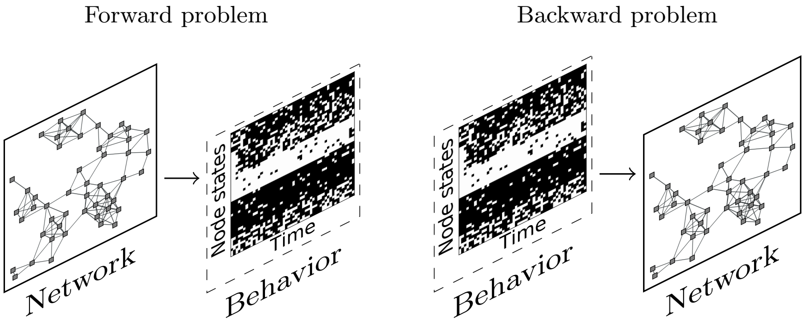
Network reconstruction from indirect measurements and dynamics

Tiago P. Peixoto

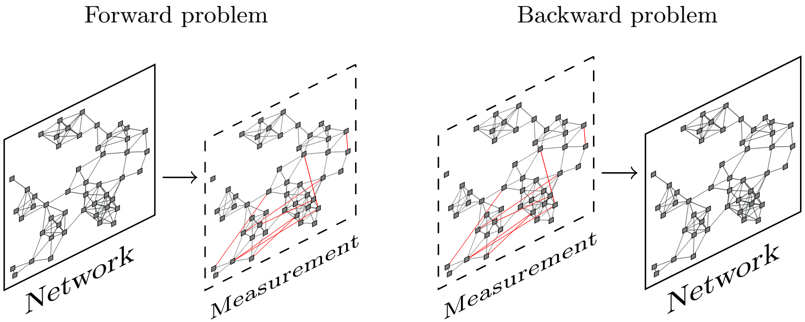
Department of Network and Data Science
Central European University, Vienna

~~Split~~ Vienna, September 2020

Network reconstruction from dynamics

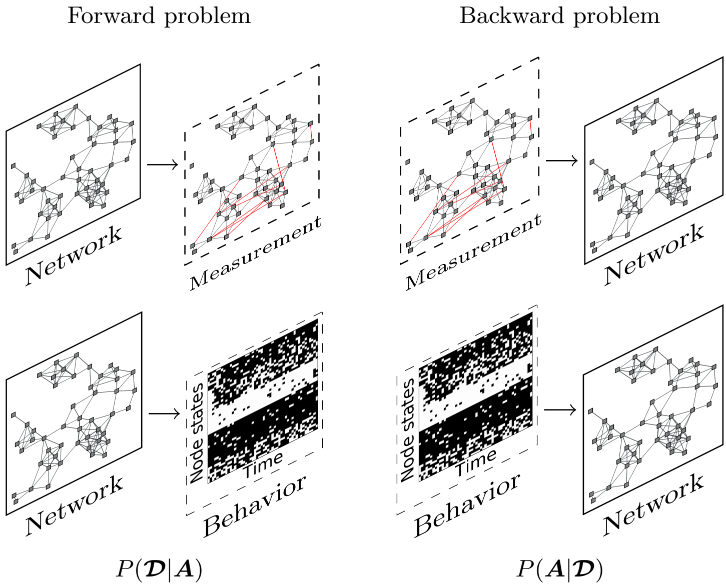


Network reconstruction from noisy measurements



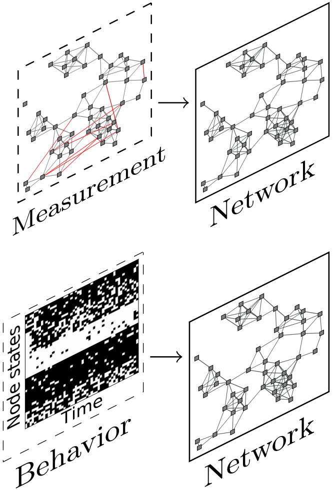
Network reconstruction

$\boldsymbol{A} \rightarrow$ Network $\boldsymbol{\mathcal{D}} \rightarrow$ Data (Measurement or Behavior)



Bayesian network reconstruction

Backward problem



$$\boxed{P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})}}$$

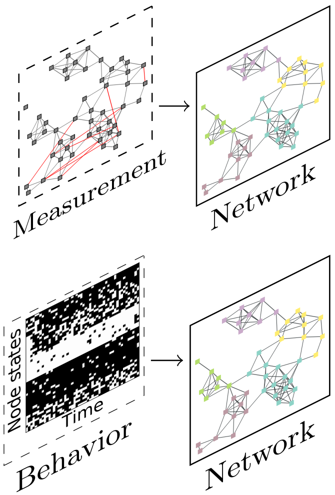
$P(\boldsymbol{\mathcal{D}}|\boldsymbol{A}) \to$ Likelihood

$P(\boldsymbol{A}) \to$ Prior

$$P(\boldsymbol{\mathcal{D}}) = \sum_{\boldsymbol{A}} P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})$$

Joint reconstruction and community detection

Backward problem



$\boldsymbol{b} \rightarrow$ Partition of the nodes into groups

$$\boxed{P(\boldsymbol{A}, \boldsymbol{b}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})}{P(\boldsymbol{\mathcal{D}})}}$$

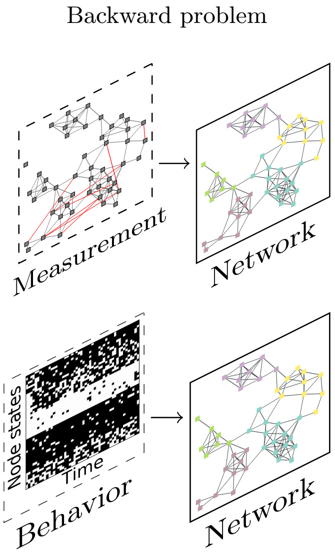
$P(\boldsymbol{\mathcal{D}}|\boldsymbol{A}) \rightarrow$ Likelihood

$P(\boldsymbol{A}|\boldsymbol{b}) \rightarrow$ Prior w/ latent communities

$P(\boldsymbol{b}) \rightarrow$ Partition prior

$$P(\boldsymbol{\mathcal{D}}) = \sum_{\boldsymbol{A},\boldsymbol{b}} P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})$$

Joint reconstruction and community detection



$\boldsymbol{b} \rightarrow$ Partition of the nodes into groups

$$\boxed{P(\boldsymbol{A},\boldsymbol{b}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})}{P(\boldsymbol{\mathcal{D}})}}$$

$P(\boldsymbol{\mathcal{D}}|\boldsymbol{A}) \rightarrow$ Likelihood

$P(\boldsymbol{A}|\boldsymbol{b}) \rightarrow$ Prior w/ latent communities

$P(\boldsymbol{b}) \rightarrow$ Partition prior

$$P(\boldsymbol{\mathcal{D}}) = \sum_{\boldsymbol{A},\boldsymbol{b}} P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})$$

"Strict" reconstruction:

$$P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \sum_{\boldsymbol{b}} P(\boldsymbol{A},\boldsymbol{b}|\boldsymbol{\mathcal{D}})$$

"Strict" community detection:

$$P(\boldsymbol{b}|\boldsymbol{\mathcal{D}}) = \sum_{\boldsymbol{A}} P(\boldsymbol{A},\boldsymbol{b}|\boldsymbol{\mathcal{D}})$$

Noisy network reconstruction task

$$\underbrace{\boldsymbol{A}^*}_{\text{True network}} \longrightarrow \boxed{\text{Noisy measurement}} \longrightarrow \underbrace{\boldsymbol{A}}_{\text{Observed network}}$$

$$\underbrace{\boldsymbol{A}}_{\text{Observed network}} \longrightarrow \boxed{\text{Reconstruction}} \longrightarrow \underbrace{\hat{\boldsymbol{A}}}_{\text{Reconstructed network}}$$

So that $\hat{\boldsymbol{A}}$ is as close as possible to $\boldsymbol{A}^*$.

