A physical model for efficient ranking in networks

Daniel Larremore
BioFrontiers Institute & Dept. of Computer Science

Apr 9, 2018
Duke - DNAC

University of Colorado **Boulder**

daniel.larremore@colorado.edu
@danlarremore

Rankings and linear hierarchies

Rankings and linear hierarchies

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)

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.

The idea of rankings—pervasive!

Assumptions:

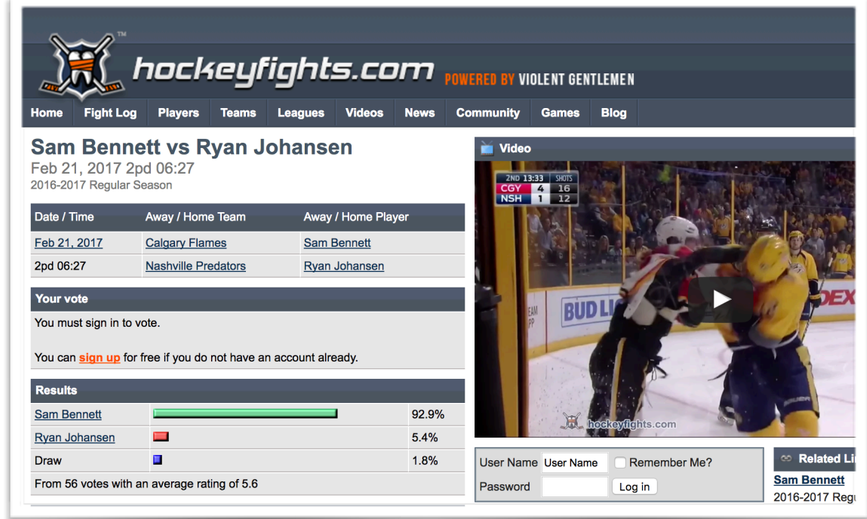
1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.
3. Competitions are pair-wise. (Lee Sedol vs. AlphaGo; Astros vs. Dodgers)

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

Systems of dominance

social

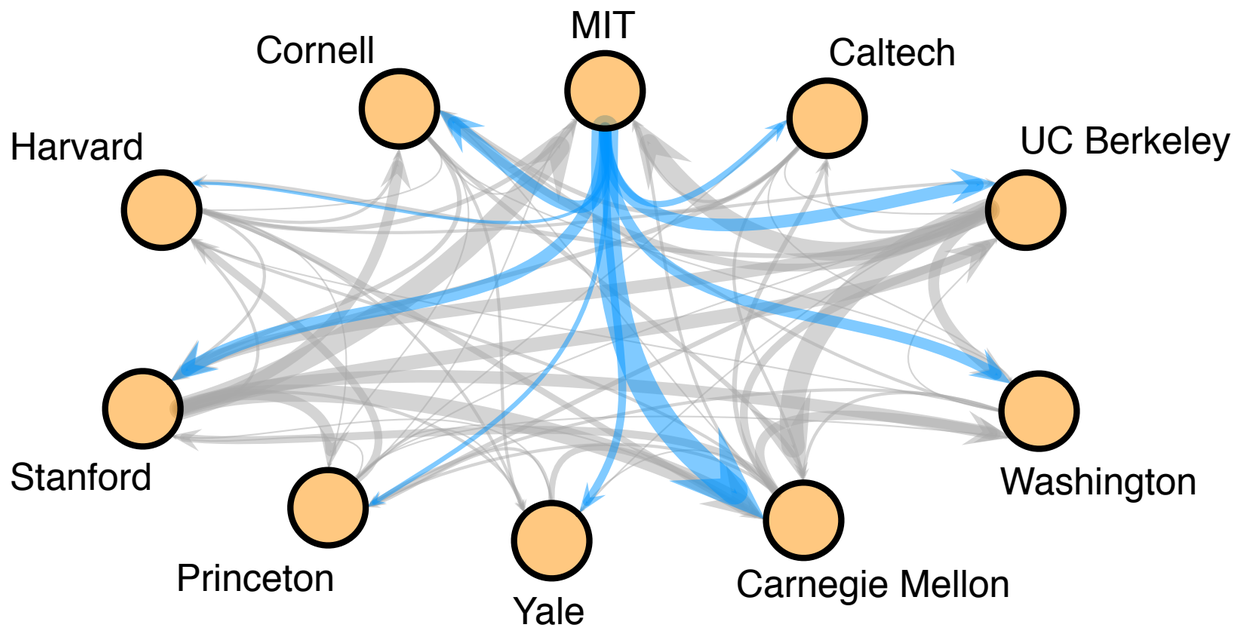
physical



mental

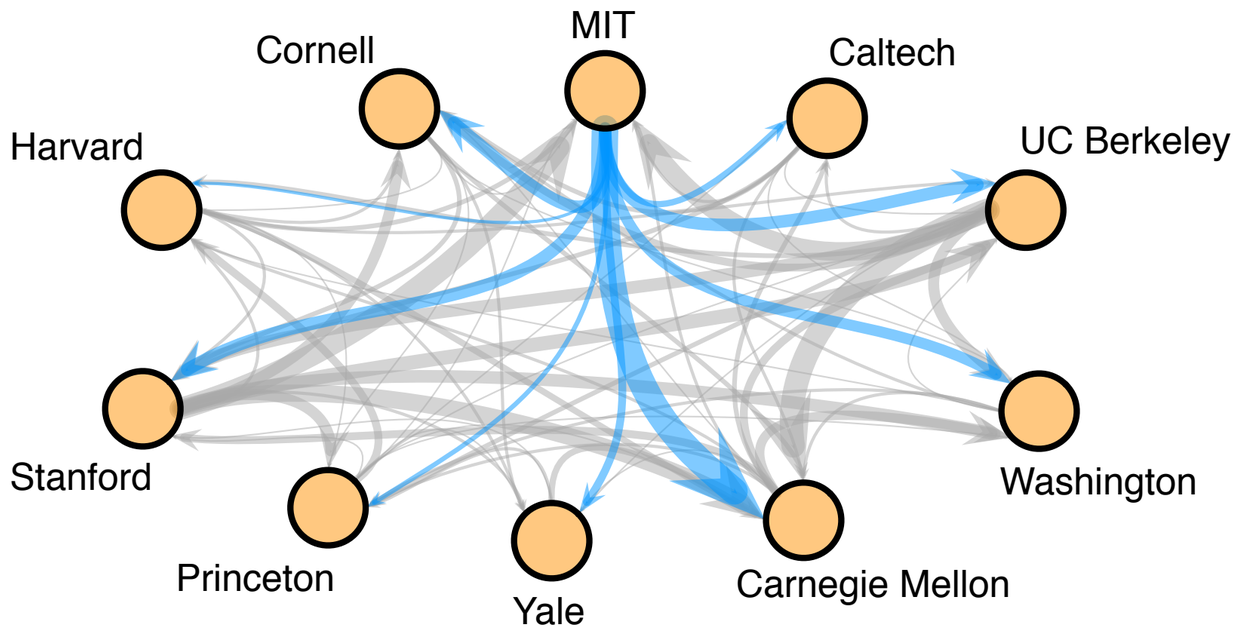
financial

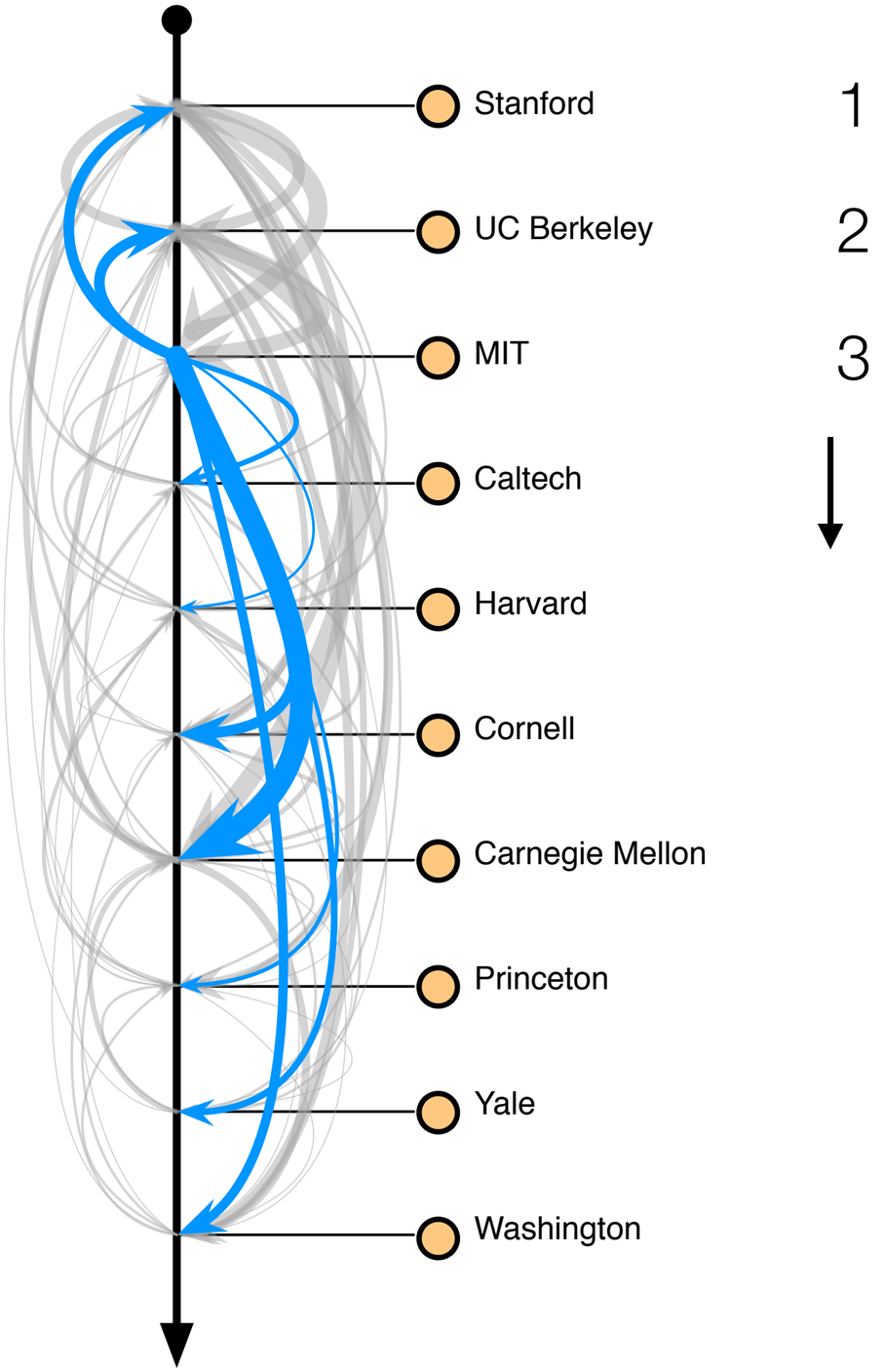
Systems of endorsement



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

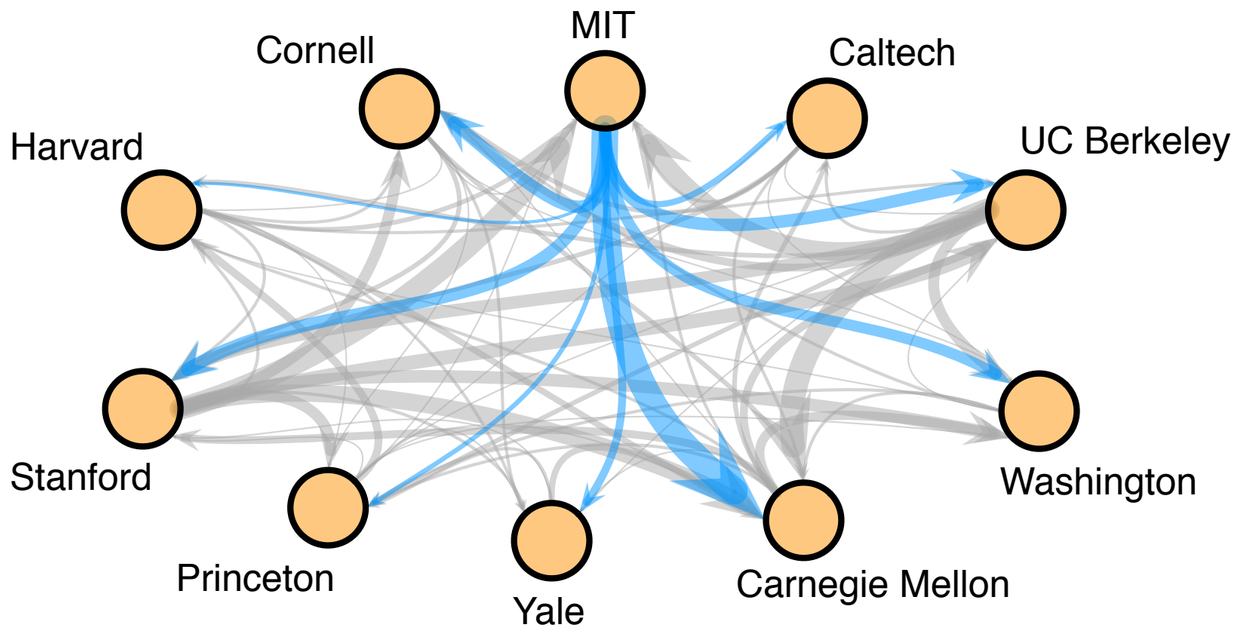
Systems of endorsement

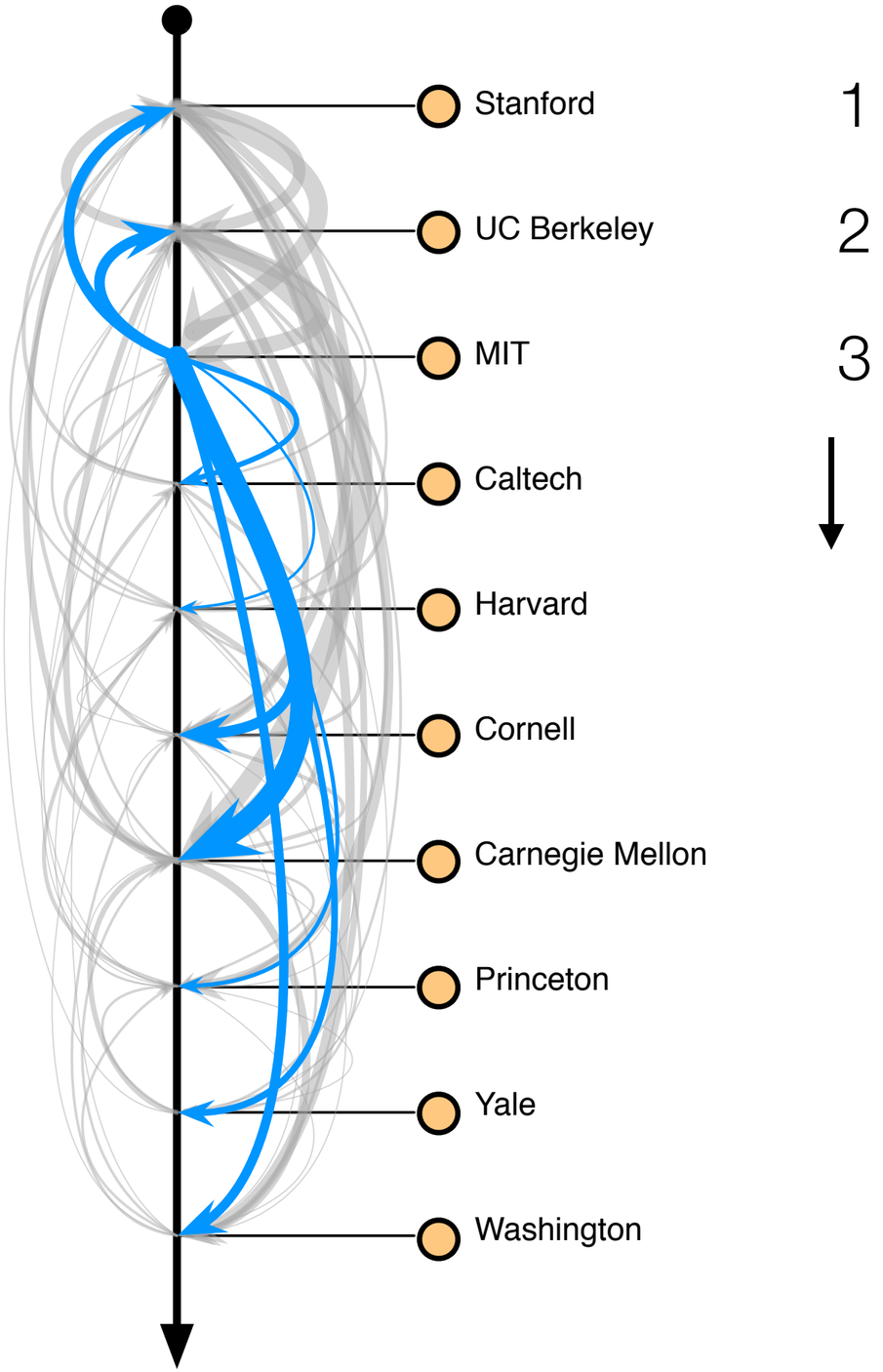




Clauset, Arbesman, Larremore. *Science Advances* 1, e1400005 (2015).

Systems of endorsement



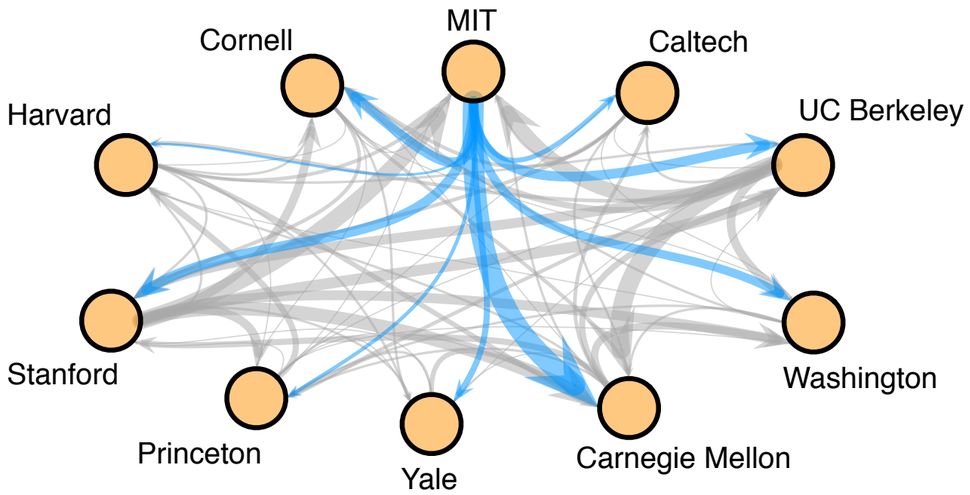


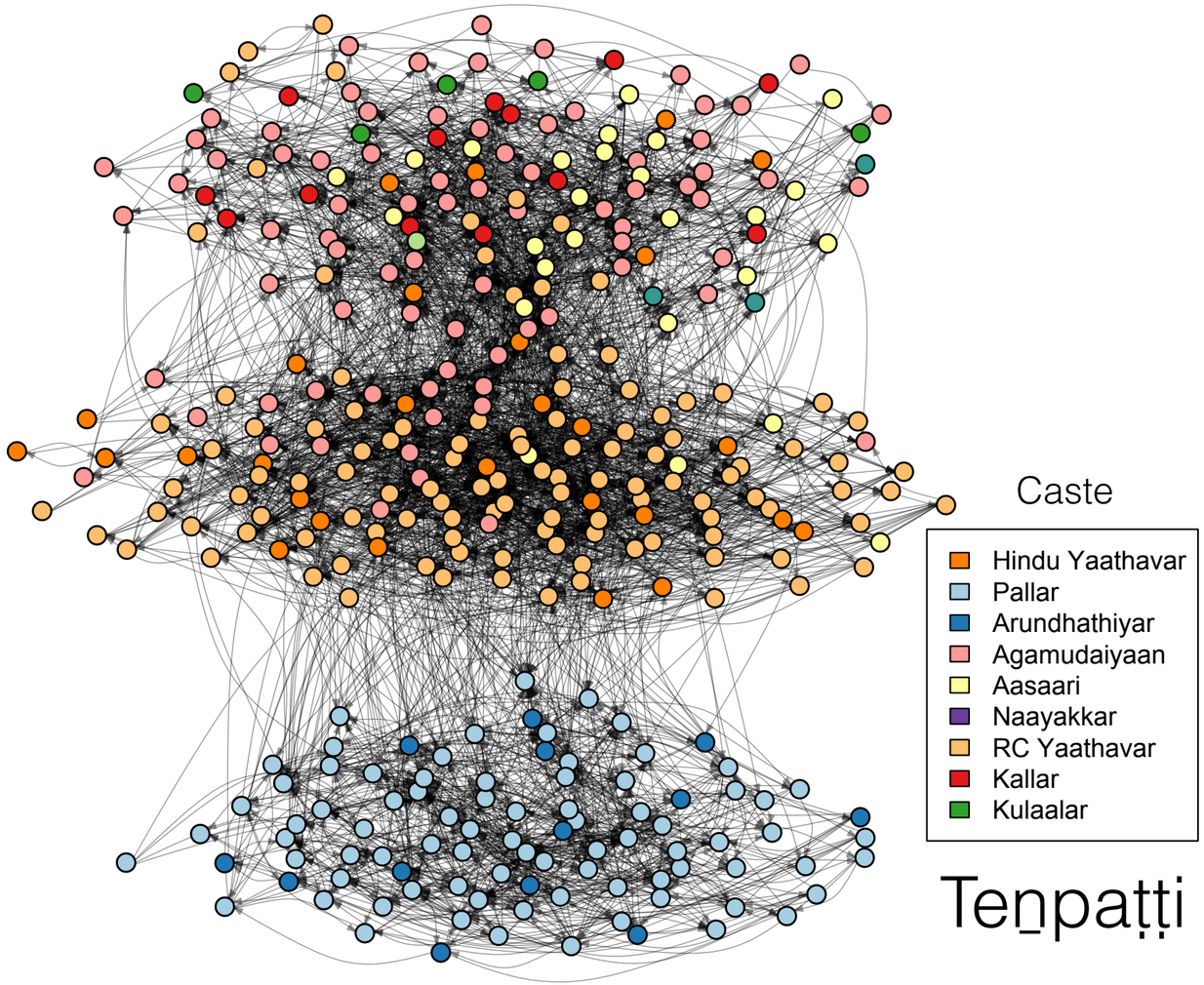
Assumptions:

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.

