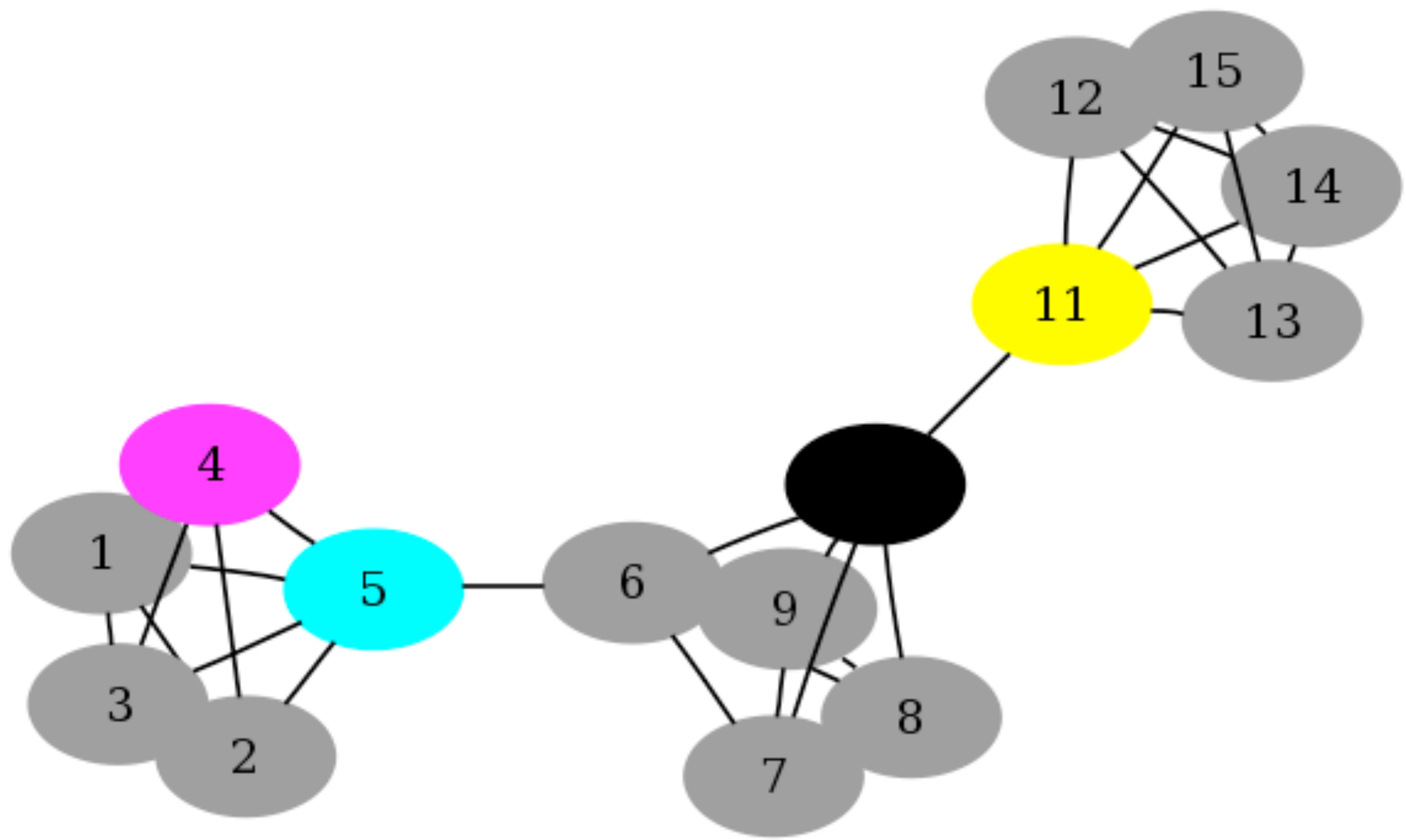
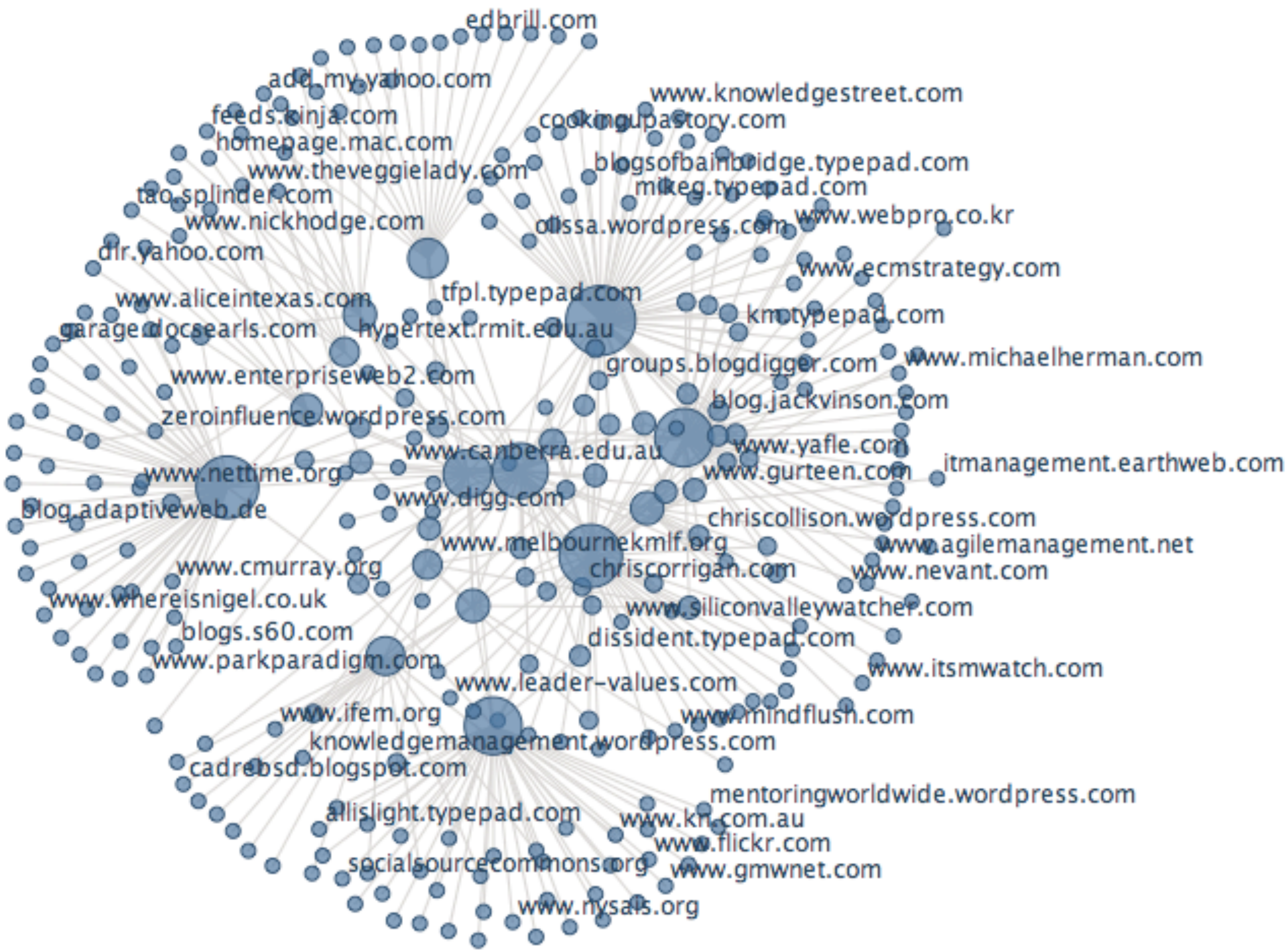


Inferring the structure and scale of modular networks

Jake Hofman
Wiggins Group
Columbia University
2008.07.04

Network modularity (community detection)



Identify groups of "similar"
nodes using network topology?

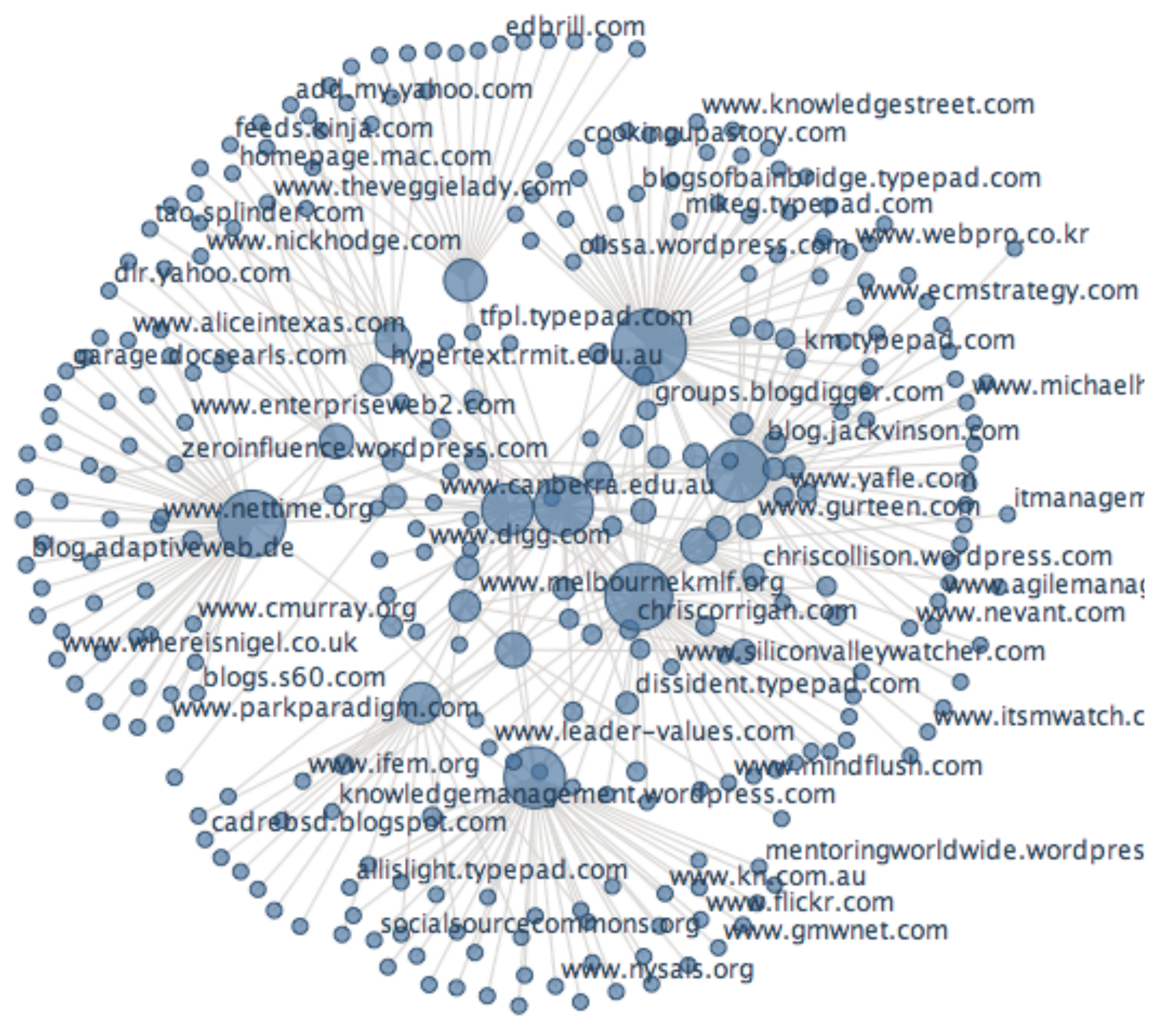
Image from IBM Manyeyes

Outline

- Motivation/background
- Bayesian inference and complexity control
- Generating and inferring modular networks
- Validation and applications

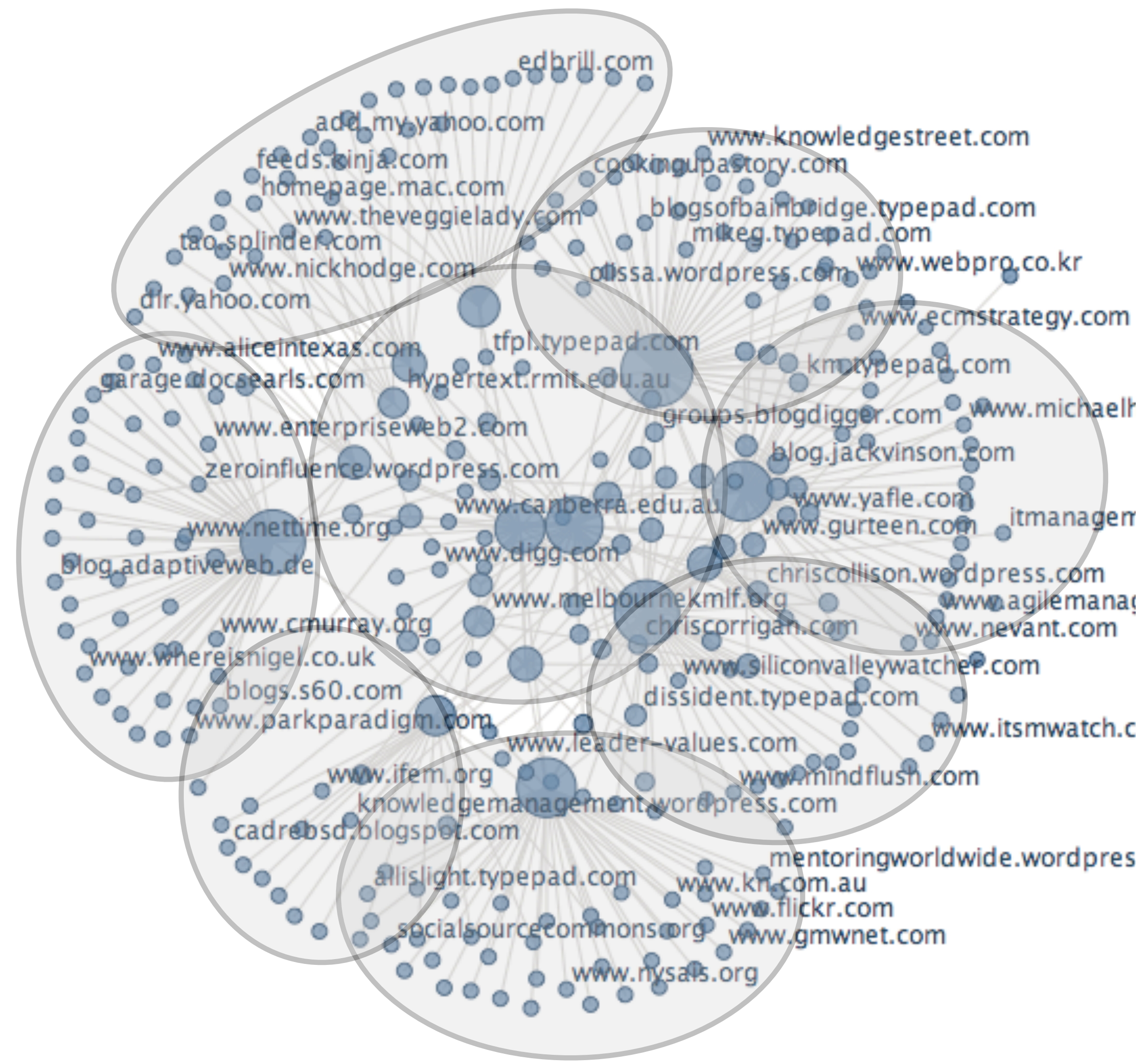
Motivation

- *Model* structure (e.g. summarize data)
- *Visualize* structure (e.g. graph layout)
- *Analyze* interactions (e.g. affinities within/between groups)
- *Explore* interactions (e.g. recommendations)



Motivation

- *Model* structure (e.g. summarize data)
- *Visualize* structure (e.g. graph layout)
- *Analyze* interactions (e.g. affinities within/between groups)
- *Explore* interactions (e.g. recommendations)

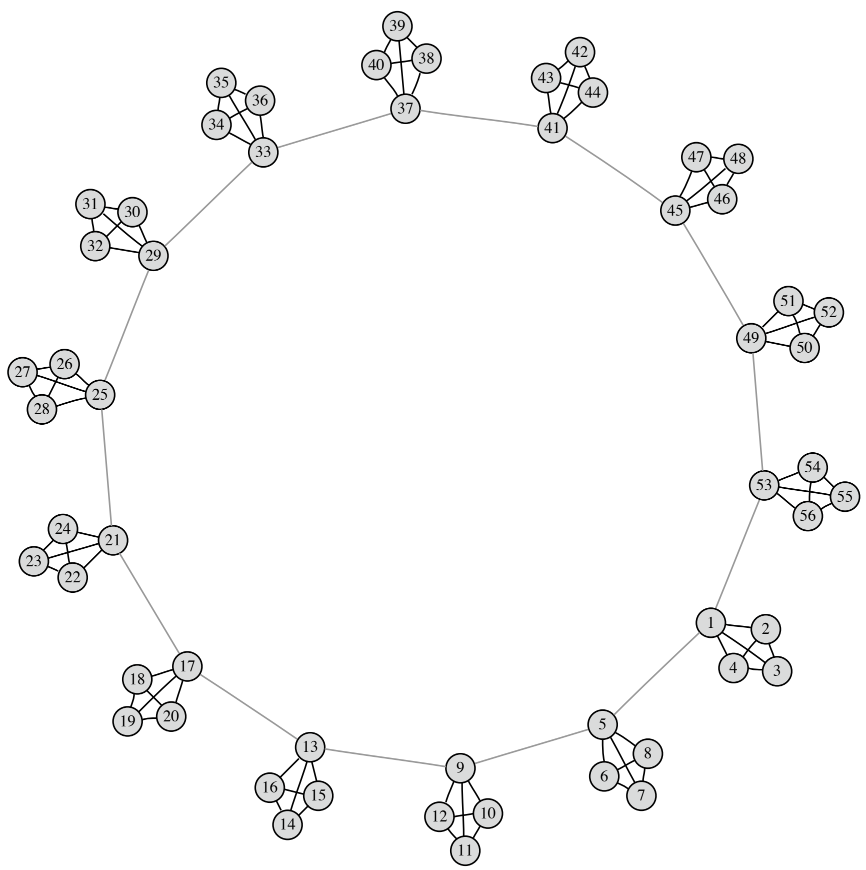


Background

- Physics literature
  - Newman et. al. (2002, 2004)
  - Bornholdt & Reichardt (2006)
  - Hastings (2006)
  - ...
- Parametrized cost function (energy), mostly focus on *how* to optimize