Caveats:

- With a single copy of $\boldsymbol{A}$.
- Without knowing how strong the noise is (i.e. the number of missing or spurious edges).

How is reconstruction possible?

(a)

(b)

How is reconstruction possible?


(a)


(b)

We need:

- A model for structure.
- A model for noise.

How is reconstruction possible?

(a) (b)

We need:

- A model for structure.
- A model for noise.

(but for networks)

Nonparametric Bayesian inference

- A model for structure, $P(\boldsymbol{A}|\theta)$
- A model for noise, $P(\boldsymbol{\mathcal{D}}|\boldsymbol{A},\phi)$

$\boldsymbol{A} \to$ Network, $\boldsymbol{\mathcal{D}} \to$ Measured data, $(\theta,\phi) \to$ Parameters

Posterior distribution:

$$\boxed{P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})}}$$

Marginal probabilities:

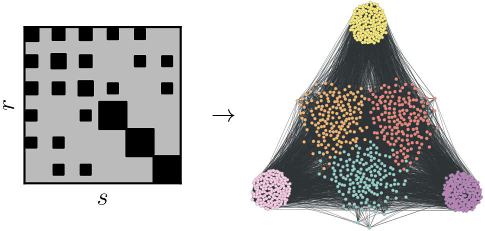
$$P(\boldsymbol{\mathcal{D}}|\boldsymbol{A}) = \int P(\boldsymbol{\mathcal{D}}|\boldsymbol{A},\phi)P(\phi)\mathrm{d}\phi$$

$$P(\boldsymbol{A}) = \int P(\boldsymbol{A}|\theta)P(\theta)\mathrm{d}\theta$$

Network structure: The stochastic block model (SBM)

Planted partition: N nodes divided into B groups.

Parameters: $b_i \to$ group membership of node i
$\omega_{rs} \to$ edge probability from group r to s.



Degree-corrected: Arbitrary degree sequence: $\{\kappa_i\}$

- Not restricted to assortative structures ("communities").
- Easily generalizable (edge direction, overlapping groups, etc.)

Bayesian SBM

$$P(\boldsymbol{A}|\boldsymbol{\lambda},\boldsymbol{\kappa},\boldsymbol{b}) = \prod_{i<j} \frac{(\kappa_i\kappa_j\lambda_{b_ib_j})^{A_{ij}} e^{-\kappa_i\kappa_j\lambda_{b_ib_j}}}{A_{ij}!} \times \prod_i \frac{(\kappa_i^2\lambda_{b_ib_i}/2)^{A_{ii}/2} e^{-\kappa_i^2\lambda_{b_ib_i}/2}}{(A_{ii}/2)!}$$

Noninformative priors:

$$P(\boldsymbol{\lambda}|\boldsymbol{b}) = \prod_{r\le s} e^{-\lambda_{rs}/(1+\delta_{rs})\bar{\lambda}}/(1+\delta_{rs})\bar{\lambda}$$

$$P(\boldsymbol{\kappa}|\boldsymbol{b}) = \prod_r (n_r-1)!\delta(\textstyle\sum_i \kappa_i\delta_{b_i,r}-1)$$

Bayesian SBM

$$P(\boldsymbol{A}|\boldsymbol{\lambda},\boldsymbol{\kappa},\boldsymbol{b}) = \prod_{i<j} \frac{(\kappa_i\kappa_j\lambda_{b_ib_j})^{A_{ij}} e^{-\kappa_i\kappa_j\lambda_{b_ib_j}}}{A_{ij}!} \times \prod_i \frac{(\kappa_i^2\lambda_{b_ib_i}/2)^{A_{ii}/2} e^{-\kappa_i^2\lambda_{b_ib_i}/2}}{(A_{ii}/2)!}$$

Noninformative priors:

$$P(\boldsymbol{\lambda}|\boldsymbol{b}) = \prod_{r\le s} e^{-\lambda_{rs}/(1+\delta_{rs})\bar\lambda}/(1+\delta_{rs})\bar\lambda$$

$$P(\boldsymbol{\kappa}|\boldsymbol{b}) = \prod_r (n_r-1)!\,\delta(\textstyle\sum_i \kappa_i\delta_{b_i,r}-1)$$

Marginal likelihood:

$$\begin{aligned}
P(\boldsymbol{A}|\boldsymbol{b}) &= \int P(\boldsymbol{A}|\boldsymbol{\lambda},\boldsymbol{\kappa},\boldsymbol{b})P(\boldsymbol{\lambda}|\boldsymbol{b})P(\boldsymbol{\kappa}|\boldsymbol{b})\,\mathrm{d}\boldsymbol{\lambda}\mathrm{d}\boldsymbol{\kappa}\\
&= \frac{\bar\lambda^E}{(\bar\lambda+1)^{E+B(B+1)/2}} \times \frac{\prod_{r<s} e_{rs}!\prod_r e_{rr}!!}{\prod_{i<j} A_{ij}!\prod_i A_{ii}!!} \times \prod_r \frac{(n_r-1)!}{(e_r+n_r-1)!} \times \prod_i k_i!\\
&= P(\boldsymbol{A}|\boldsymbol{k},\boldsymbol{e},\boldsymbol{b})P(\boldsymbol{k}|\boldsymbol{e},\boldsymbol{b})P(\boldsymbol{e})
\end{aligned}$$

Bayesian SBM

$$P(\boldsymbol{A}|\boldsymbol{\lambda},\boldsymbol{\kappa},\boldsymbol{b}) = \prod_{i<j} \frac{(\kappa_i\kappa_j\lambda_{b_ib_j})^{A_{ij}} e^{-\kappa_i\kappa_j\lambda_{b_ib_j}}}{A_{ij}!} \times \prod_i \frac{(\kappa_i^2\lambda_{b_ib_i}/2)^{A_{ii}/2} e^{-\kappa_i^2\lambda_{b_ib_i}/2}}{(A_{ii}/2)!}$$

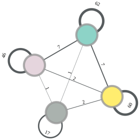
Noninformative priors:

$$P(\boldsymbol{\lambda}|\boldsymbol{b}) = \prod_{r\le s} e^{-\lambda_{rs}/(1+\delta_{rs})\bar{\lambda}}/(1+\delta_{rs})\bar{\lambda}$$

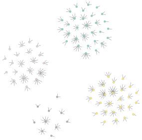
$$P(\boldsymbol{\kappa}|\boldsymbol{b}) = \prod_r (n_r-1)!\,\delta(\textstyle\sum_i \kappa_i\delta_{b_i,r} - 1)$$

Marginal likelihood:

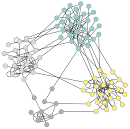
$$\begin{aligned} P(\boldsymbol{A}|\boldsymbol{b}) &= \int P(\boldsymbol{A}|\boldsymbol{\lambda},\boldsymbol{\kappa},\boldsymbol{b})P(\boldsymbol{\lambda}|\boldsymbol{b})P(\boldsymbol{\kappa}|\boldsymbol{b})\,\mathrm{d}\boldsymbol{\lambda}\mathrm{d}\boldsymbol{\kappa} \\ &= \frac{\bar{\lambda}^E}{(\bar{\lambda}+1)^{E+B(B+1)/2}} \times \frac{\prod_{r<s} e_{rs}!\prod_r e_{rr}!!}{\prod_{i<j} A_{ij}!\prod_i A_{ii}!!} \times \prod_r \frac{(n_r-1)!}{(e_r+n_r-1)!} \times \prod_i k_i! \\ &= P(\boldsymbol{A}|\boldsymbol{k},\boldsymbol{e},\boldsymbol{b})P(\boldsymbol{k}|\boldsymbol{e},\boldsymbol{b})P(\boldsymbol{e}) \end{aligned}$$



Edge counts $\boldsymbol{e}$.