Clauset, Arbesman, Larremore. *Science Advances* (2015). Fig: Larremore, Hébert-Dufresne, Power. *Draft.* Data: Power. *Nature Human Behaviour* 1, 0057 (2017)

Win-Loss is not satisfactory: schedule matters

Beating the grandmaster counts for more than beating a novice.

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.**

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.**

Ranking methods: many competing ideas!

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.**

Ranking methods: many competing ideas!

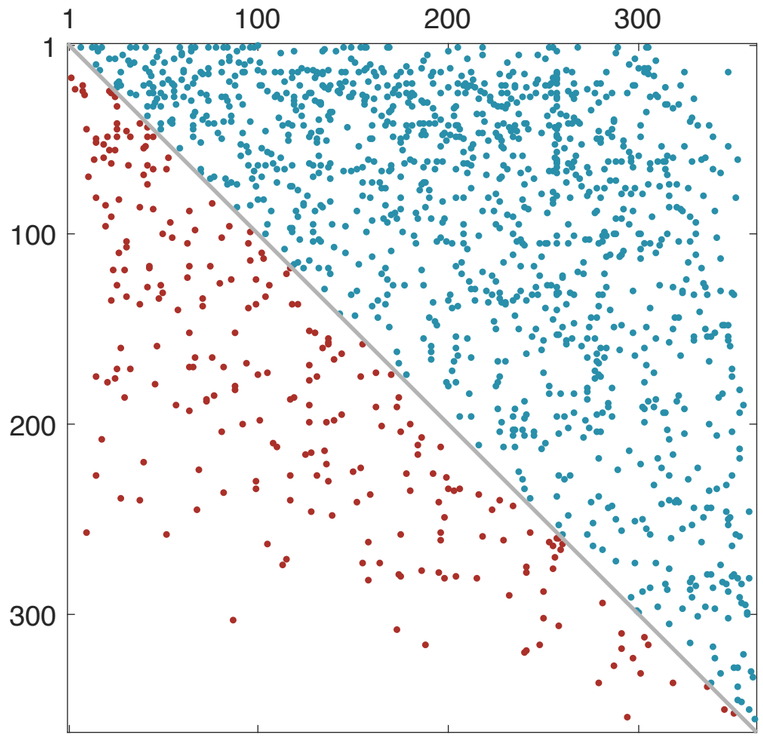
Ordinal rankings (order nodes; integer values)

Embeddings (place nodes; continuous values)

Ranking methods: many competing ideas!

Minimum Violations Rank finds an order of nodes to minimize “upsets”:

- No ties (but see Agony methods), many minima, and provably NP hard.



minimum violation ranking: sort *A.*

Ranking methods: many competing ideas!

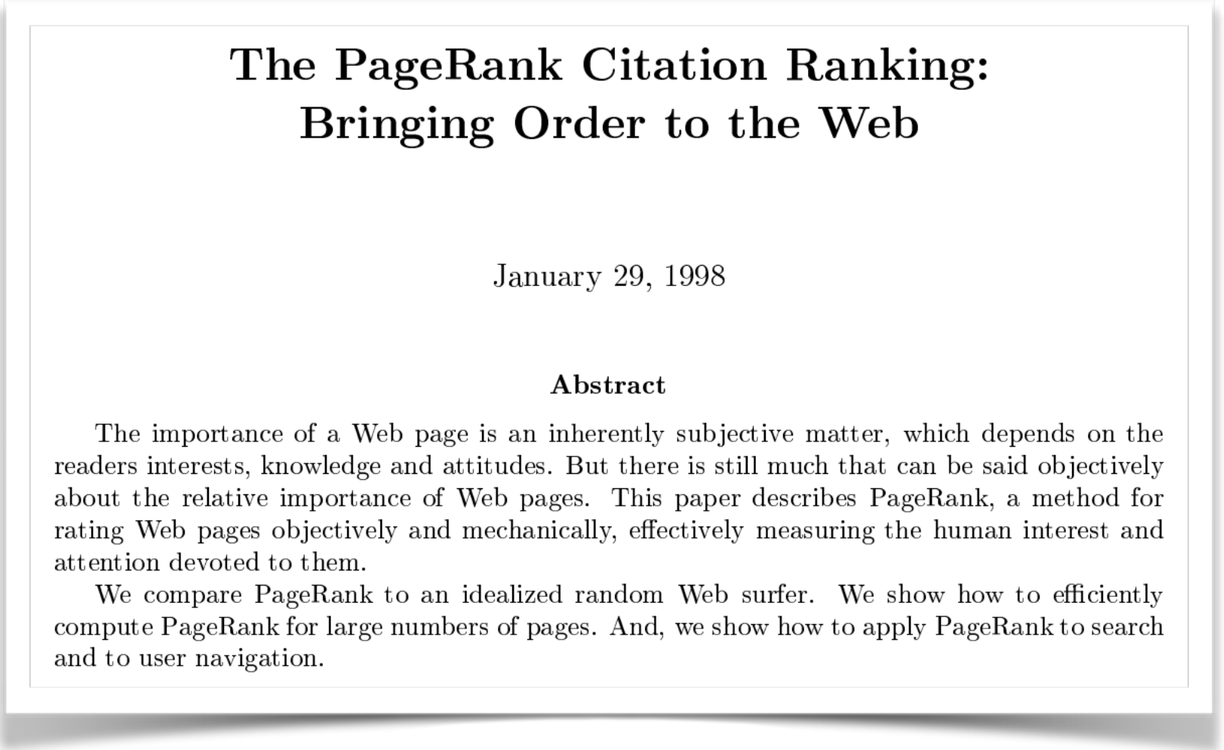
Minimum Violations Rank finds an order of nodes to minimize “upsets”:

- No ties (but see Agony methods), many minima, and provably NP hard.

PageRank defines scalar rank recursively:

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

- Great at finding the top 3 but low interpretability of 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.

Ranking methods: many competing ideas!

Minimum Violations Rank finds an order of nodes to minimize “upsets”:

- No ties (but see Agony methods), many minima, and provably NP hard.

PageRank defines scalar rank recursively:

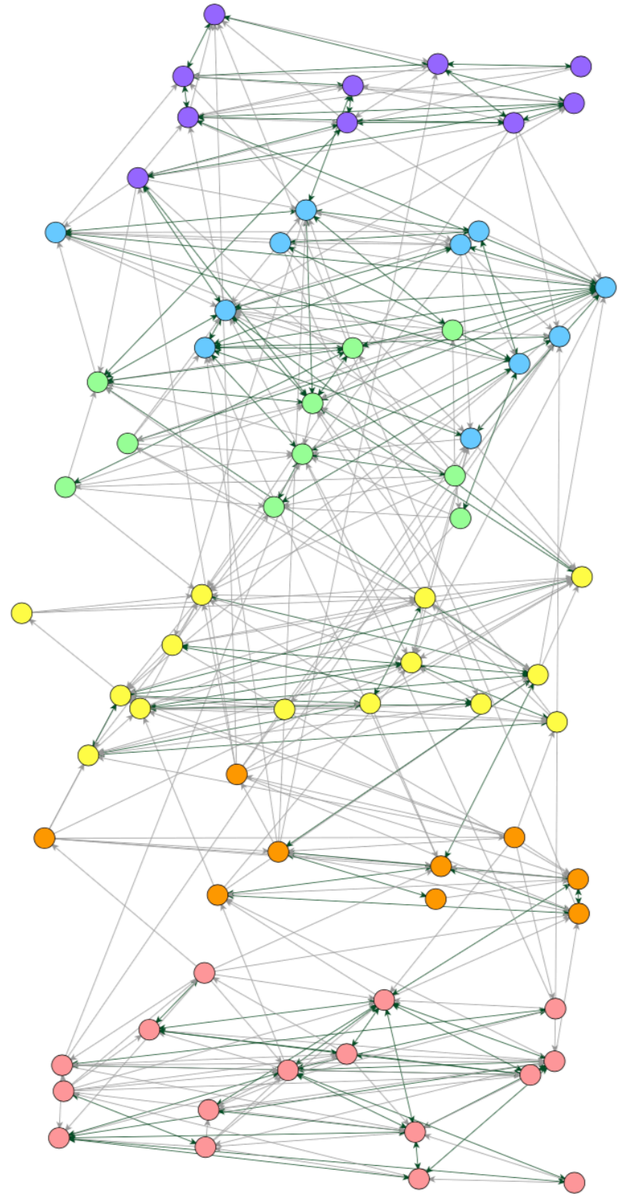
important pages are those that are linked to by important pages.

- Great at finding the top 3 but low interpretability of the PageRank scores.

Ball & Newman inferred ordinal rank via random graph + ranks model:

infer parameters of people’s attachment preferences & ranks.

- Found that in AddHealth data, teens link to others of *nearby* social status.



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

Ranking methods: many competing ideas!

Minimum Violations Rank finds an order of nodes to minimize "upsets":

- No ties (but see Agony methods), many minima, and provably NP hard.

PageRank defines scalar rank recursively:

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

- Great at finding the top 3 but low interpretability of the PageRank scores.

Ball & Newman inferred ordinal rank via random graph + ranks model:

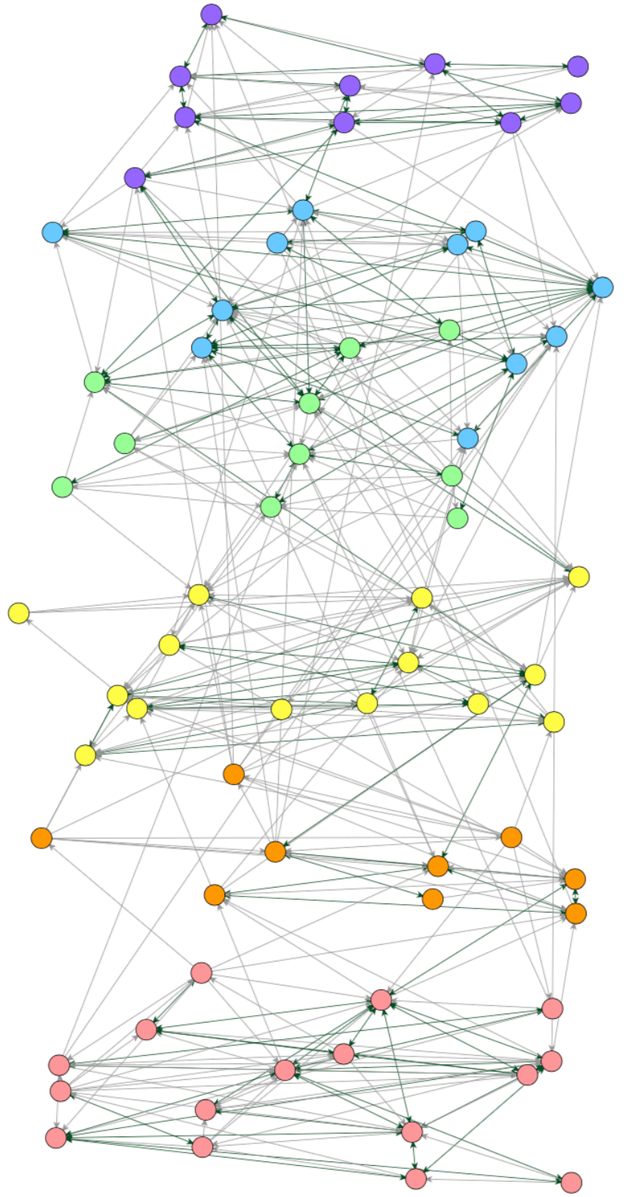
infer parameters of people's attachment preferences & ranks.

- Found that in AddHealth data, teens link to others of *nearby* social status.

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.



Williams & Martinez. Nature 404.6774 (2000). Ball, Newman. Network Science 1, 16-30 (2013)

Ranking methods: many competing ideas!

Minimum Violations Ra
- No ties (but see Ag

PageRank defines scalar
important pages are th
- Great at finding the

Ball & Newman inferred
infer parameters of pe
- Found that in Addh

Niche Models embed sp
most species feed fron
- Great for food web

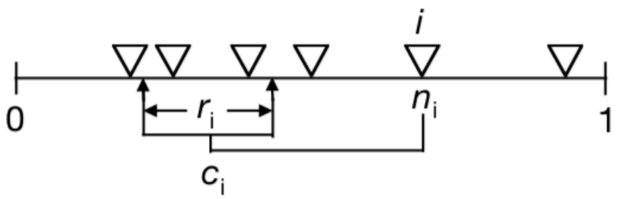
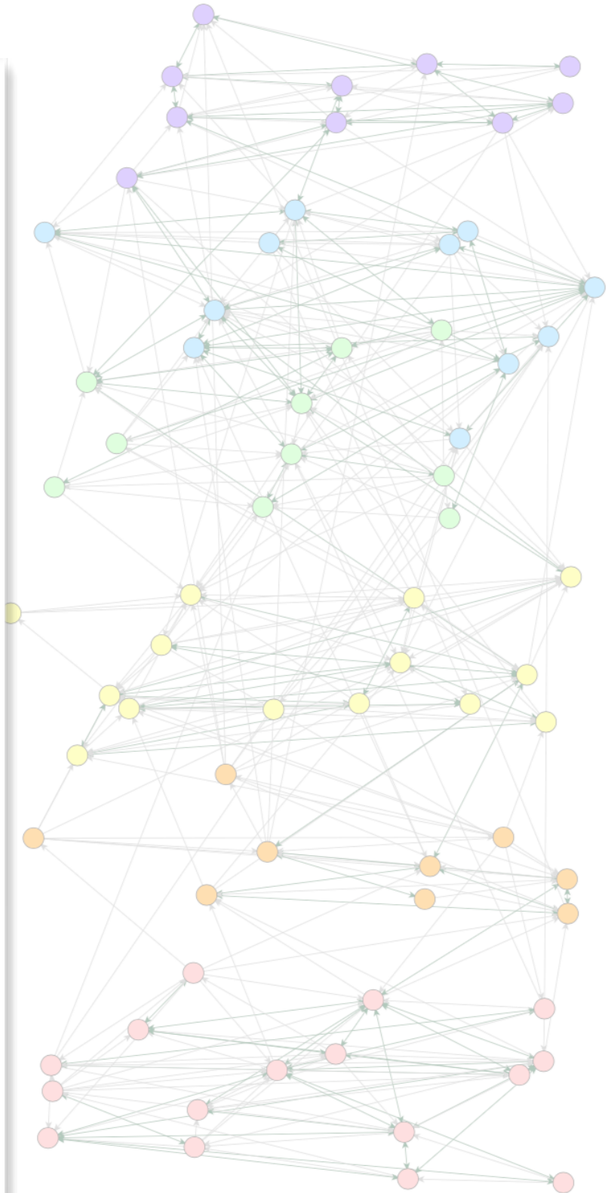


Figure 1 Diagram of the niche model. Each of ***S*** species (for example, ***S*** = 6, each shown as an inverted triangle) is assigned a 'niche value' parameter (n_i) drawn uniformly from the interval [0,1]. Species ***i*** consumes all species falling in a range (r_i) that is placed by uniformly drawing 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). Ball, Newman. Network Science 1, 16-30 (2013)

Ranking methods: many competing ideas!

Minimum Violations Rank finds an order of nodes to minimize "upsets":

- No ties (but see Agony methods), many minima, and provably NP hard.

PageRank defines scalar rank recursively:

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

- Great at finding the top 3 but low interpretability of the PageRank scores.

Ball & Newman inferred ordinal rank via random graph + ranks model:

infer parameters of people's attachment preferences & ranks.

- Found that in AddHealth data, teens link to others of *nearby* social status.

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).

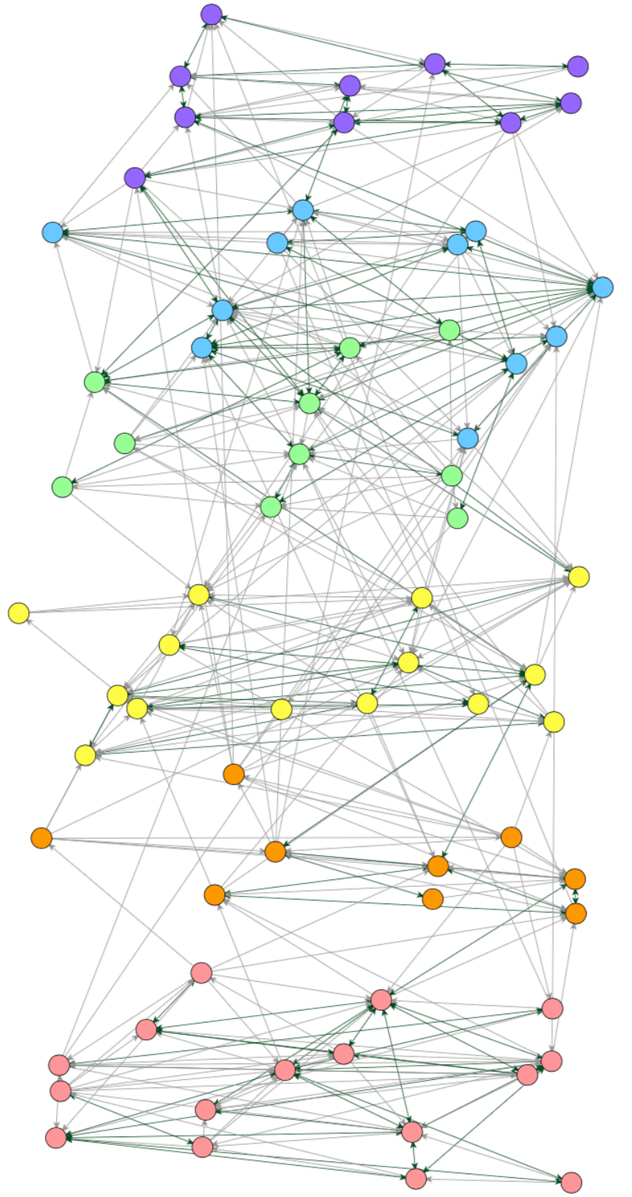
- Great for food webs. Inference models v slow for all but small networks.

Discrete Choice Models (BTL, Thurstone V) embed products in a 1D space.

