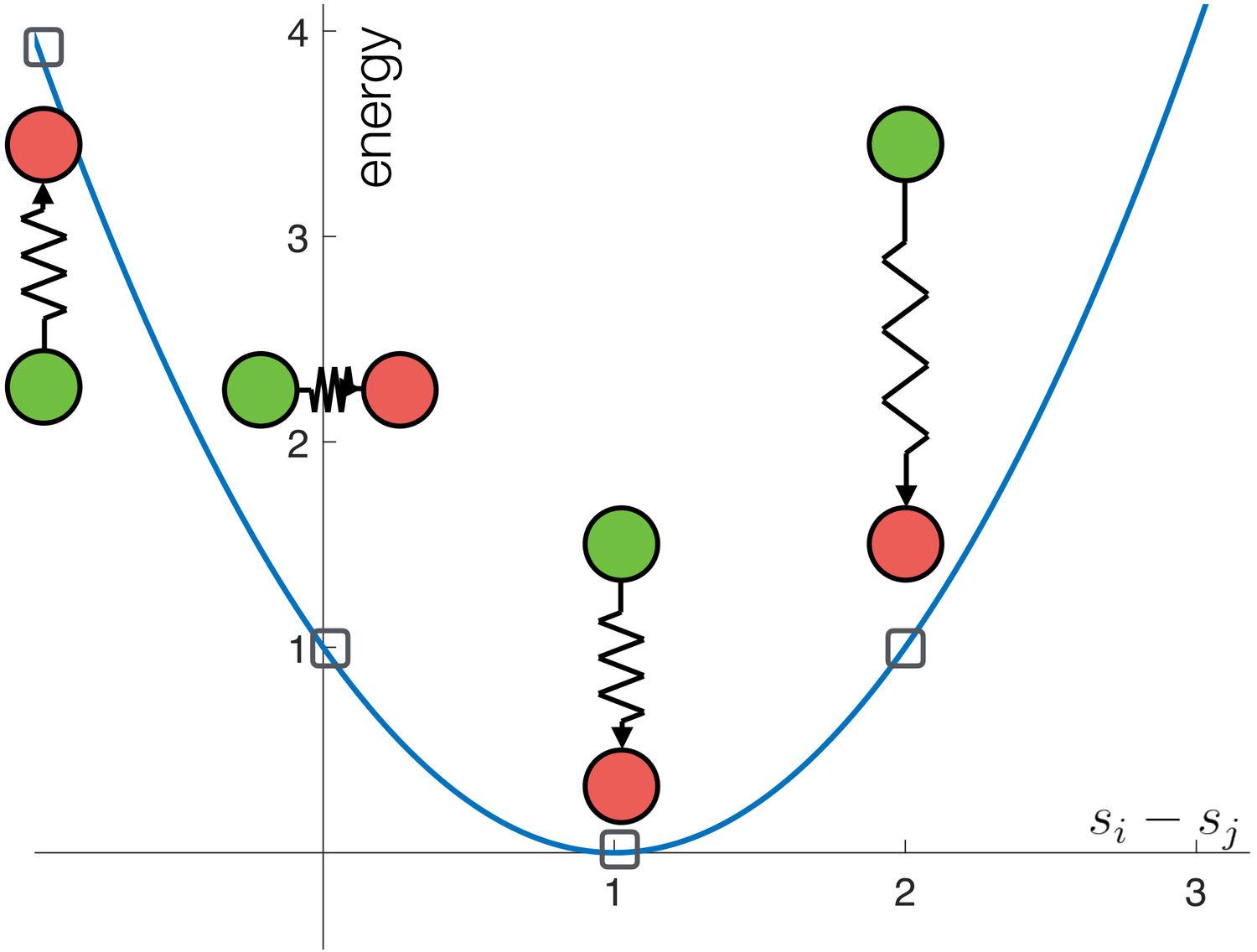
Introductory tutorial:
http://mayagupta.org/publications/PairedComparisonTutorialTsukidaGupta.pdf

Discrete choice today:
https://web.stanford.edu/~jugander/papers/nips16-pcmc-slides.pdf

Embeddings & Orderings 2: SpringRank

Each directed edge = directed spring





How much energy is this system of springs?

Relax and let the springs decide the ranks

$$H(s) = \frac{1}{2} \sum_{i,j=1}^{N} A_{ij}(s_i - s_j - 1)^2$$

SpringRank Hamiltonian = energy of the system, given the node positions *s*.

Because the springs are linear, the potential is quadratic.

The SR Hamiltonian is *convex* in s.

$$\nabla H(s) = 0$$

The solution is unique…up to an additive constant. (Why?)

Derivatives work out nicely

$$0 = \frac{\partial H}{\partial s_i} = \sum_j A_{ij}(s_i - s_j - 1) - A_{ji}(s_j - s_i - 1)$$

Rewrite as a linear algebra problem.

$$\left[D^{\text{out}} + D^{\text{in}} - \left(A + A^T\right)\right] s^* = \left[D^{\text{out}} - D^{\text{in}}\right] \mathbf{1}$$

We know *a priori* that the matrix on the left is singular: translational invariance of H(s).
[if *s* is a solution, then *s* + *k* is a solution for any constant *k;* eigenvalue 0, eigenvector **1**]

Notice: the matrix on the left is the *graph Laplacian* of the *undirected* network.