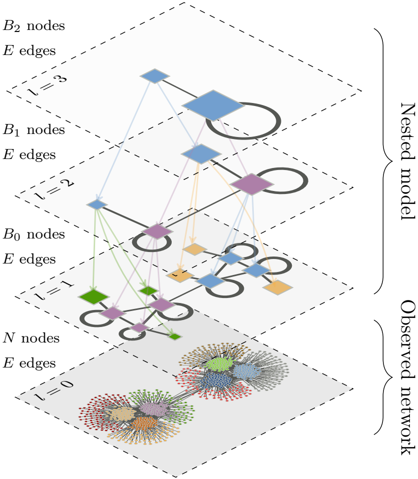


Degrees, $\boldsymbol{k}$.



Network, $\boldsymbol{A}$.

Nested SBM: Group hierarchies



Deeper Bayesian hierarchy:

- Prevents underfitting.
- Multiple scales of description.

Measurement model

$\boxed{\text{Edge-o-meter}}$

$p \to$ probability of a missing edge $(1 \to 0)$
$q \to$ probability of a spurious edge $(0 \to 1)$.

$n_{ij} \to$ number of measurements of pair (i, j)
$x_{ij} \to$ number of edges recorded

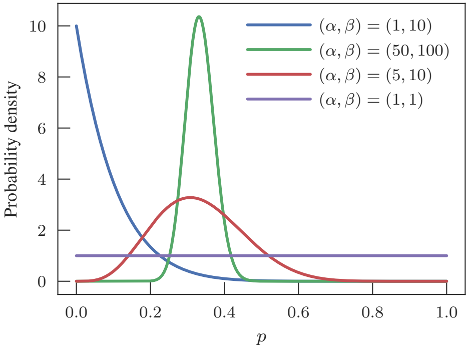
$$P(x_{ij}|n_{ij}, A_{ij}, p, q) = \binom{n_{ij}}{x_{ij}} \left[(1-p)^{x_{ij}} p^{n_{ij}-x_{ij}}\right]^{A_{ij}} \left[q^{x_{ij}} (1-q)^{n_{ij}-x_{ij}}\right]^{1-A_{ij}}$$

$$P(\boldsymbol{x}|\boldsymbol{n}, \boldsymbol{A}, \alpha, \beta, \mu, \nu) = \int P(\boldsymbol{x}|\boldsymbol{n}, \boldsymbol{A}, p, q) P(p|\alpha, \beta) P(q|\mu, \nu)\, \mathrm{d}p\, \mathrm{d}q$$

$$P(p|\alpha, \beta), P(q|\mu, \nu) \to \text{Beta priors}$$

The Edge-o-meter

$$P(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\alpha-1}}{\mathcal{B}(\alpha,\beta)} \qquad P(q|\mu,\nu) = \frac{q^{\mu-1}(1-q)^{\mu-1}}{\mathcal{B}(\mu,\nu)}$$



- $(\alpha,\beta) = (1,10) \rightarrow$ accurate measurement (low noise)
- $(\alpha,\beta) = (50,100) \rightarrow$ high noise, good calibration
- $(\alpha,\beta) = (5,10) \rightarrow$ high noise, bad calibration
- $(\alpha,\beta) = (1,1) \rightarrow$ <u>noninformative</u> (i.e. uniform distribution)

The full reconstruction method

Posterior distribution:

$$P(\boldsymbol{A},\boldsymbol{b}|\boldsymbol{n},\boldsymbol{x},\alpha,\beta,\mu,\nu) = \frac{P(\boldsymbol{x}|\boldsymbol{n},\boldsymbol{A},\alpha,\beta,\mu,\nu)P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})}{P(\boldsymbol{x}|\alpha,\beta,\mu,\nu)}.$$

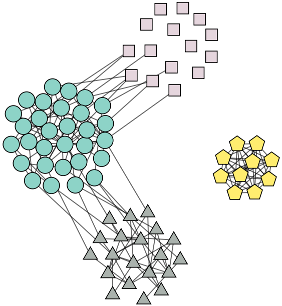
We infer both the network $\boldsymbol{A}$ as well as the SBM latent variables $\boldsymbol{b}$ via MCMC:

Move proposals $P(\boldsymbol{b}'|\boldsymbol{A},\boldsymbol{b})$ and $P(\boldsymbol{A}'|\boldsymbol{A},\boldsymbol{b})$, accept with probability

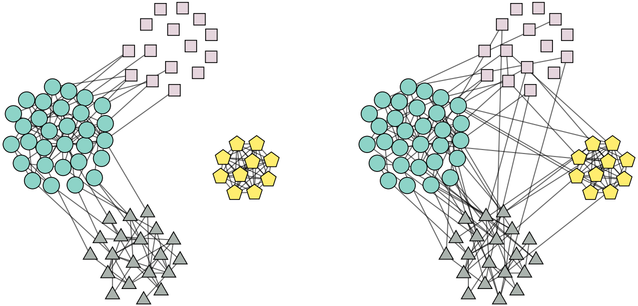
$$\min\left(1, \frac{P(\boldsymbol{A}',\boldsymbol{b}'|\boldsymbol{\mathcal{D}})P(\boldsymbol{A}|\boldsymbol{A}',\boldsymbol{b}')P(\boldsymbol{b}|\boldsymbol{A}',\boldsymbol{b}')}{P(\boldsymbol{A},\boldsymbol{b}|\boldsymbol{\mathcal{D}})P(\boldsymbol{A}'|\boldsymbol{A},\boldsymbol{b})P(\boldsymbol{b}'|\boldsymbol{A},\boldsymbol{b})}\right).$$

(Efficient, scales to very large networks.)

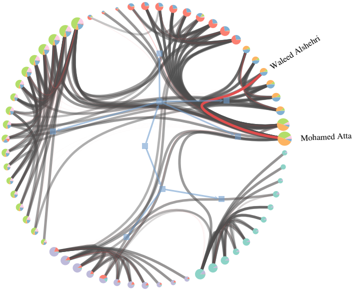
How does it work?

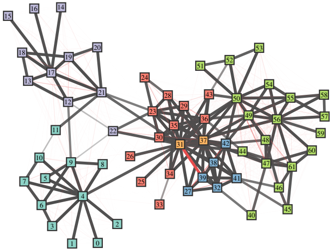


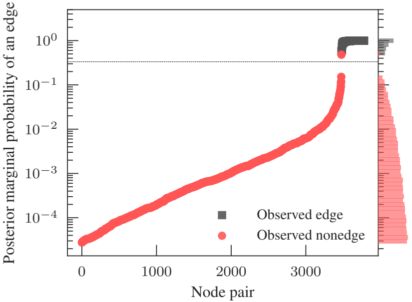
How does it work?



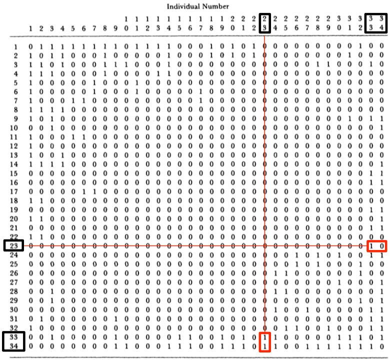
Example: Terrorist associations







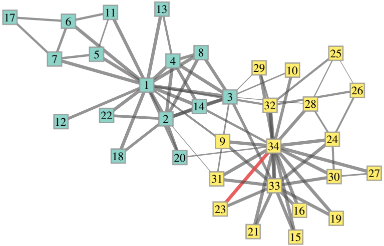
Example: Zachary's karate club

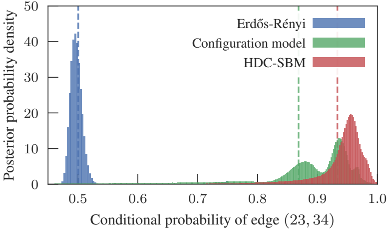


Individual Number

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0
2	1	0	1	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0
3	1	1	0	1	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0
4	1	1	1	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
11	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
17	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
20	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
22	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	1
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
28	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1
29	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1
31	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
32	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	1	1
33	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	1	0	1	0	1	1	0	0	0	0	0	1	1	1	0	1
34	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	1	1	1	1	1	1	0

(credit: Aaron Clauset)





Wait! Is this just edge prediction?