For BTL, outcome direction is simply:

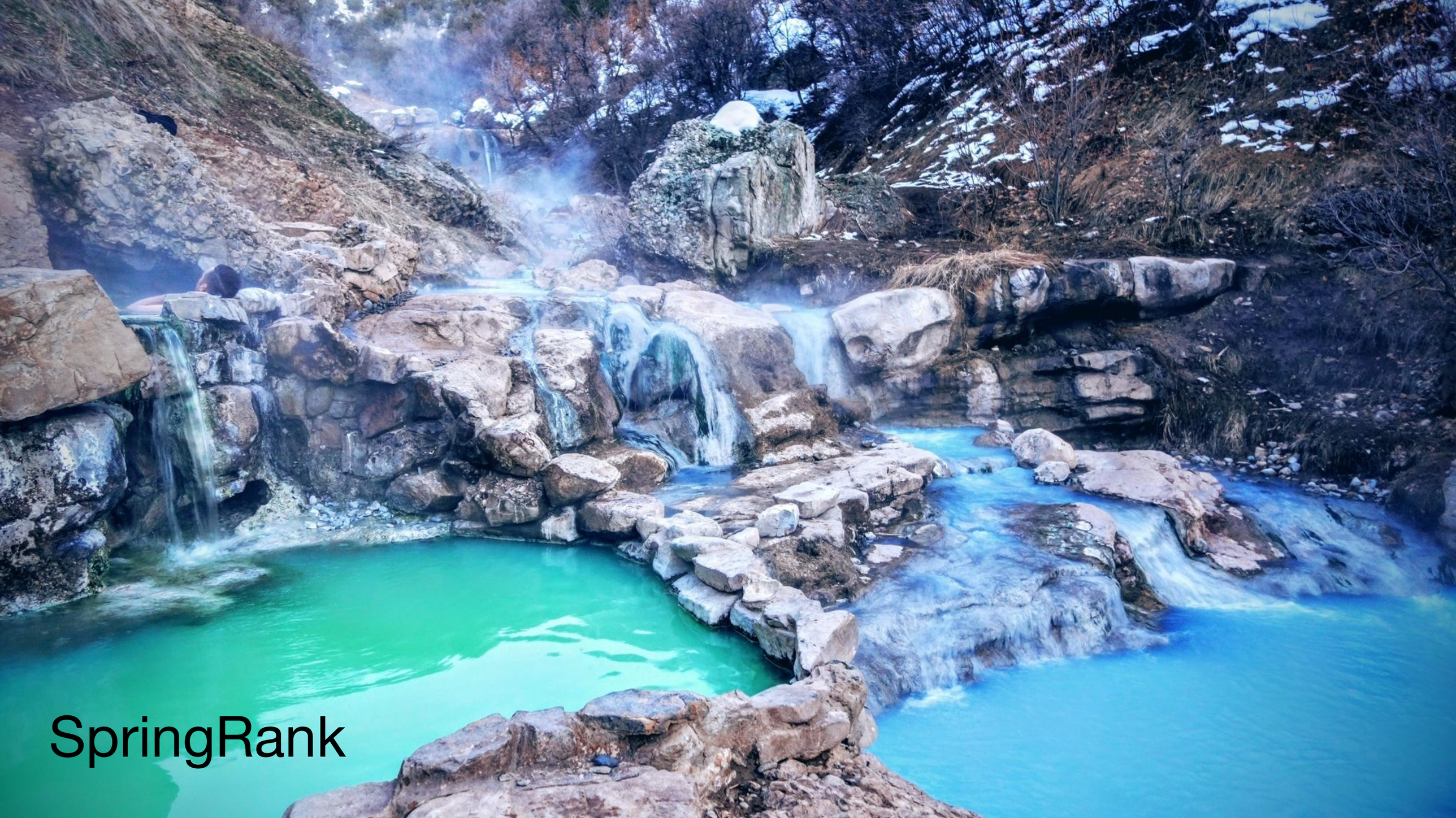
$$P(i \to j) = \frac{\gamma_i}{\gamma_i + \gamma_j}$$

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



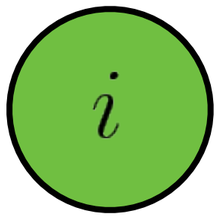
Honestly? Best to google "BTL model." Williams & Martinez. Nature 404.6774 (2000). Ball, Newman. Network Science 1, 16-30 (2013)

The rest of this talk:
SpringRank—fast, interpretable, predictive.