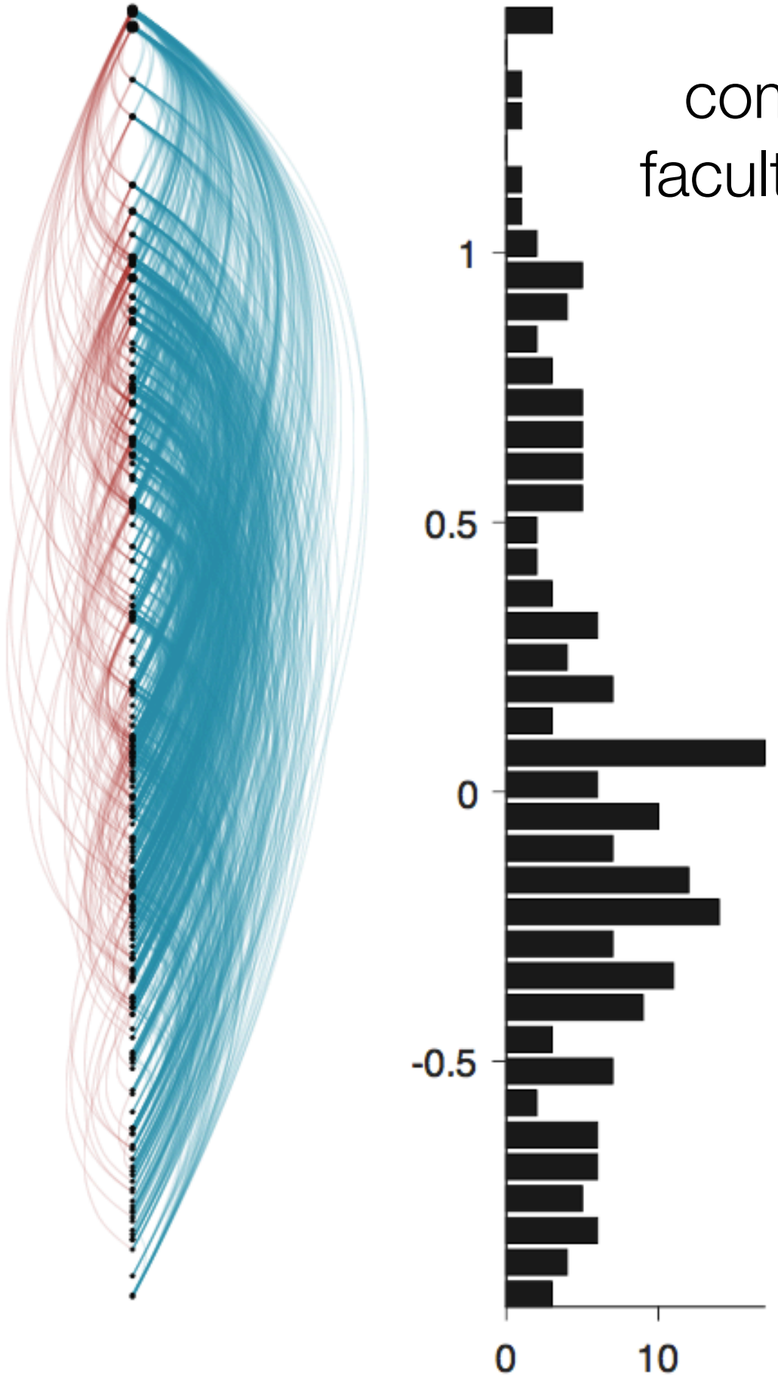
Uniqueness: Set s_1=0, min(*s*)=0, or mean(*s*)=0. Or use a pseudoinverse. Or regularize.

It works!