- Machine learning literature
  - Nowicki & Snijders (2001)
  - Kemp et. al. (2004)
  - Airoldi et. al. (2007)
  - Xu et. al. (2007)
  - Sinkkonen et. al. (2007)
  - ...
- Complex models, approximate inference (often expensive)

The “resolution limit” problem

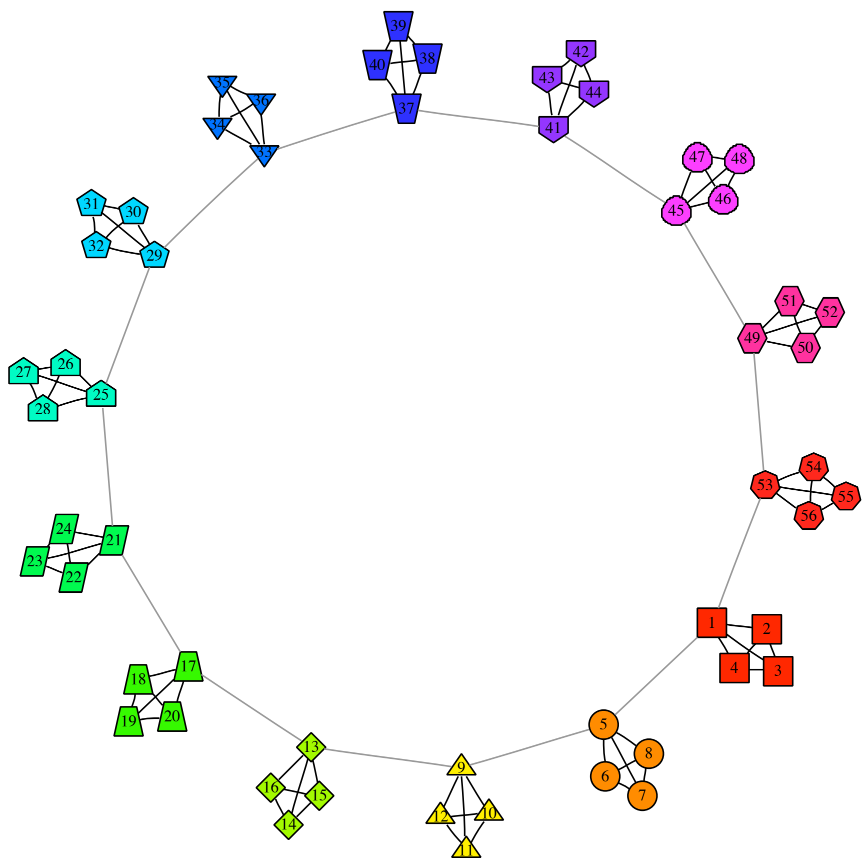


$$\mathcal{H} = -\sum_{ij}(A_{ij} - \gamma p_{ij})\delta_{z_i,z_j}$$

Girvan & Newman (2004), Reichardt & Bornholdt (2006)

Fortunato et. al. (2007), Kumpula et. al. (2007)

The “resolution limit” problem

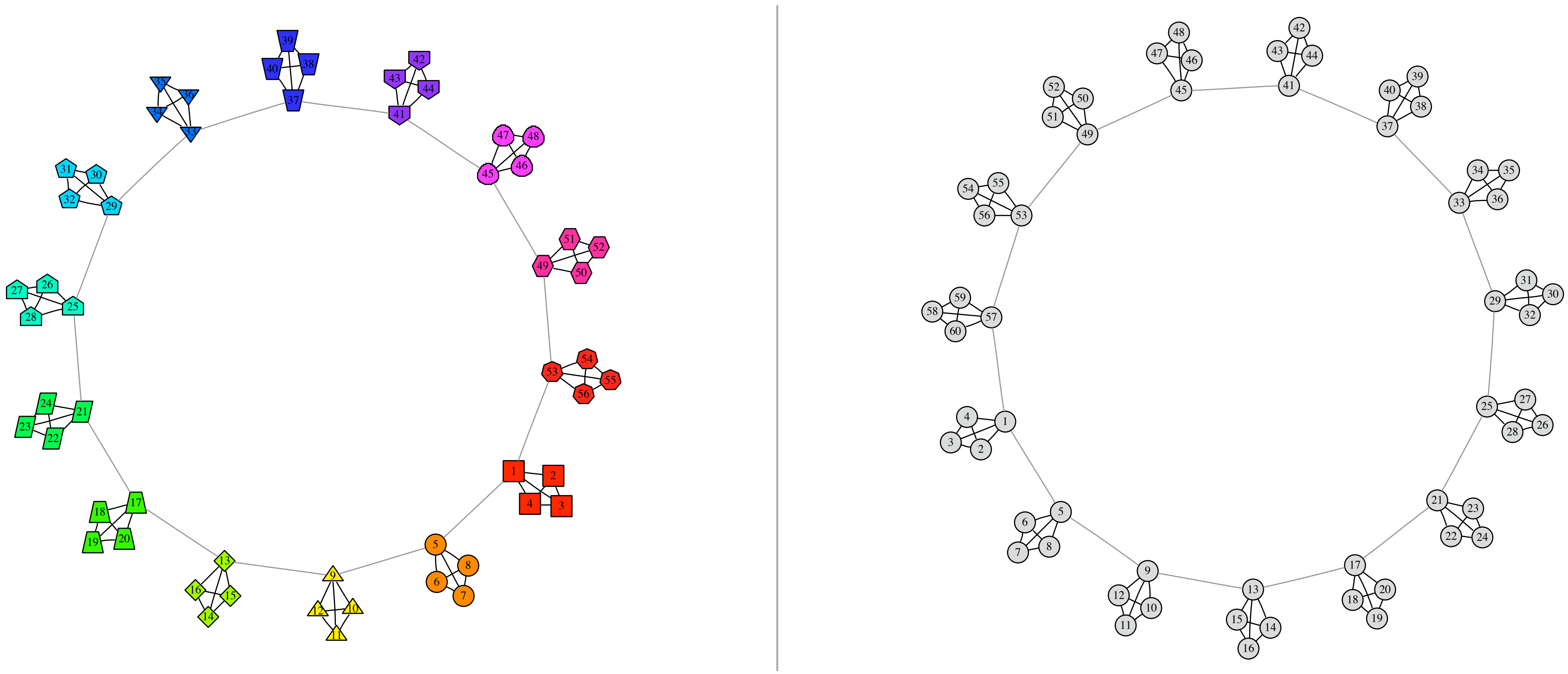


$$\mathcal{H} = -\sum_{ij} (A_{ij} - \gamma p_{ij}) \delta_{z_i, z_j}$$

Girvan & Newman (2004), Reichardt & Bornholdt (2006)

Fortunato et. al. (2007), Kumpula et. al. (2007)

The “resolution limit” problem



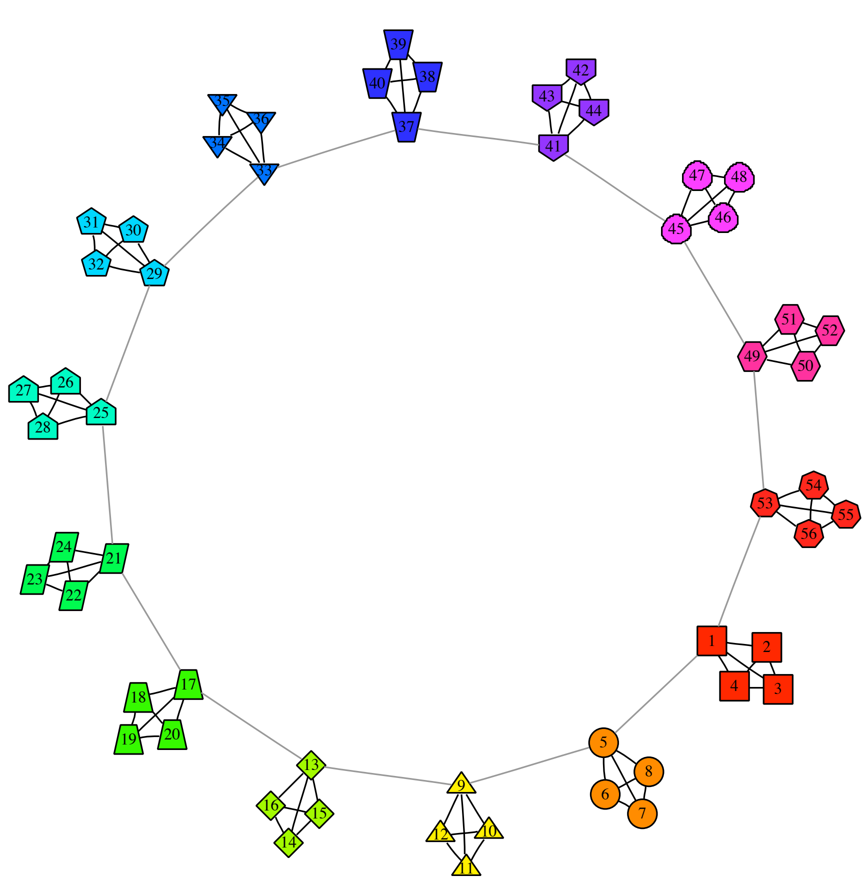
$$\mathcal{H} = -\sum_{ij}(A_{ij} - \gamma p_{ij})\delta_{z_i,z_j}$$

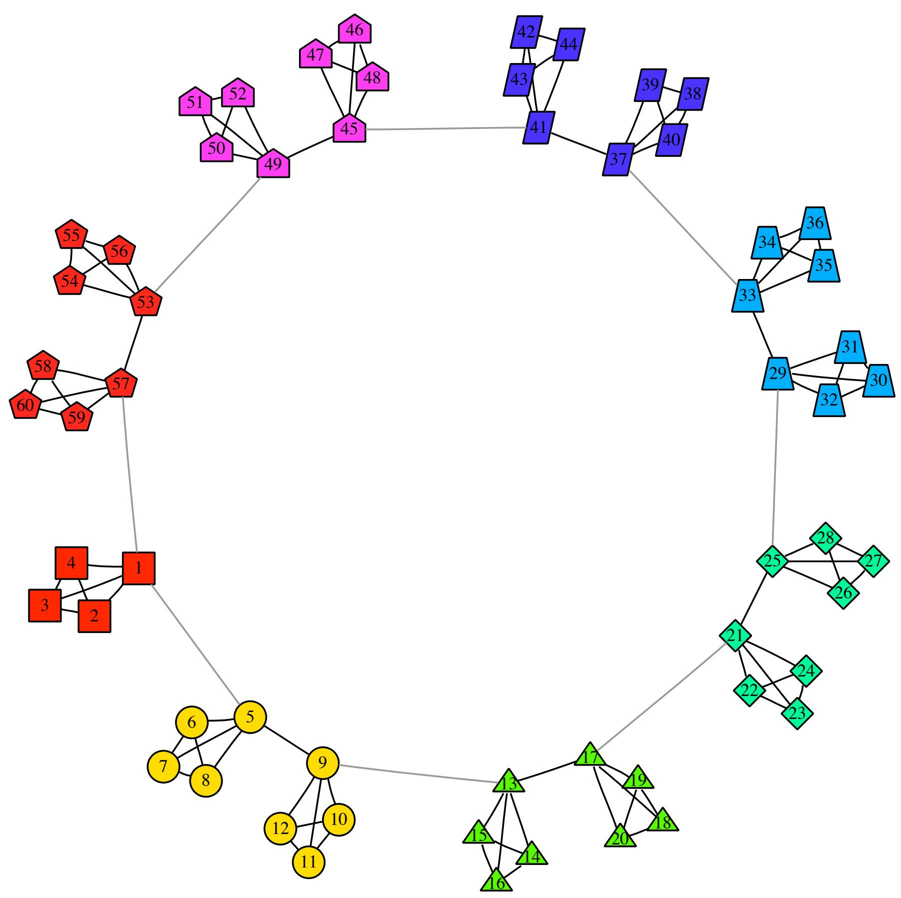
Girvan & Newman (2004), Reichardt & Bornholdt (2006)

Fortunato et. al. (2007), Kumpula et. al. (2007)

The "resolution limit" problem

Fixed parameters -> fixed resolution or *complexity*



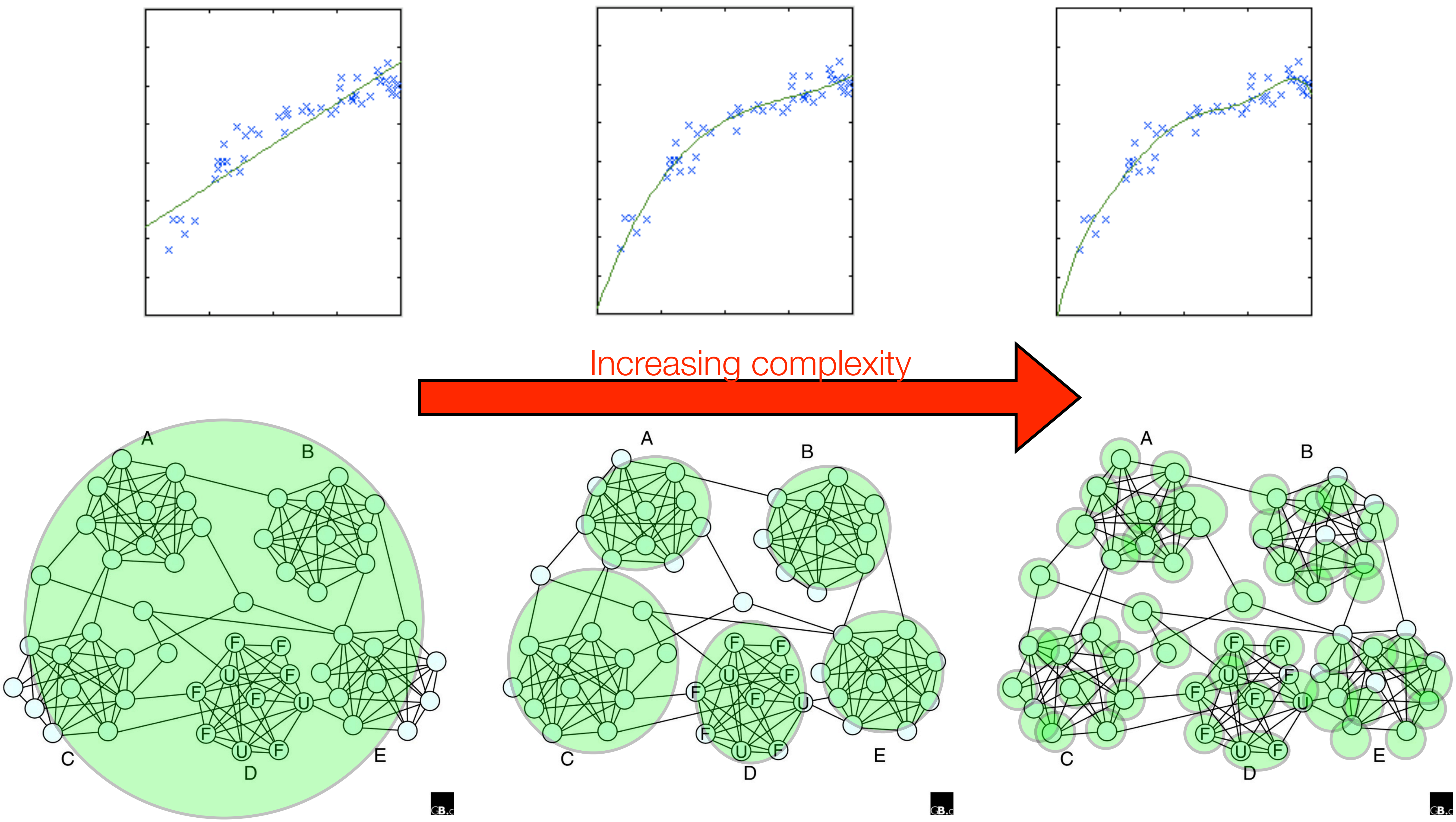


$$\mathcal{H} = -\sum_{ij} (A_{ij} - \gamma p_{ij})\delta_{z_i,z_j}$$

Girvan & Newman (2004), Reichardt & Bornholdt (2006)

Fortunato et. al. (2007), Kumpula et. al. (2007)