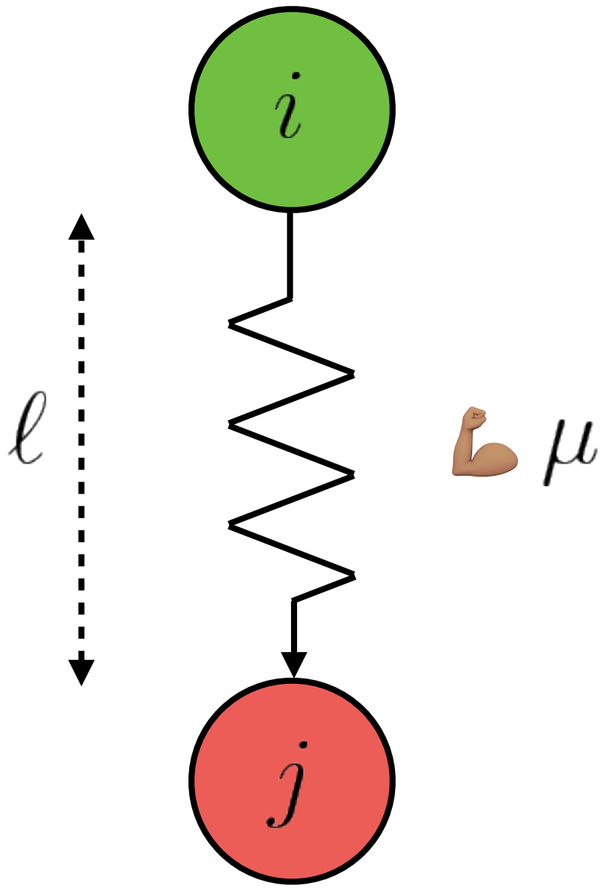

SpringRank

SpringRank

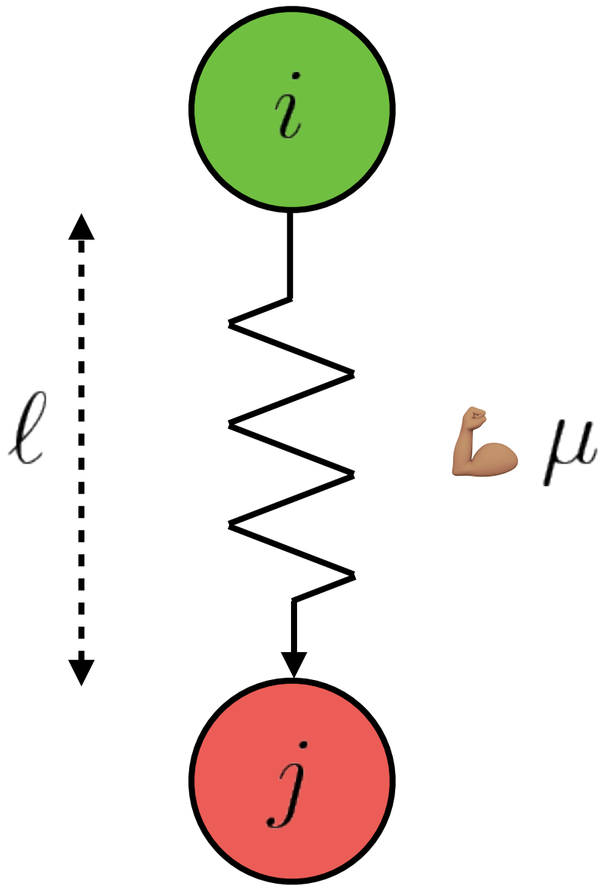
Each directed edge = directed spring

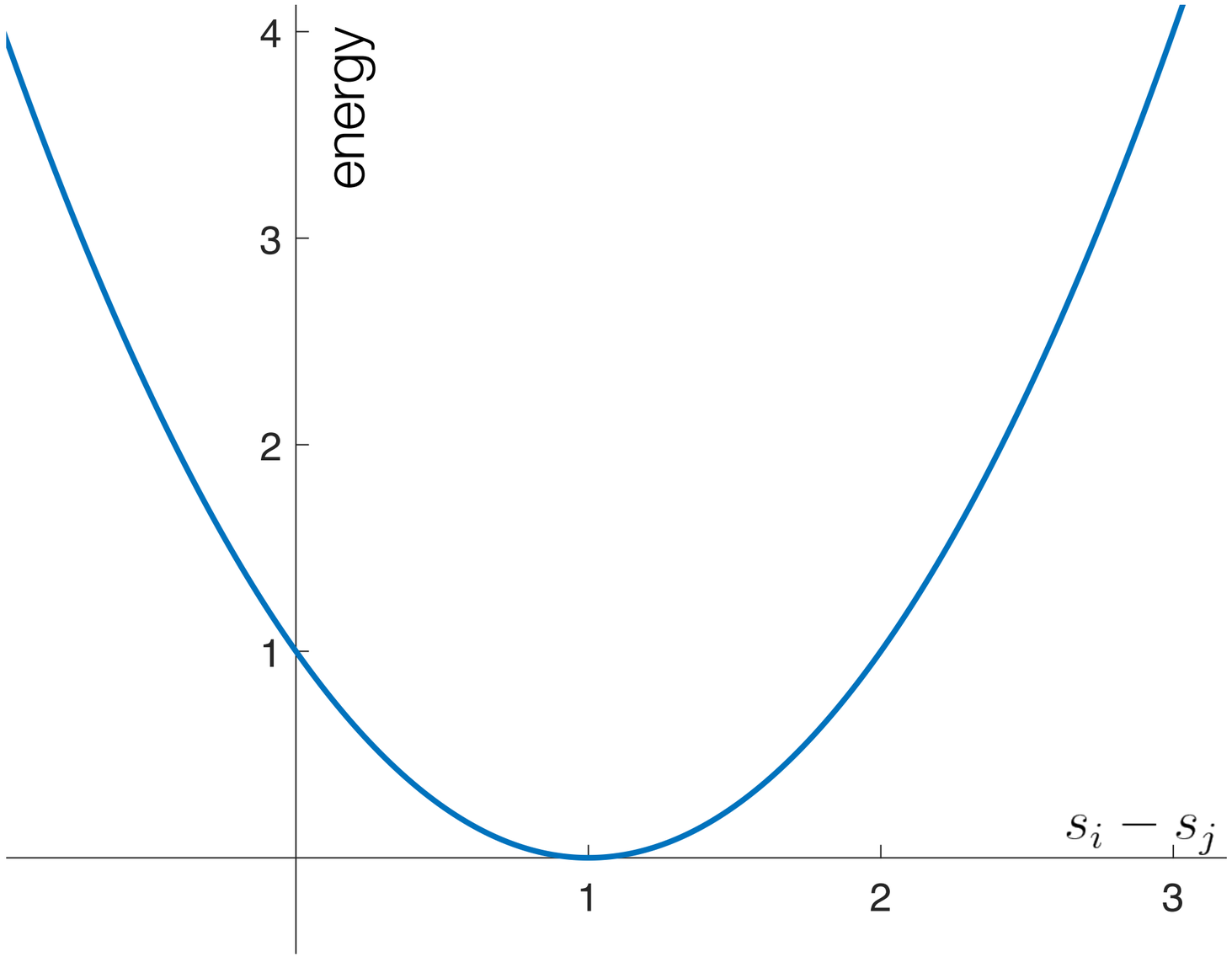


Each directed edge = directed spring

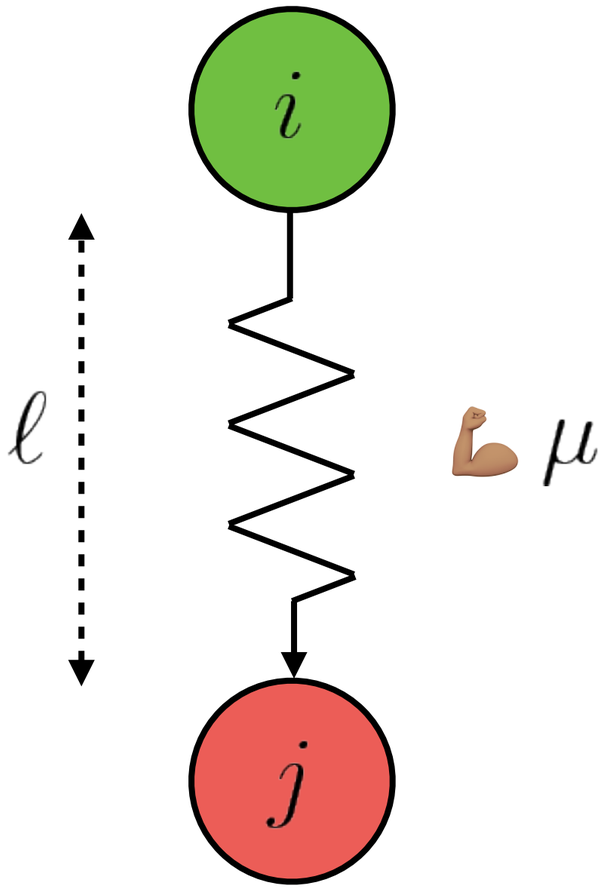


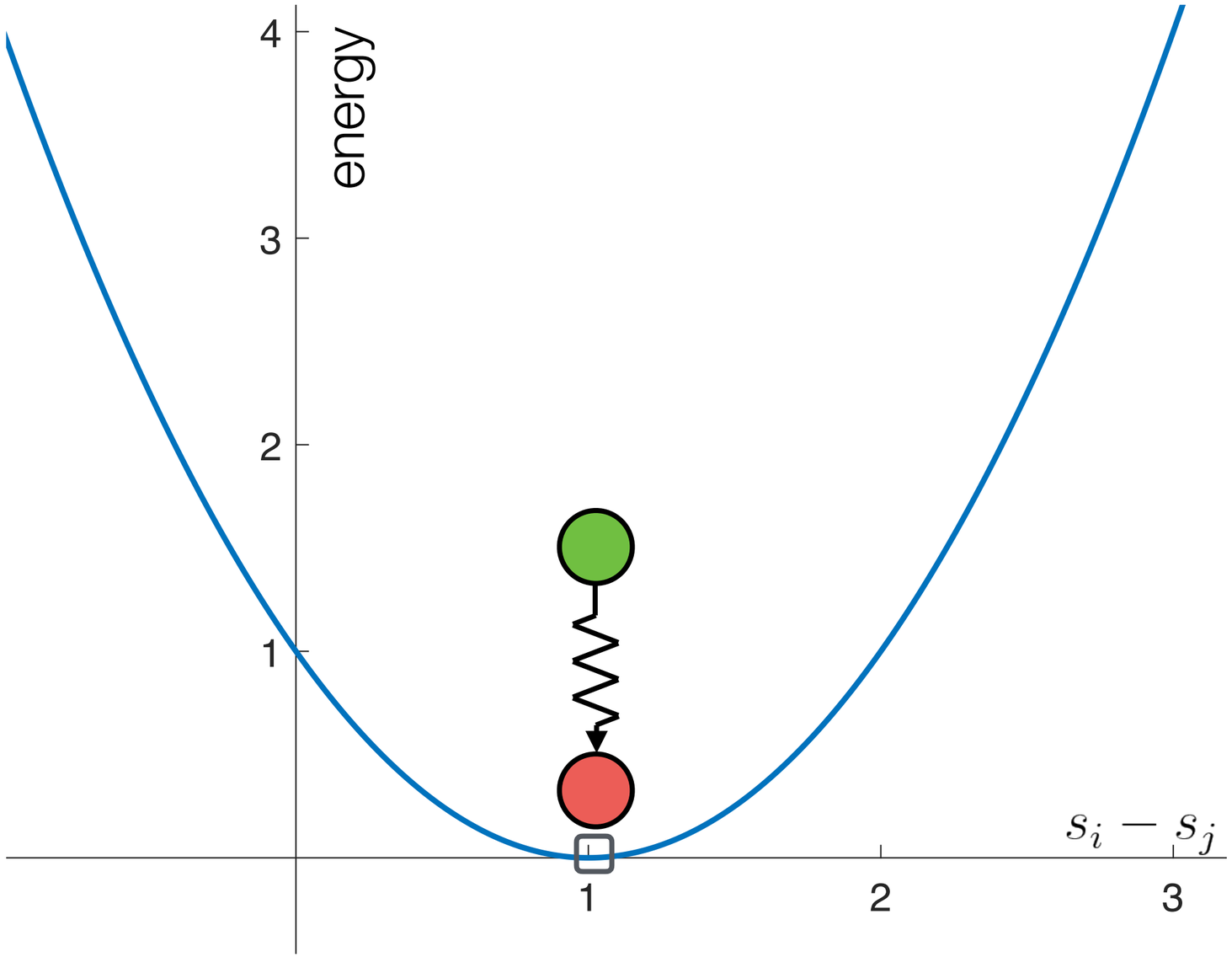
Each directed edge = directed spring



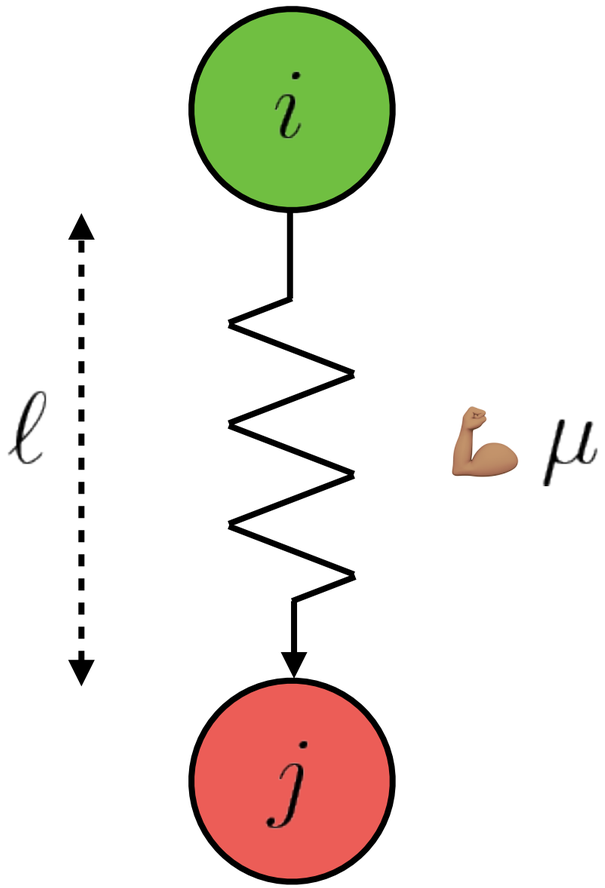


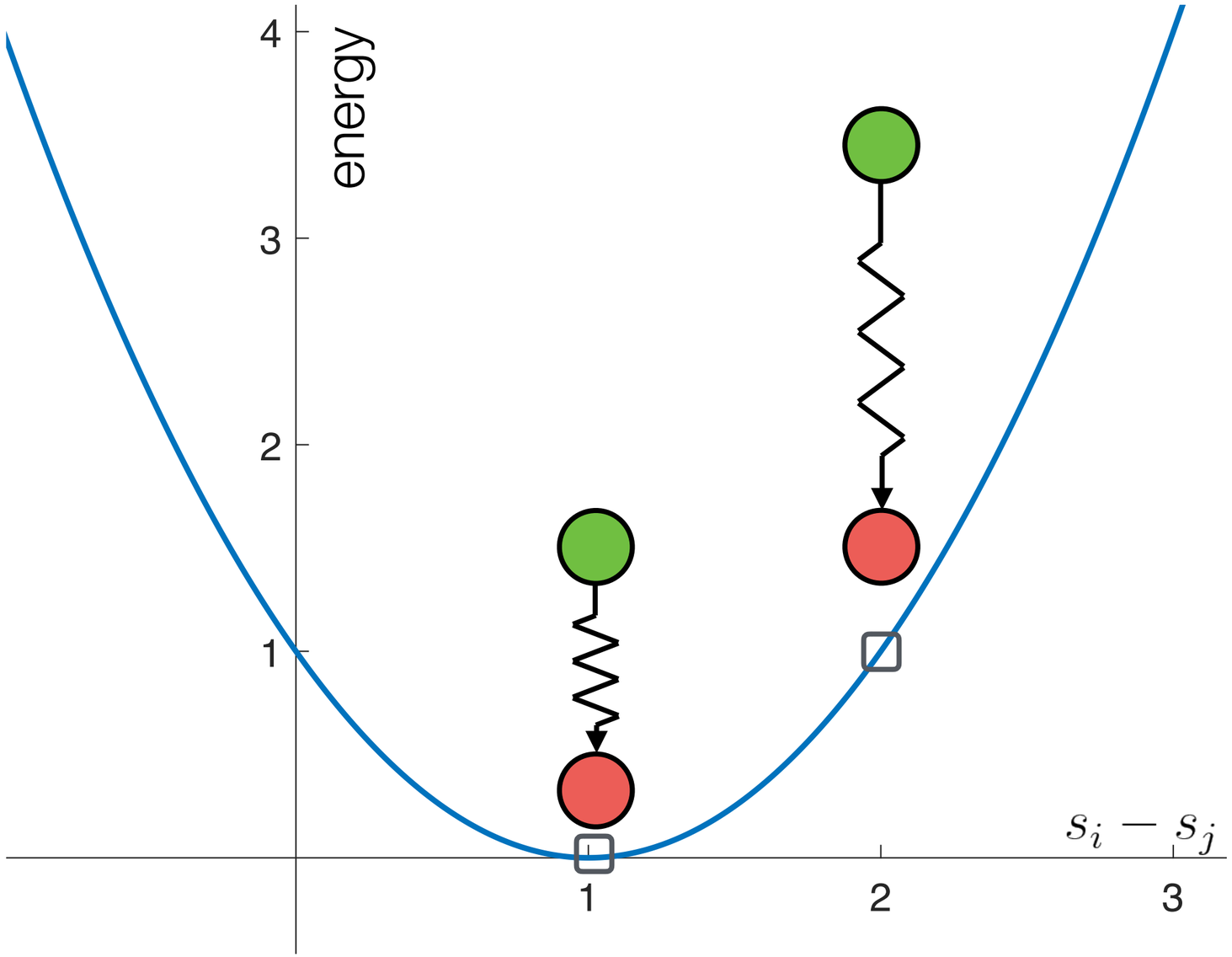
Each directed edge = directed spring



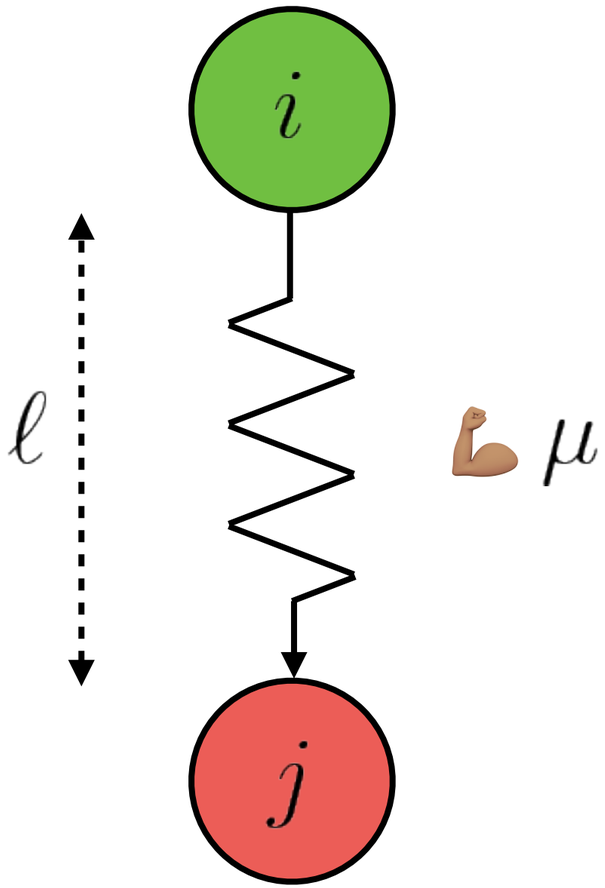


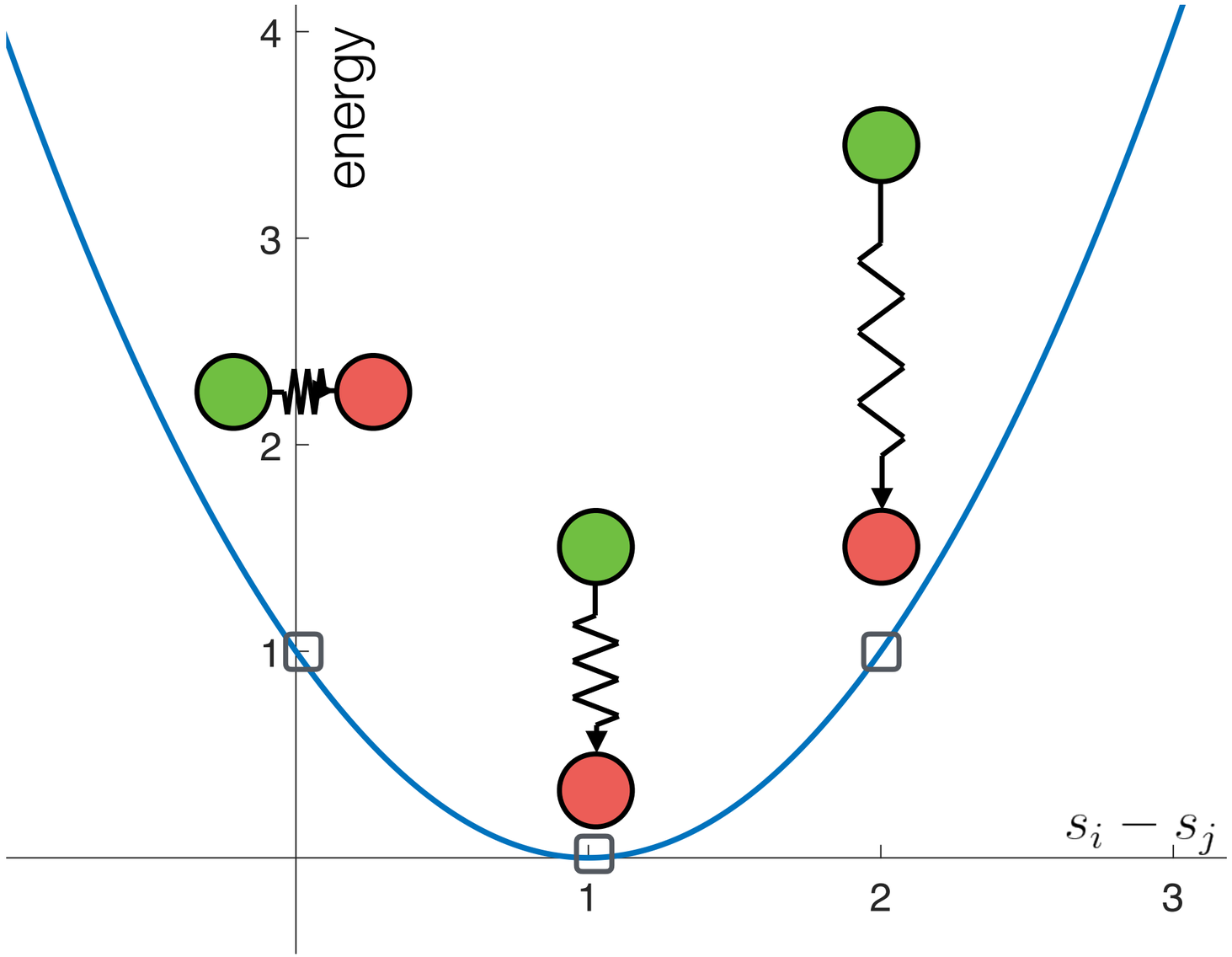
Each directed edge = directed spring



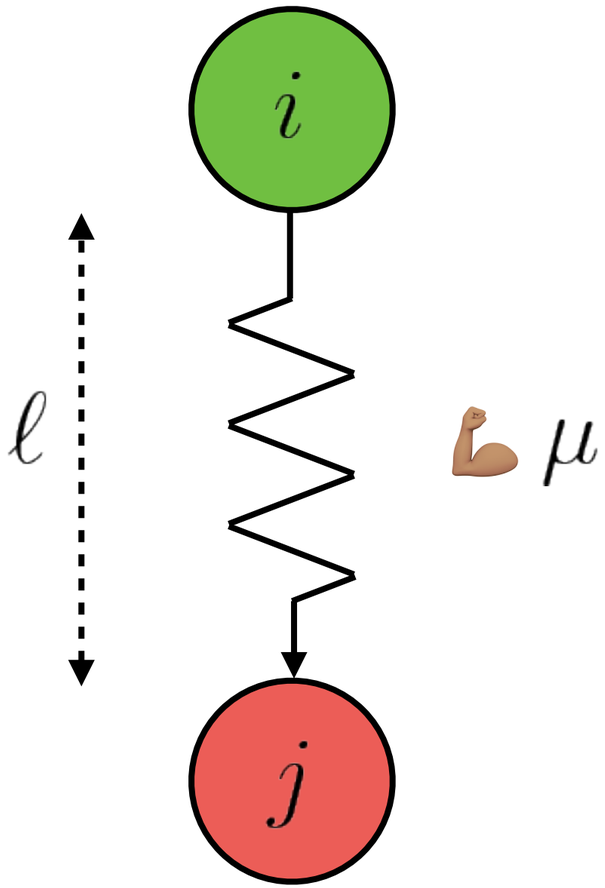


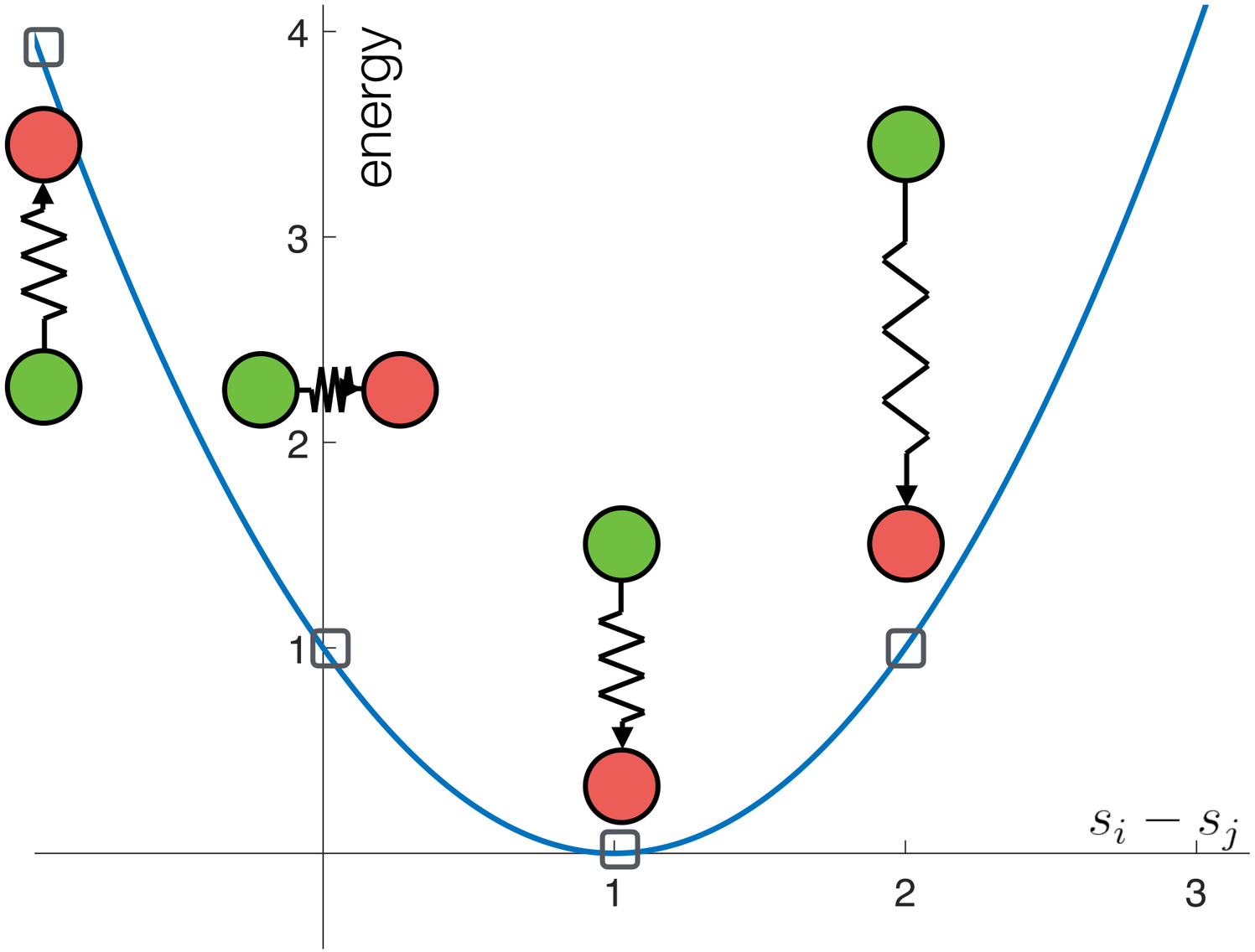
Each directed edge = directed spring





Each directed edge = directed spring





Relax and let the springs decide the ranks

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

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

Because the springs are linear, the potential is quadratic.

Relax and let the springs decide the ranks

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

SpringRank Hamiltonian: energy of the system, given the 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

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

NB: the matrix on the left is also called the *Network Laplacian matrix.*

Derivatives work out nicely

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

Rewrite as a linear algebra problem.

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

NB: the matrix on the left is also called the *Network Laplacian matrix.*

Derivatives work out nicely

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

Rewrite as a linear algebra problem.

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

We know *a priori* that the matrix on the left is singular: translational invariance of H(s).

NB: the matrix on the left is also called the *Network Laplacian matrix.*

Derivatives work out nicely

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

Rewrite as a linear algebra problem.

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

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**]

NB: the matrix on the left is also called the *Network Laplacian matrix.*

Dealing with translational invariance

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

1. Invert this matrix on the *n-1* dimensional subspace orthogonal to **1**. (pseudoinverse)

Dealing with translational invariance

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

1. Invert this matrix on the *n-1* dimensional subspace orthogonal to **1**. (pseudoinverse)

2. Fix a rank. Say s_1=0. Or min(s)=0. Or mean(s)=0.

Dealing with translational invariance

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

1. Invert this matrix on the *n-1* dimensional subspace orthogonal to **1**. (pseudoinverse)

2. Fix a rank. Say s_1=0. Or min(s)=0. Or mean(s)=0.

3. Impose some regularization to prevent wandering.
   You could call it an external field.
   Or an undirected spring attached to the origin.

$$H_{\text{reg}}(s) = H(s) + \frac{\alpha}{2} \sum_{i=1}^{N} s_i^2$$

Dealing with translational invariance

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

1. Invert this matrix on the *n-1* dimensional subspace orthogonal to **1**. (pseudoinverse)

2. Fix a rank. Say s_1=0. Or min(s)=0. Or mean(s)=0.

3. Impose some regularization to prevent wandering.
   You could call it an external field.
   Or an undirected spring attached to the origin.

$$H_{\text{reg}}(s) = H(s) + \frac{\alpha}{2} \sum_{i=1}^{N} s_i^2$$

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

now the eigenvector **1** has eigenvalue α

Dealing with translational invariance

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

1. Invert this matrix on the *n-1* dimensional subspace orthogonal to **1**. (pseudoinverse)

2. Fix a rank. Say s_1=0. Or min(*s*)=0. Or mean(*s*)=0.

3. Impose some regularization to prevent wandering.
   You could call it an external field.
   Or an undirected spring attached to the origin.

$$H_{\text{reg}}(s) = H(s) + \frac{\alpha}{2} \sum_{i=1}^{N} s_i^2$$

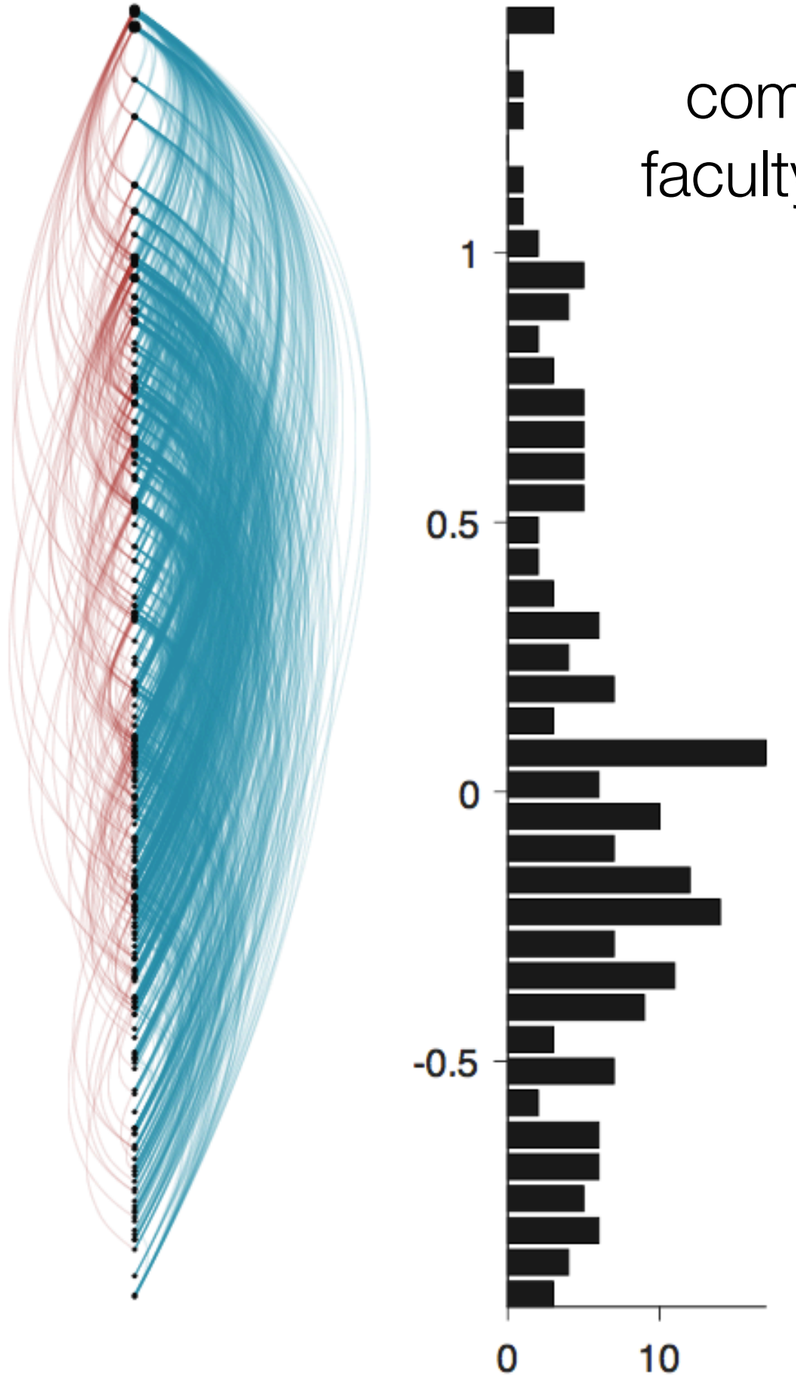
$$\left[D^{\text{out}} + D^{\text{in}} - \left(A + A^T\right) + \alpha I\right] s^* = \left[D^{\text{out}} - D^{\text{in}}\right] \mathbf{1}$$

now the eigenvector **1** has eigenvalue α

$\alpha \to 0$

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

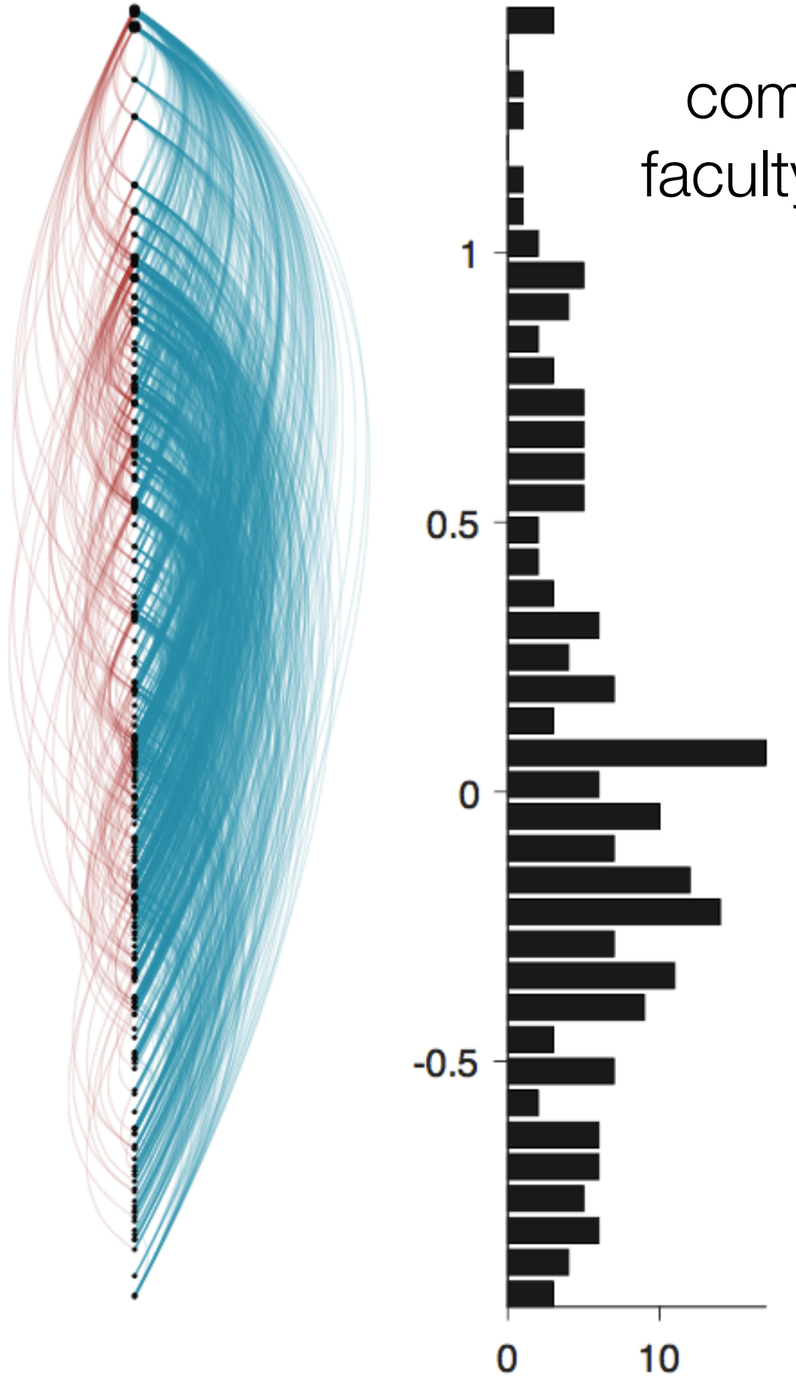
It works!