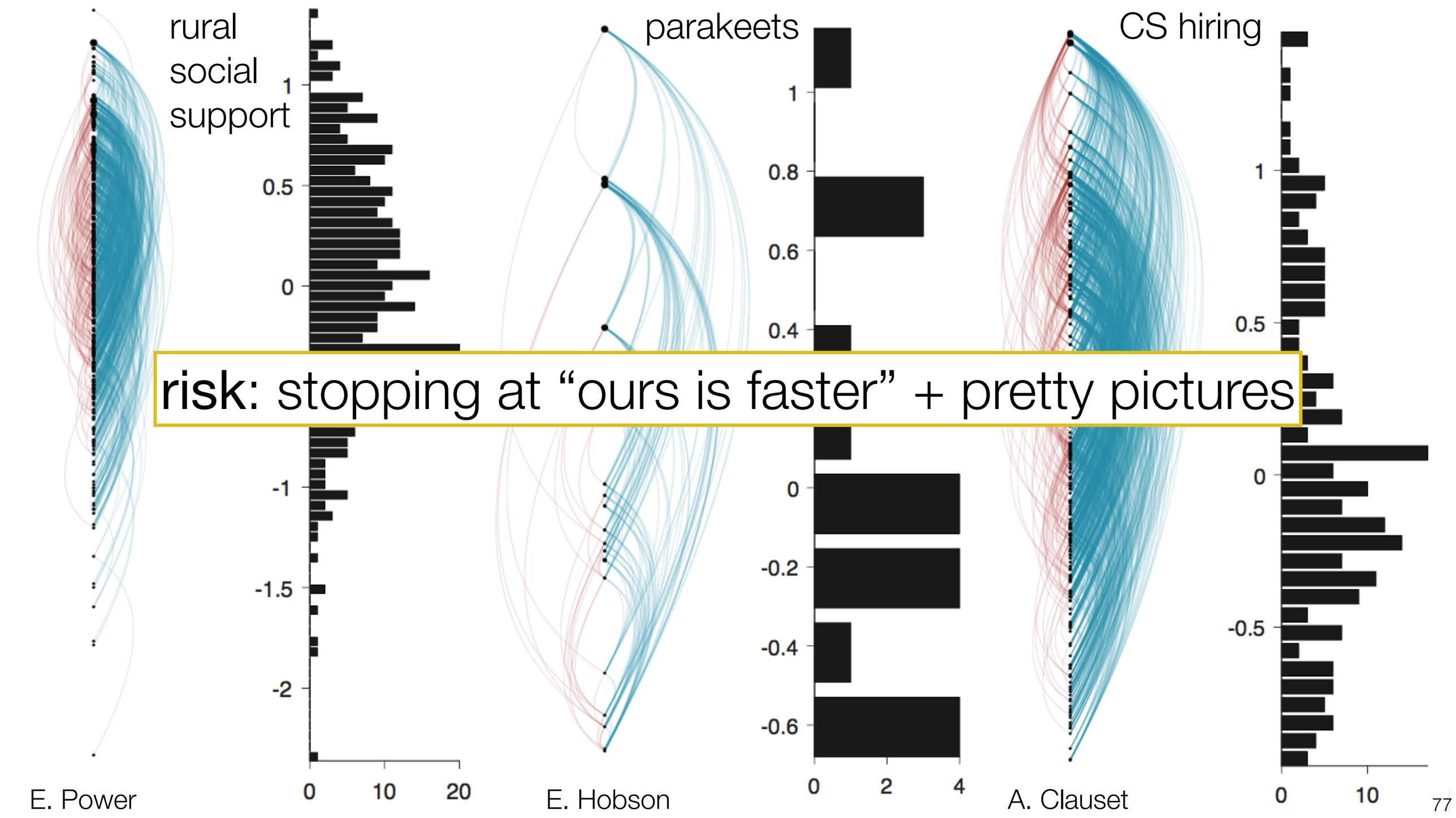
Real networks tend to be sparse…
our linear algebra problem is sparse…
we can use sparse iterative solvers…
millions of edges in **seconds.**

Even better: it's a linear-Laplacian system.
🚀 Near-linear-time (in |edges|) solutions.

Note that node positions can be clumpy,
since this is an *embedding*.



computer science
faculty hiring network



risk: stopping at “ours is faster” + pretty pictures

E. Power

E. Hobson

A. Clauset

Cross validation: train on 80%, predict 20%

In a linear hierarchy the key quantity to predict is *edge direction,* given *edge existence*.

If *i* and *j* were to face off, who would win?

I'll give you *undirected(A)*, and you predict *directed(A).*

Setup: learn *s* from 80% of *A*. Then predict edge directions for remaining 20% of *A.*

SpringRank predicts edge direction based on the relative direction probabilities:

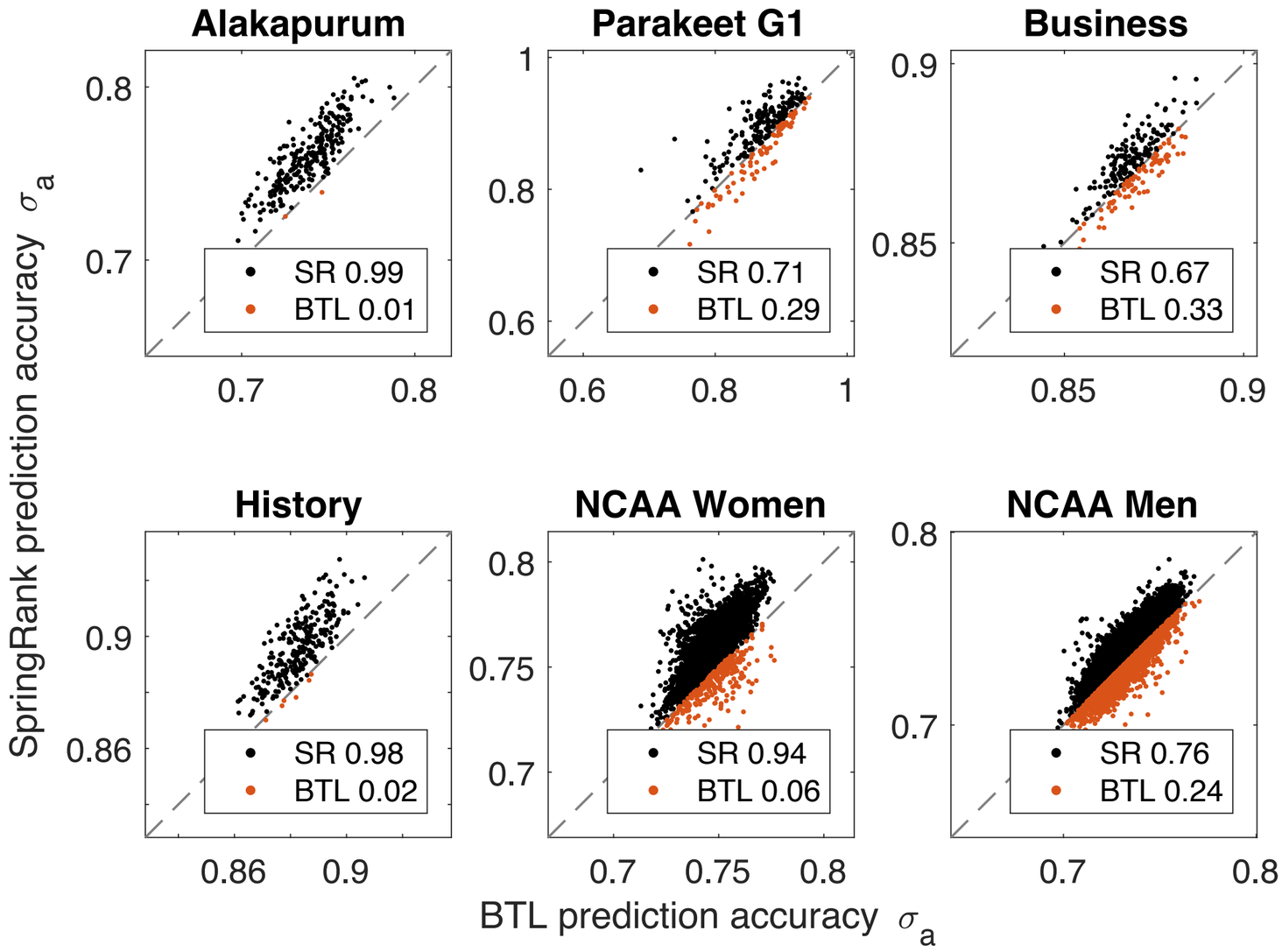
$$P_{ij}(\beta) = \frac{e^{-\beta H_{ij}}}{e^{-\beta H_{ij}} + e^{-\beta H_{ji}}} = \frac{1}{1 + e^{-2\beta(s_i - s_j)}}$$