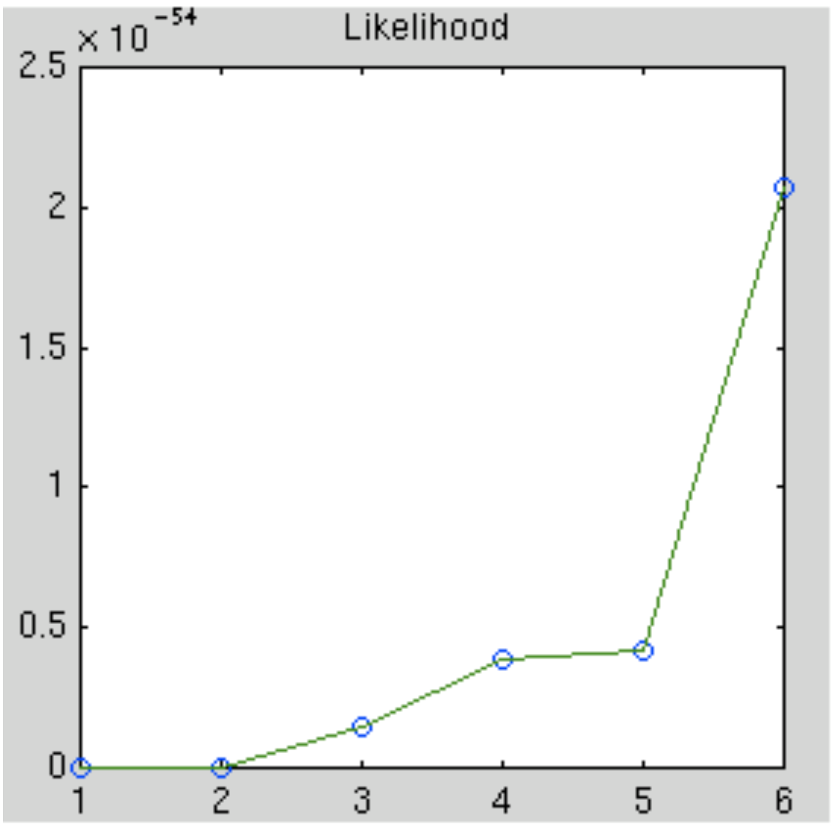
Complexity control in probabilistic models

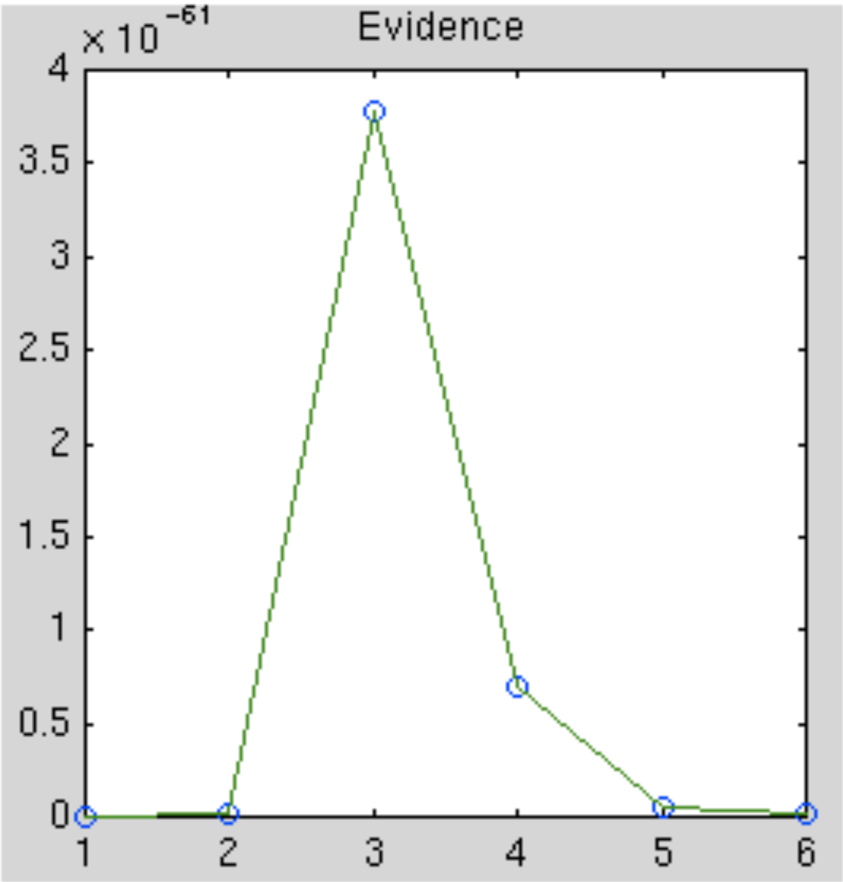


Images from http://research.microsoft.com/~minka/statlearn/demo/ and Giorgini, et. al., Genome Biol, 2005

Bayesian complexity control

- Maximize evidence (integrating over unknown parameters and latent variables) to infer most probable model complexity





$$\hat{\theta} = \arg\max_{\theta} p(\mathcal{D}|\theta, K)$$
$$= \arg\max_{\theta} \sum_{Z} p(\mathcal{D}, Z|\hat{\theta}, K)$$

$$\hat{K} = \arg\max_{K} p(\mathcal{D}|K)$$
$$= \arg\max_{K} \sum_{Z} \int d\theta \; p(\mathcal{D}, Z|\theta, K) p(\theta|K)$$

http://research.microsoft.com/~minka/statlearn/demo/

Bayesian complexity control

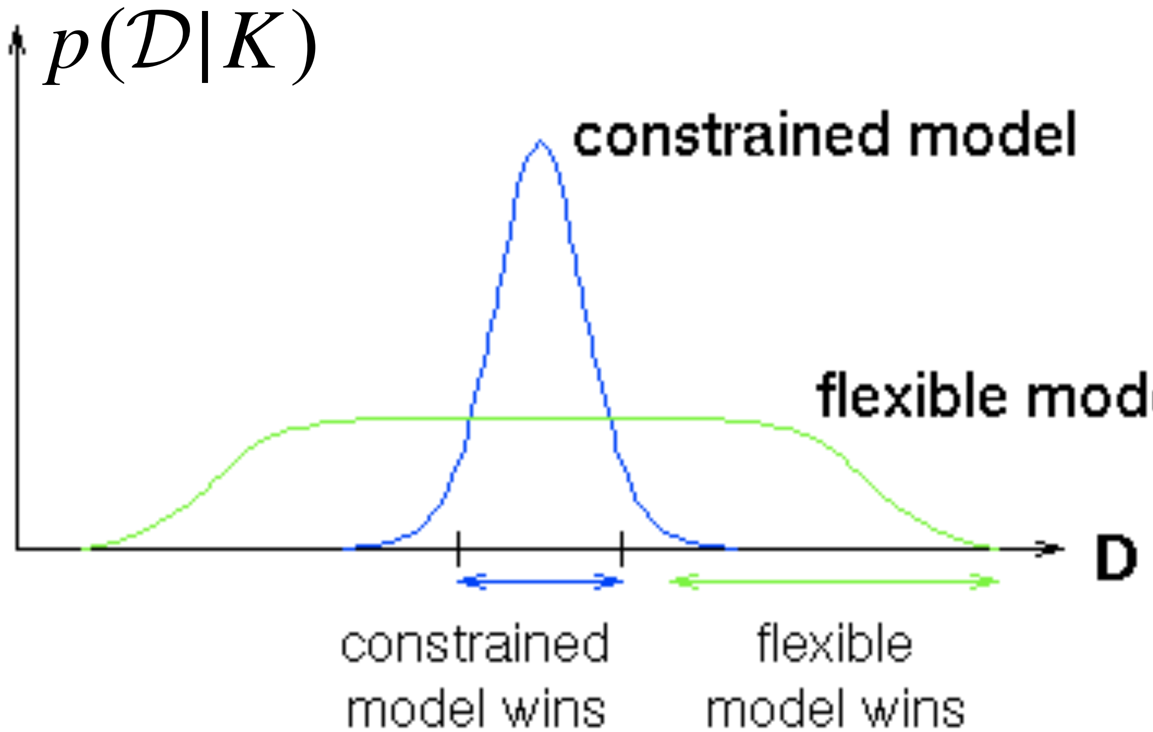
- Find ***most probable complexity*** K, given data D, integrating over unknowns

$$p(K|\mathcal{D}) = \frac{p(\mathcal{D}|K)\,p(K)}{p(\mathcal{D})}$$

- If p(K) sufficiently weak, maximize evidence to find optimal complexity

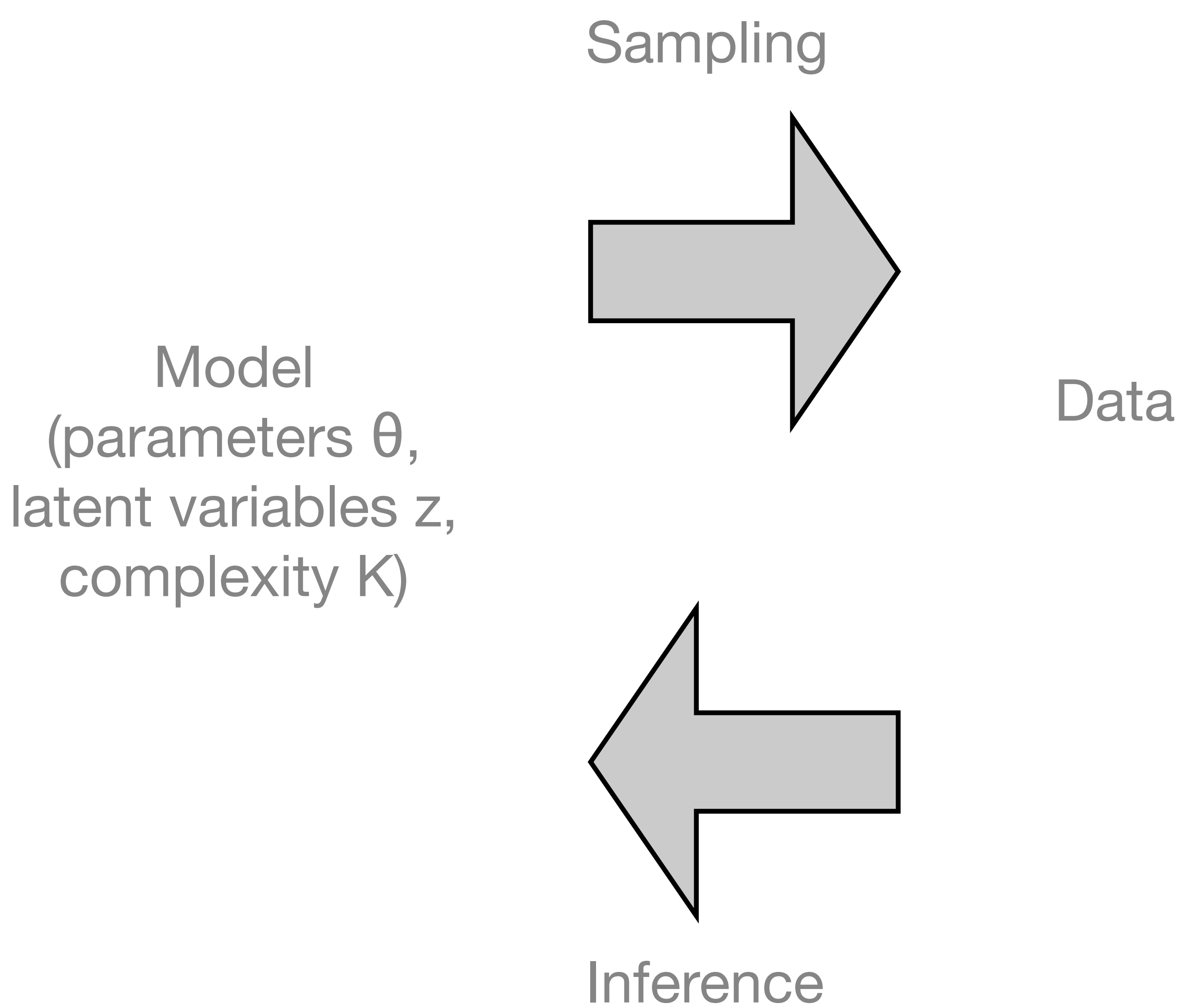
evidence

$$p(\mathcal{D}|K) = \int d\theta \; p(\mathcal{D}|\theta, K)\,p(\theta|K)$$



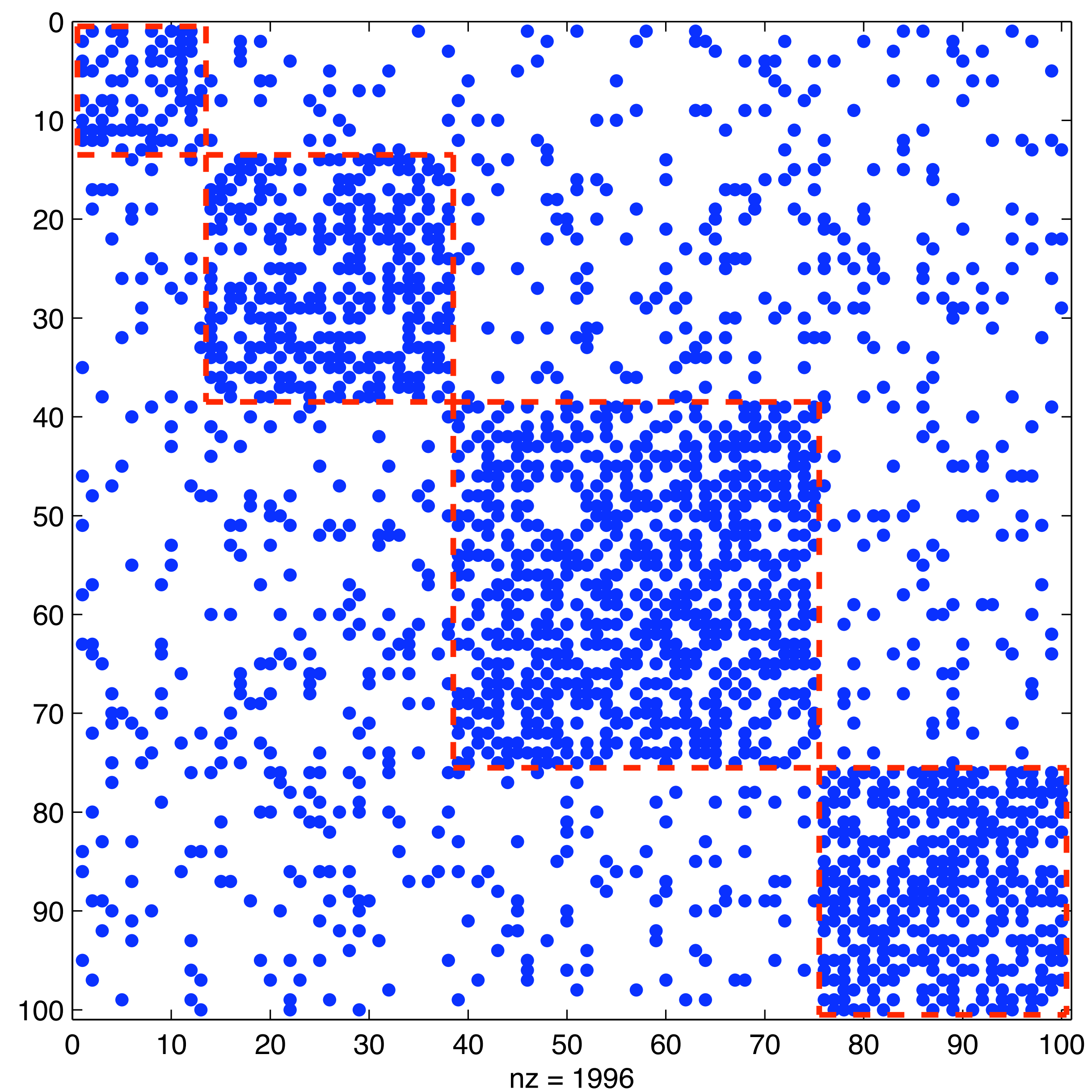
http://research.microsoft.com/~minka/statlearn/demo/

Community detection as inference



Stochastic Block Models

- **Nodes belong to "blocks"** of varying size
  - Roll die for assignment of nodes to blocks
- Probability of **edge** between two nodes **depends only on block membership**
  - Flip (one of two) coins for edges
- Result: **mixture of Erdos-Renyi** graphs



Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987

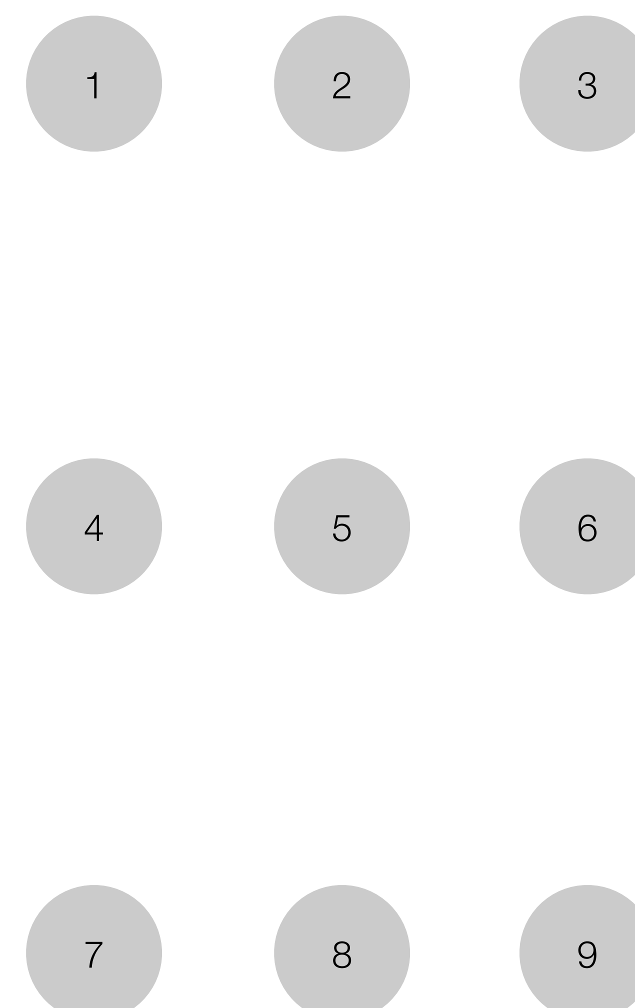
Generating modular networks

- For each node:
  - **Roll K-sided die** with bias π to determine $z_i=1,\ldots,K$, the (unobserved) module assignment for i^{th} node
- For each pair of nodes (i,j):
  - If $z_i=z_j$, **flip "in community" coin** with bias θ_c to determine edge A_{ij}
  - If $z_i \neq z_j$, **flip "between communities" coin** with bias θ_d to determine edge A_{ij}

