Testing for commonality among graphs and subgraphs: The cortical column conjecture

Daniel Sussman

Harvard University, Statistics Department

August 4, 2014

Collaborators at JHU: Avanti Athreya, Donniell Fishkind, Vince Lyzinski, Carey Priebe, Minh Tang, Joshua Vogelstein, . . .

Cortical Column Conjecture

- Many contemporary theories of neural information processing suggest that the neocortex employs hierarchical algorithms composed of repeated instances of **a limited set of computing primitives**.

EPFL/Blue Brain Project

- The cortical column conjecture suggests that neurons are connected in a graph that exhibits motif representing **repeated processing modules**.

Cortical Column Conjecture

Computer Analogy ...

- Repeated logic gates form,
- repeated logic curcuits,
- which form larger units (microprocessor, memory, etc).

Repetitions are nearly **exact**.

Cortical Column Conjecture

Computer Analogy ...

- Repeated logic gates form,
- repeated logic curcuits,
- which form larger units (microprocessor, memory, etc).

Repetitions are nearly **exact**.

... breaks down

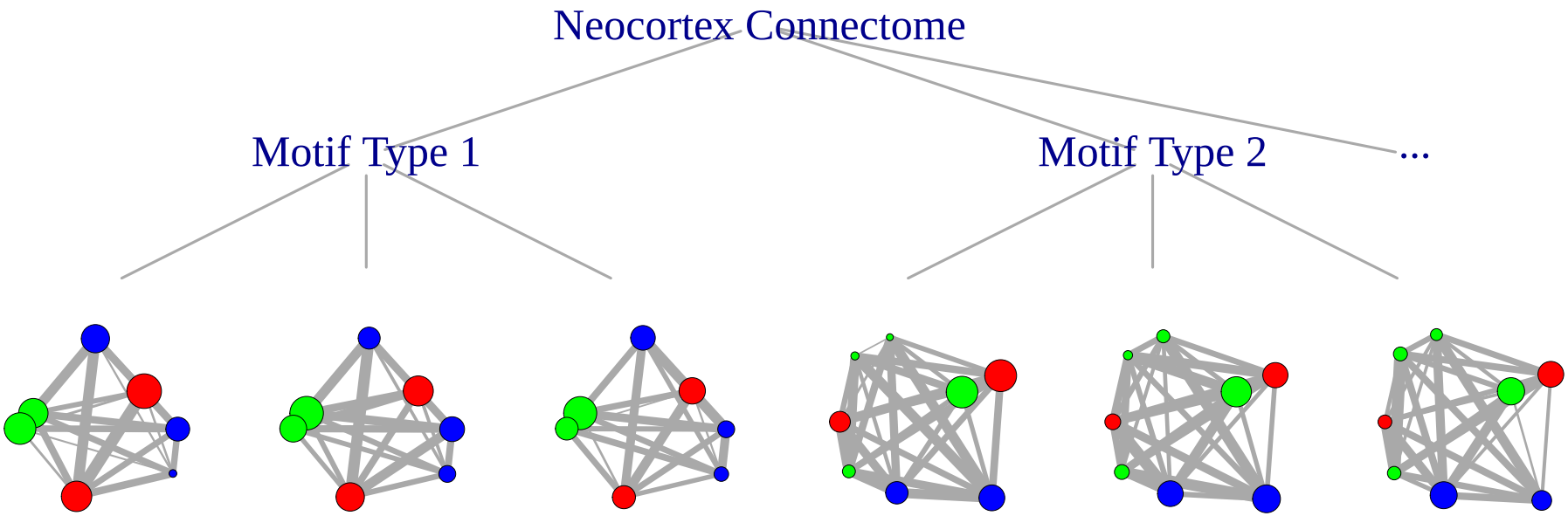
- For a brain, complex and noisy biological processes occur during developement.
- We cannot assume the motifs will be exact repetitions but they will be **noisy repetitions**.

We will model the repeated motifs as noisily repeated random graphs.

Motif Hierarchy

Hiearchical model for connectome graph

- Level 1 Motif Types
- Level 2 Repetitions/Variations of Motifs
- Level 3 "Block Structure" within each motif
- Level 4 Neuron Level Variation