Cross validation vs BTL: SR makes better predictions

Accuracy:

$$\sigma_a = 1 - \frac{1}{2M} \sum_{i,j} |A_{ij} - (A_{ij} + A_{ji})\, P_{ij}|$$

Goal: maximize the number of correctly predicted edge directions.



50 independent trials of 5-fold cross validation (250 folds)

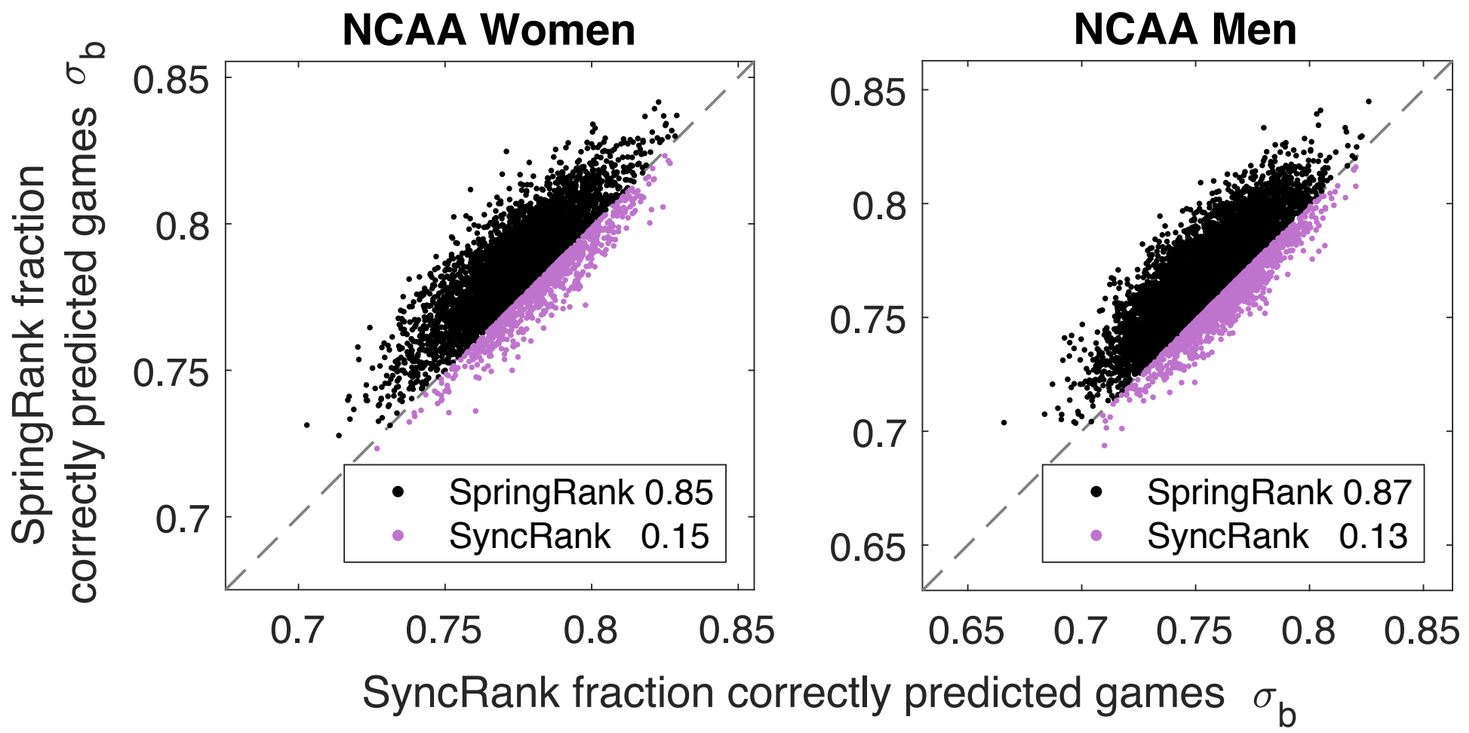
Cross validation vs SyncRank: SR makes better predictions

"One-bit" Accuracy:

Higher ranked player always wins.

- No probabilistic prediction.
- Bad for gambling.