It *is* edge prediction, but it yields a full posterior distribution $P(\boldsymbol{A}|\boldsymbol{n},\boldsymbol{x})$ that is **nonparametric**.

We can:

- Perform maximum marginal posterior estimation,

$$\hat{A}_{ij} = \begin{cases} 1 & \text{if } \pi_{ij} > 1/2 \\ 0 & \text{if } \pi_{ij} < 1/2, \end{cases}$$

  where $\pi_{ij} = \sum_{\boldsymbol{A}} A_{ij} P(\boldsymbol{A}|\boldsymbol{n},\boldsymbol{x})$ is the marginal posterior edge probability.

- Estimate network properties $y(\boldsymbol{A})$ and their error estimates:

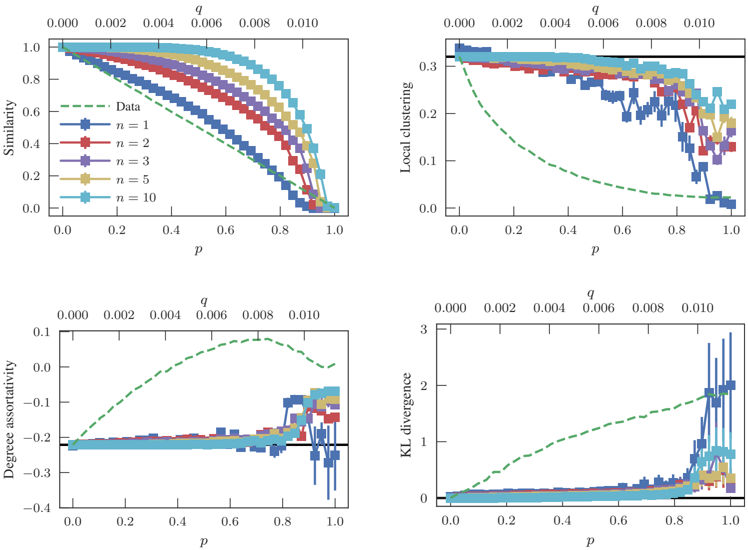
$$\hat{y} = \sum_{\boldsymbol{A}} y(\boldsymbol{A}) P(\boldsymbol{A}|\boldsymbol{n},\boldsymbol{x})$$

$$\sigma_y^2 = \sum_{\boldsymbol{A}} (\hat{y} - y(\boldsymbol{A}))^2 P(\boldsymbol{A}|\boldsymbol{n},\boldsymbol{x}).$$

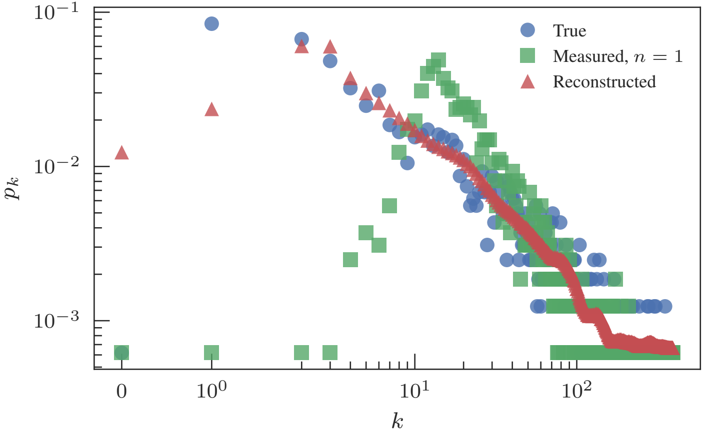
Reconstruction performance

Real network (political blogs) + simulated noise:

$$p \in [0,1],\ q = pE/[\binom{N}{2} - E]$$

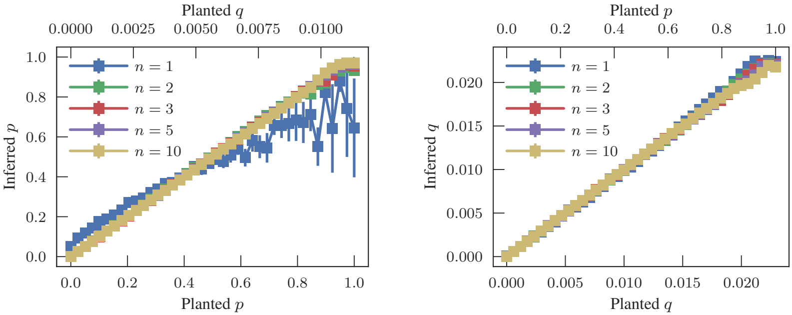


Reconstruction performance: Degrees



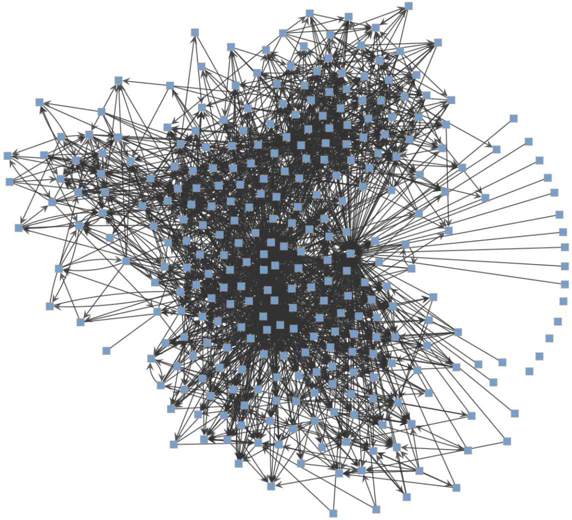
$(p, q) = (0.41, 0.0094)$

Inferring the noise



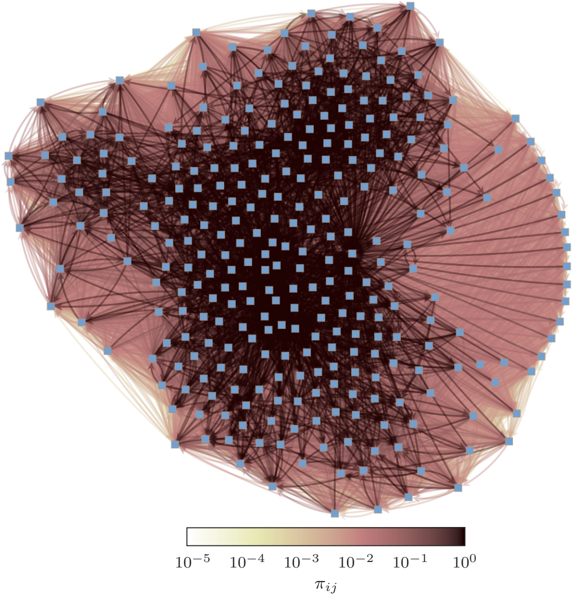
Empirical reconstruction redux

C. elegans neural network



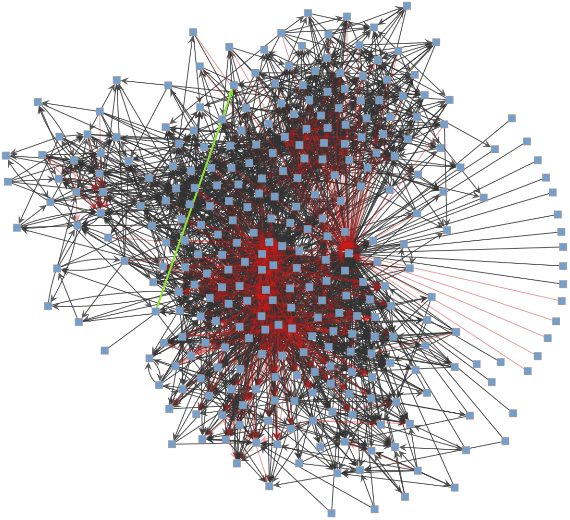
Empirical reconstruction redux

C. elegans neural network

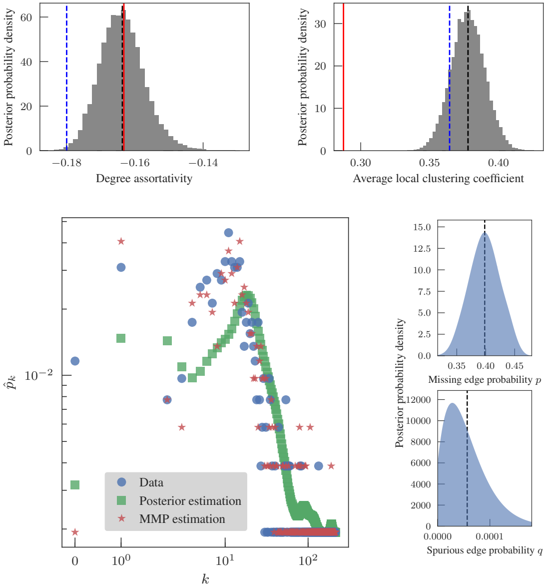


Empirical reconstruction redux

C. elegans neural network



C. elegans NEURAL NETWORK



Multiple measurements and heterogeneous errors

Observational error does not need to be uniform for every pair (i,j).
Non-uniform model, w/ pair-specific error rates: p_{ij} and q_{ij}

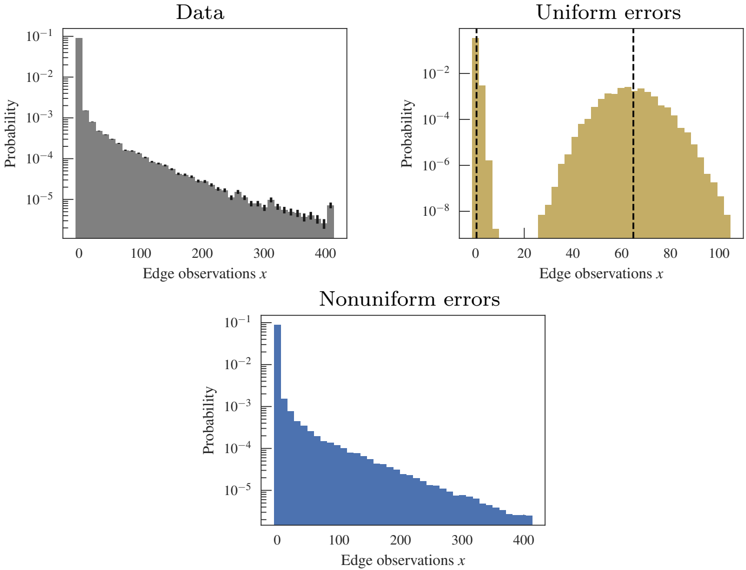
$$P(x_{ij}|n_{ij},A_{ij},p_{ij},q_{ij}) = \binom{n_{ij}}{x_{ij}} \left[(1-p_{ij})^{x_{ij}} p_{ij}^{n_{ij}-x_{ij}}\right]^{A_{ij}} \left[q_{ij}^{x_{ij}}(1-q_{ij})^{n_{ij}-x_{ij}}\right]^{1-A_{ij}}$$

Marginal probability,

$$\begin{aligned} &P(x_{ij}|n_{ij},A_{ij},\alpha,\beta,\mu,\nu) \\ &\quad= \int P(x_{ij}|n_{ij},A_{ij},p_{ij},q_{ij}) P(p_{ij}|\alpha,\beta) P(q_{ij}|\mu,\nu)\, \mathrm{d}p_{ij} \mathrm{d}q_{ij} \\ &\quad= \binom{n_{ij}}{x_{ij}} \left[\frac{\mathcal{B}(n_{ij}-x_{ij}+\alpha, x_{ij}+\beta)}{\mathcal{B}(\alpha,\beta)}\right]^{A_{ij}} \times \\ &\qquad\qquad \left[\frac{\mathcal{B}(x_{ij}+\mu, n_{ij}-x_{ij}+\nu)}{\mathcal{B}(\mu,\nu)}\right]^{1-A_{ij}}. \end{aligned}$$

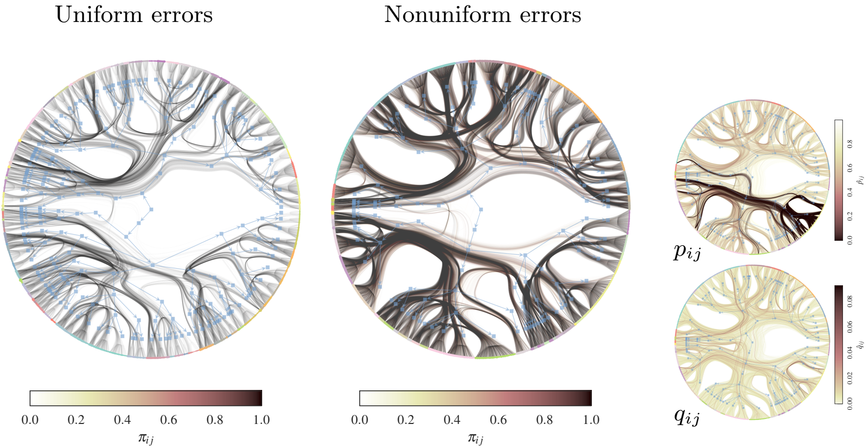
Human connectome

418 individuals



Human connectome

418 individuals



Extrinsic error estimates

$Q_{ij} \in [0,1] \to$ experimentally determined uncertainties

$$P_Q(\boldsymbol{A}|\boldsymbol{Q}) = \prod_{i<j} Q_{ij}^{A_{ij}} (1 - Q_{ij})^{1-A_{ij}}.$$

Example:

STRING Protein-Protein interaction network database, Szklarczyk et al, Nucleic Acids Research 45, D362–D368 (2017).

Errors are estimated via a combination of: (i) direct experiments, (ii) database curation, (iii) publication text-mining, (iv) co-expression data, (v) genome proximity, (vi) ortholog fusion, (vii) phylogenetic co-ocurrence.

Extrinsic error estimates

The distribution $P_Q(\boldsymbol{A}|\boldsymbol{Q})$ implies the following noisy measurement process,

$$P(\boldsymbol{Q}|\boldsymbol{A}) = \frac{P_Q(\boldsymbol{A}|\boldsymbol{Q})P_Q(\boldsymbol{Q})}{P_Q(\boldsymbol{A})},$$

with prior

$$P_Q(\boldsymbol{Q}) = \prod_{i<j} P(Q_{ij}),$$

and normalization constant

$$P_Q(\boldsymbol{A}) = \int P_Q(\boldsymbol{A}|\boldsymbol{Q})P_Q(\boldsymbol{Q})\,\mathrm{d}\boldsymbol{Q} = \prod_{i<j} \bar{Q}^{A_{ij}}(1-\bar{Q})^{1-A_{ij}},$$

with $\bar{Q} = \int_0^1 QP(Q)\mathrm{d}Q$. Combining these together we have

$$P(\boldsymbol{Q}|\boldsymbol{A}) = P_Q(\boldsymbol{Q}) \prod_{i<j} \left(\frac{Q_{ij}}{\bar{Q}}\right)^{A_{ij}} \left(\frac{1-Q_{ij}}{1-\bar{Q}}\right)^{1-A_{ij}},$$