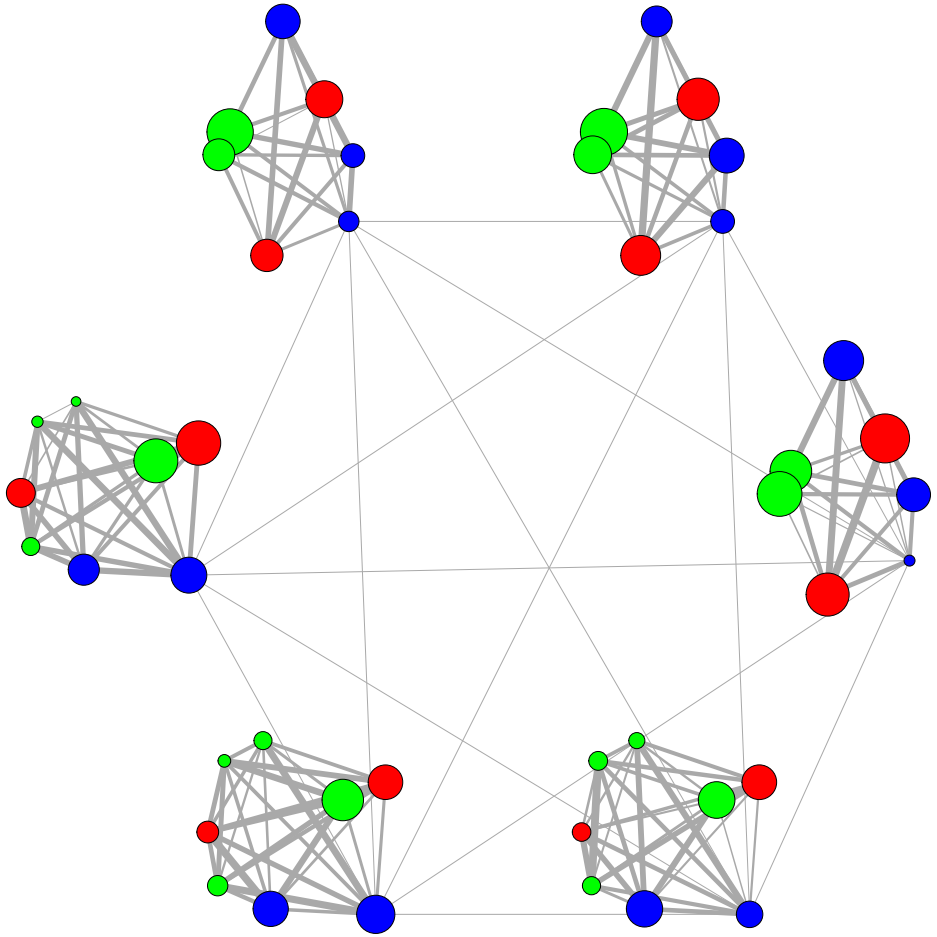


Connectome Graph

Disjoint union of the repetitions/variations of motif graphs

$+$

Sparse connectivity between theses



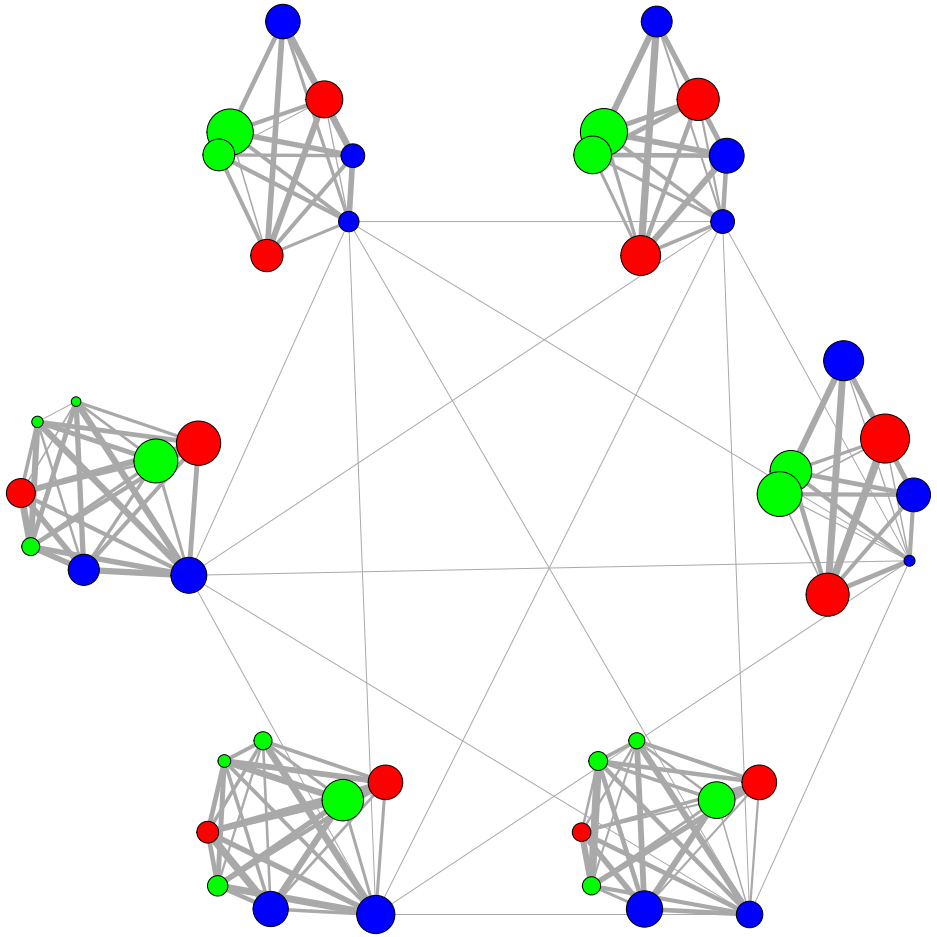
Goals

Given an observed graph we seek to

Step 1. Cluster vertices to into *candidate* repeated motifs

Step 2. Cluster candidate induced subgraphs into motif types

Step 3. Test for how closely motifs are repeated and estimate parameters



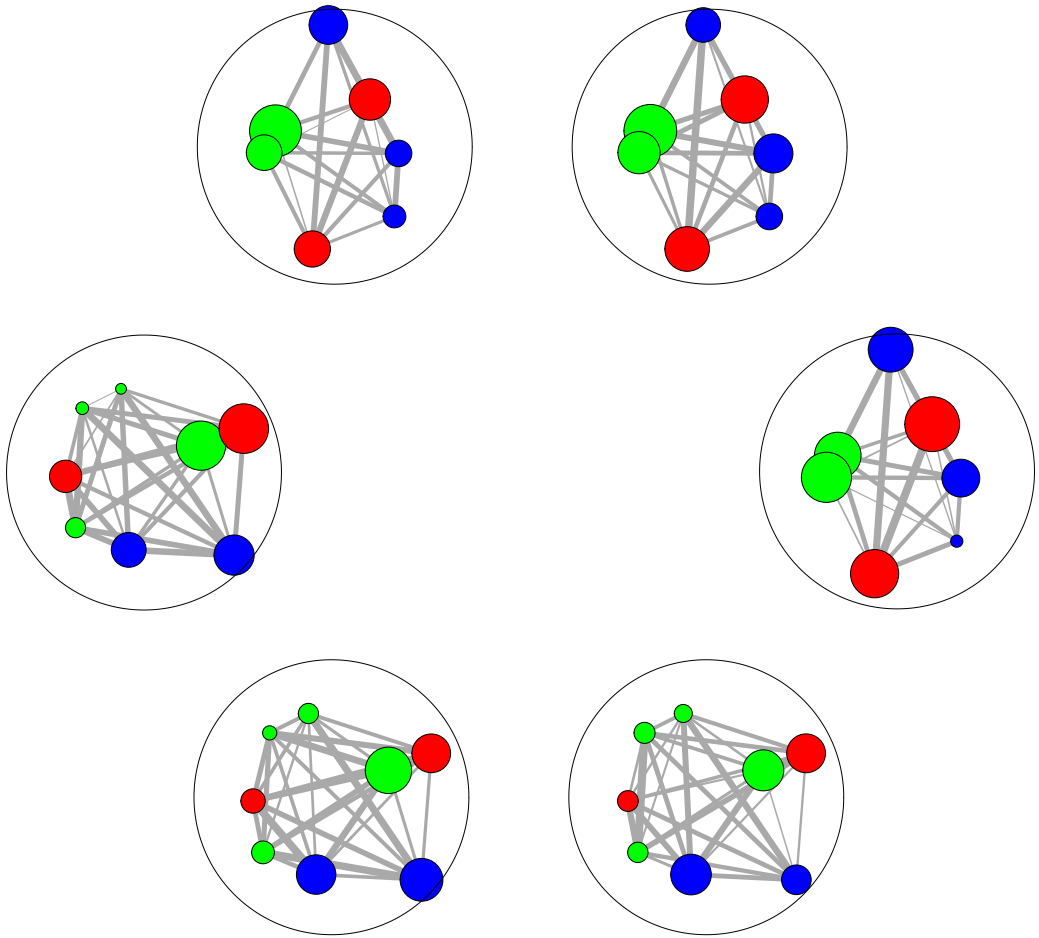
Goals

Given an observed graph we seek to

Step 1. Cluster vertices to into *candidate* repeated motifs

Step 2. Cluster candidate induced subgraphs into motif types