Stochastic block models (Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987)

Generating modular networks

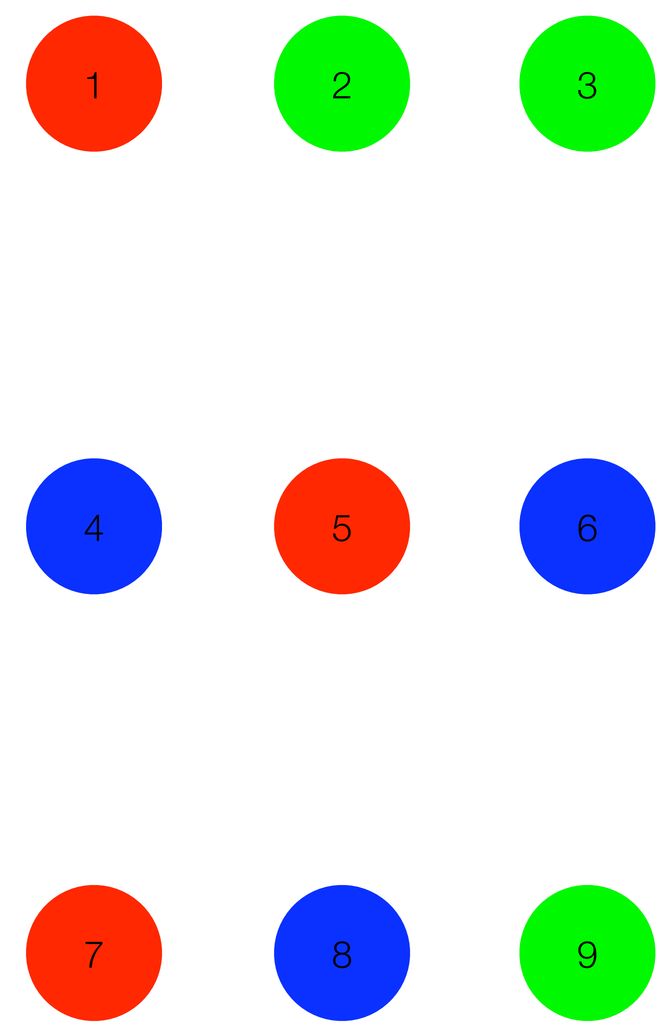
- For each node:
  - **Roll K-sided die** with bias π to determine $z_i=1,...,K$, the (unobserved) module assignment for i^{th} node
- For each pair of nodes (i,j):
  - If $z_i=z_j$, **flip "in community" coin** with bias θ_c to determine edge A_{ij}
  - If $z_i \neq z_j$, **flip "between communities" coin** with bias θ_d to determine edge A_{ij}



Stochastic block models (Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987)

Generating modular networks

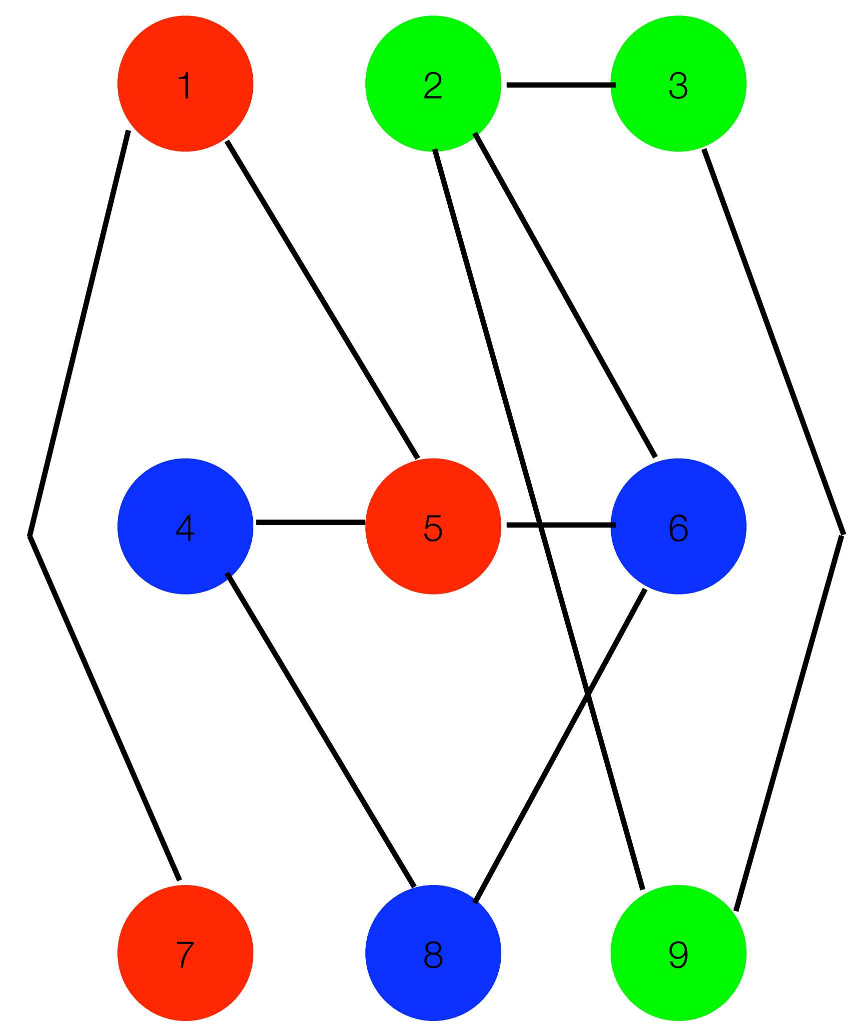
- For each node:
  - **Roll K-sided die** with bias π to determine z_i=1,...,K, the (unobserved) module assignment for i^{th} node
- For each pair of nodes (i,j):
  - If $z_i = z_j$, **flip "in community" coin** with bias θ_c to determine edge A_{ij}
  - If $z_i \neq z_j$, **flip "between communities" coin** with bias θ_d to determine edge A_{ij}



Stochastic block models (Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987)

Generating modular networks

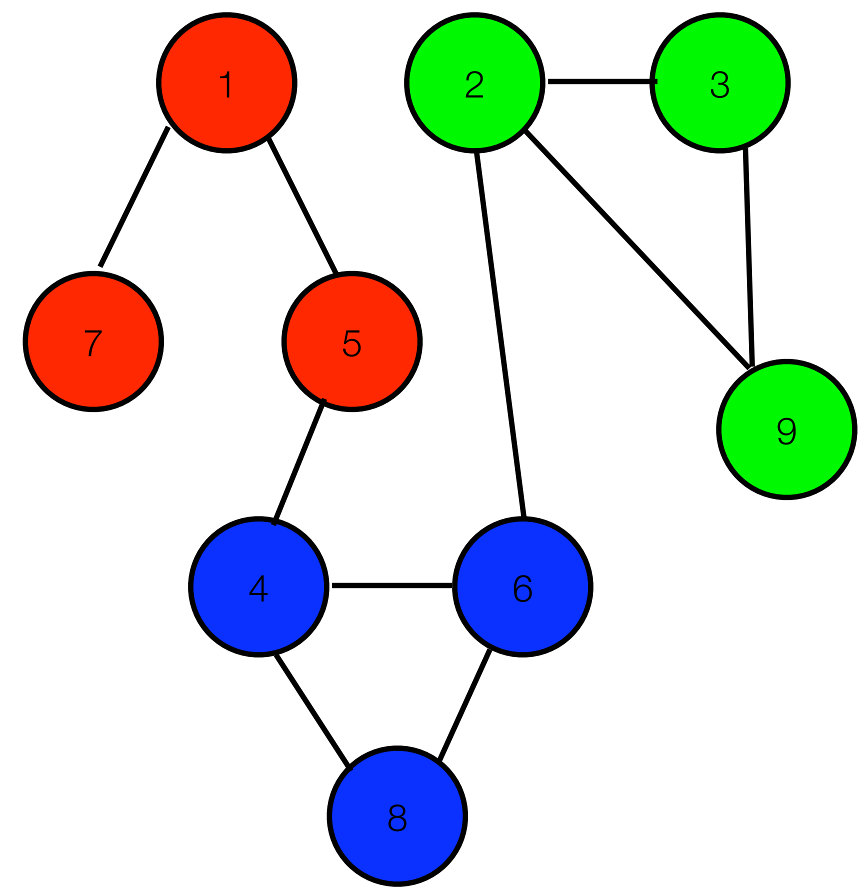
- For each node:
  - **Roll K-sided die** with bias π to determine z_i=1,...,K, the (unobserved) module assignment for i^{th} node
- For each pair of nodes (i,j):
  - If $z_i=z_j$, **flip "in community" coin** with bias θ_c to determine edge A_{ij}
  - If $z_i \neq z_j$, **flip "between communities" coin** with bias θ_d to determine edge A_{ij}



Stochastic block models (Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987)

Generating modular networks

- For each node:
  - **Roll K-sided die** with bias π to determine z_i=1,...,K, the (unobserved) module assignment for i^{th} node
- For each pair of nodes (i,j):
  - If $z_i=z_j$, **flip "in community" coin** with bias θ_c to determine edge A_{ij}
  - If $z_i \neq z_j$, **flip "between communities" coin** with bias θ_d to determine edge A_{ij}



Stochastic block models (Holland, Laskey, Leinhardt 1983; Wang and Wong, 1987)

Generating modular networks

Die rolling, coin flipping, and priors:

$$p(\vec{z}|\vec{\pi}) \equiv \prod_{\mu=1}^{K} \pi_{\mu}^{n_\mu}$$

$$p(\mathbf{A}|\vec{z},\vec{\pi},\vec{\theta}) \equiv \theta_c^{c_+}(1-\theta_c)^{c_-}\theta_d^{d_+}(1-\theta_d)^{d_-}$$

$$p(\vec{\theta}) \equiv \mathcal{B}(\theta_c;\tilde{c}_{+_0},\tilde{c}_{-_0})\mathcal{B}(\theta_d;\tilde{d}_{+_0},\tilde{d}_{-_0})$$

$$p(\vec{\pi}) \equiv \mathcal{D}(\vec{\pi};\tilde{\vec{n}})$$

where counts are:

edges within modules
$$c_+ \equiv \sum_{i,j} A_{ij}\delta_{z_i,z_j}$$

non-edges within modules
$$c_- \equiv \sum_{i,j}(1-A_{ij})\delta_{z_i,z_j}$$

edges between modules
$$d_+ \equiv \sum_{i,j} A_{ij}(1-\delta_{z_i,z_j})$$

non-edges between modules
$$d_- \equiv \sum_{i,j}(1-A_{ij})(1-\delta_{z_i,z_j})$$

nodes in each module
$$n_\mu \equiv \sum_{i=1}^{N} \delta_{z_i,\mu}$$

Physical analogy

- Statistical mechanics: infinite-range spin-glass Potts model

$$\mathcal{H} \equiv -\ln p(\mathbf{A}, \vec{z}|\vec{\pi}, \vec{\theta}) = -\sum_{i,j}(J_L A_{ij} - J_G)\delta_{z_i,z_j} + \sum_{\mu=1}^{K} h_\mu \sum_{i=1}^{N} \delta_{z_i,\mu}$$

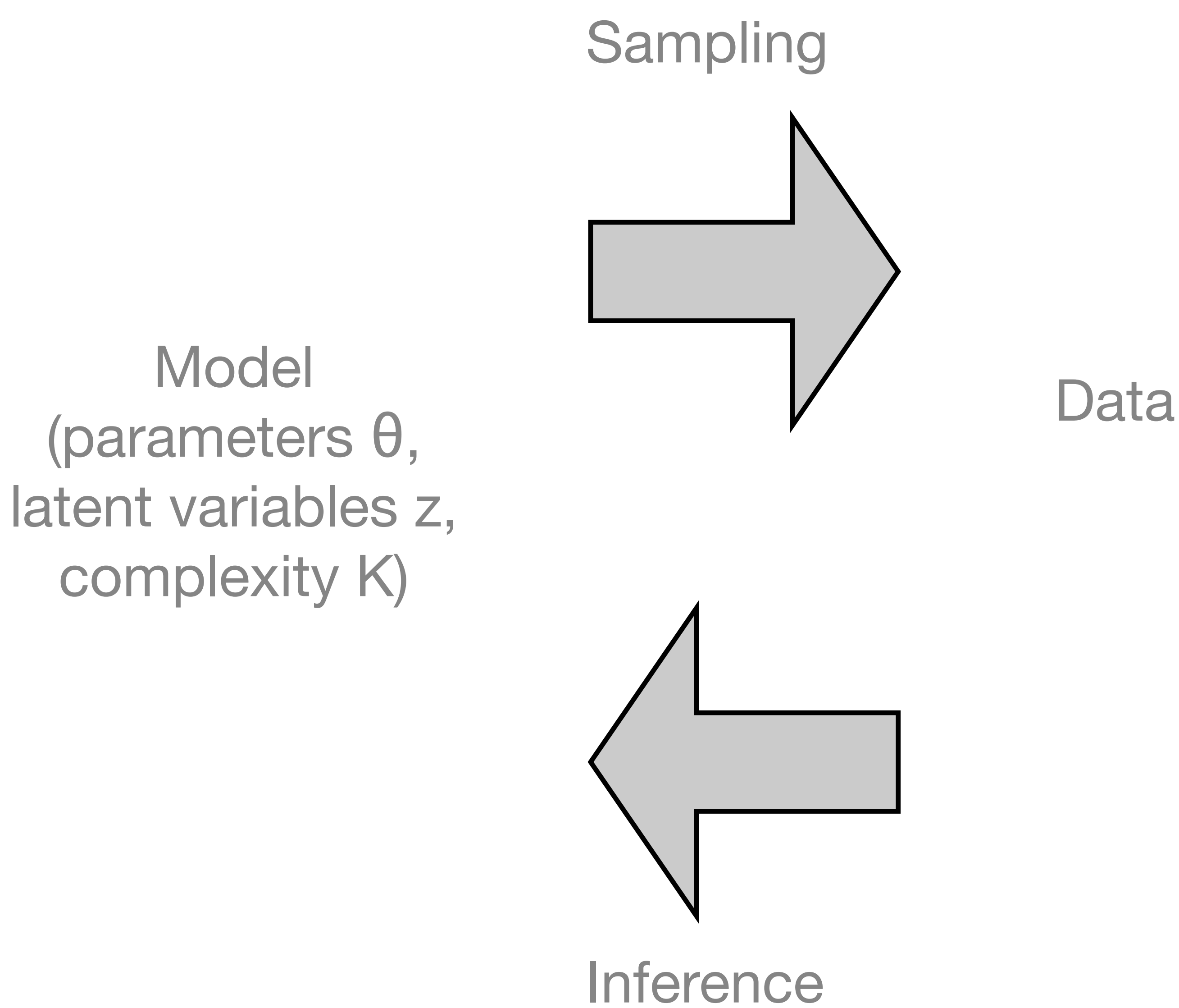
$$\begin{aligned} J_G &\equiv \ln \vartheta_c/\vartheta_d \\ J_L &\equiv \ln(1-\vartheta_d)/(1-\vartheta_c) + J_G \\ h_\mu &\equiv -\ln \pi_\mu \end{aligned}$$

- Infer *distributions* over spin assignments, coupling constants, and chemical potentials and find number of occupied spin states

- Bayesian inference corresponds to calculation of disorder-averaged partition function

Extends Newman (2004,2006), Hastings (2006), Bornholdt & Reichardt (2006)