computer science
faculty hiring network

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.**

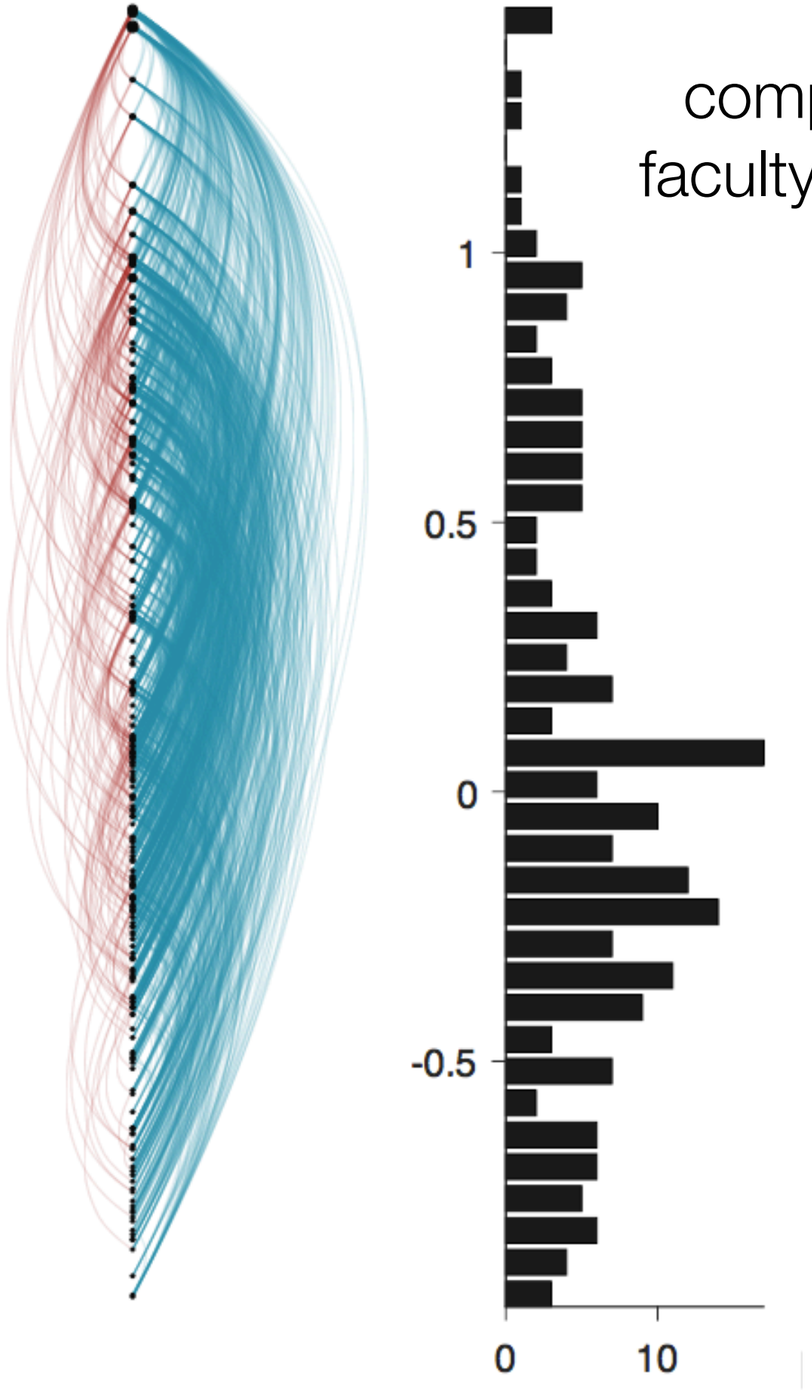


computer science
faculty hiring network

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: Spielman & Teng showed that the solution to these Laplacian systems can be found in *O(edges)* time.]



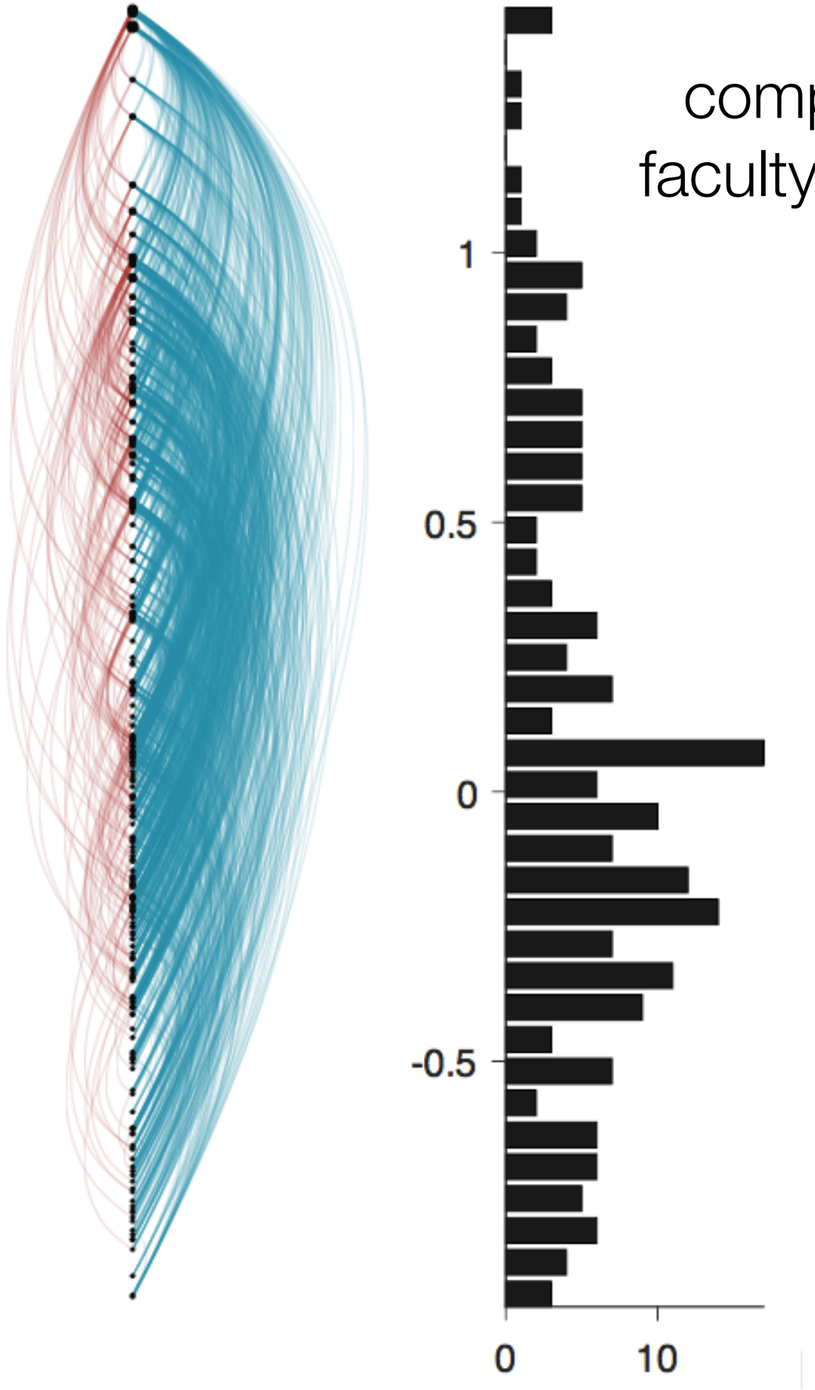
computer science
faculty hiring network

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: Spielman & Teng showed that the solution to these Laplacian systems can be found in *O(edges)* time.]

Note that node positions can be clumpy.



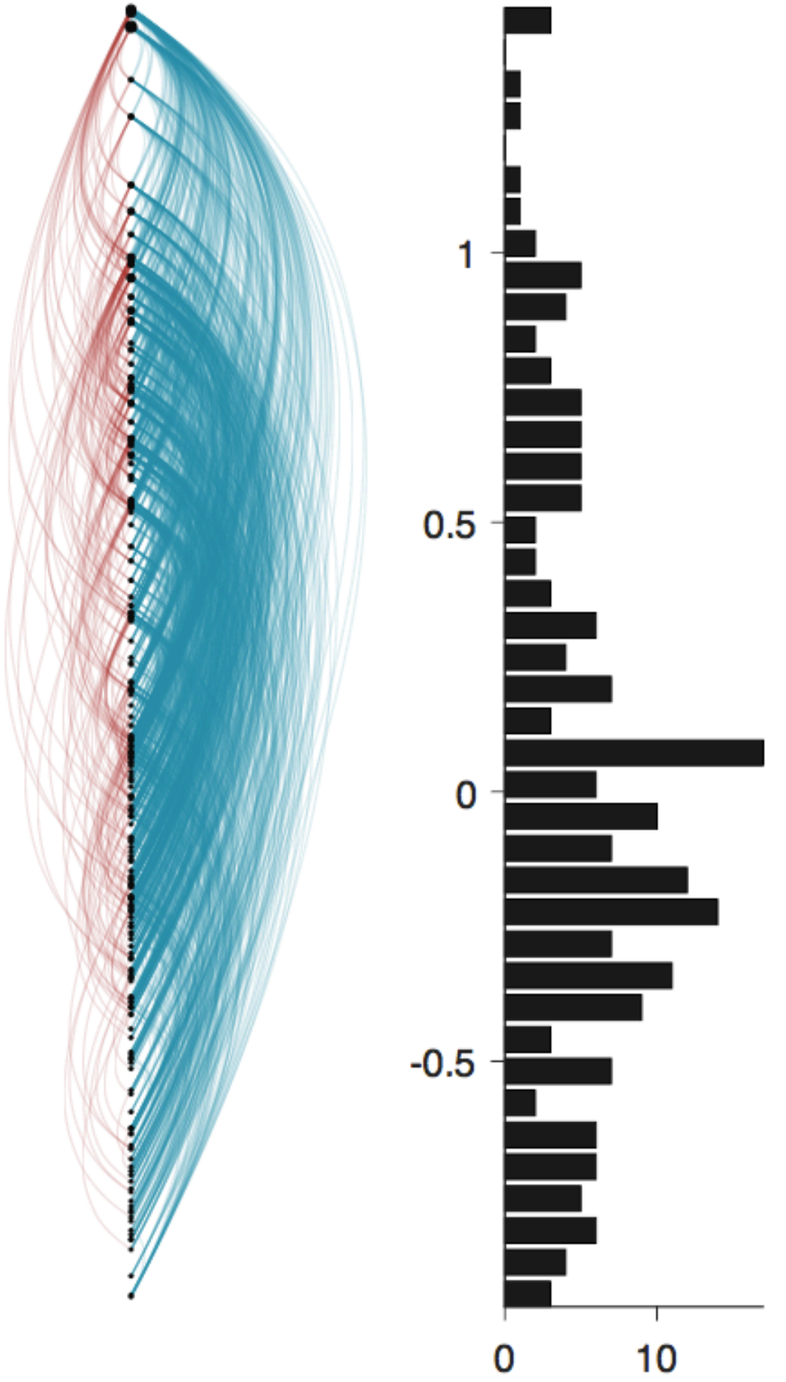
computer science faculty hiring network

How certain are we?

$$P(s) \propto e^{-\beta H(s)}$$

Boltzmann distribution: probability around ranks s

$$P(s) \propto \exp -\frac{\beta}{2} \sum_{i,j=1}^{N} A_{ij} \left(s_i - s_j - \ell\right)^2$$



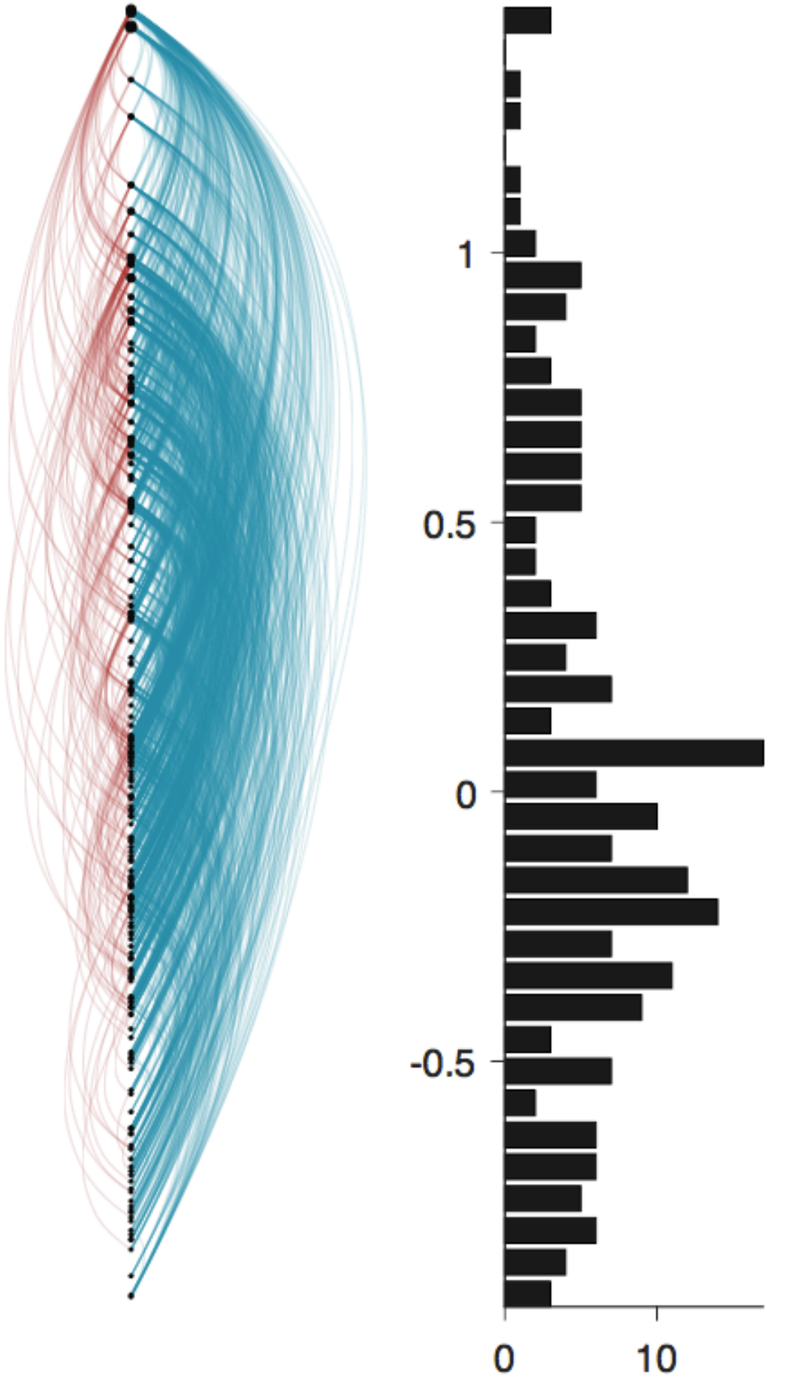
How certain are we?

$$P(s) \propto e^{-\beta H(s)}$$

Boltzmann distribution: probability around ranks s

$$P(s) \propto \exp -\frac{\beta}{2} \sum_{i,j=1}^{N} A_{ij} \left(s_i - s_j - \ell\right)^2$$

consider:
$\beta \to 0$
$\beta \to \infty$



How certain are we?