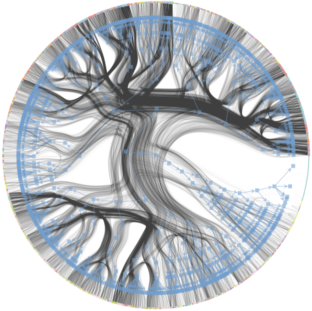
$$P(\boldsymbol{A}|\boldsymbol{Q}) = \frac{P(\boldsymbol{Q}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{Q})}, \qquad \bar{Q} = \frac{\sum_{i<j} Q_{ij}}{\binom{N}{2}}.$$

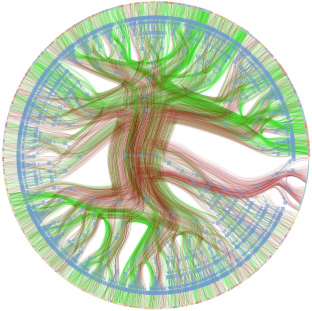
Extrinsic error estimates

$$P(\boldsymbol{A}|\boldsymbol{Q}) = \frac{P(\boldsymbol{Q}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{Q})}, \qquad P(\boldsymbol{A}|\boldsymbol{Q}) \neq P_Q(\boldsymbol{A}|\boldsymbol{Q})!$$

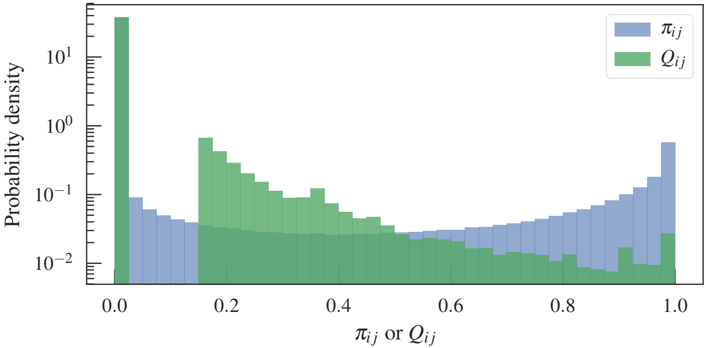
We are keeping the same noise generating process, but changing our prior assumption about the data.

E. coli proteins:





Extrinsic error estimates



Reconstruction from epidemic dynamics

$\boxed{\text{SIS epidemic model}}$

$$\sigma_i(t) \in \{0,1\} \rightarrow \text{ State of node } i \text{ at time } t$$

$$\tau_{ij} \rightarrow \text{ Transmission probability via edge } i \leftarrow j$$

$$\gamma \rightarrow \; 1 \rightarrow 0 \text{ Recovery probability}$$

Reconstruction from epidemic dynamics

$\boxed{\text{SIS epidemic model}}$

$$\sigma_i(t) \in \{0, 1\} \to \text{ State of node } i \text{ at time } t$$

$$\tau_{ij} \to \text{ Transmission probability via edge } i \leftarrow j$$

$$\gamma \to \; 1 \to 0 \text{ Recovery probability}$$

$$P(\boldsymbol{\sigma}|\boldsymbol{A}, \boldsymbol{\tau}, \gamma) = \prod_t \prod_i P(\sigma_i(t)|\boldsymbol{\sigma}(t-1))$$

$$P(\sigma_i(t)|\boldsymbol{\sigma}(t-1)) = f(\mathrm{e}^{m_i(t-1)}, \sigma_i(t))^{1-\sigma_i(t-1)} \times f(\gamma, \sigma_i(t))^{\sigma_i(t-1)}$$

$$f(p, \sigma) = (1-p)^\sigma p^{1-\sigma}, \qquad m_i(t) = \sum_j A_{ij} \ln(1 - \tau_{ij}) \sigma_j(t)$$

Reconstruction from epidemic dynamics

SIS epidemic model

$$\sigma_i(t) \in \{0, 1\} \to \text{ State of node } i \text{ at time } t$$

$$\tau_{ij} \to \text{ Transmission probability via edge } i \leftarrow j$$

$$\gamma \to \ 1 \to 0 \text{ Recovery probability}$$

$$P(\boldsymbol{\sigma}|\boldsymbol{A}, \boldsymbol{\tau}, \gamma) = \prod_t \prod_i P(\sigma_i(t)|\boldsymbol{\sigma}(t-1))$$

$$P(\sigma_i(t)|\boldsymbol{\sigma}(t-1)) = f(\mathrm{e}^{m_i(t-1)}, \sigma_i(t))^{1-\sigma_i(t-1)} \times f(\gamma, \sigma_i(t))^{\sigma_i(t-1)}$$

$$f(p, \sigma) = (1-p)^\sigma p^{1-\sigma}, \qquad m_i(t) = \sum_j A_{ij} \ln(1-\tau_{ij})\sigma_j(t)$$

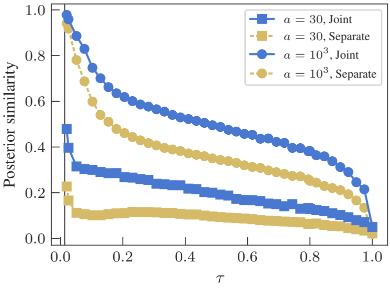
Posterior:

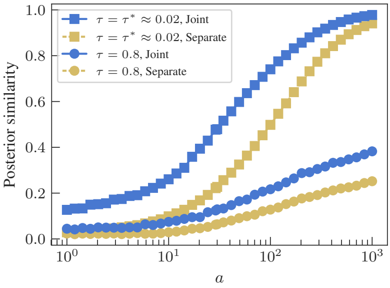
$$\boxed{P(\boldsymbol{A}, \boldsymbol{\tau}, \boldsymbol{b}|\boldsymbol{\sigma}) = \frac{P(\boldsymbol{\sigma}|\boldsymbol{A}, \boldsymbol{\tau}, \gamma)P(\boldsymbol{A}|\boldsymbol{b})P(\boldsymbol{b})P(\boldsymbol{\tau})}{P(\boldsymbol{\sigma}|\gamma)}}$$

Reconstruction from epidemic dynamics

Joint reconstruction with community detection improves accuracy.

Example: Simulated SIS for the worldwide directed network of $N = 3,286$ airports with $E = 39,430$ edges.





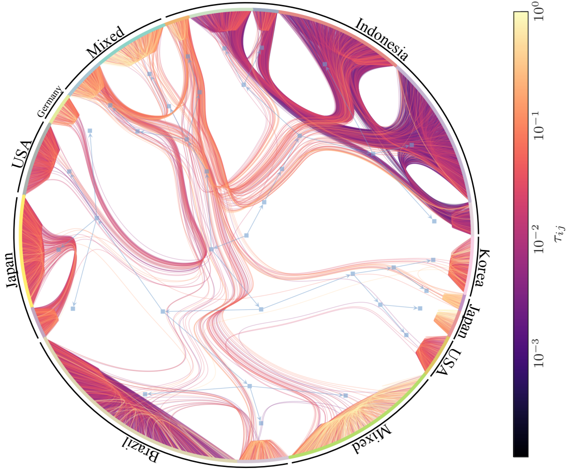
$a \to$ activity, i.e. number of $0 \to 1$ transitions per node

"Separate" means sampling from the posterior

$$P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})}, \qquad P(\boldsymbol{A}) \to \text{Erdős-Rényi}$$

Example: Reconstruction from twitter cascades

Reconstruction of the directed network of influence between $N = 1,833$ twitter users from $58,224$ re-tweets, using a SI infection model.