Goal: maximize the number of correctly predicted edge directions.



50 independent trials of 5-fold cross validation (250 folds)

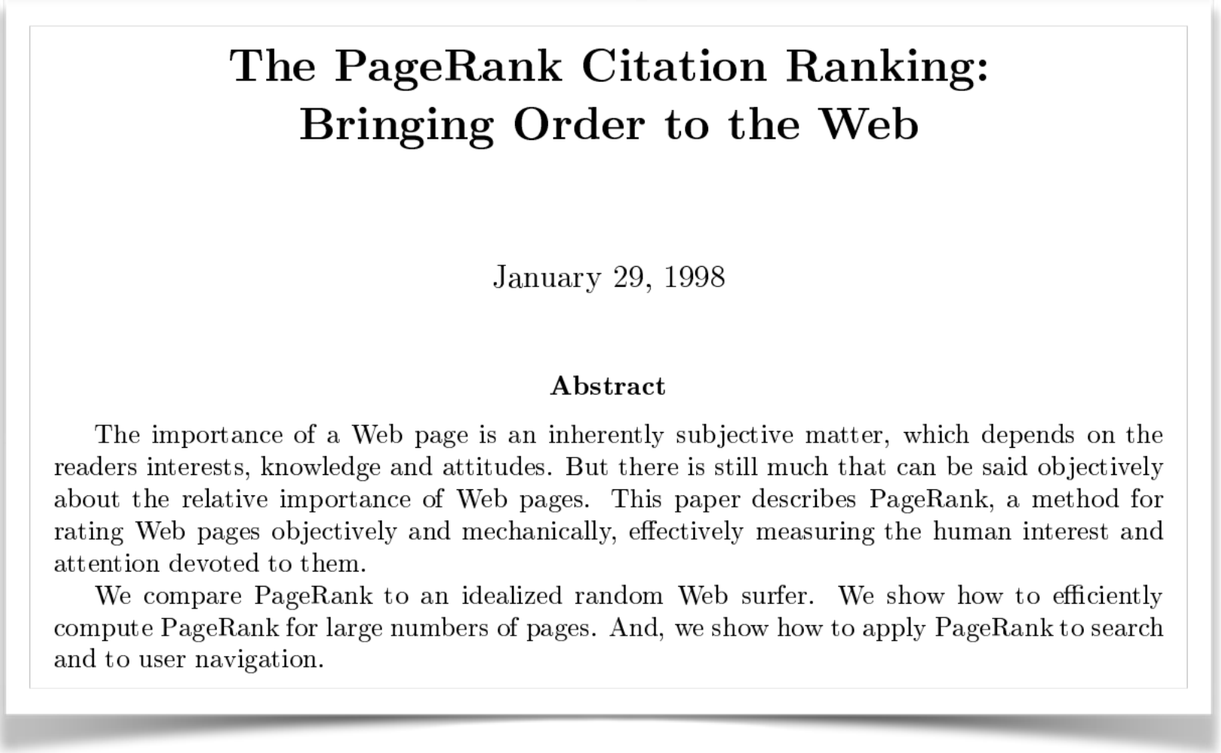
Why/when would a model of springs make better predictions than a model of the choices themselves?🧐

Embeddings and Orderings 3: PageRank

PageRank defines scalar rank recursively:

important pages are those that are linked to by important pages.

- Great at finding the top 3 but limited predictions available using the PageRank scores.



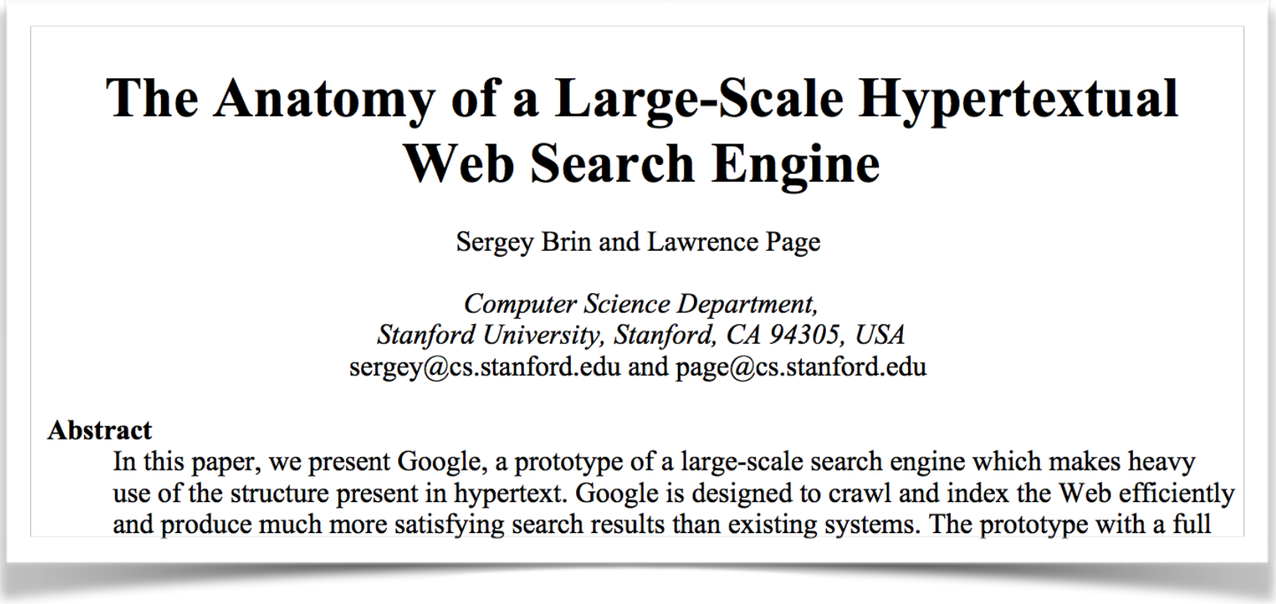
The PageRank Citation Ranking: Bringing Order to the Web

January 29, 1998

Abstract

The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes PageRank, a method for rating Web pages objectively and mechanically, effectively measuring the human interest and attention devoted to them.