Step 3. Test for how closely motifs are repeated and estimate parameters



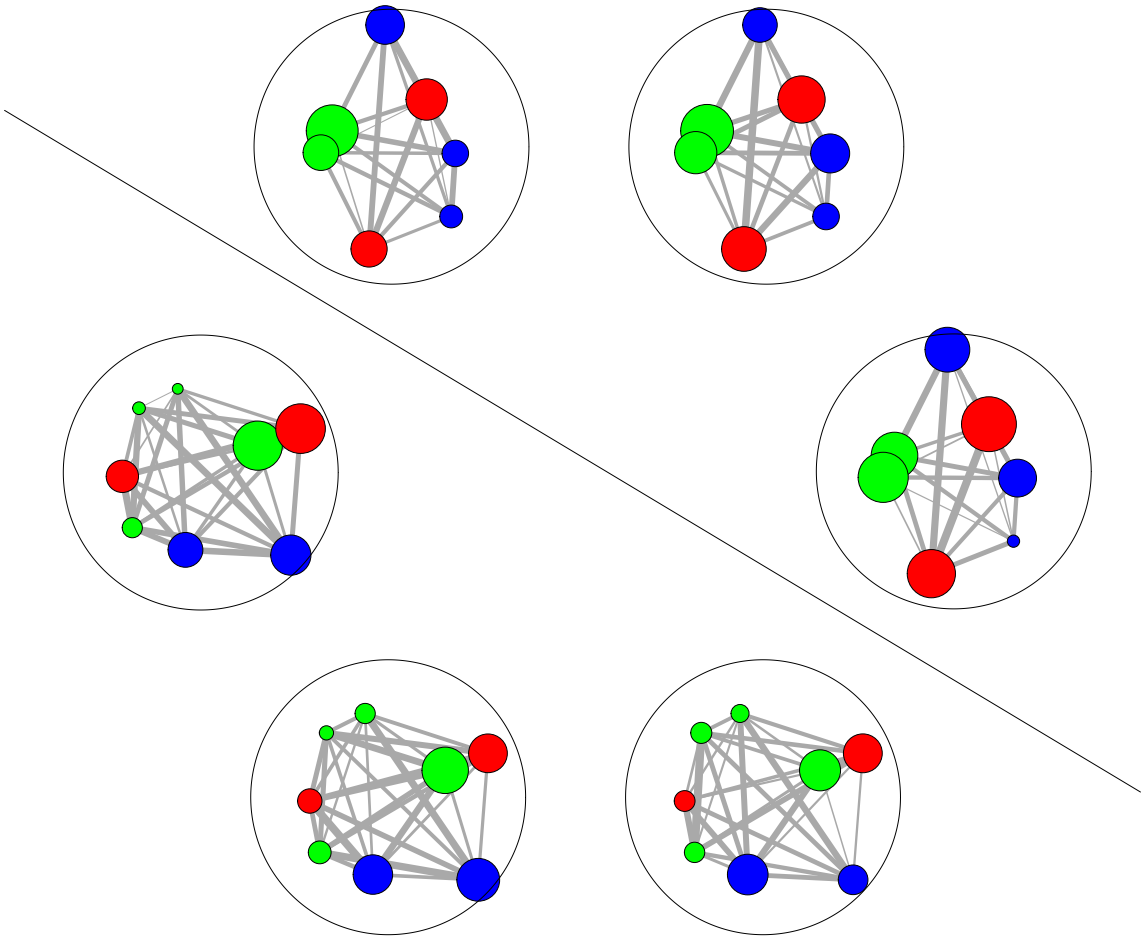
Goals

Given an observed graph we seek to

Step 1. Cluster vertices to into *candidate* repeated motifs

Step 2. Cluster candidate induced subgraphs into motif types

Step 3. Test for how closely motifs are repeated and estimate parameters



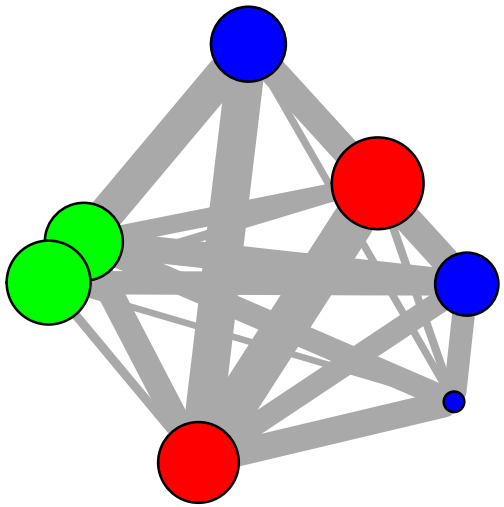
Goals

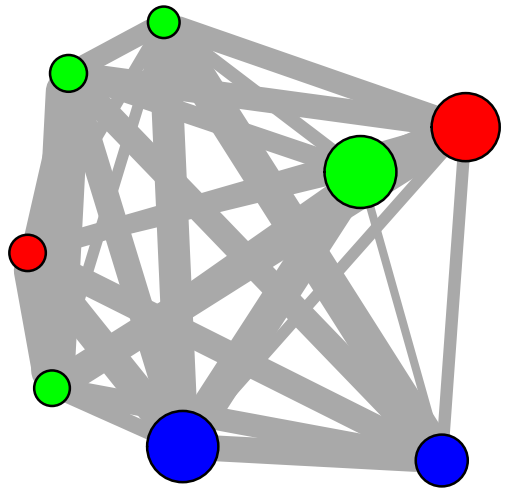
Given an observed graph we seek to

Step 1. Cluster vertices to into *candidate* repeated motifs

Step 2. Cluster candidate induced subgraphs into motif types

Step 3. Test for how closely motifs are repeated and estimate parameters





Data?

Does it exists? On a small scale ... almost.
High resolution data consists of only a few hundred partial neurons

What will the data look like? ... of course we don't know but ...

We think ...

- Motif may consist of O(100) neurons
- Brains contain billions of neurons and an order of magnitude more synapses.

For this talk, we will create a pseudo-real data example that captures our modeling framework and uses neuroscientific data.

Pseudo-real Data Example

KKI Data (openconnecto.me)

- 21 subjects were each scanned twice using DTMRI
- 42 graphs on 70 vertices
  - Subject and vertex correspondence known
- We focus on 6 graphs corresponding to 3 subjects