Community detection as inference

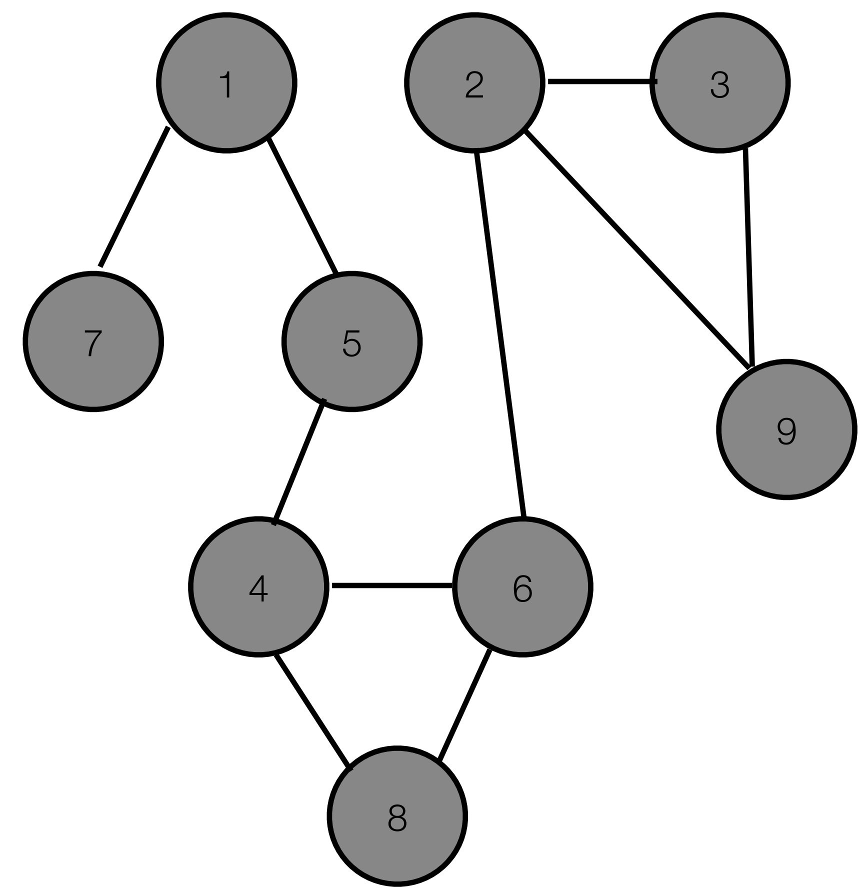


Community detection as inference

- From observed graph structure, infer distributions over module assignments, model parameters, and model complexity

$$p(\vec{\pi},\vec{\theta}|\mathbf{A},K) = \frac{p(\mathbf{A}|\vec{\pi},\vec{\theta},K)p(\vec{\pi},\vec{\theta}|K)}{p(\mathbf{A}|K)}$$

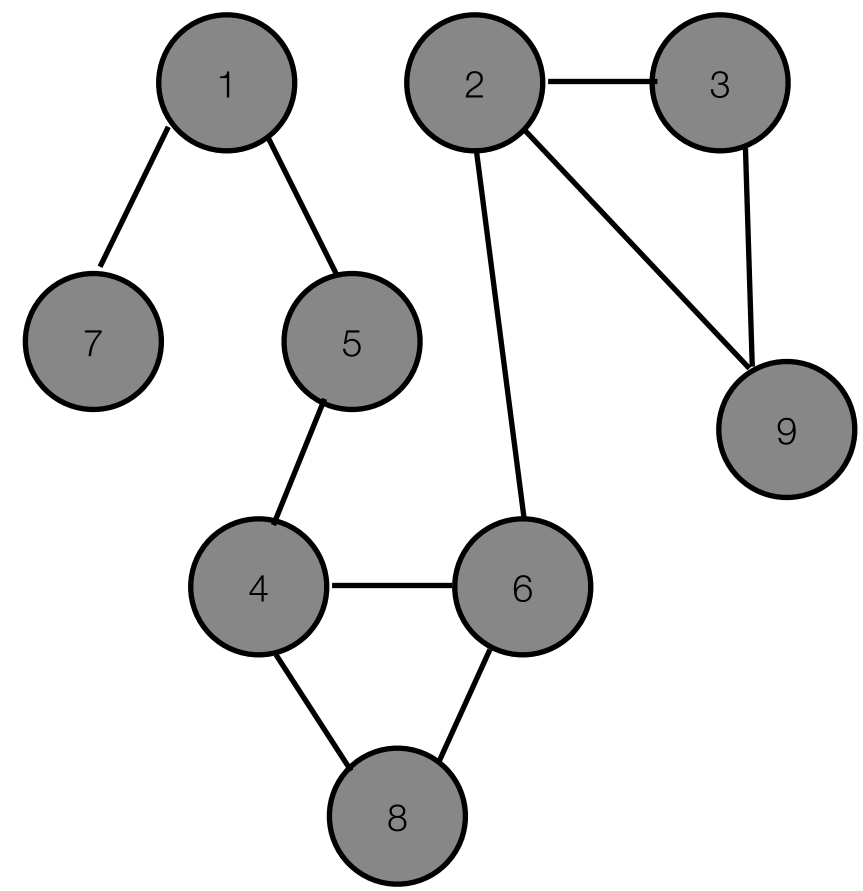
$$p(\vec{z}|\mathbf{A},K) = \frac{p(\mathbf{A}|\vec{z},K)p(\vec{z}|K)}{p(\mathbf{A}|K)}$$



$$p(A|K) = \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ p(\mathbf{A},\vec{z},\vec{\pi},\vec{\theta}) = \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ e^{-\mathcal{H}} p(\vec{\theta})p(\vec{\pi})$$

Community detection as inference

- From observed graph structure, infer distributions over module assignments, model parameters, and model complexity



$$p(\vec{\pi}, \vec{\theta}|\mathbf{A}, K) = \frac{p(\mathbf{A}|\vec{\pi}, \vec{\theta}, K)p(\vec{\pi}, \vec{\theta}|K)}{p(\mathbf{A}|K)}$$

$$p(\vec{z}|\mathbf{A}, K) = \frac{p(\mathbf{A}|\vec{z}, K)p(\vec{z}|K)}{p(\mathbf{A}|K)}$$

$$p(A|K) = \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ p(\mathbf{A}, \vec{z}, \vec{\pi}, \vec{\theta}) = \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ e^{-\mathcal{H}} p(\vec{\theta}) p(\vec{\pi})$$

← Can do integrals, but sum is intractable, $O(K^N)$

Variational Bayes

- Jensen's inequality (log of expected value bounds expected value of log) for any distribution q

$$
\begin{aligned}
-\ln p(\mathbf{A}|K) &= -\ln \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ p(\mathbf{A},\vec{z},\vec{\pi},\vec{\theta}|K) \\
&= -\ln \sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ q(\vec{z},\vec{\pi},\vec{\theta}) \frac{p(\mathbf{A},\vec{z},\vec{\pi},\vec{\theta}|K)}{q(\vec{z},\vec{\pi},\vec{\theta})} \\
&\leq \underbrace{-\sum_{\vec{z}} \int d\vec{\theta} \int d\vec{\pi}\ q(\vec{z},\vec{\pi},\vec{\theta}) \ln \frac{p(\mathbf{A},\vec{z},\vec{\pi},\vec{\theta}|K)}{q(\vec{z},\vec{\pi},\vec{\theta})}}_{F\{q;A\}}
\end{aligned}
$$

Variational Bayes (Feynman 1972; MacKay, Jordan, Ghahramani, Jaakola, Saul 1999)

Approximate inference for modular networks

Initialization.—Initialize the N-by-K matrix $\mathbf{Q} = \mathbf{Q}_0$ and set pseudocounts $\tilde{c}_+ = \tilde{c}_{+_0}, \tilde{c}_- = \tilde{c}_{-_0}, \tilde{d}_+ = \tilde{d}_{+_0}, \tilde{d}_- = \tilde{d}_{-_0}$, and $\tilde{n}_\mu = \tilde{n}_{\mu_0}$.

Main Loop.—Until convergence in $F\{q; \mathbf{A}\}$:

(i) Update the expected value of the coupling constants and chemical potentials

$$\begin{aligned} \langle J_L \rangle &= \psi(\tilde{c}_+) - \psi(\tilde{c}_-) - \psi(\tilde{d}_+) + \psi(\tilde{d}_-) & (8) \\ \langle J_G \rangle &= \psi(\tilde{d}_-) - \psi(\tilde{d}_+ + \tilde{d}_-) \\ &\quad -\psi(\tilde{c}_-) + \psi(\tilde{c}_+ + \tilde{c}_-) & (9) \\ \langle h_\mu \rangle &= \psi\left(\sum_\mu \tilde{n}_\mu\right) - \psi(\tilde{n}_\mu), & (10) \end{aligned}$$

where $\psi(x)$ is the digamma function;

(ii) Update the variational distribution over each spin σ_i

$$Q_{i\mu} \propto \exp\left\{\sum_{j \neq i} \left[\langle J_L \rangle A_{ij} - \langle J_G \rangle\right] Q_{j\mu} - \langle h_\mu \rangle\right\} \quad (11)$$

normalized such that $\sum_\mu Q_{i\mu} = 1$, for all i;

(iii) Update the variational distribution over parameters from the expected counts and pseudocounts

$$\begin{aligned} \tilde{n}_\mu &= \langle n_\mu \rangle + \tilde{n}_{\mu_0} = \sum_{i=1}^N Q_{i\mu} + \tilde{n}_{\mu_0} & (12) \\ \tilde{c}_+ &= \langle c_+ \rangle + \tilde{c}_{+_0} = \frac{1}{2} Tr(\mathbf{Q}^T \mathbf{A} \mathbf{Q}) + \tilde{c}_{+_0} & (13) \\ \tilde{c}_- &= \langle c_- \rangle + \tilde{c}_{-_0} \\ &= \frac{1}{2} Tr(\mathbf{Q}^T (\vec{u} \langle \vec{n} \rangle^T - \mathbf{Q})) - \langle c_+ \rangle + \tilde{c}_{-_0} & (14) \\ \tilde{d}_+ &= \langle d_+ \rangle + \tilde{d}_{+_0} = M - \langle c_+ \rangle + \tilde{d}_{+_0} & (15) \\ \tilde{d}_- &= \langle d_- \rangle + \tilde{d}_{-_0} = C - M - \langle c_- \rangle + \tilde{d}_{-_0}, & (16) \end{aligned}$$

where $C = N(N-1)/2$, $M = \sum_{i>j} A_{ij}$, and $\vec{u}$ is a N-by-1 vector of 1's;

(iv) Calculate the updated optimized free energy

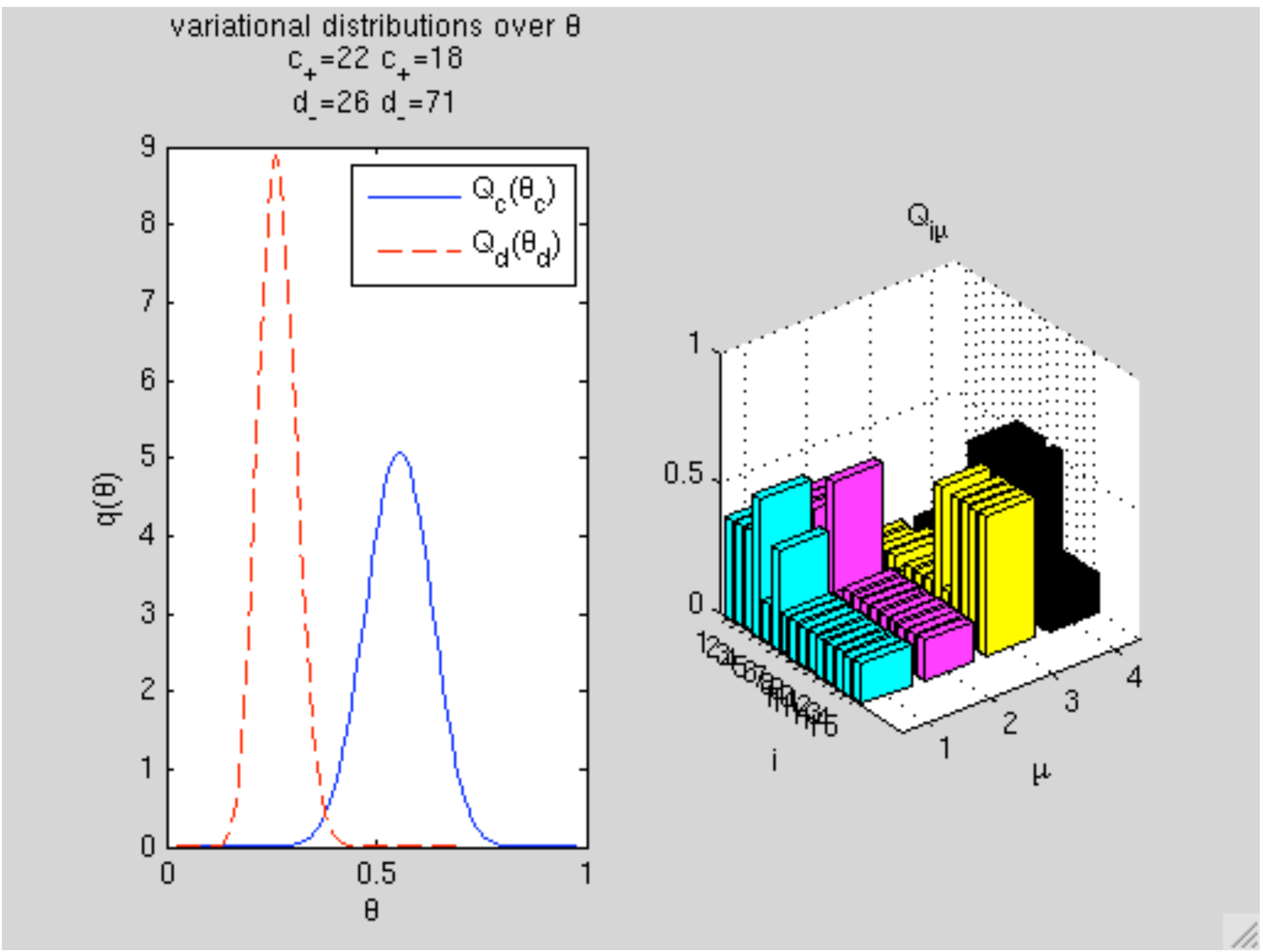
$$F\{q; \mathbf{A}\} = -\ln \frac{\mathcal{Z}_c \mathcal{Z}_d \mathcal{Z}_{\vec{\pi}}}{\tilde{\mathcal{Z}}_c \tilde{\mathcal{Z}}_d \tilde{\mathcal{Z}}_{\vec{\pi}}} + \sum_{\mu=1}^K \sum_{i=1}^N Q_{i\mu} \ln Q_{i\mu}, \quad (17)$$

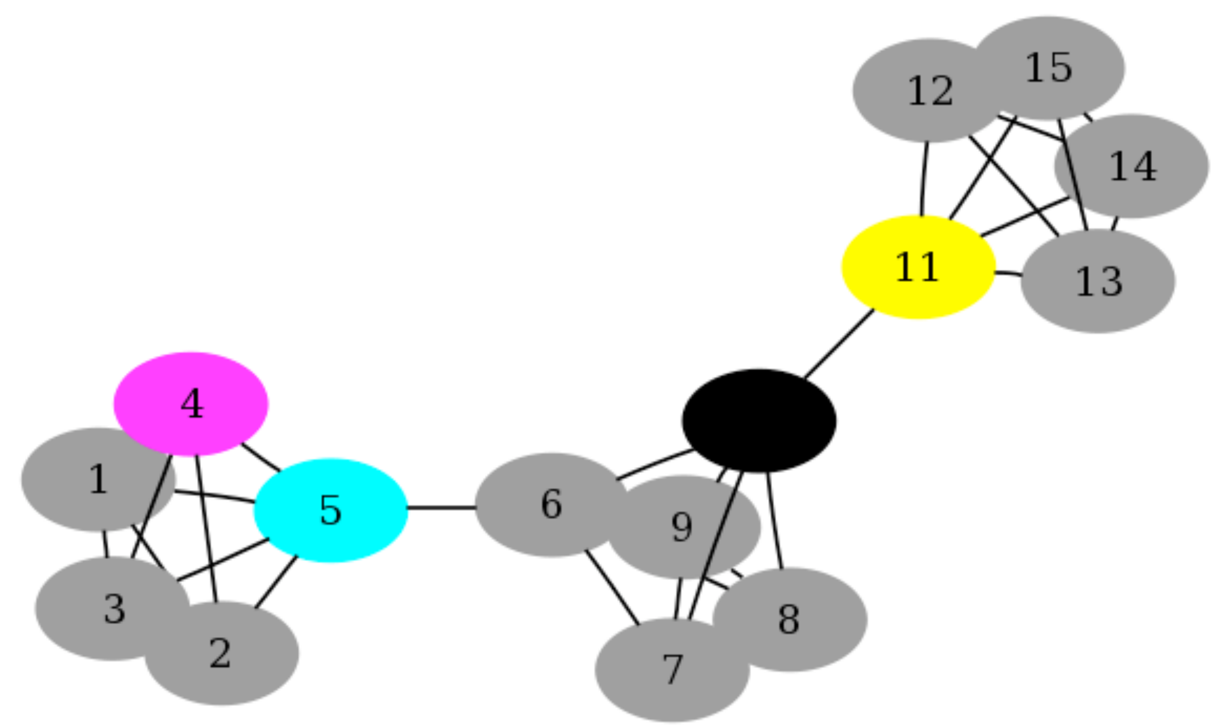
where $\mathcal{Z}_{\vec{\pi}} = B(\tilde{\vec{n}})$ is the beta function with a vector-valued argument, the partition function for the Dirichlet distribution $q_{\vec{\pi}}(\vec{\pi})$ (likewise for $q_c(\vartheta_c), q_d(\vartheta_d)$).