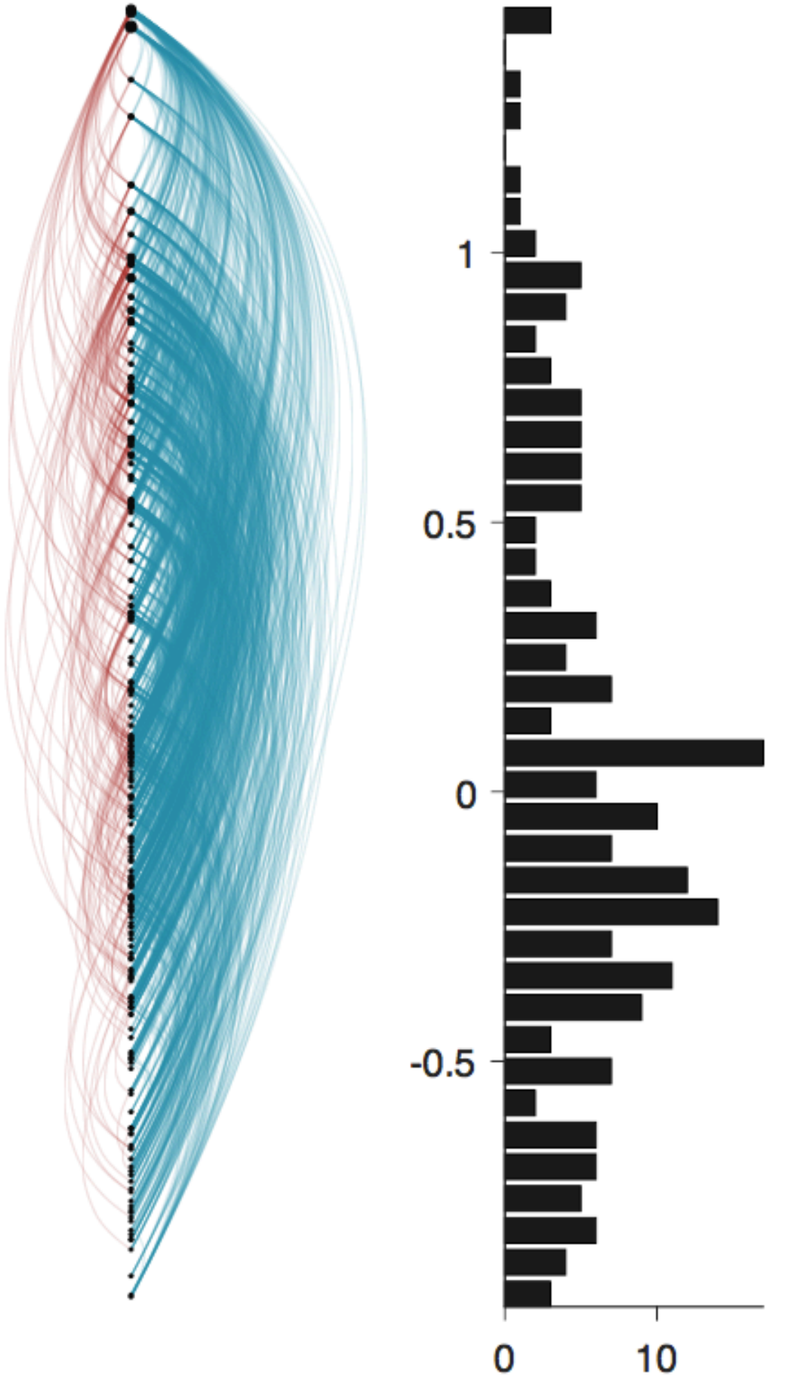
$$P(s) \propto e^{-\beta H(s)}$$

Boltzmann distribution: probability around ranks s

$$P(s) \propto \exp -\frac{\beta}{2} \sum_{i,j=1}^{N} A_{ij} \left(s_i - s_j - \ell\right)^2$$

consider:
$\beta \to 0$
$\beta \to \infty$

$$P(s) \propto \exp\left(-\frac{1}{2}(s - \underline{s})^{\intercal}\Sigma^{-1}(s - \underline{s})\right)$$



How certain are we?

$$P(s) \propto e^{-\beta H(s)}$$

Boltzmann distribution: probability around ranks s

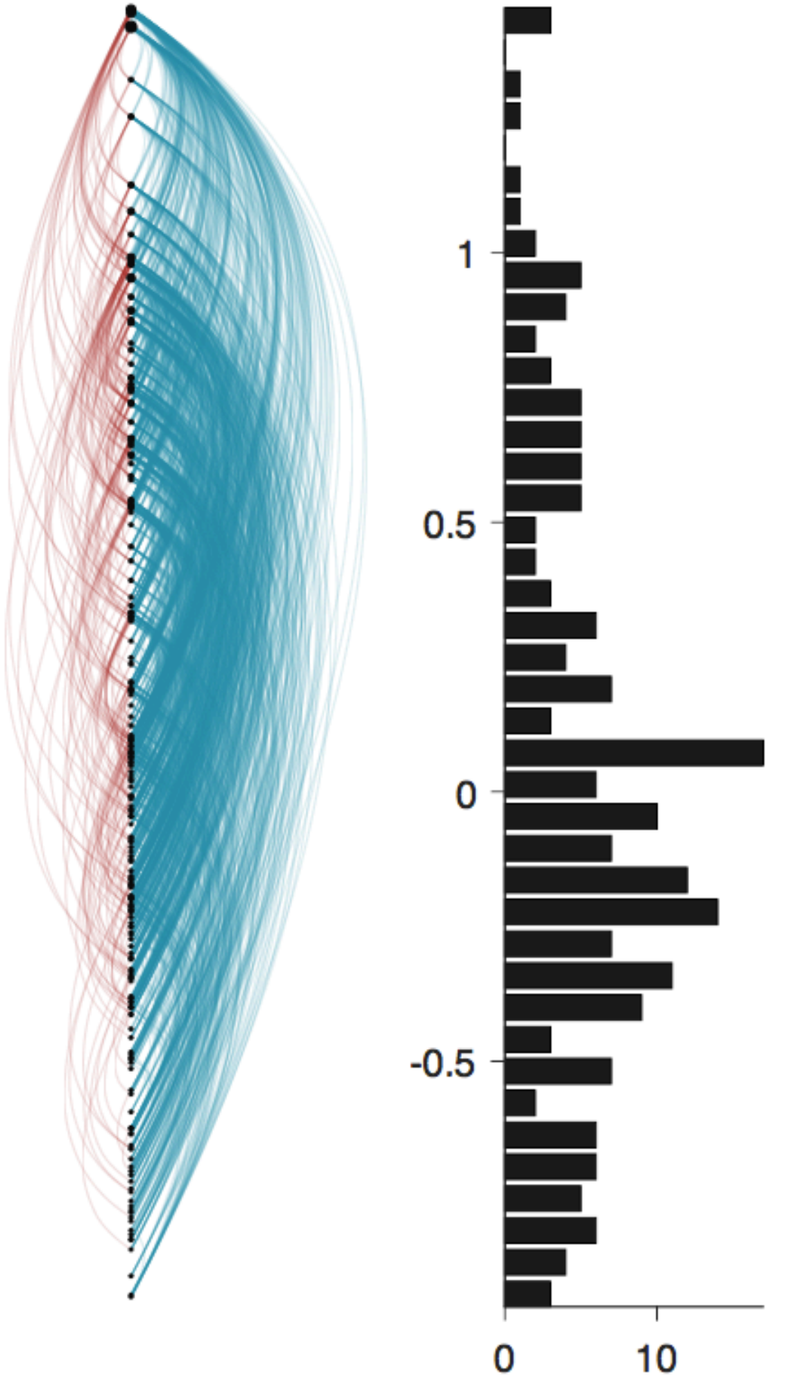
$$P(s) \propto \exp -\frac{\beta}{2} \sum_{i,j=1}^{N} A_{ij} \left(s_i - s_j - \ell\right)^2$$

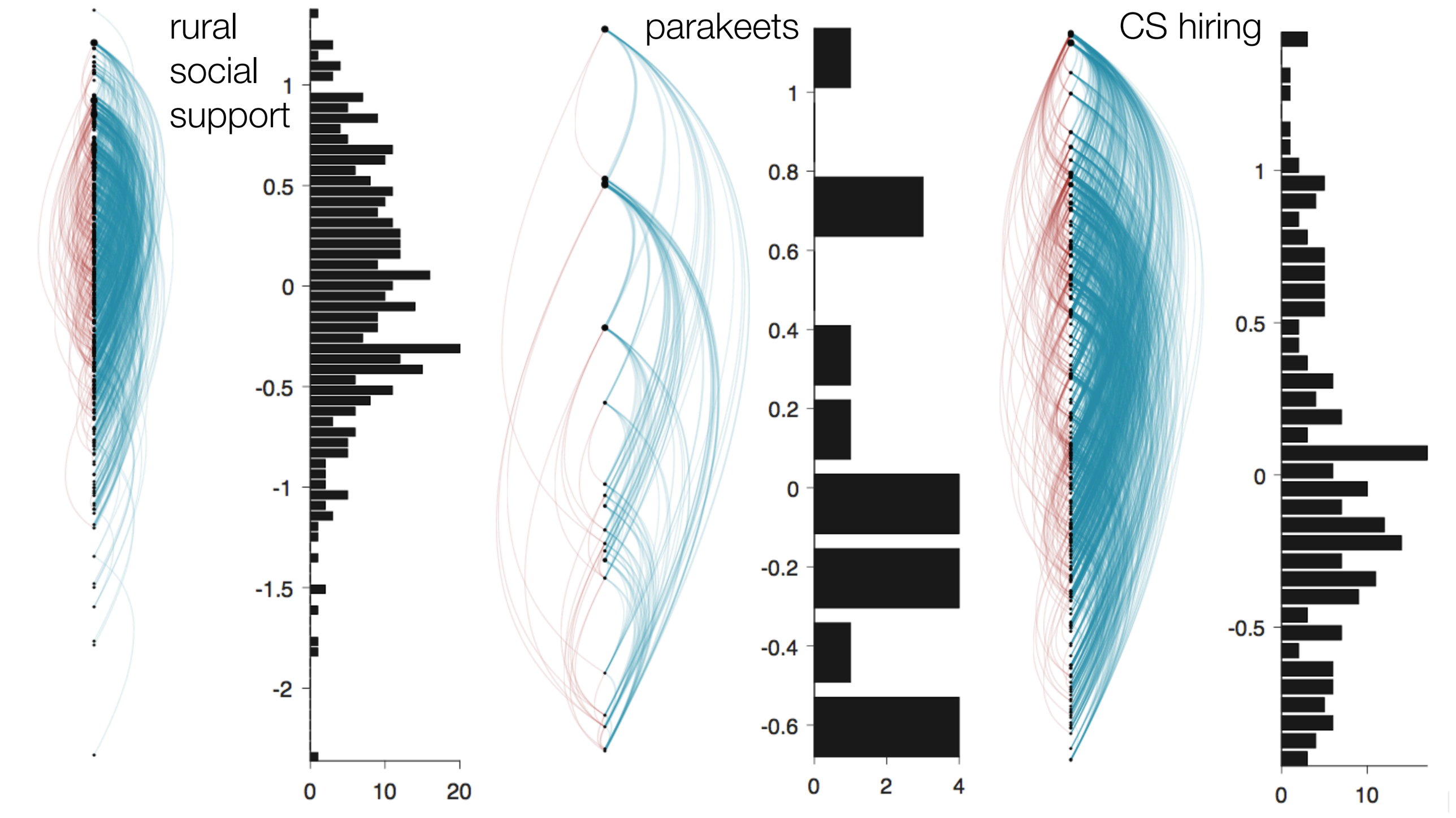
consider:
$\beta \to 0$
$\beta \to \infty$

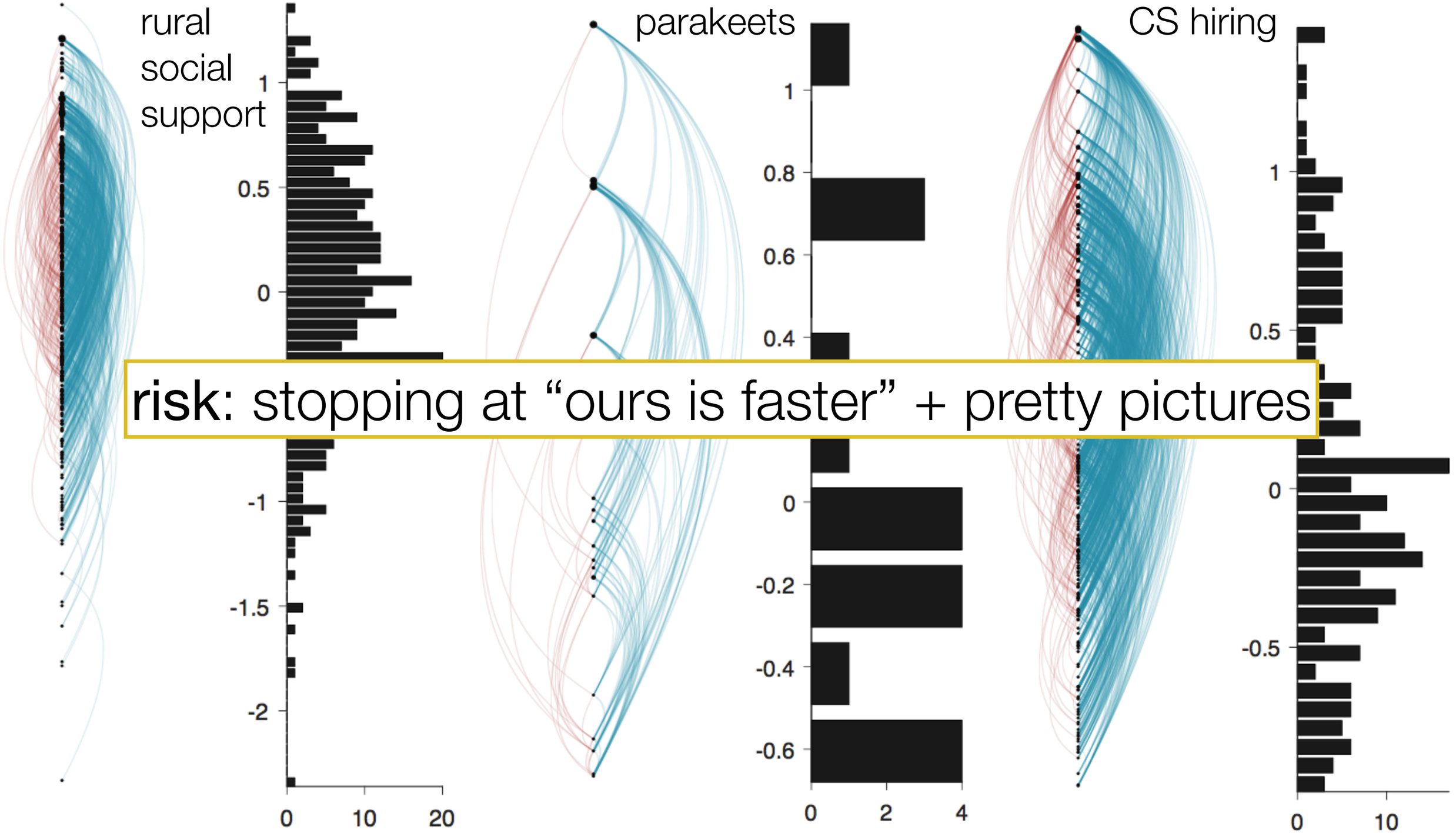
$$P(s) \propto \exp\left(-\frac{1}{2}(s - \underline{s})^{\intercal}\Sigma^{-1}(s - \underline{s})\right)$$

Comparison shows that mean rank is the SR solution,
and ranks are distributed according to n-variate Gaussian!

$\Sigma = \frac{1}{\beta}\left[D^{\text{out}} + D^{\text{in}} - \left(A + A^T + \alpha I\right)\right]^{-1}$ covariance matrix




rural
social
support
1
0.5
0
-0.5
-1
-1.5
-2
0
10
20
parakeets
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
0
2
4
CS hiring
1
0.5
0
-0.5
0
10



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

Generative models

Generative models

Recall assumptions:

1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.

Generative models

Recall assumptions:

1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.

This actually suggests a generative model:

*On the **first** day, the Research Team initialized an n-vector of qualities, s.*

Generative models

Recall assumptions:

1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.

This actually suggests a generative model:

*On the **first** day, the Research Team initialized an n-vector of qualities, s.*

*On the **second** day, a number of interactions occurred, whose outcomes were chosen by the Research Team in order to reflect the qualities, s.*

Generative models

Recall assumptions:

1. Competitors have some intrinsic quality (or vector of qualities).
2. Interactions can (stochastically) reveal differences in qualities.

This actually suggests a generative model:

*On the **first** day, the Research Team initialized an n-vector of qualities, s.*

*On the **second** day, a number of interactions occurred, whose outcomes were chosen by the Research Team in order to reflect the qualities, s.*

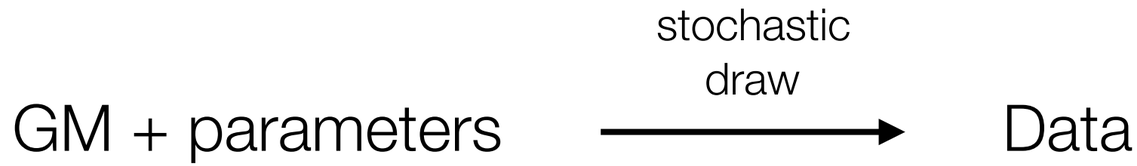
*On the **third** day, these interactions were written down as an adjacency matrix, and the Reviewers saw that it was good.* 😌

Generative models

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

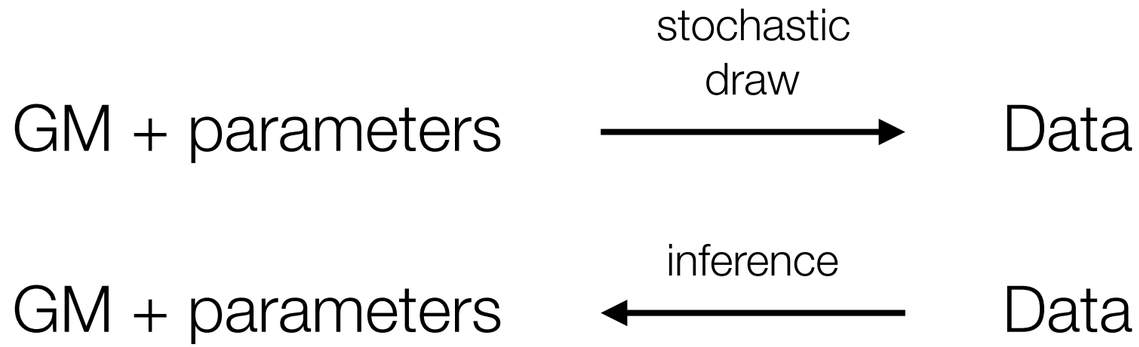
Generative models

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



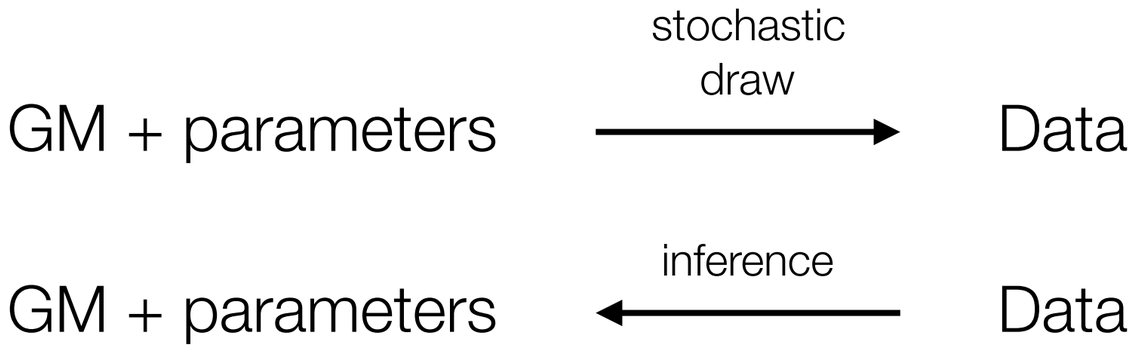
Generative models

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



Generative models

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



Let the expected number of edges from *i* to *j* be:

$$\mathrm{E}[A_{ij}] = c \exp\left[-\frac{1}{2}\beta(s_i - s_j - 1)^2\right]$$

And let the actual number of edges be drawn from a Poisson distribution with that mean.

Parameters…

$$\mathrm{E}[M] = \sum_{i,j}[A_{ij}] = c\sum_{i,j}\exp\left[-\frac{1}{2}\beta(s_i - s_j - 1)^2\right]$$

Thus, the total number of edges is proportional to *c*, meaning *c* controls for sparsity.

β specifies the disorder in the system; the extent to which edges respect the ranks *s*.

Parameters…

$$\mathrm{E}[M] = \sum_{i,j}[A_{ij}] = c\sum_{i,j}\exp\left[-\frac{1}{2}\beta(s_i - s_j - 1)^2\right]$$

Thus, the total number of edges is proportional to *c*, meaning *c* controls for sparsity.

β specifies the disorder in the system; the extent to which edges respect the ranks *s*.

The likelihood of observing/generating data *A*, given the parameters is:

$$P(A \mid s, \beta, c) = \prod_{i,j} \frac{\left[ce^{-\frac{\beta}{2}(s_i - s_j - 1)^2}\right]^{A_{ij}}}{A_{ij}!} \exp\left[-ce^{-\frac{\beta}{2}(s_i - s_j - 1)^2}\right]$$

Maximize the likelihood to find the ranks!

the usual tricks:

maximize the *log*; take derivatives & set to zero; chuck out additive constants.

$$\mathcal{L}(A \mid s, \beta) = -\beta H(s) - M \log \left[\sum_{i,j} e^{-\frac{\beta}{2}(s_i - s_j - 1)^2} \right]$$

Maximize the likelihood to find the ranks!

the usual tricks:

maximize the *log*; take derivatives & set to zero; chuck out additive constants.

$$\mathcal{L}(A \mid s, \beta) = -\beta H(s) - M \log \left[\sum_{i,j} e^{-\frac{\beta}{2}(s_i - s_j - 1)^2} \right]$$

For low temperature (β large) or a sparse network (M small) ,
the ML ranks *are equal to* the ranks minimizing the SR Hamiltonian!

Maximize the likelihood to find the ranks!

the usual tricks:

maximize the *log*; take derivatives & set to zero; chuck out additive constants.

$$\mathcal{L}(A \mid s, \beta) = -\beta H(s) - M \log \left[\sum_{i,j} e^{-\frac{\beta}{2}(s_i - s_j - 1)^2} \right]$$

For low temperature (β large) or a sparse network (M small) ,
the ML ranks *are equal to* the ranks minimizing the SR Hamiltonian!