Pseudo-data Hierarchical Model

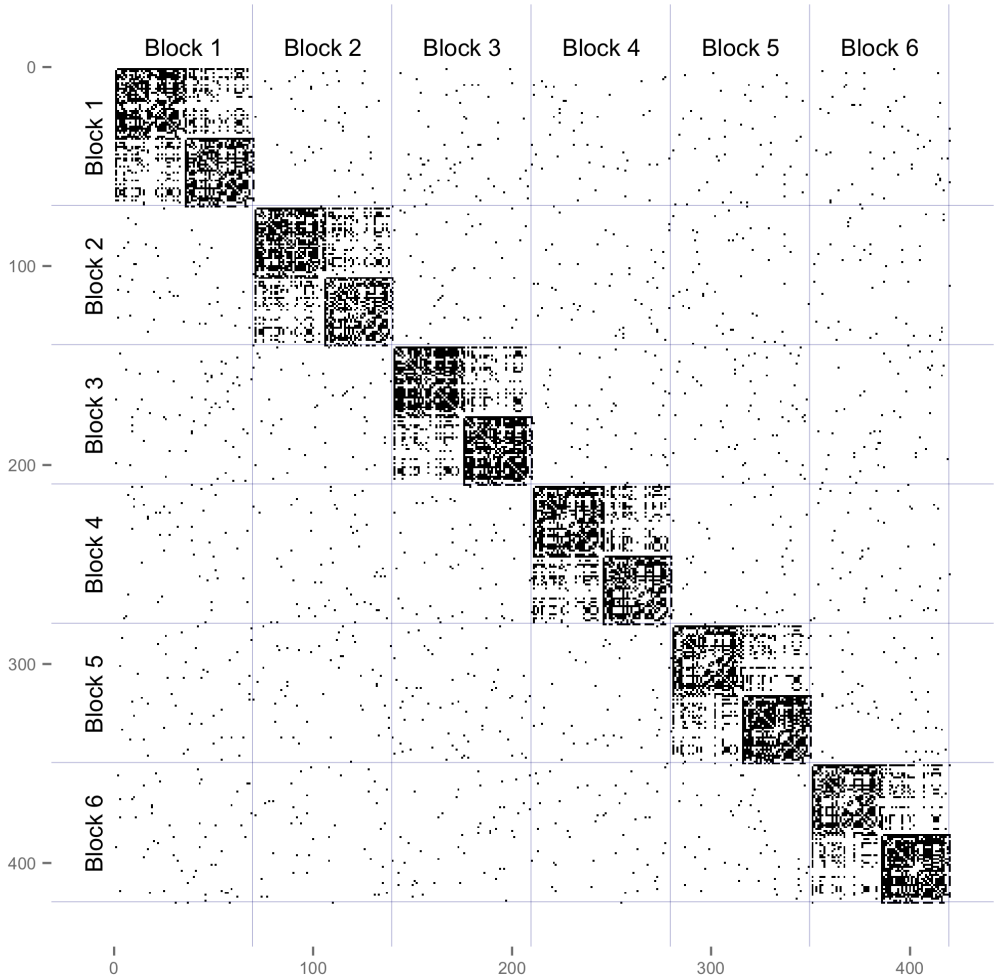
Disjoint union of 6 graphs

+

Erdos-Renyi(p) between graphs

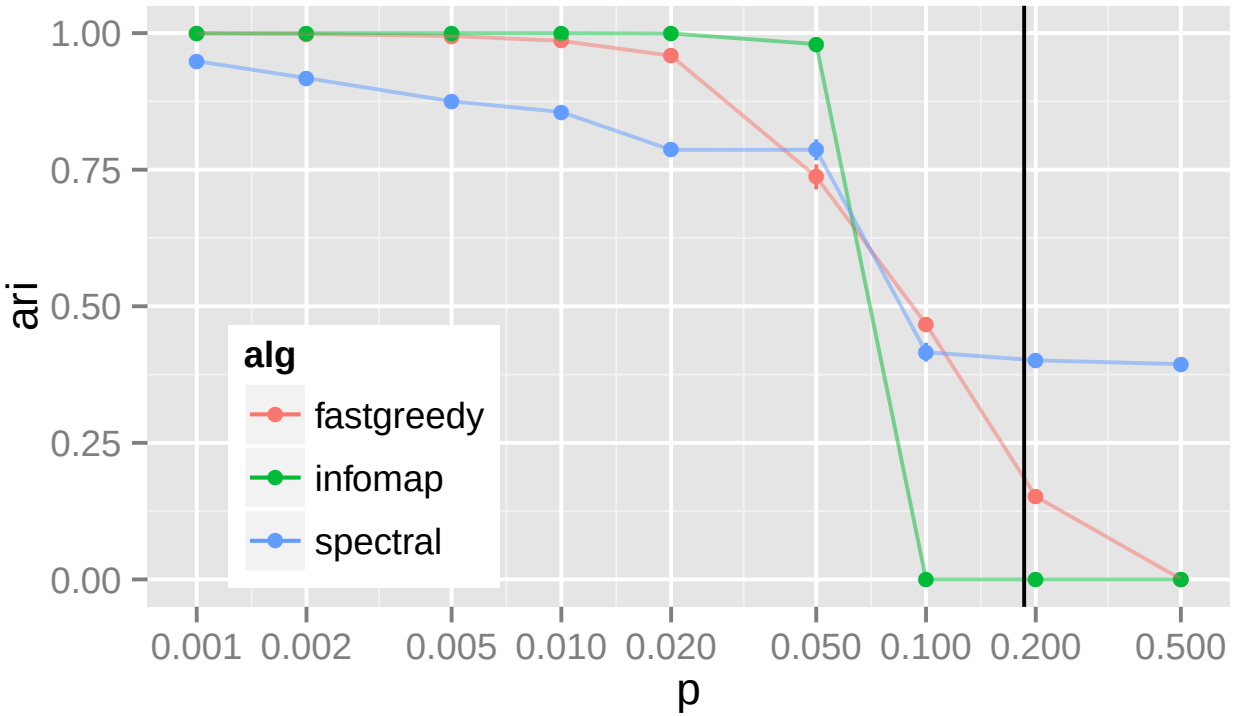
=

420 Vertex Hierarchical Graph
with 2 repeats of 3 motifs



Can we recover original graphs by clustering vertices?

Step 1: Cluster Vertices



fastgreedy	**infomap**	**spectral**
Clauset, et al. (2004)	Rosvall, et al. (2009)	Fishkind et. al. (2013)