- Iteratively optimize F{q;A} by updating distributions over parameters {π, θ} and latent variables {z}

Validation: complexity control

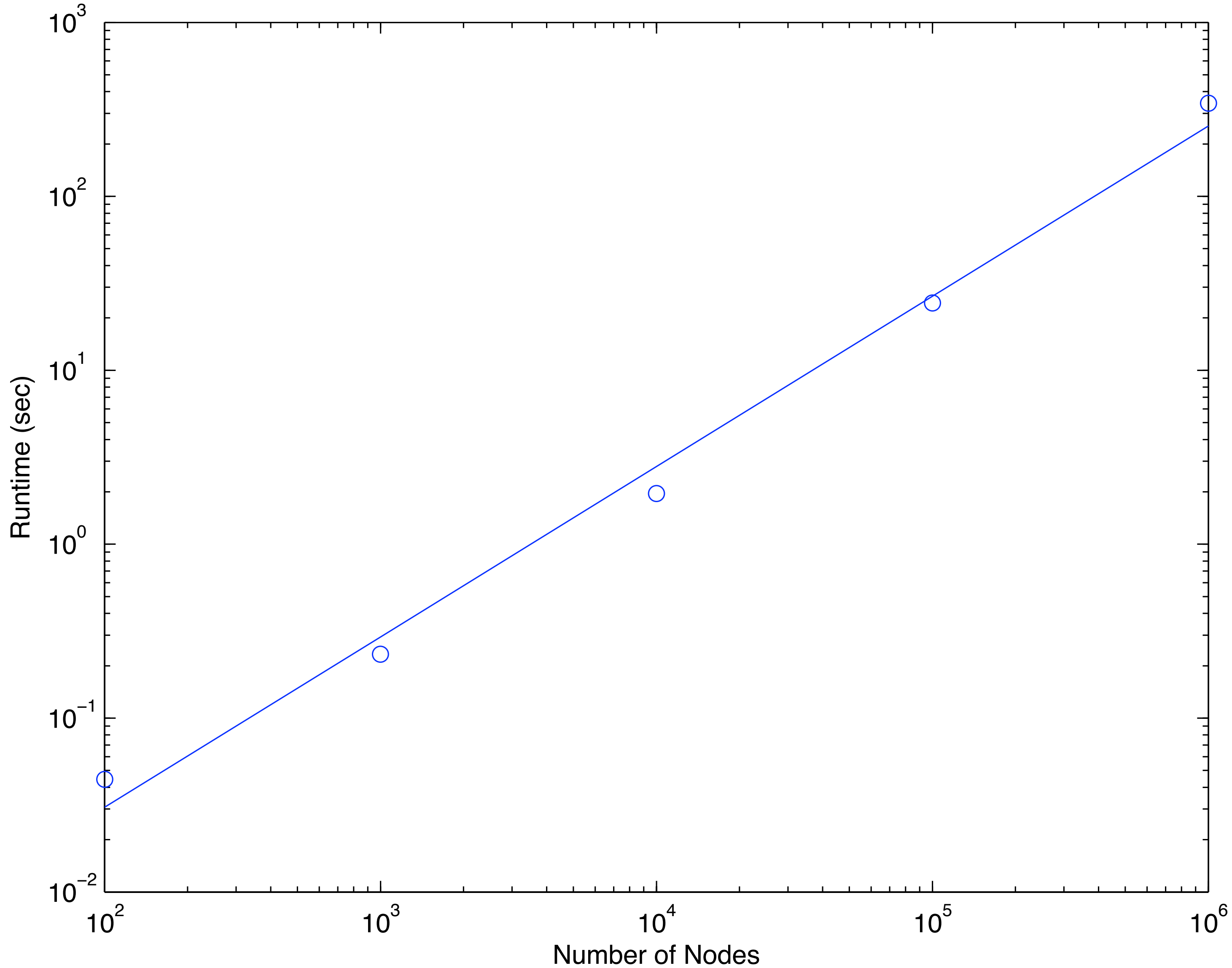
- Automatic complexity control: probability of occupation for extraneous modules goes to zero



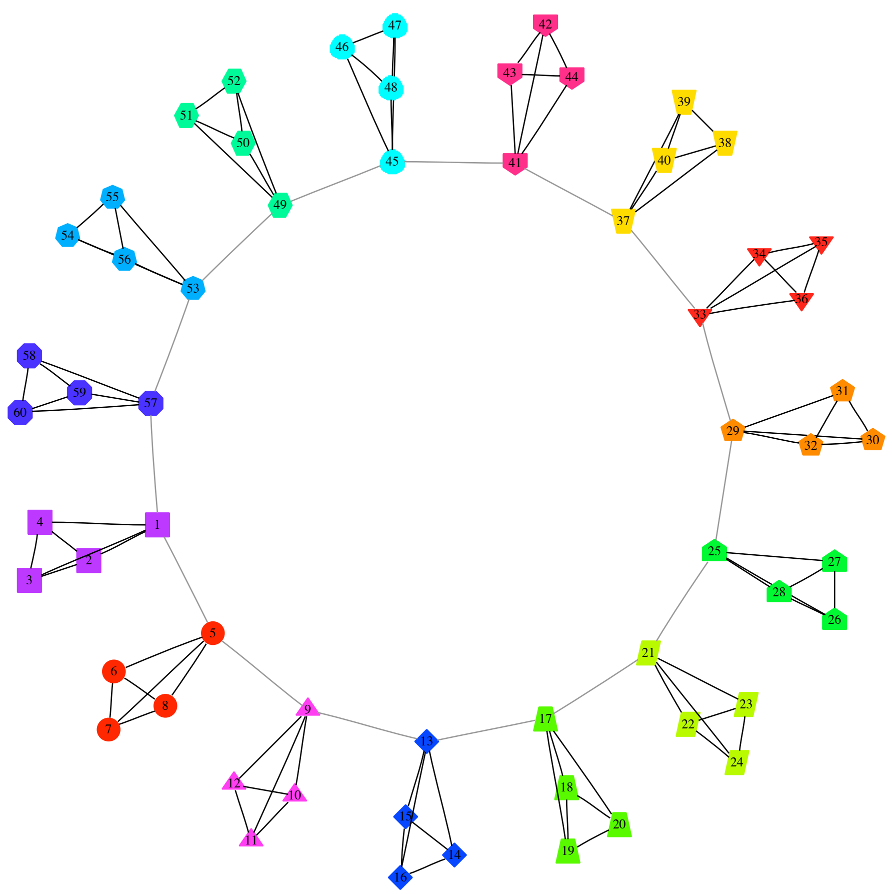


Validation: Runtime

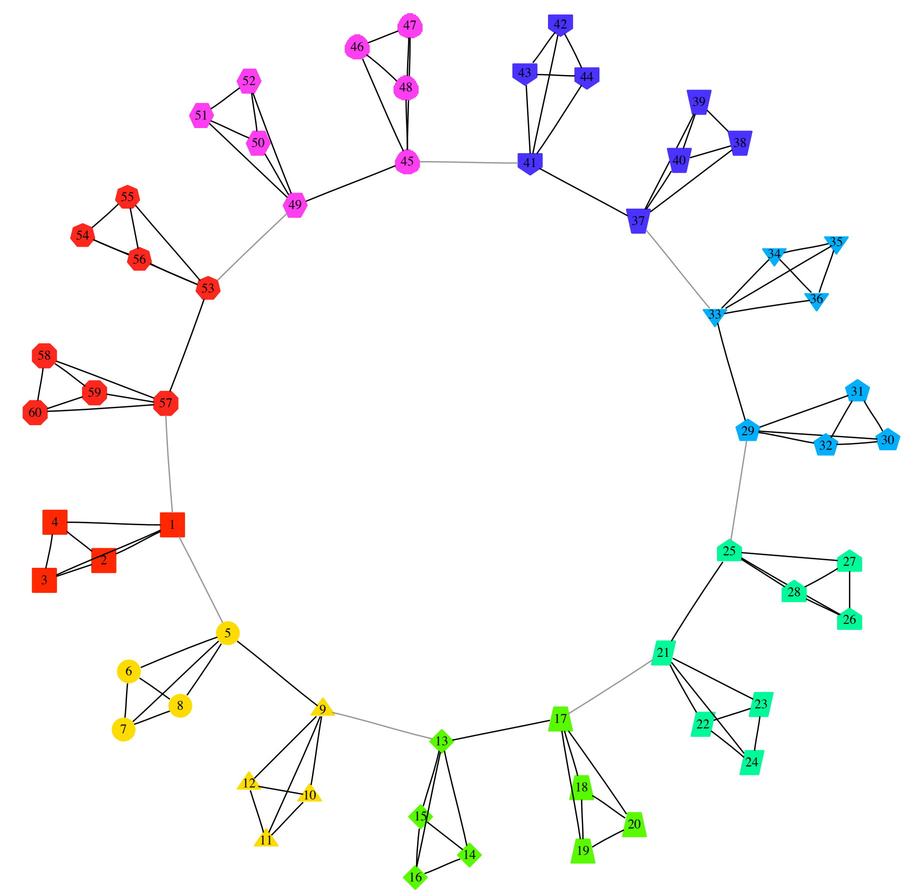
- O(MK) runtime; ~400 sec for $N=10^6$ nodes, K=4 modules, average node degree 16



The “resolution limit” problem



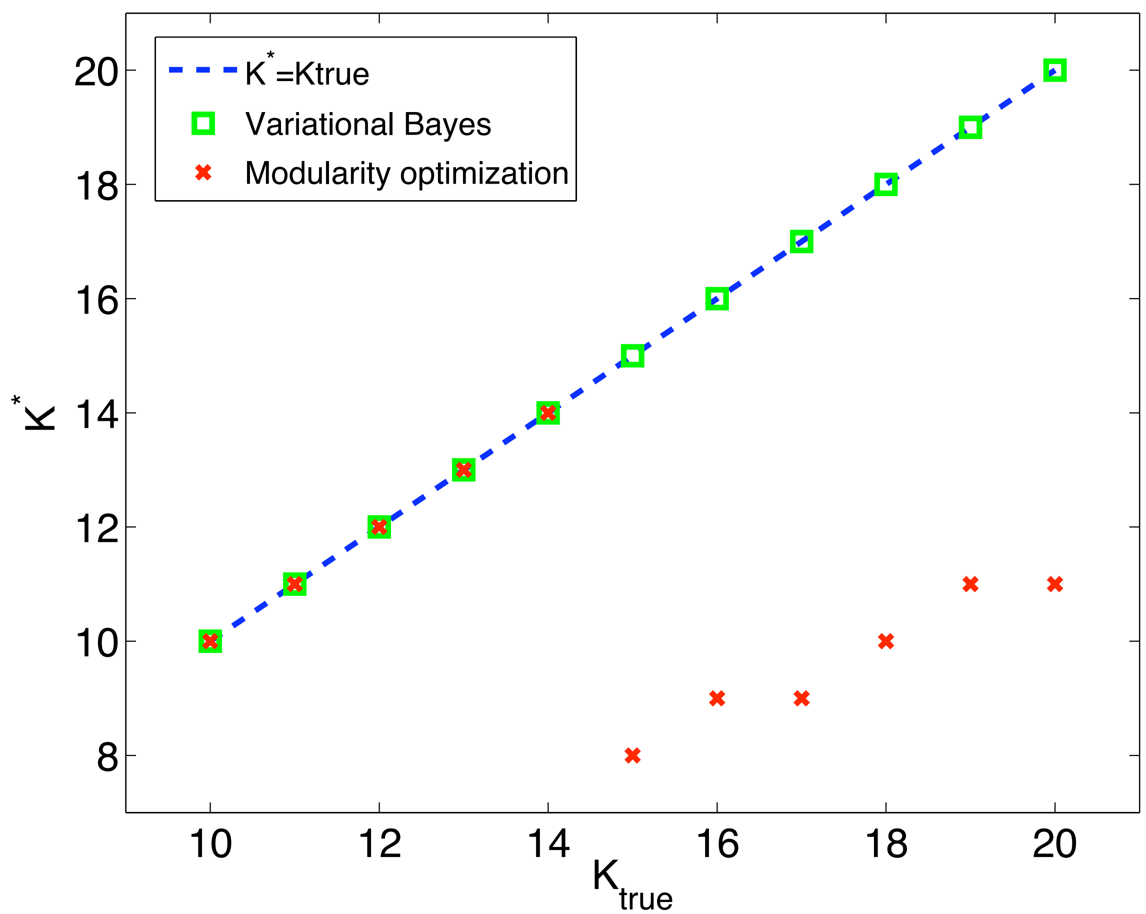
Variational Bayes

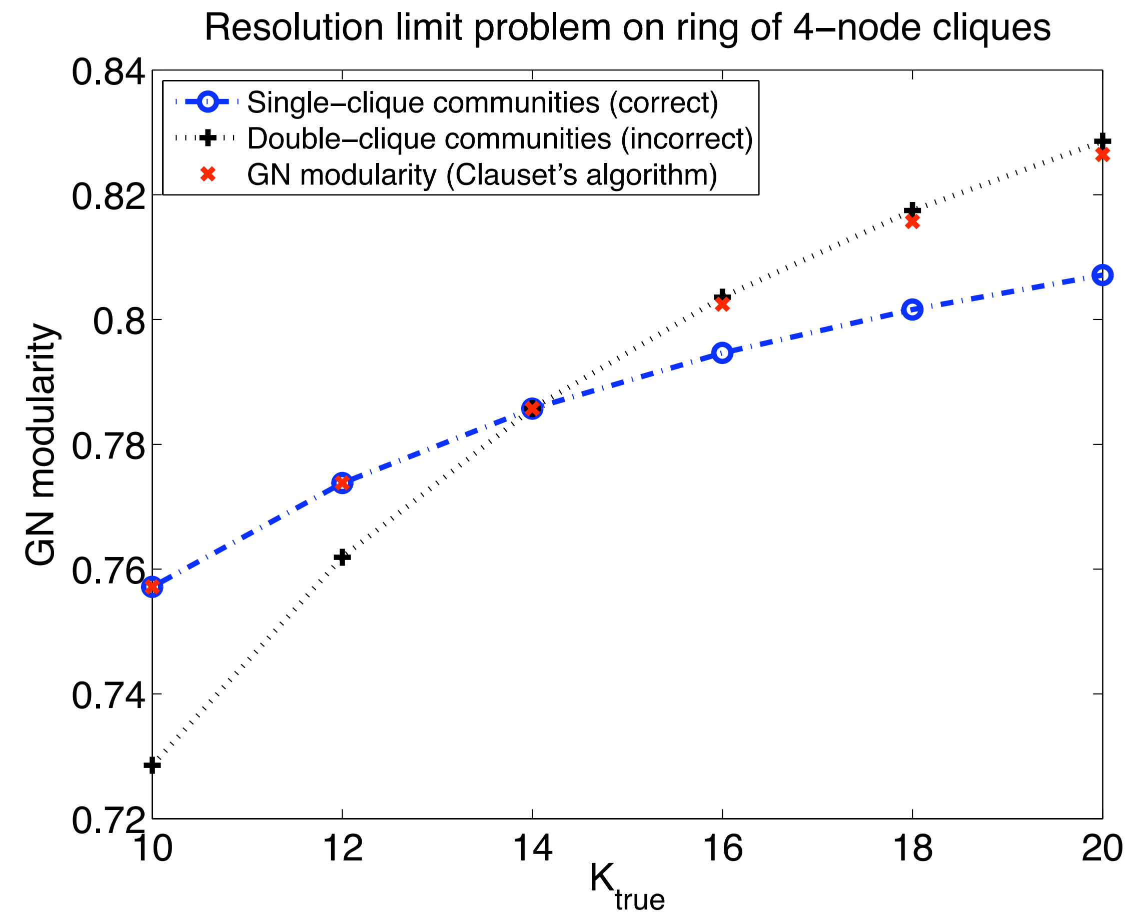


Girvan-Newman modularity

Variational Bayes overcomes the resolution limit by inferring distributions over parameter values as opposed to asserting them

The "resolution limit" problem

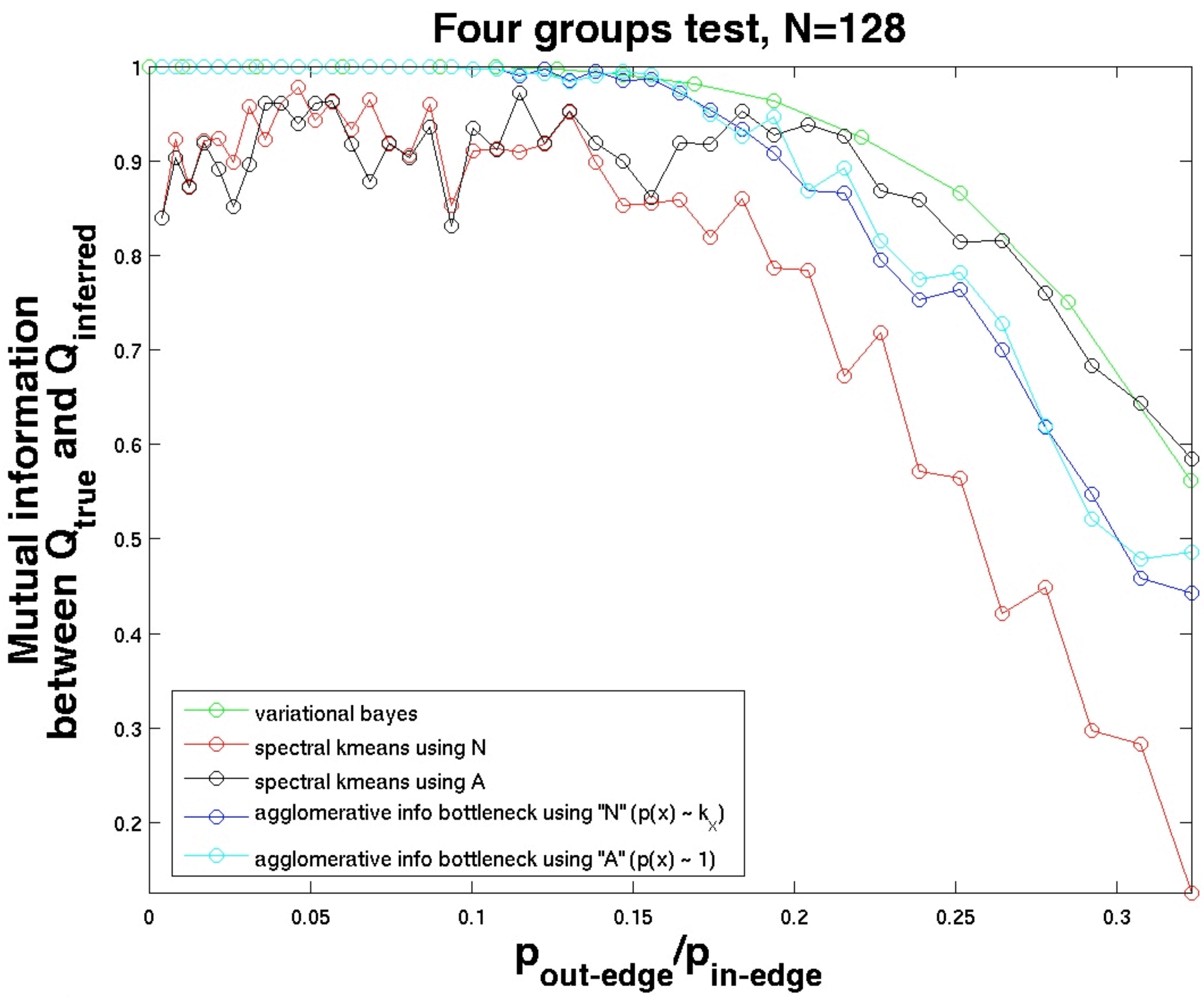




Variational Bayes overcomes the resolution limit by inferring distributions over parameter values as opposed to asserting them