We compare PageRank to an idealized random Web surfer. We show how to efficiently compute PageRank for large numbers of pages. And, we show how to apply PageRank to search and to user navigation.



The Anatomy of a Large-Scale Hypertextual Web Search Engine

Sergey Brin and Lawrence Page

Computer Science Department,
Stanford University, Stanford, CA 94305, USA
sergey@cs.stanford.edu and page@cs.stanford.edu

Abstract

In this paper, we present Google, a prototype of a large-scale search engine which makes heavy use of the structure present in hypertext. Google is designed to crawl and index the Web efficiently and produce much more satisfying search results than existing systems. The prototype with a full

Embeddings and Orderings 3: PageRank

We imagine a web surfer who choose a starting webpage at random.

From that webpage, she looks at the links on the page, and either
(a) clicks on a random link or
(b) stops surfing; when she returns, she starts at a new random page.

What's the probability that she's at a particular page? *That's PageRank*.

$$\pi_{ji} = \frac{A_{ji}}{k_j}$$

define a transition matrix

$$p_i = \frac{1-d}{N} + d\sum_j p_j \pi_{ji}$$

write the equation

$$\mathbf{p} = \left(\frac{1-d}{N}\right)\mathbf{1} + d\pi^T\mathbf{p}$$

matrix-vector form

Alternative: stationary distribution of random walk on the network + weak all-to-all links

Jeremy Kun: http://www.infinitelooper.com/?v=K3pT0gTaDec&p=n

Embeddings and Orderings 4: Ball & Newman

Generative model:
Generate the patterns that you want to identify.

Create N nodes.
Assign each node an integer rank *r*, from 1 to N.

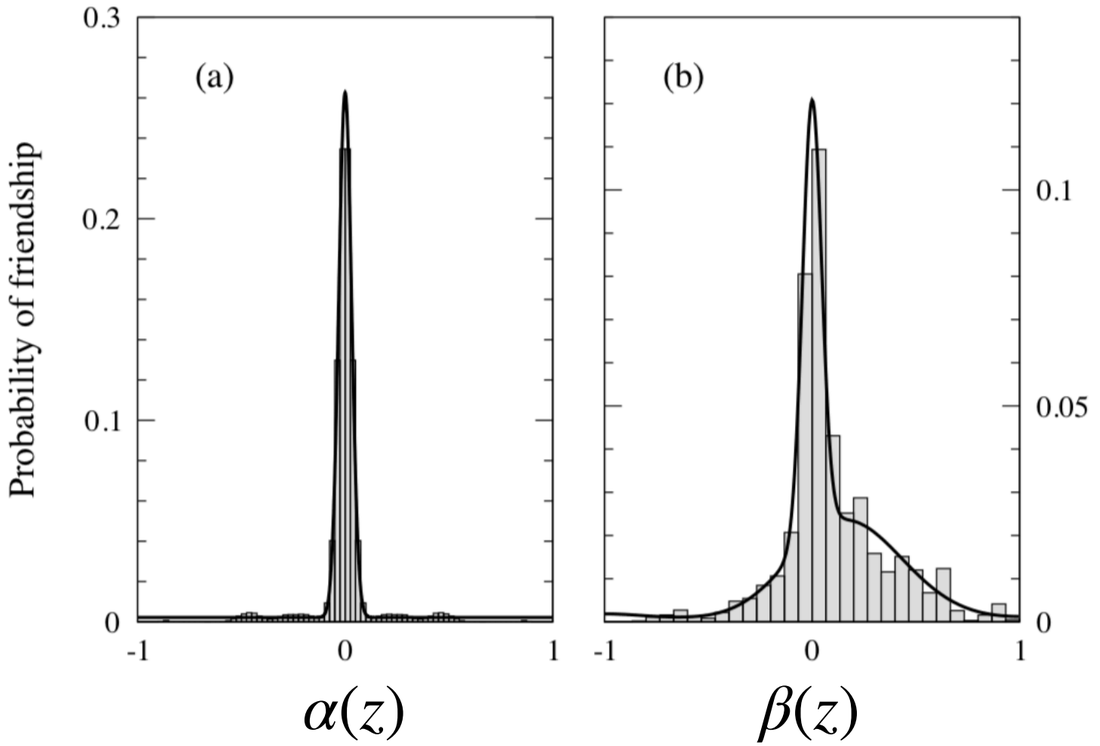
IRL, not all friendships are reciprocated 😭
So let's generate undirected AND directed edges:

$$P(i \leftrightarrow j) \;=\; \alpha(r_i - r_j)$$

A gaussian centered at 0

$$P(i \rightarrow j) \;=\; \beta(r_i - r_j)$$

Fourier cosine series, keeping five terms & squaring to enforce nonnegativity, plus an additional Gaussian peak at the origin.