Modularity Based	Random Walk Based	Spectral Based

Step 2: Cluster Subgraphs

Suppose we cluster perfectly in Step 1
and we know correspondence between vertices. Then we can

- Compute pairwise distances (matrix norms/graph metrics)
- Vectorize adjacency matrices
- Estimate graph parameters

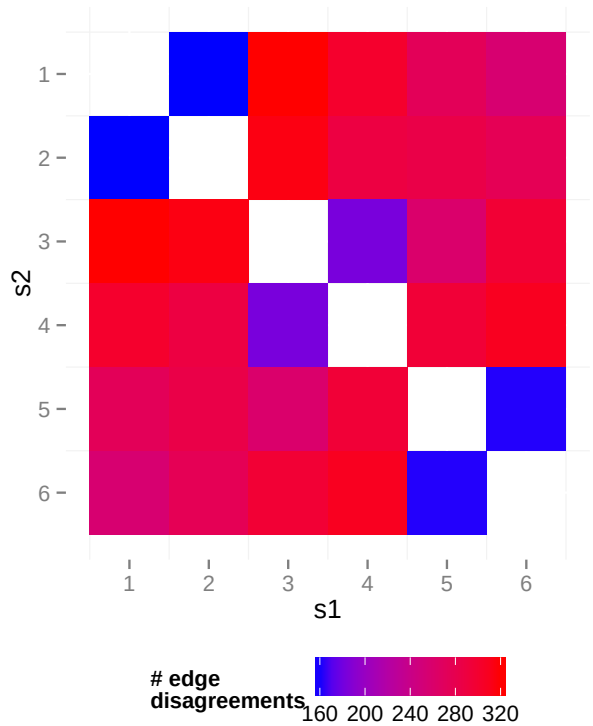


Fig: Pairwise Distance Matrix

and cluster via . . . .