Validation: “four groups” test

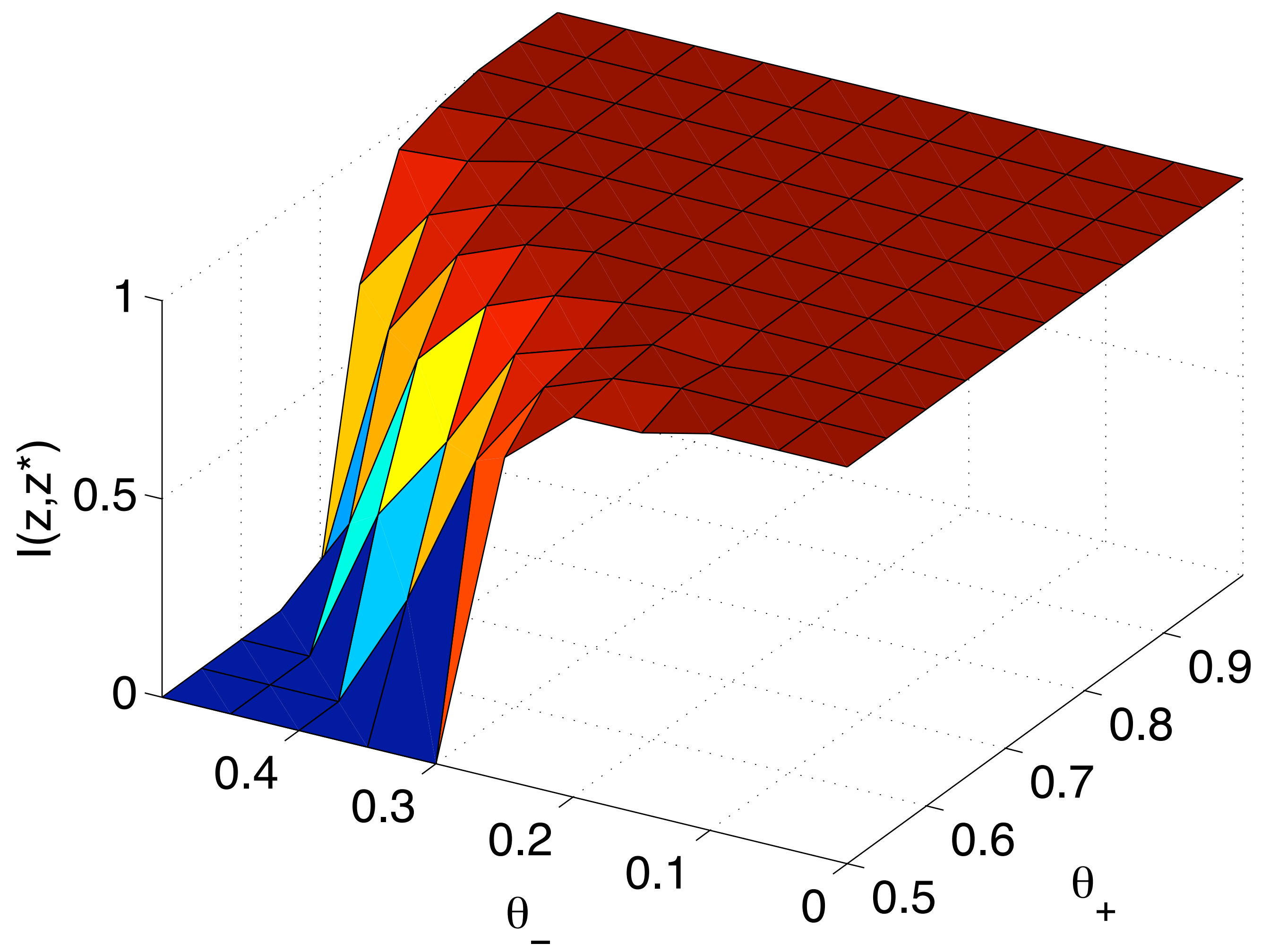
- Mutual information between true and inferred latent variable assignments for N=128 nodes, K=4 modules, average node degree 16



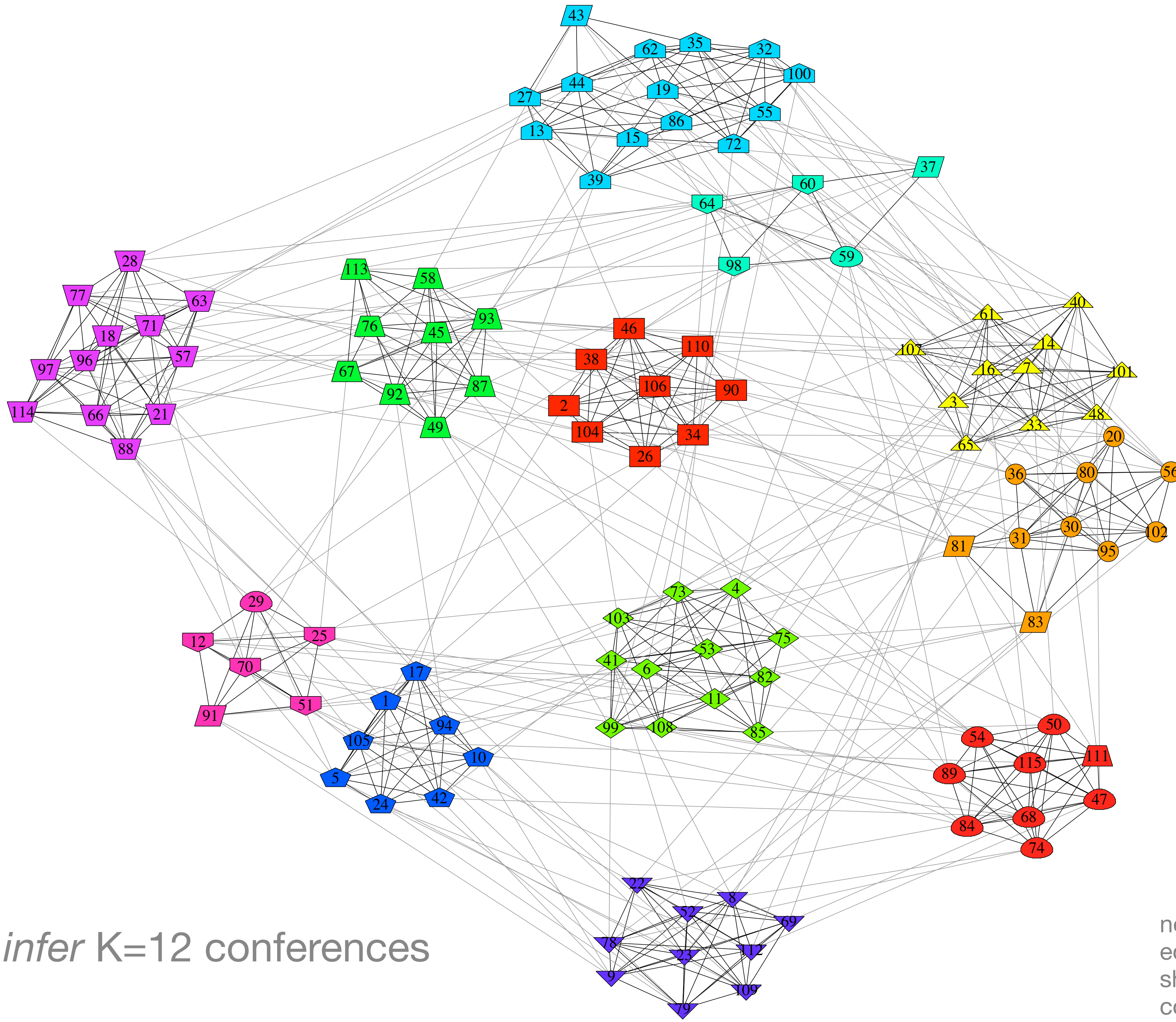
Danon et. al (2005)

Validation: synthetic data

- Mutual information between true and inferred latent variable assignments for N=128 nodes, K=4 modules



Validation: NCAA football schedule



- Correctly *infer* K=12 conferences

nodes: teams
edges: games
shape: conference
color: inferred module

Application: APS March Meeting 2008 co-authorship

APS Meetings

- Login
- Create Account
- Meeting Home
- APS Home
  - Meeting Announcement
- Invited Speakers
- Author Index
- Session Index
- Epitome
- Session Chairs
- Word Search
- Affiliation Search
- Using the Scheduler
- BAPS PDFs

2008 APS March Meeting

Monday–Friday, March 10–14, 2008; New Orleans, Louisiana

Session P39: Applications of Complex Networks

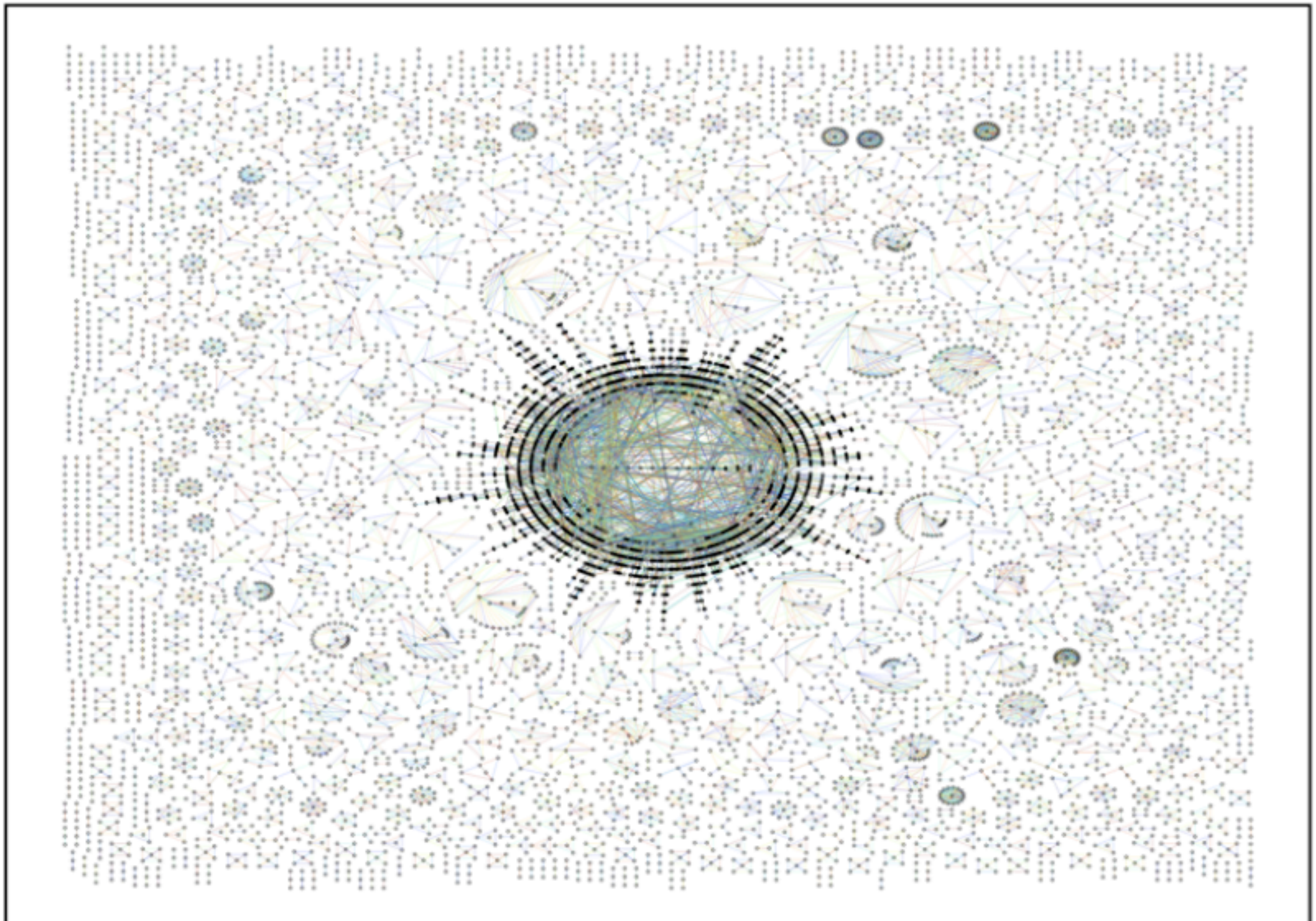
Show Abstracts

Sponsoring Units: GSNP
Chair: Narayan Menon, University of Massachusetts, Amherst
Morial Convention Center - 231

Wednesday, March 12, 2008 8:00AM - 8:12AM	P39.00001: Effects of quenched randomness on predator-prey interactions in a stochastic Lotka-Volterra lattice model Uwe C. Tauber , Ulrich Dobramysl Preview Abstract
Wednesday, March 12, 2008 8:12AM - 8:24AM	P39.00002: Dynamical Clustering in Reaction-Dispersal Processes on Complex Networks Vincent David , Marc Timme , Theo Geisel , Dirk Brockmann Preview Abstract
Wednesday, March 12, 2008 8:24AM - 8:36AM	P39.00003: Fluctuations and Food-web Structures in Individual-based Models of Biological Coevolution Per Arne Rikvold , Volkan Sevim Preview Abstract
Wednesday, March 12, 2008 8:36AM - 8:48AM	P39.00004: Metabolic disease network and its implication for disease comorbidity Deok-Sun Lee , Zoltan Oltvai , Nicholas Christakis , Albert-Laszlo Barabasi Preview Abstract
Wednesday, March 12, 2008 8:48AM - 9:00AM	P39.00005: The Human Phenotypic Disease Network Cesar Hidalgo , Nicholas Blumm , Albert-Laszlo Barabasi , Nicholas Christakis Preview Abstract