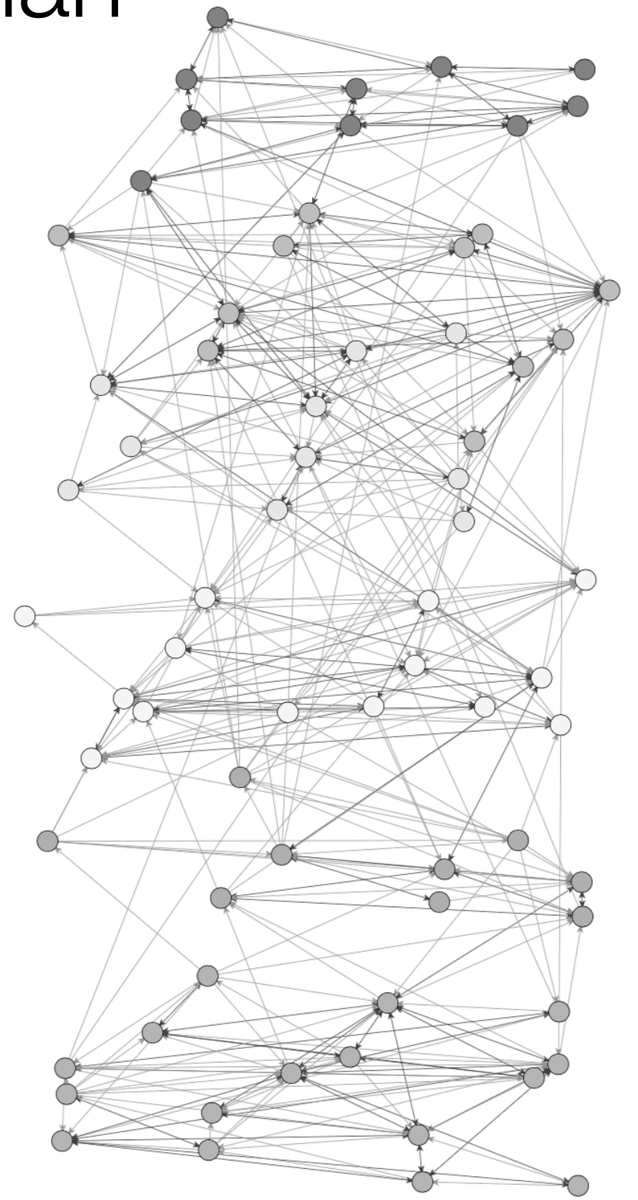
Ball, Newman. Network Science 1, 16-30 (2013)

Embeddings and Orderings 4: Ball & Newman

GM + parameters
α
β
ranks

inference ←

Data



Inferred parameters of people's attachment preferences & ranks.

- Identified the need to learn from reciprocated friendships.
- Found that in AddHealth data, teens link to others of *nearby* social status.

Ball, Newman. Network Science 1, 16-30 (2013)

Embeddings and Orderings 5: Niche Models

Niche Models embed species in a latent space based on feeding preferences:

most species feed from narrow range in a 1-dim. space (~body size).

- Great for food webs. Inference models v slow for all but small networks.

Want more? Jen Dunne, Cris Moore

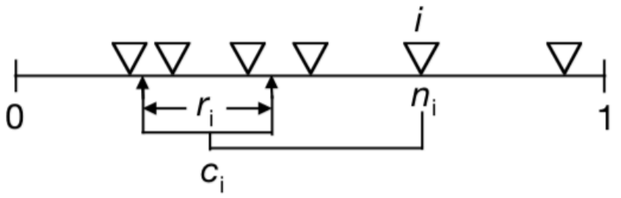


Figure 1 Diagram of the niche model. Each of ***S*** species (for example, ***S*** = 6, each shown as an inverted triangle) is assigned a 'niche value' parameter (n_i) drawn uniformly from the interval [0,1]. Species ***i*** consumes all species falling in a range (r_i) that is placed by uniformly drawing the centre of the range (c_i) from [r_i/2, n_i]. This permits looping and cannibalism by allowing up to half of r_i to include values $\geqslant n_i$. The size of r_i is assigned by using a beta function to randomly draw values from [0,1] whose expected value is 2***C*** and then multiplying that value by n_i [expected E(n_i) = 0.5] to obtain the desired ***C***. A beta distribution with $\alpha = 1$ has the form $f(x|1, \beta) = \beta(1-x)^{\beta-1}$, $0 < x < 1$, 0 otherwise, and $E(X) = 1/(1+\beta)$. In this case, $x = 1-(1-y)^{1/\beta}$ is a random variable from the beta distribution if y is a uniform random variable and β is chosen to obtain the desired expected value. We chose this form because of its simplicity and ease of calculation. The fundamental generality of species i is measured by r_i. The number of species falling within r_i measures realized generality. Occasionally, model-generated webs contain completely disconnected species or trophically identical species. Such species are eliminated and replaced until the web is free of such species. The species with the smallest n_i has $r_i = 0$ so that every web has at least one basal species.

Williams & Martinez. Nature 404.6774 (2000).

Many uses for the same techniques. cf regression

Treat the network like a system:

Extrapolation. Make predictions for as-yet unseen nodes (in “space” or time).
Interpolation. Identify missing links.
Generalization. Nodes of this type are like others of the same type.

Treat the network like an artifact:

Mechanisms. How did this network arise? What rules governed its assembly?
Explanations. Coarse-graining or compression.

Treat the network like a means to an end; an intermediate data structure:

Useful division. Need groups so that we can assign treatments in an A/B test.
Simplification. Downstream regression model needs ranks or groups.