Step 2: Cluster Subgraphs

Suppose we cluster perfectly in Step 1
and we know correspondence between vertices. Then we can

- Compute pairwise distances (matrix norms/graph metrics)
- Vectorize adjacency matrices
- Estimate graph parameters

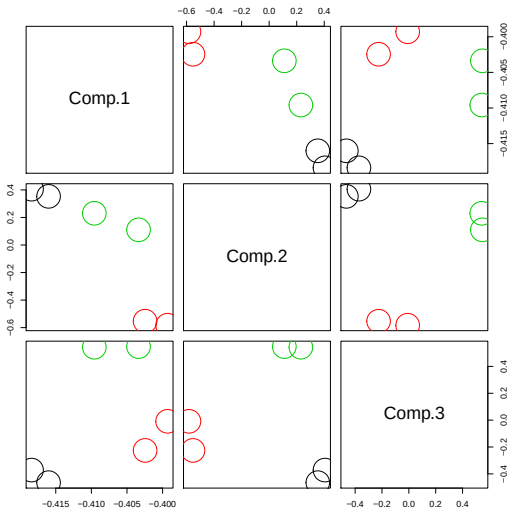


Fig: PCA of vectorized adjacency

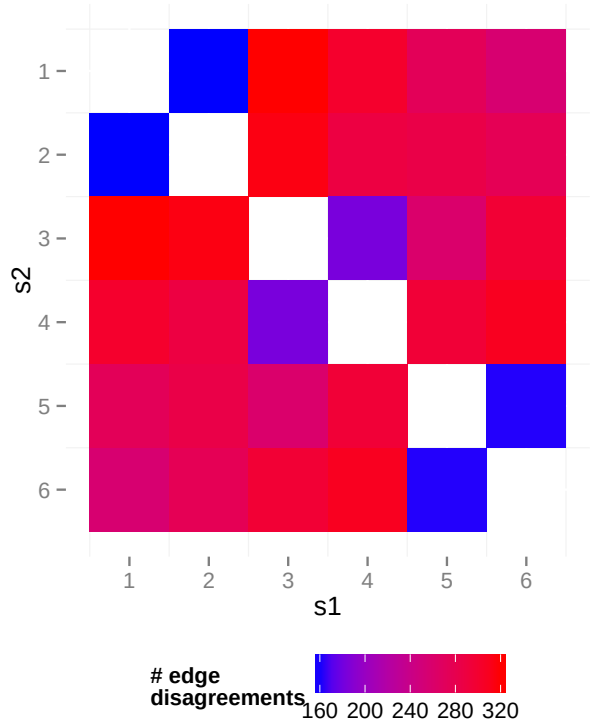


Fig: Pairwise Distance Matrix

and cluster via . . . .

Step 2: Cluster Subgraphs

Suppose we cluster perfectly in Step 1
and we know correspondence between vertices. Then we can

- Compute pairwise distances (matrix norms/graph metrics)
- Vectorize adjacency matrices
- Estimate graph parameters