In other words: solving SpringRank is asymptotically equivalent to inferring *s* using the GM.

What about priors?

$$P(s) \propto \prod_{i=1}^{N} e^{-\frac{\alpha\beta}{2} s_i^2}$$

For example, we believe the ranks were drawn from a Gaussian.

What about priors?

$$P(s) \propto \prod_{i=1}^{N} e^{-\frac{\alpha\beta}{2} s_i^2}$$

For example, we believe the ranks were drawn from a Gaussian.

Then the **maximum** a posteriori estimate, asymptotically, becomes the **minimizer** of

$$H_{\text{reg}}(s) = H(s) + \frac{\alpha}{2} \sum_{i=1}^{N} s_i^2$$

What about priors?

$$P(s) \propto \prod_{i=1}^{N} e^{-\frac{\alpha\beta}{2} s_i^2}$$

For example, we believe the ranks were drawn from a Gaussian.

Then the **maximum** a posteriori estimate, asymptotically, becomes the **minimizer** of

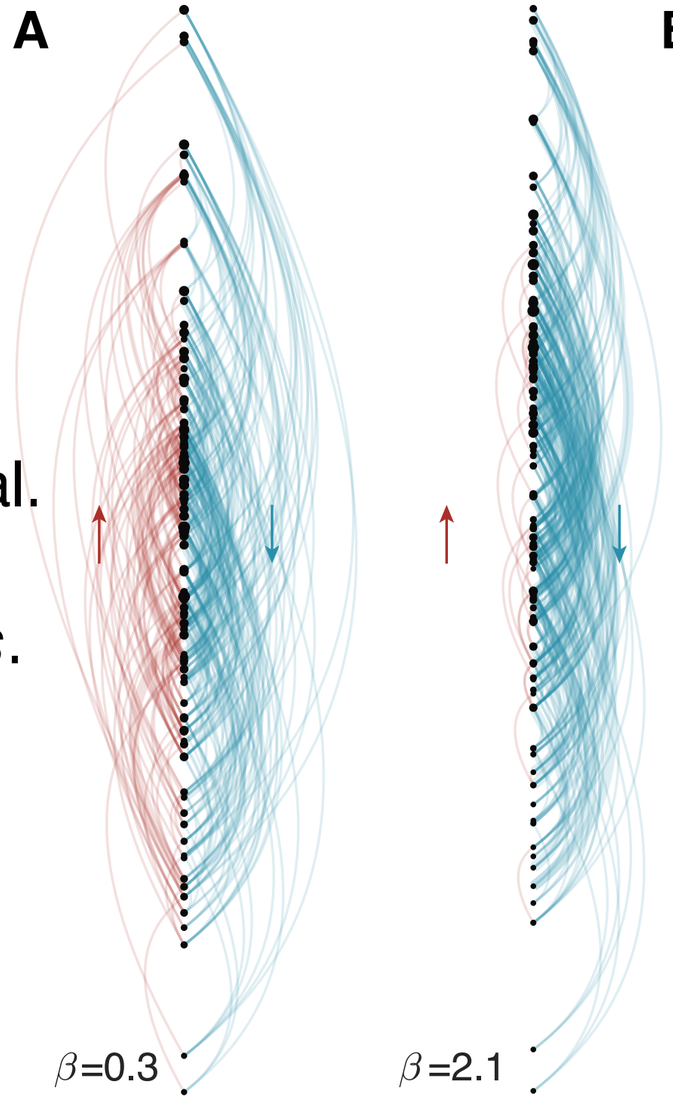
$$H_{\text{reg}}(s) = H(s) + \frac{\alpha}{2} \sum_{i=1}^{N} s_i^2$$

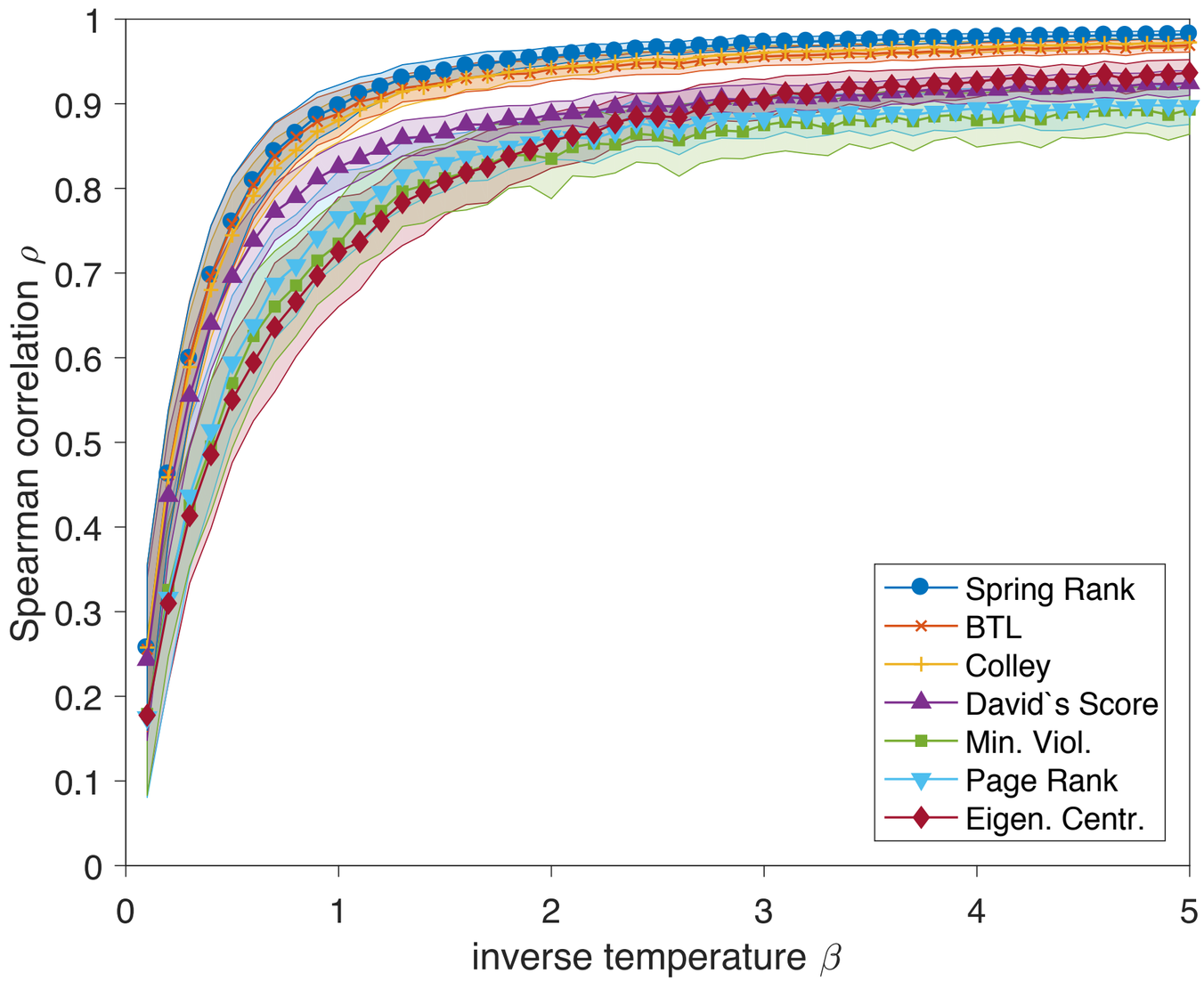
Put differently: a Gaussian prior over *s* corresponds to the regularization from before!

Consistency: Test SpringRank using GM synthetic data

Easy:

1. Plant *s* from **standard normal.**
2. Infer ranks.
3. Compare to *s*.

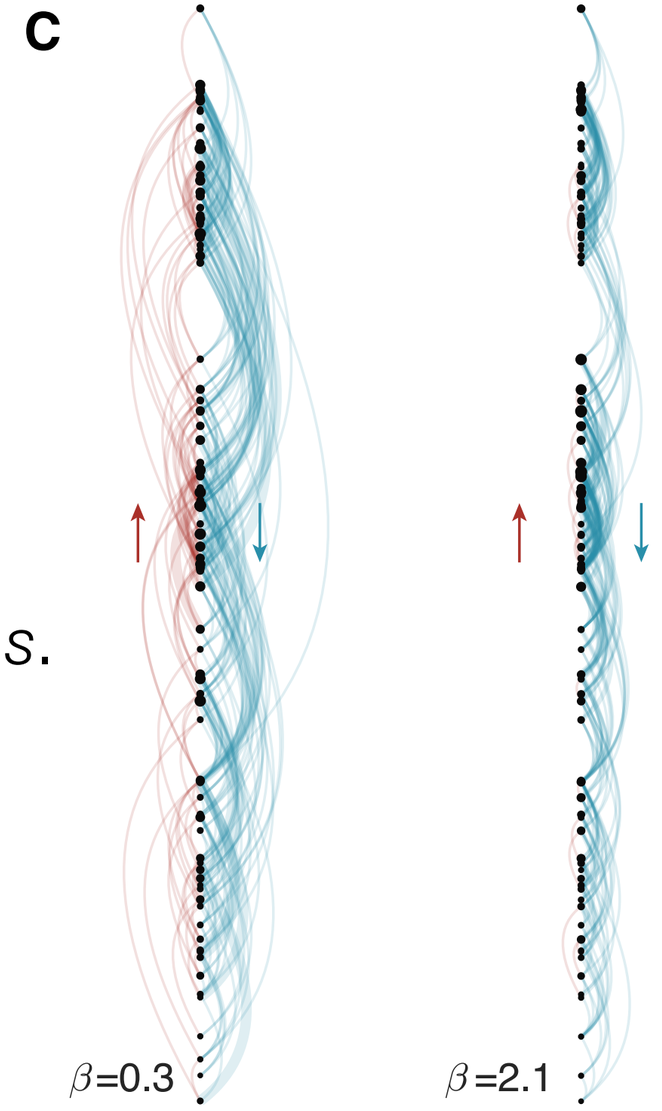


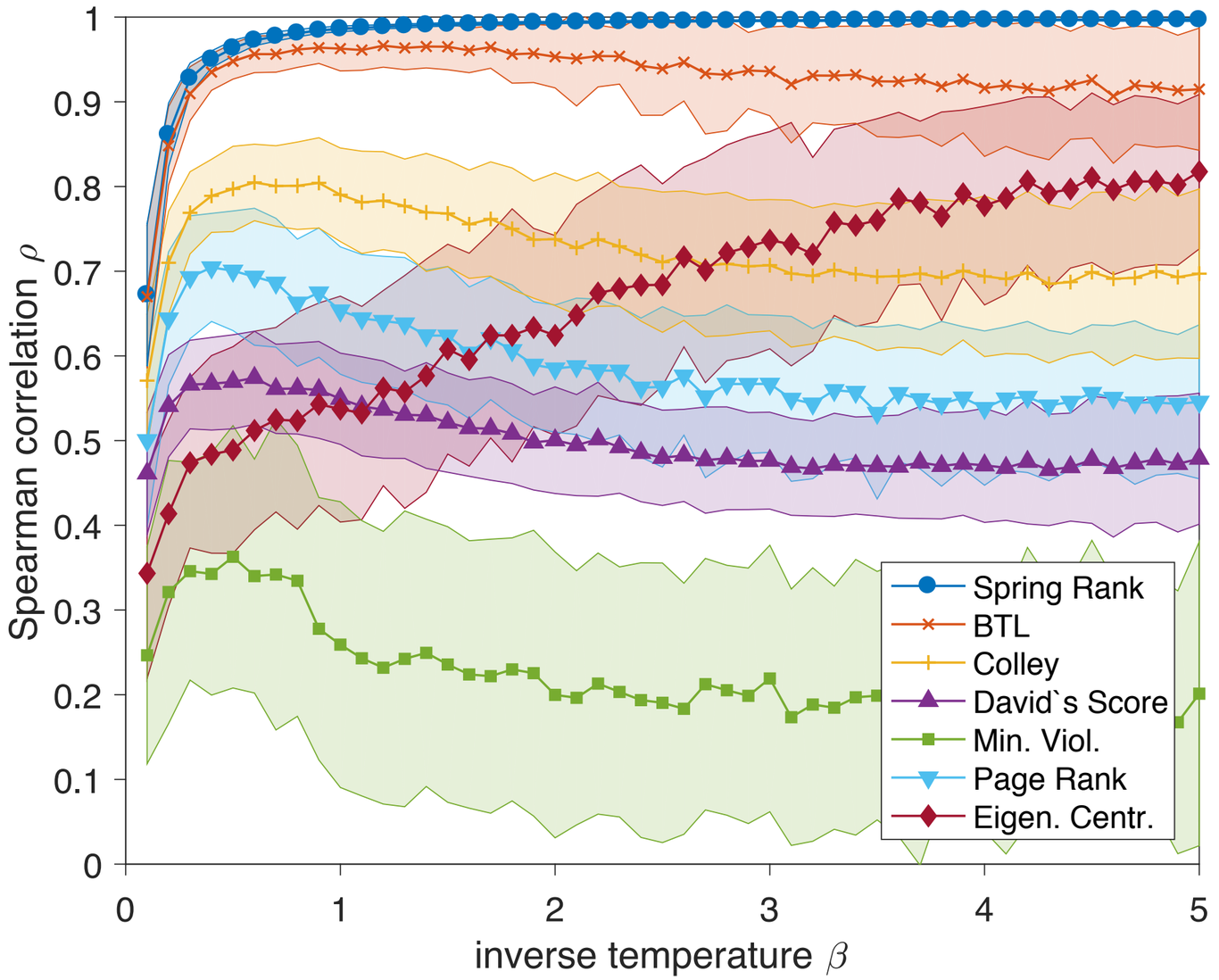


Consistency: Test SpringRank using GM synthetic data

Hard:

1. Plant *s* from **three normals.**
2. Infer ranks.
3. Compare to *s*.





Edge prediction & cross validation

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?

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).*

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.*

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:

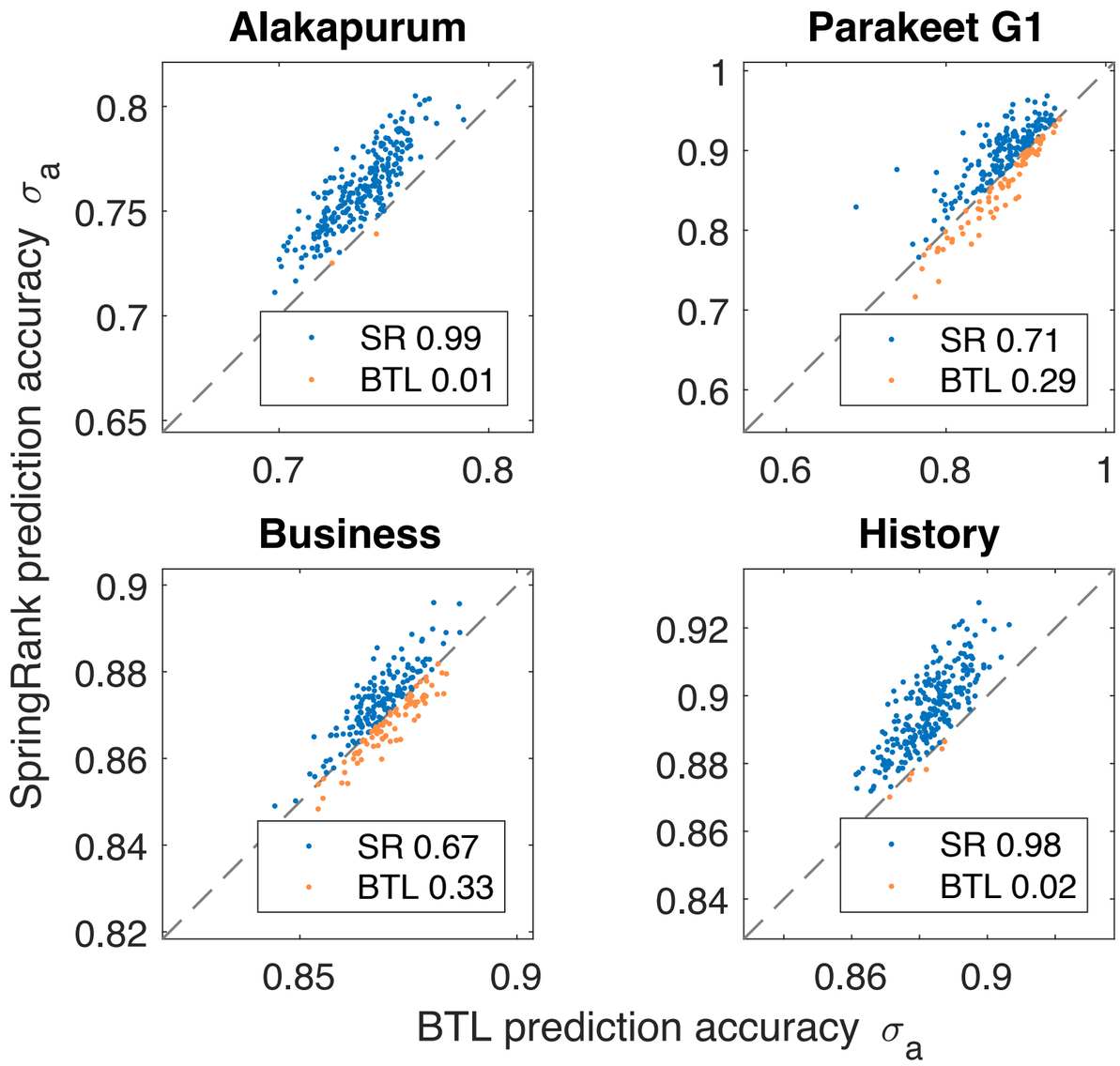
$$P_{ij}(\beta) = \frac{e^{-\beta H_{ij}}}{e^{-\beta H_{ij}} + e^{-\beta H_{ji}}} = \frac{1}{1 + e^{-2\beta(s_i - s_j)}}$$

Cross validation results

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



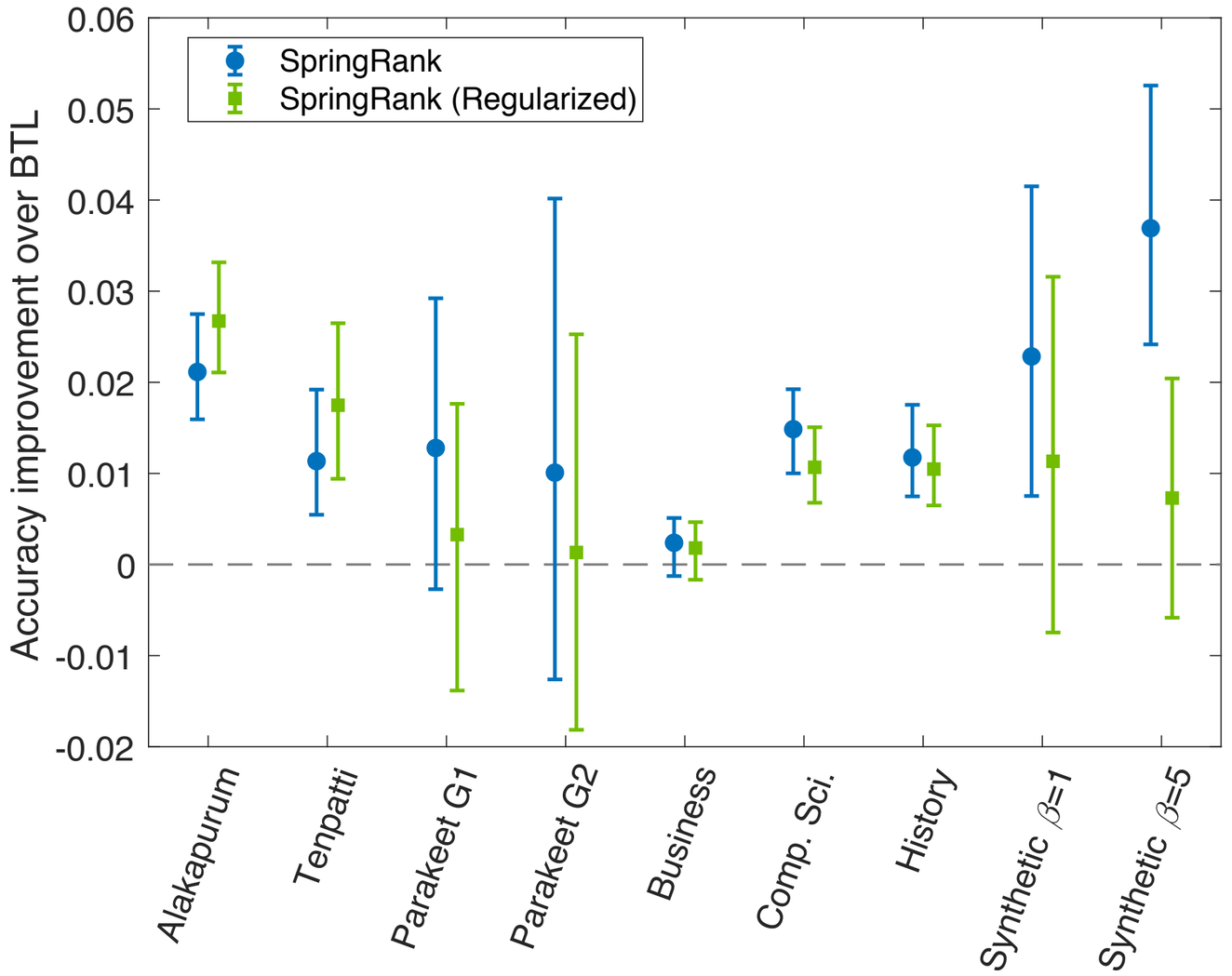
50 independent trials of 5-fold cross validation (250 folds)

Cross validation results

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.