The inverse Ising model

Spins $s_i \in \{-1, 1\}$ sampled with probability

$$P(\boldsymbol{s}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \frac{\exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)}{Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})}$$

Partition function:

$$Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \sum_{\boldsymbol{s}} \exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)$$

The inverse Ising model

Spins $s_i \in \{-1, 1\}$ sampled with probability

$$P(\boldsymbol{s}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \frac{\exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)}{Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})}$$

Partition function:

$$Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \sum_{\boldsymbol{s}} \exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)$$

Unfortunately computing $Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})$ is intractable. Fortunatelly, we can resort to an accurate approximation called ***pseudo-likelihood***:

$$P(\boldsymbol{s}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \prod_i P(s_i|\boldsymbol{s}\setminus s_i, \boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \prod_i \frac{\exp(\beta s_i \sum_j J_{ij} A_{ij} s_j + h_i s_i)}{2\cosh(\beta \sum_j J_{ij} A_{ij} s_j + h_i)}$$

The inverse Ising model

Spins $s_i \in \{-1, 1\}$ sampled with probability

$$P(\boldsymbol{s}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \frac{\exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)}{Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})}$$

Partition function:

$$Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \sum_{\boldsymbol{s}} \exp\left(\beta \sum_{i<j} J_{ij} A_{ij} s_i s_j + \sum_i h_i s_i\right)$$

Unfortunately computing $Z(\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})$ is intractable. Fortunatelly, we can resort to an accurate approximation called ***pseudo-likelihood***:

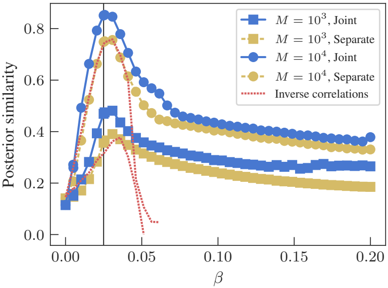
$$P(\boldsymbol{s}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \prod_i P(s_i|\boldsymbol{s} \setminus s_i, \boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \prod_i \frac{\exp(\beta s_i \sum_j J_{ij} A_{ij} s_j + h_i s_i)}{2\cosh(\beta \sum_j J_{ij} A_{ij} s_j + h_i)}$$

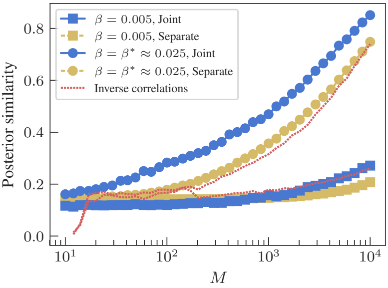
We observe a set of M microstates $\bar{\boldsymbol{s}} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_M\}$, with likelihood $P(\bar{\boldsymbol{s}}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) = \prod_l P(\boldsymbol{s}_l|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h})$, and perform reconstruction with the posterior

$$\boxed{P(\boldsymbol{A}, \boldsymbol{b}, \beta, \boldsymbol{J}, \boldsymbol{h}|\bar{\boldsymbol{s}}) = \frac{P(\bar{\boldsymbol{s}}|\boldsymbol{A}, \beta, \boldsymbol{J}, \boldsymbol{h}) P(\beta) P(\boldsymbol{h}) P(\boldsymbol{J}|\boldsymbol{A}) P(\boldsymbol{A}|\boldsymbol{b}) P(\boldsymbol{b})}{P(\bar{\boldsymbol{s}})}}$$

The inverse Ising model

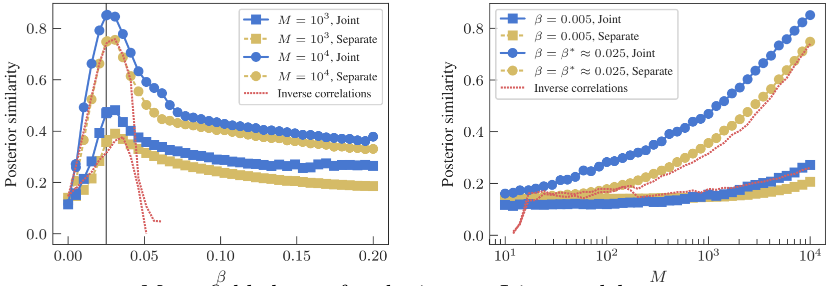
Simulated Ising model on a food web from Little Rock Lake, containing $N = 183$ nodes and $E = 2,434$ edges.





The inverse Ising model

Simulated Ising model on a food web from Little Rock Lake, containing $N = 183$ nodes and $E = 2,434$ edges.

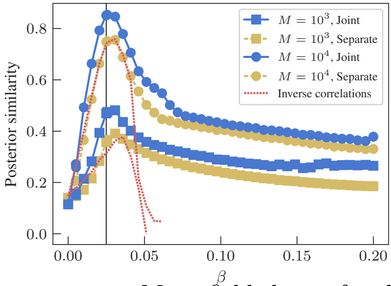


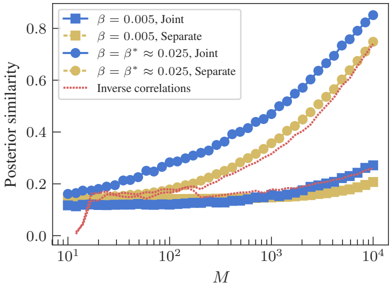
Mean-field theory for the inverse Ising model:

$$P(\boldsymbol{s}) = \prod_i \frac{1 + m_i s_i}{2}, \quad m_i = \langle s_i \rangle$$

The inverse Ising model

Simulated Ising model on a food web from Little Rock Lake, containing $N = 183$ nodes and $E = 2,434$ edges.





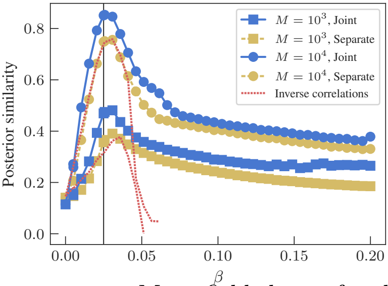
Mean-field theory for the inverse Ising model:

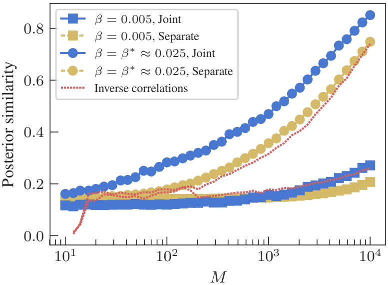
$$P(\boldsymbol{s}) = \prod_i \frac{1 + m_i s_i}{2}, \quad m_i = \langle s_i \rangle$$

$$G(\boldsymbol{J}, \boldsymbol{m}) = -\sum_{i<j} J_{ij} m_i m_j + \sum_i \left[\frac{1+m_i}{2} \ln \frac{1+m_i}{2} + \frac{1-m_i}{2} \ln \frac{1-m_i}{2} \right]$$

The inverse Ising model

Simulated Ising model on a food web from Little Rock Lake, containing $N = 183$ nodes and $E = 2,434$ edges.