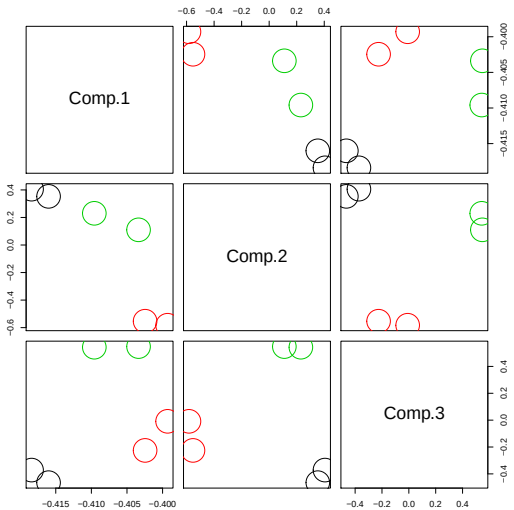


Fig: PCA of vectorized adjacency

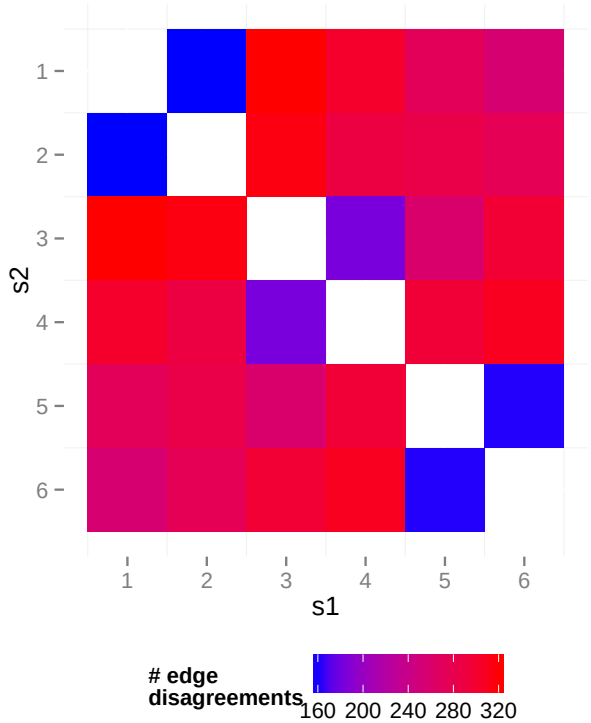


Fig: Pairwise Distance Matrix

and cluster via . . . .

Step 3: Testing and Estimation

Once we have clustered the subgraphs, we want to test for how closely the motifs are repeated and estimated the repeated motif distributions.

Testing

Within each cluster we can consider test statistic

$$T = \sum_{r \in \text{cluster}} \|A_r - \bar{A}_{cluster}\|$$

so that small values of this test statistic indicate a tight cluster of repeated motifs while large values indicate that the motifs are not very repeated.

Estimation

We can use established methods for estimating random graph distributions based on $\bar{A}_{cluster}$, the mean adjacency matrix for the cluster.
Estimate variation within each motif.

Hierarchical SBM

$$\begin{pmatrix} \left[\begin{array}{c} \\ \mathrm{SBM}(\theta_1) \\ \\ \end{array}\right] & \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] & \cdots & \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] \\ \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] & \left[\begin{array}{c} \\ \mathrm{SBM}(\theta_2) \\ \\ \end{array}\right] & \cdots & \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] \\ \vdots & \vdots & \ddots & \vdots \\ \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] & \left[\begin{array}{c} \\ \mathrm{ER}(p) \\ \\ \end{array}\right] & \cdots & \left[\begin{array}{c} \\ \mathrm{SBM}(\theta_Q) \\ \\ \end{array}\right] \end{pmatrix}$$

wbere $\theta_i = (K_i, B_i, \vec{n}_i)$ and

$$\theta_1, \ldots, \theta_Q \overset{iid}{\sim} \sum_{r=1}^{R} \rho_r G_r$$

where the G_r are distributions on SBM parameters and $\sum_r \rho_r = 1$.
Various theoretical results depending on the asymptotic regime.

Challenges

Have made quite a few assumptions that make our lives easy.

Step 1 (Large literature)

- Potentially a large number of repeated motifs
- Deeper hierarchy with varying levels of interconnectivity?
- How to leverage edge and vertex covariates (spatial location, neuron type, . . .)

Step 2

- Contaminated by errors from Step 1
- Don't know vertex correspondence or non-existent vertex correspondence (Graph Matching or Parameter Matching)
- Not necessarily the same # vertices in each motif
- How to better leverage network structures

Step 3

Same as in Step 2 +

- Contaminated by errors from Step 2
- Test statistic distributions unknown in general
- Estimate parameters for mixtures of SBMs or more complicated

Challenges

All three steps must be able to contend with

1. graphs at the massive scale (human brain has 100 billion neurons/vertices and $\approx 10^{14}$ synapses/edges)

and

2. in the presence of errors such as
   - missing edges
   - extra edges
   - merged vertices
   - split vertices
   - sampling bias

NB: Recent efforts took $\approx$5 years to "perfectly" reconstruct a graph with $O(100)$ neurons and $O(1000)$ synapses.

Big Five of Graph Analysis

Clustering the vertices in a graph Identify "communities" within the graph via a partition or a clustering of the vertex.

Clustering a collection of graphs Cluster graphs $\{G_r = (V_r, E_r)\}_{r \in [R]}$ them into groups that share similar structures.

Graph matching Given graphs $\{G_r = (V_r, E_r)\}_{r \in [R]}$ with presumed shared structure, identify a correspondence between the vertex sets across the multiple graphs, i.e. mappings from $V_r \mapsto V_{r'}$ for $r, r' \in [R]$, that match graph structure.

Testing and estimation Given graphs $\{G_r = (V_r, E_r)\}_{r \in [R]}$, test or estimate structural parameters to obtain meaningful actionable information.

Robustness to errorfully observed graphs Graphs are observed with error and sampling bias so we need methods that are robust to deviations from idealized random graph models. Sampling and data collection designs that are optimized for inference given computation constraints.