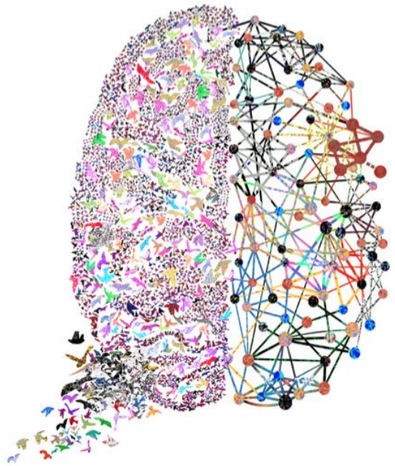


medians & quartiles; 50 independent trials of 5-fold cross validation (250 folds)

Conference on Complex Networks

COMPLENET '18

Hosted by Northeastern University Network Science Institute



BOSTON, MA APRIL 2018

Xindi Wang

netsi | Northeastern University
Network Science Institute

LEARNING TO PLACE OBJECTS: A NETWORK-BASED APPROACH

Xindi Wang

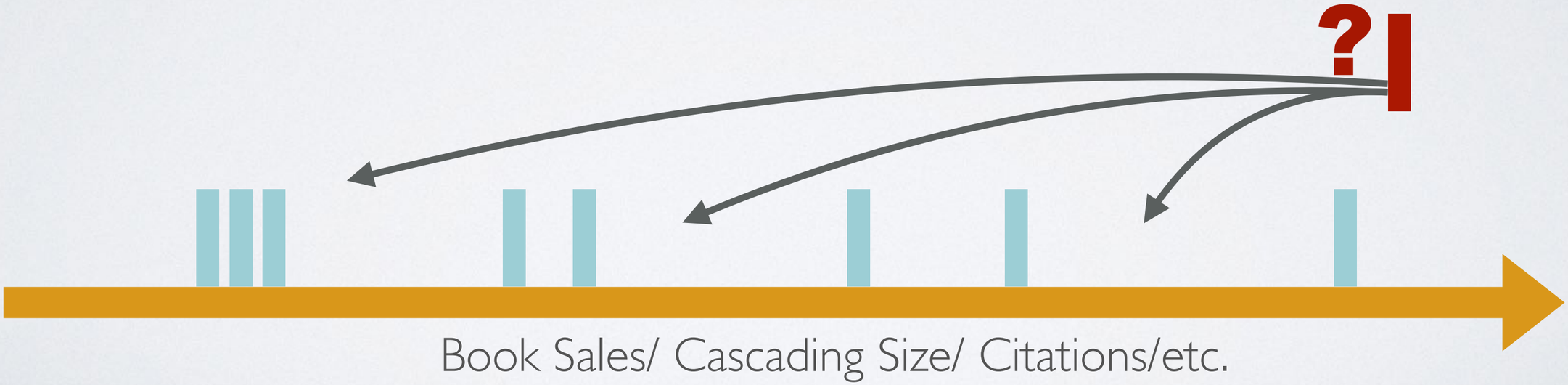
Onur Varol

Tina Eliassi-Rad

Albert-László Barabási

PROBLEM DEFINITION

Learning to place: Given a sequence of ordered items, where should we place a new item?



AUC

Method		AUC on Fiction	AUC on Biography
	KNN	0.759	0.815
Pairwise +	Cohen et al.	0.892	0.871
	WTG wave	**0.910**	**0.892**
	Voting	**0.915**	**0.891**
	FAS-PIVOT	**0.907**	**0.892**
	SpringRank	**0.908**	**0.893**

AUC

Method		AUC on Fiction	AUC on Biography
	KNN	0.759	0.815
Pairwise +	Cohen et al.	0.892	0.871
	WTG wave	**0.910**	**0.892**
	Voting	**0.915**	**0.891**
	FAS-PIVOT	**0.907**	**0.892**
	SpringRank	🥉 **0.908**	🥇 **0.893**

But is there *really* a linear hierarchy?

If ranks are meaningful, then they are related to *outcomes,* i.e. the directions of edges.

Use the ground-state energy as a test statistic, under an edge-direction randomized null.

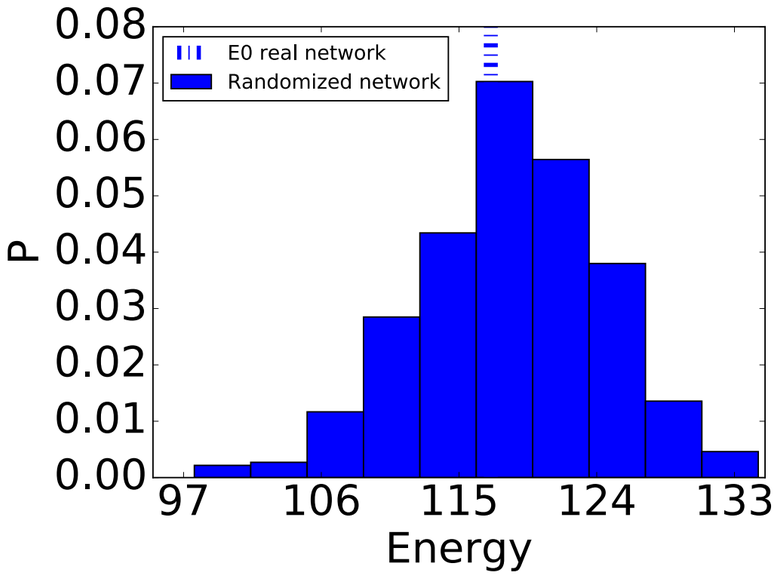
$$\frac{H(s^*)}{M} = \frac{1}{2M} \sum_i \left(d_i^{\text{in}} - d_i^{\text{out}} \right) s_i^* + \frac{1}{2}$$

But is there *really* a linear hierarchy?

If ranks are meaningful, then they are related to *outcomes,* i.e. the directions of edges.

Use the ground-state energy as a test statistic, under an edge-direction randomized null.

$$\frac{H(s^*)}{M} = \frac{1}{2M}\sum_i \left(d_i^{\text{in}} - d_i^{\text{out}}\right) s_i^* + \frac{1}{2}$$



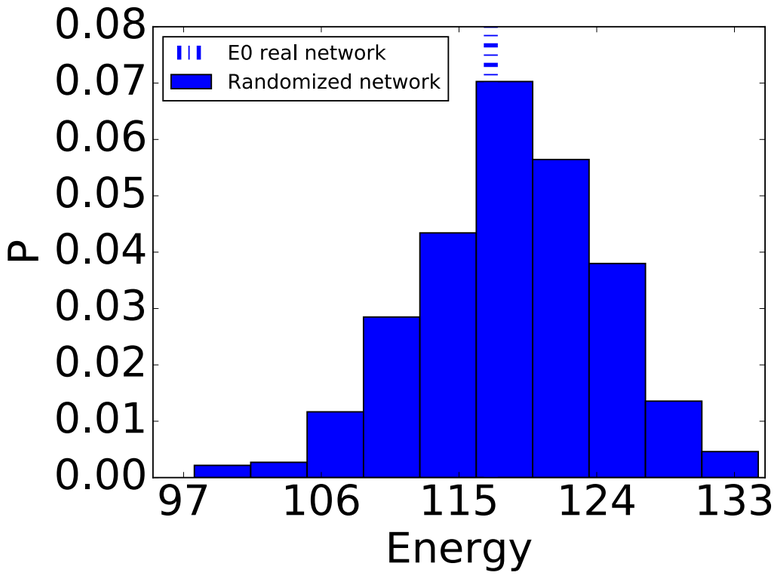
ER Network

But is there *really* a linear hierarchy?

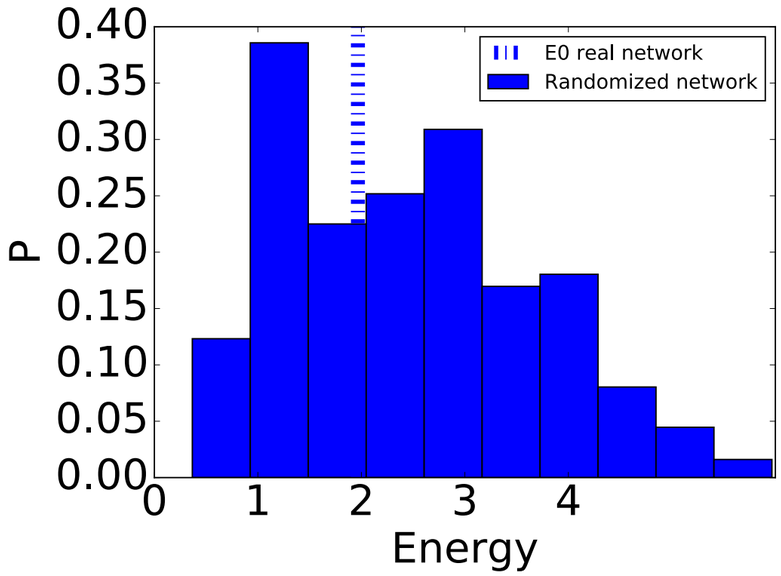
If ranks are meaningful, then they are related to *outcomes,* i.e. the directions of edges.

Use the ground-state energy as a test statistic, under an edge-direction randomized null.

$$\frac{H(s^*)}{M} = \frac{1}{2M}\sum_i \left(d_i^{\text{in}} - d_i^{\text{out}}\right) s_i^* + \frac{1}{2}$$



ER Network



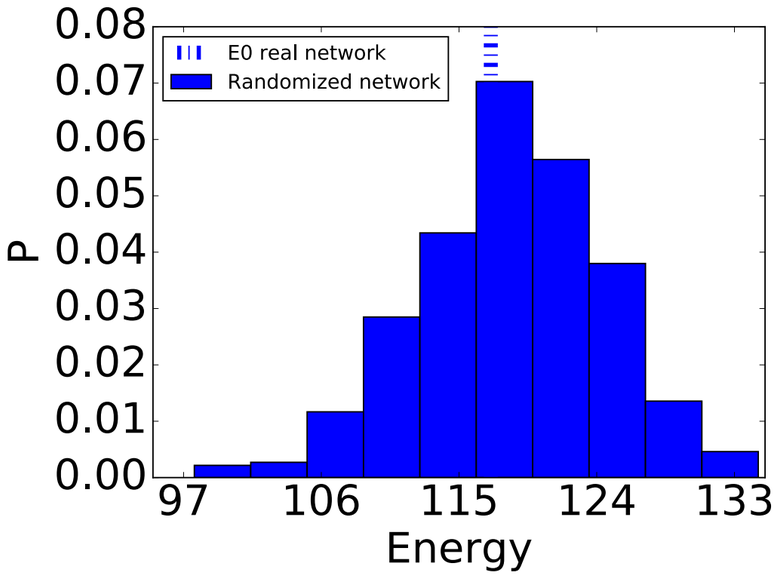
Asian Elephants

But is there *really* a linear hierarchy?

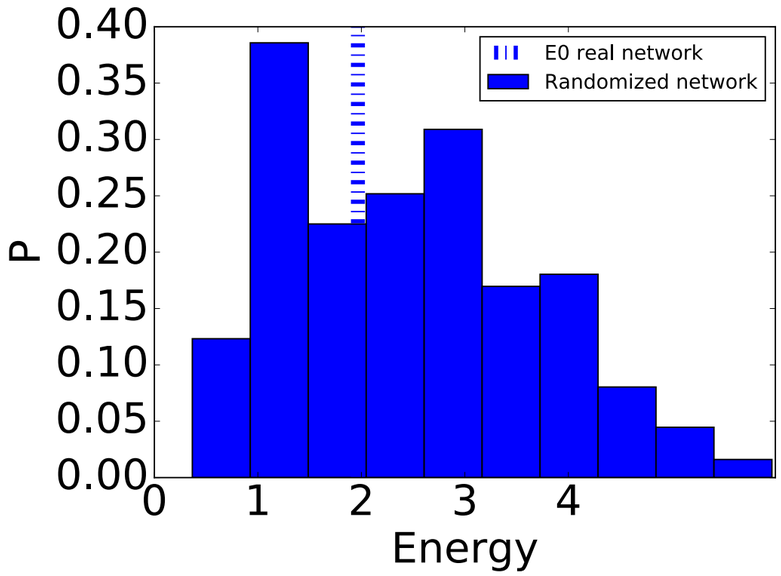
If ranks are meaningful, then they are related to *outcomes,* i.e. the directions of edges.

Use the ground-state energy as a test statistic, under an edge-direction randomized null.

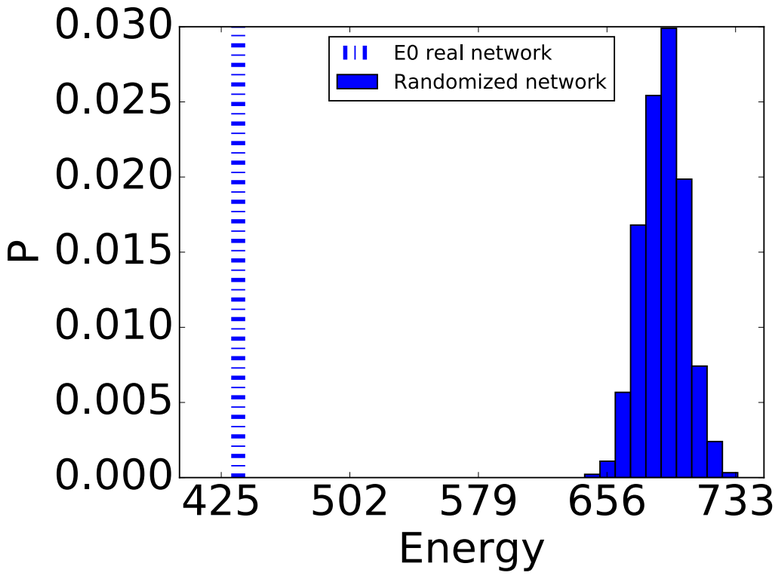
$$\frac{H(s^*)}{M} = \frac{1}{2M}\sum_i \left(d_i^{\text{in}} - d_i^{\text{out}}\right) s_i^* + \frac{1}{2}$$



ER Network



Asian Elephants



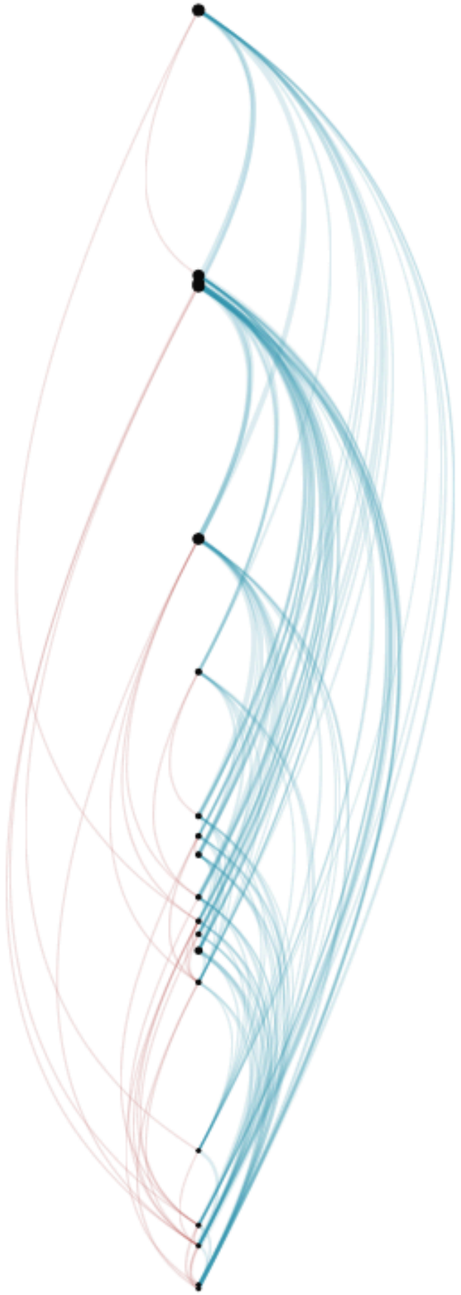
Social Support

Conclusions and extensions

1. SpringRank is a fast, scalable, *predictive* approach to ranking in directed networks. Works for Dominance or Endorsement.

2. It formalizes the idea that we can learn from not just the *direction* of edges, but the *existence* of edges as well.

3. There is a corresponding generative model whose ML solution is asymptotically equivalent. SpringRank approximates this ML solution.



Conclusions and extensions

1. SpringRank is a fast, scalable, *predictive* approach to ranking in directed networks. Works for Dominance or Endorsement.

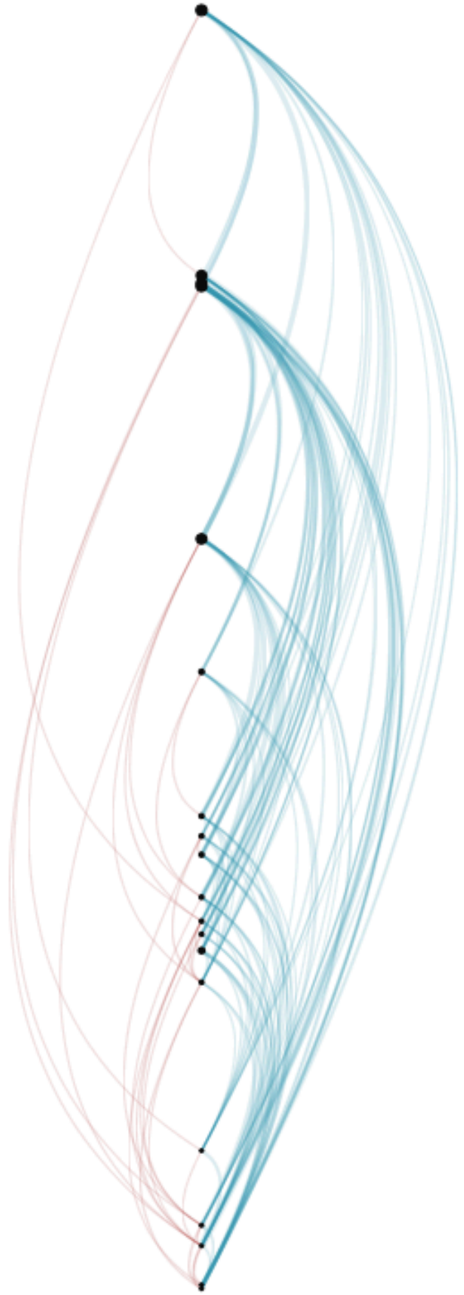
2. It formalizes the idea that we can learn from not just the *direction* of edges, but the *existence* of edges as well.

3. There is a corresponding generative model whose ML solution is asymptotically equivalent. SpringRank approximates this ML solution.

a. What if the hierarchy **branches**, like a Y?

b. What about other convex **spring potential functions**?

c. When might the **spring constants** depend on *i* or *j* ?



Caterina De Bacco

COLUMBIA UNIVERSITY

Cris Moore

SANTA FE INSTITUTE

A physical model for efficient ranking in networks
De Bacco*, Larremore*, and Moore. In Revision.
{preprint, python, matlab, viz} → danlarremore.com

NSF

Thank you

@danlarremore
daniel.larremore@colorado.edu

Supplement: 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?