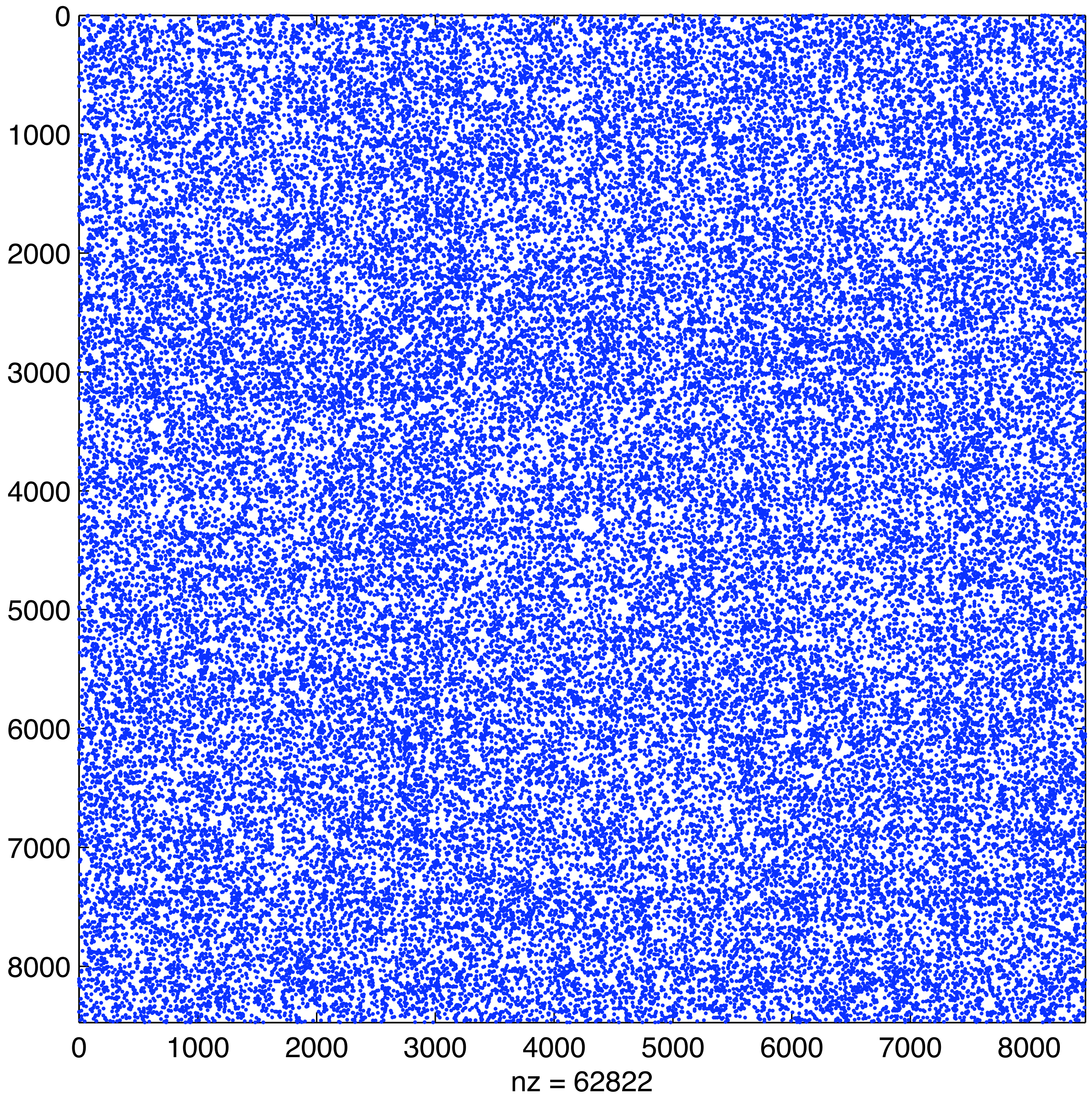
http://meetings.aps.org/Meeting/MAR08

Application: APS March Meeting 2008 co-authorship

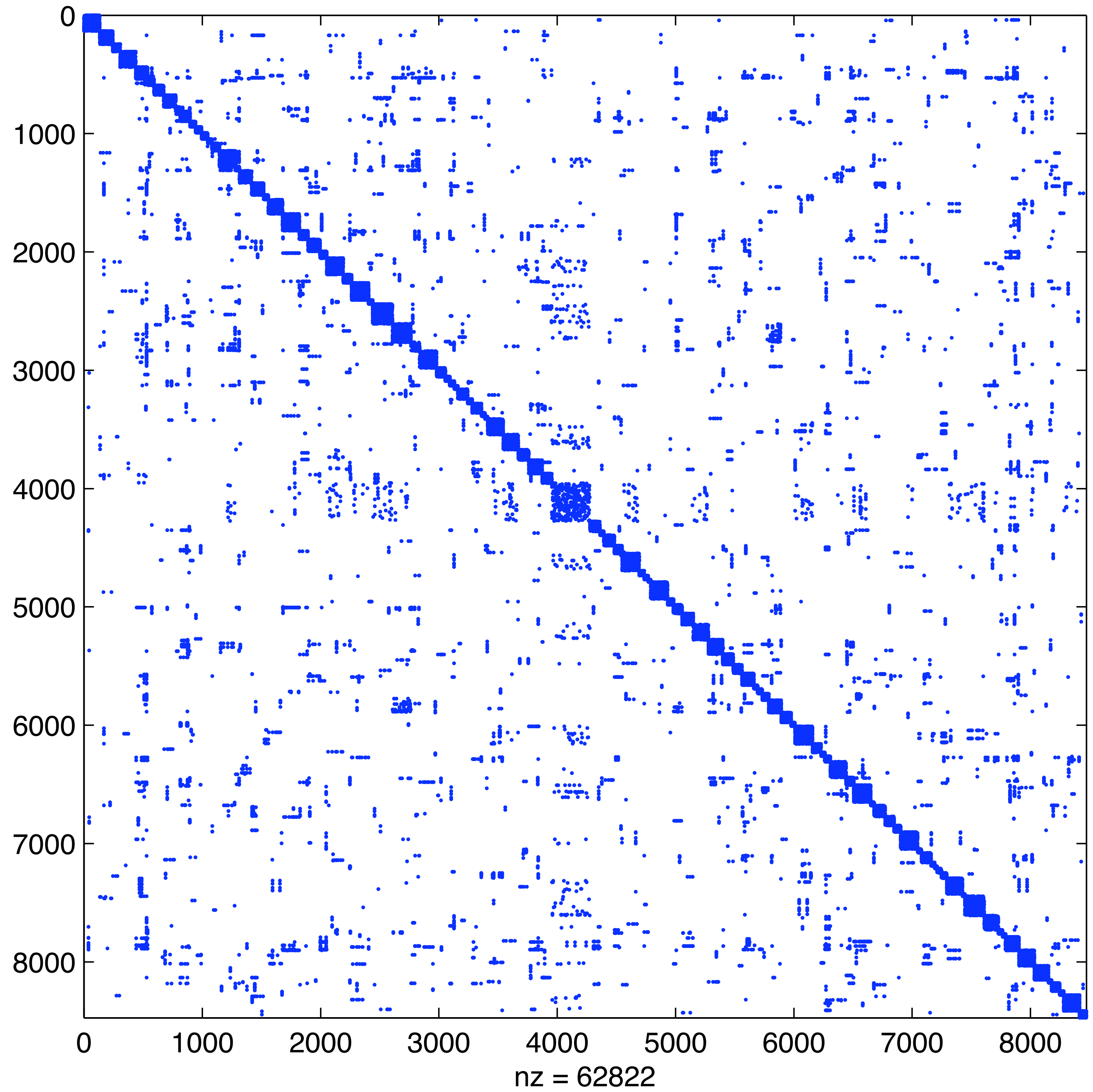


nodes: authors
edges: co-authored papers

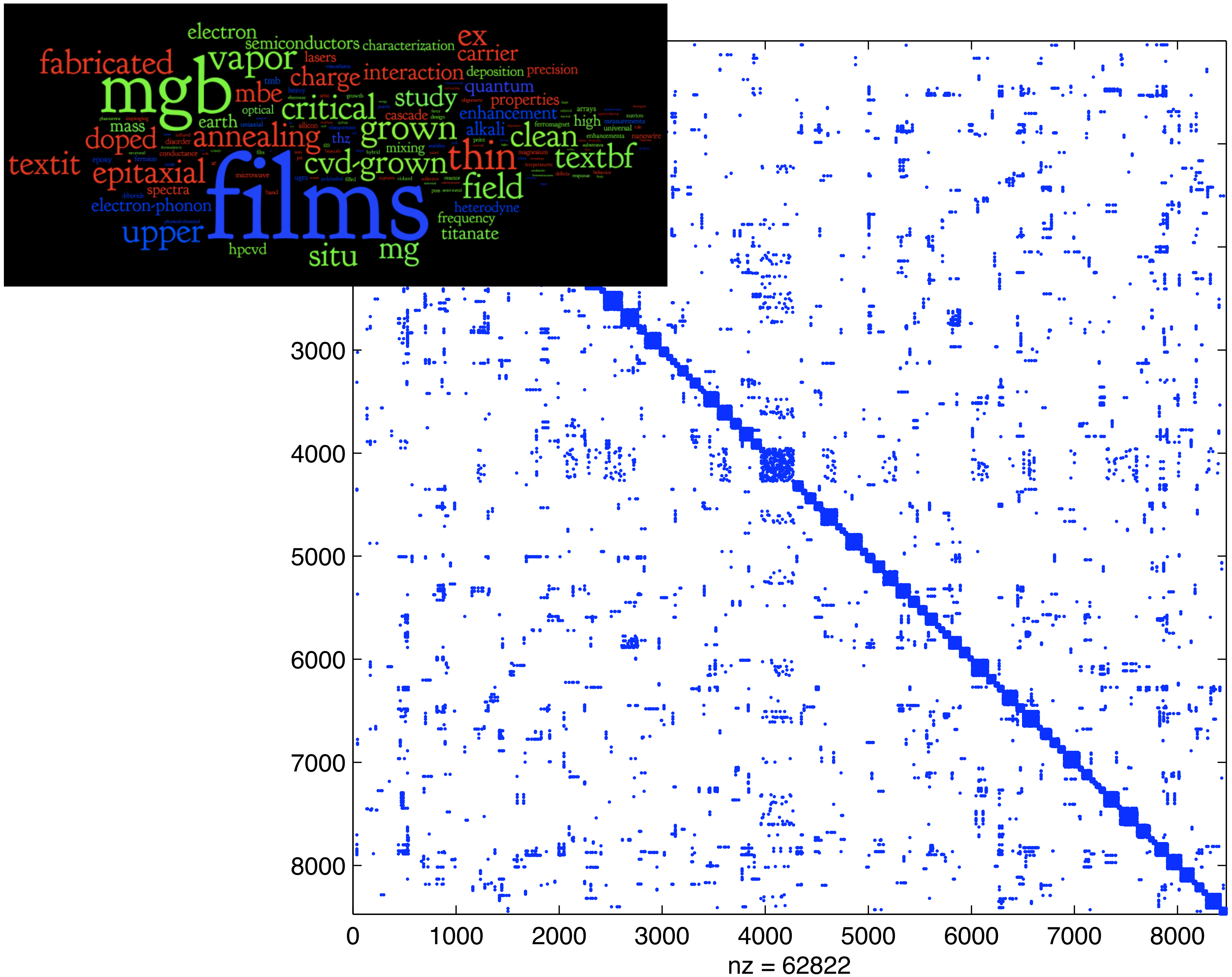
APS March Meeting 2008 co-authorship network



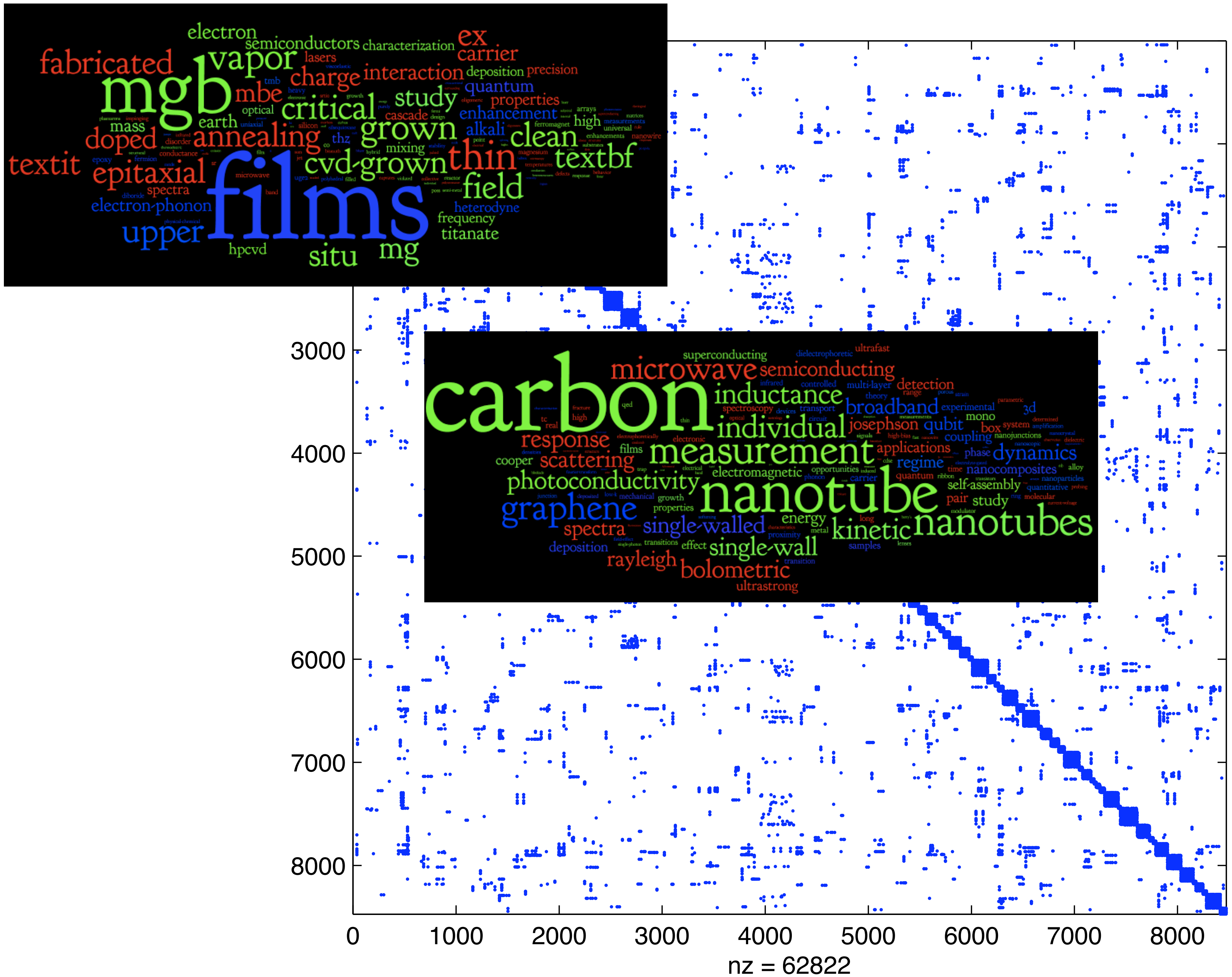
APS March Meeting 2008 co-authorship network



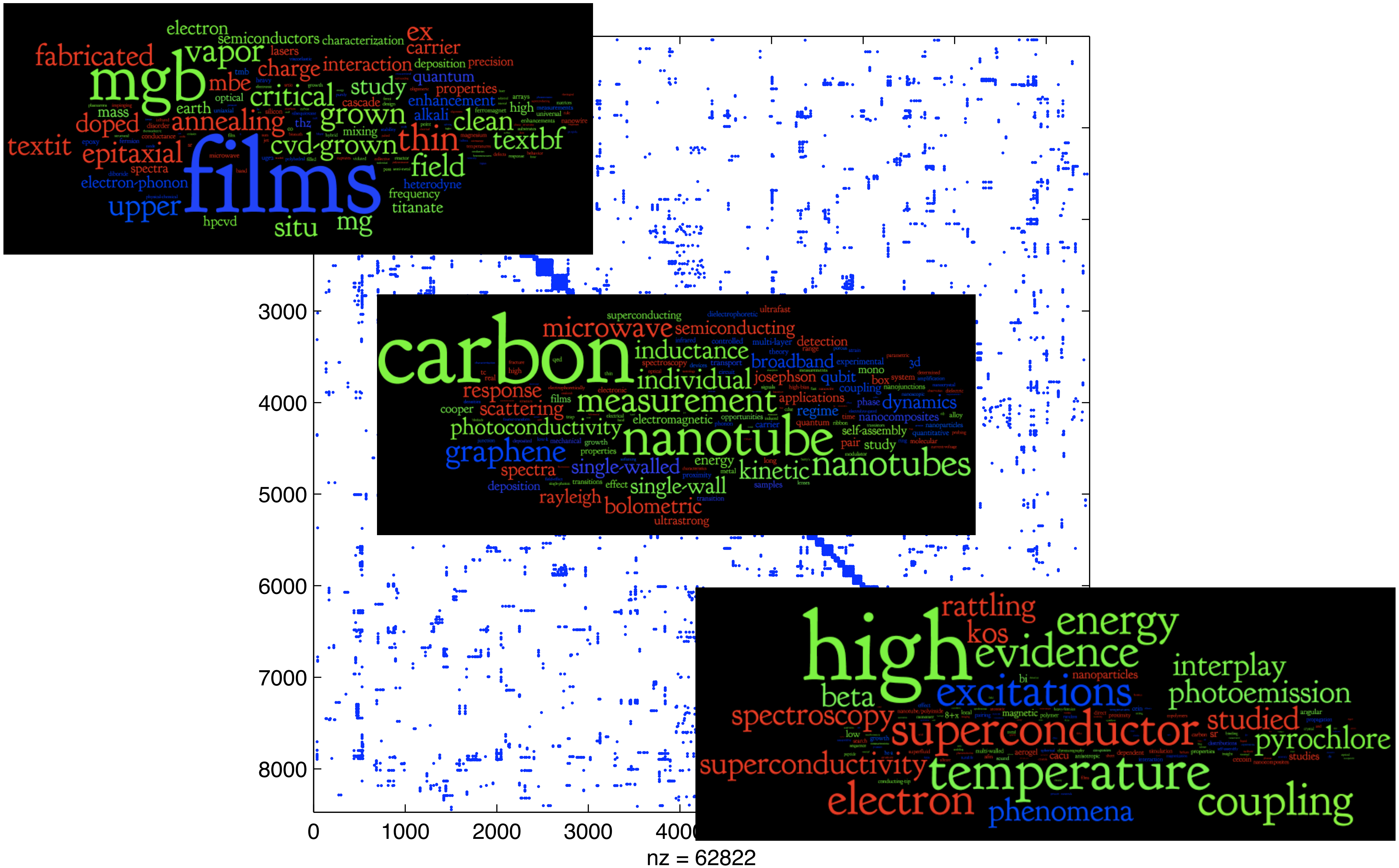
APS March Meeting 2008 co-authorship network



APS March Meeting 2008 co-authorship network



APS March Meeting 2008 co-authorship network



Conclusions

- Phrased network modularity as a modeling problem
- Resulted in a interpretable, accurate, and scalable algorithm which addresses the resolution limit problem
- Validated technique on synthetic and real networks
- Future: extend model to handle alternative network structure, using same *framework*
- Paper: Physical Review Letters, Vol.100, No.25 (258701)
- Software: http://vbmod.sourceforge.net

Acknowledgements

- **Wiggins Lab**
  - Chris Wiggins
  - Anil Raj
  - Andrew Mugler

- **Useful discussions**
  - Jonathan Goodman (NYU)
  - Joel Bader (Hopkins)
  - Matt Hastings (LANL)
  - Aaron Clauset (SFI)
  - David Blei (Princeton)
  - Edo Airoldi (Princeton)