PDF of slides available
http://danlarremore.com/**CommunityDetection_and_Ranking_Larremore_2019.pdf**

Goals for this talk:

1. **Why** do we look for large-scale structure? 🤔
2. **How** do we find communities and hierarchies? 🧐
3. **Where** can we read more details? 📚

Thank you

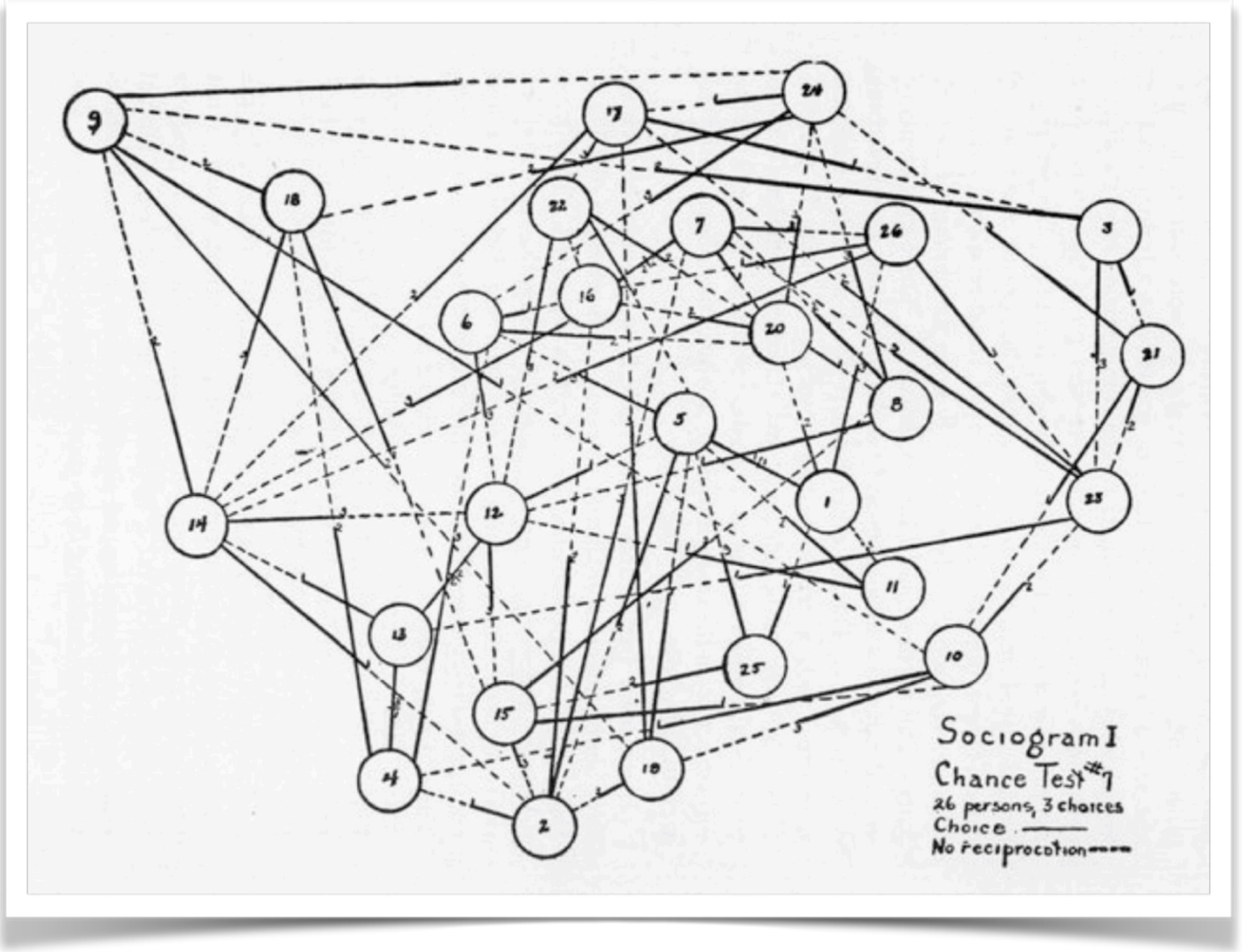
@danlarremore
daniel.larremore@colorado.edu

Aside: the birth of null models & chance sociograms

Who shall survive? Moreno, 1936

Moreno wondered if there were structural explanations for why certain young girls were running away from the school, and thought that sociographic analysis might hold an answer.

chance sociograms



SIAM Review: Configuring random graph models with fixed degree sequences. http://arxiv.org/abs/1608.00607

The Book: http://www.asgpp.org/docs/wss/Book%20VI/index.html

Johan Ugander's Post: https://jugander.wordpress.com/2014/08/07/computational-perspectives-on-large-scale-social-networks-a-brief-history/

Aside:

Here is one of my favorite papers of all time:

JMLR: Workshop and Conference Proceedings 27:65–79, 2012 Workshop on Unsupervised and Transfer Learning

Clustering: Science or Art?

Ulrike von Luxburg ULRIKE.LUXBURG@TUEBINGEN.MPG.DE
Max Planck Institute for Intelligent Systems, Tübingen, Germany

Robert C. Williamson BOB.WILLIAMSON@ANU.EDU.AU
Australian National University and NICTA, Canberra ACT 0200, Australia

Isabelle Guyon ISABELLE@CLOPINET.COM
ClopiNet, 955 Creston Road, Berkeley, CA 94708, USA

http://proceedings.mlr.press/v27/luxburg12a/luxburg12a.pdf