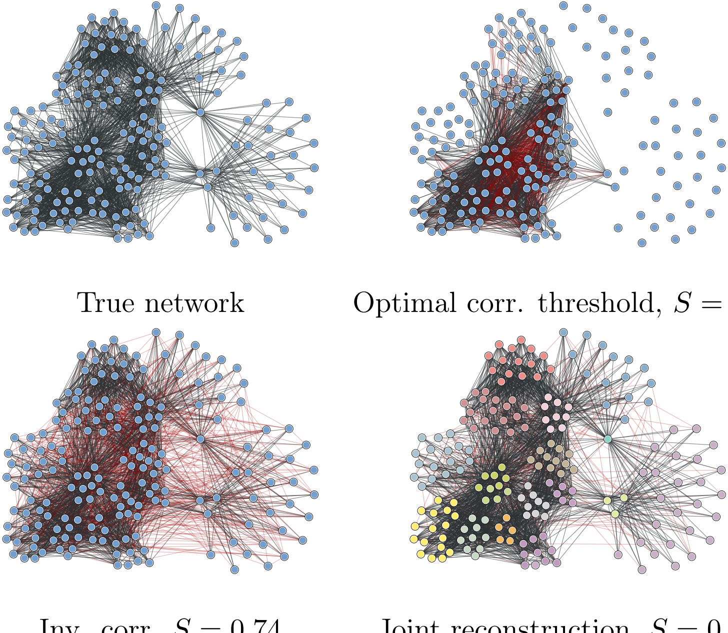
Mean-field theory for the inverse Ising model:

$$P(\boldsymbol{s}) = \prod_i \frac{1 + m_i s_i}{2}, \quad m_i = \langle s_i \rangle$$

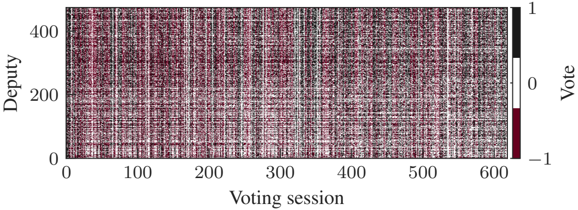
$$G(\boldsymbol{J}, \boldsymbol{m}) = -\sum_{i<j} J_{ij} m_i m_j + \sum_i \left[\frac{1+m_i}{2} \ln \frac{1+m_i}{2} + \frac{1-m_i}{2} \ln \frac{1-m_i}{2} \right]$$

$$\frac{\partial^2 G(\boldsymbol{J}, \boldsymbol{m})}{\partial m_i \partial m_j} = [\boldsymbol{C}^{-1}]_{ij}, \quad C_{ij} = \langle s_i s_j \rangle - m_i m_j \quad \rightarrow \quad \boxed{J_{ij} = -[\boldsymbol{C}^{-1}]_{ij}}$$

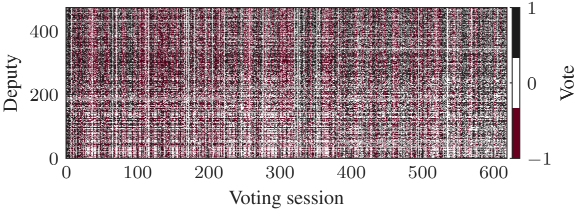
Inverse Ising model

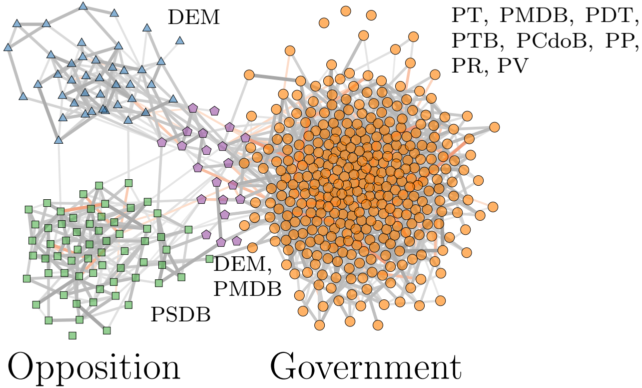


Interactions between members of congress



Interactions between members of congress





Synergy in joint reconstruction

> Finding communities helps reconstruction. But the joint approach also helps to find communities in the first place.

"Planted partition" SBM with

$$\lambda_{rs} = \lambda_{\text{in}}\delta_{rs} + \lambda_{\text{out}}(1 - \delta_{rs}),$$

where

$$\epsilon = N(\lambda_{\text{in}} - \lambda_{\text{out}})/\langle k\rangle B$$

Detectability transition exists at $\epsilon^* = 1/\sqrt{\langle k\rangle}$.

Synergy in joint reconstruction

Finding communities helps reconstruction. But the joint approach also helps to find communities in the first place.

"Planted partition" SBM with

$$\lambda_{rs} = \lambda_{\text{in}}\delta_{rs} + \lambda_{\text{out}}(1-\delta_{rs}),$$

where

$$\epsilon = N(\lambda_{\text{in}} - \lambda_{\text{out}})/\langle k\rangle B$$

Detectability transition exists at $\epsilon^* = 1/\sqrt{\langle k\rangle}$.

Compare joint approach with the "separate" approach:

$$P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})},\ P(\boldsymbol{A}) \to \text{Erdős-Rényi}$$

Synergy in joint reconstruction

Finding communities helps reconstruction. But the joint approach also helps to find communities in the first place.

"Planted partition" SBM with

$$\lambda_{rs} = \lambda_{\text{in}}\delta_{rs} + \lambda_{\text{out}}(1 - \delta_{rs}),$$

where

$$\epsilon = N(\lambda_{\text{in}} - \lambda_{\text{out}})/\langle k\rangle B$$

Detectability transition exists at $\epsilon^* = 1/\sqrt{\langle k\rangle}$.

Compare joint approach with the "separate" approach:

$$P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})},\ P(\boldsymbol{A}) \to \text{Erdős-Rényi}$$

Point estimate:

$$\hat{A}_{ij} = \begin{cases} 1 & \text{if } \pi_{ij} > 1/2 \\ 0 & \text{if } \pi_{ij} < 1/2, \end{cases}$$

Synergy in joint reconstruction

Finding communities helps reconstruction. But the joint approach also helps to find communities in the first place.

"Planted partition" SBM with

$$\lambda_{rs} = \lambda_{\text{in}}\delta_{rs} + \lambda_{\text{out}}(1-\delta_{rs}),$$

where

$$\epsilon = N(\lambda_{\text{in}} - \lambda_{\text{out}})/\langle k\rangle B$$

Detectability transition exists at $\epsilon^* = 1/\sqrt{\langle k\rangle}$.

Compare joint approach with the "separate" approach:

$$P(\boldsymbol{A}|\boldsymbol{\mathcal{D}}) = \frac{P(\boldsymbol{\mathcal{D}}|\boldsymbol{A})P(\boldsymbol{A})}{P(\boldsymbol{\mathcal{D}})},\ P(\boldsymbol{A}) \to \text{Erdős-Rényi}$$

Point estimate:

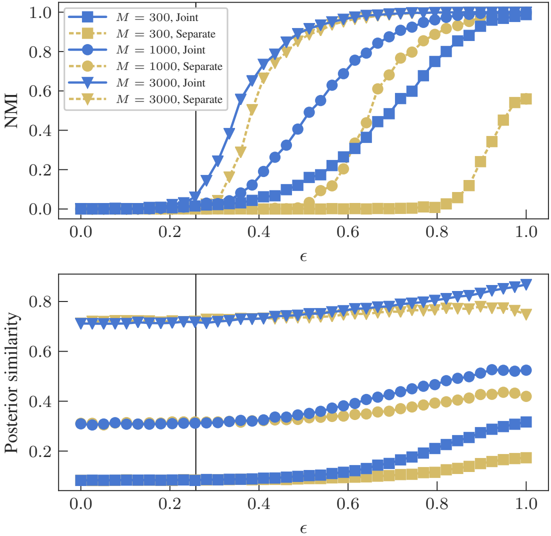
$$\hat{A}_{ij} = \begin{cases} 1 & \text{if } \pi_{ij} > 1/2 \\ 0 & \text{if } \pi_{ij} < 1/2, \end{cases}$$

SBM inference:

$$P(\boldsymbol{b}|\hat{\boldsymbol{A}}) = \frac{P(\hat{\boldsymbol{A}}|\boldsymbol{b})P(\boldsymbol{b})}{P(\hat{\boldsymbol{A}})}$$

Synergy in joint reconstruction

$N = 1000$, $\langle k \rangle = 15$, $B = 10$, Ising model at critical temperature.



T. P. P., Phys. Rev. X 8 041011 (2018)

T. P. P., Phys. Rev. Lett. 123 128301 (2019)

For code, see:

`https://graph-tool.skewed.de`

(See also HOWTO at: `https://graph-tool.skewed.de/static/doc/demos/inference/inference.html`)