

# Configuring random graph models with fixed degree sequences

**Daniel B. Larremore**  
**Santa Fe Institute**

June 21, 2017  
NetSci



**SANTA FE INSTITUTE**

larremore@santafe.edu  
@danlarremore



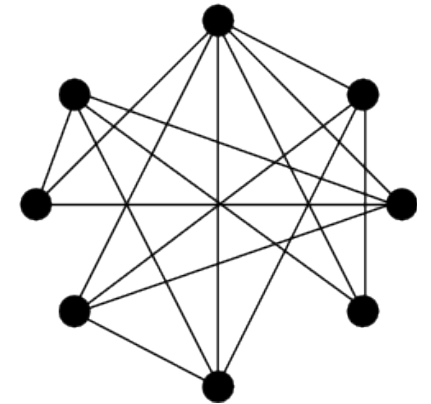
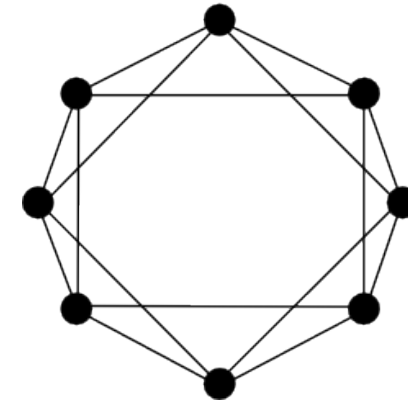
**AMS**

Brief note on references:

This talk does not include references to literature, which are numerous and important. Most (but not all) references *are* included in the arXiv paper: [arxiv.org/abs/1608.00607](https://arxiv.org/abs/1608.00607)

# Stochastic models, sets, and distributions

- a generative model is just a recipe:  
choose parameters → make the network
- a **stochastic** generative model is also just a recipe:  
choose parameters → draw **a** network
- since a single stochastic generative model can generate many networks, the model itself corresponds to a **set of networks**.
- and since the generative model itself is some combination or composition of random variables, a **random graph model** is a set of possible networks, each with an associated probability, i.e., a distribution.



this talk:

**configuration models:** uniform distributions over networks w/ fixed deg. seq.

# Why care about random graphs w/ fixed degree sequence?

Since many networks have broad or peculiar degree sequences, these random graph distributions are commonly used for:

## **Hypothesis testing:**

Can a particular network's properties be explained by the degree sequence alone?

## **Modeling:**

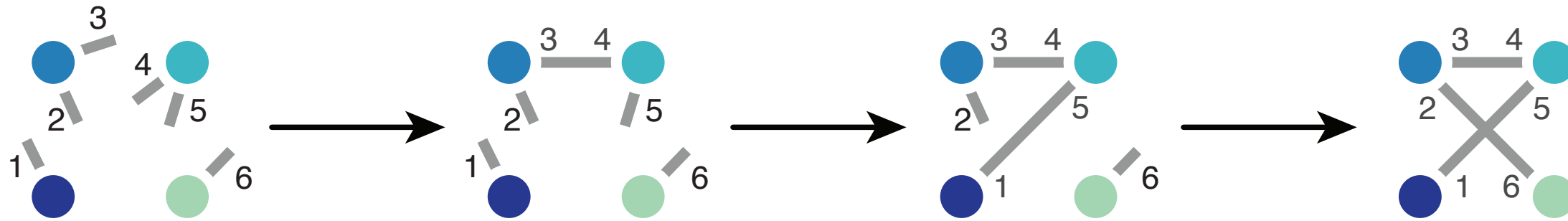
How does the degree distribution affect the epidemic threshold for disease transmission?

## **Null model for Modularity, Stochastic Block Model:**

Compare an empirical graph with (possibly) community structure to the ensemble of random graphs with the same vertex degrees.

# Stub Matching to draw from the config. model

$$\vec{k} = \{1, 2, 2, 1\}$$

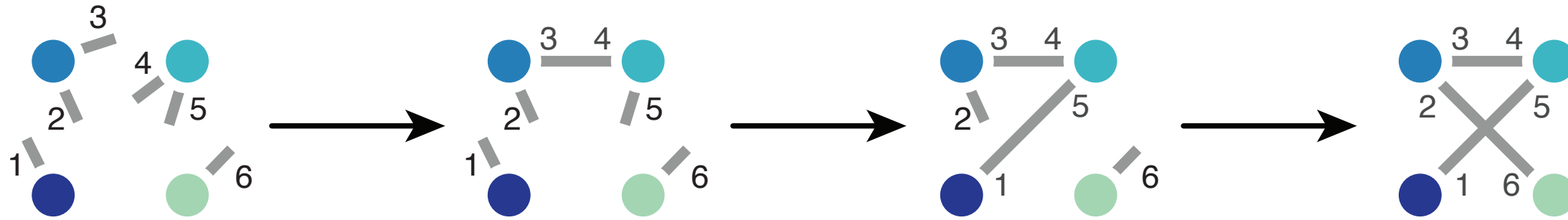


the standard algorithm:  
draw from the distribution by sequential “Stub Matching”

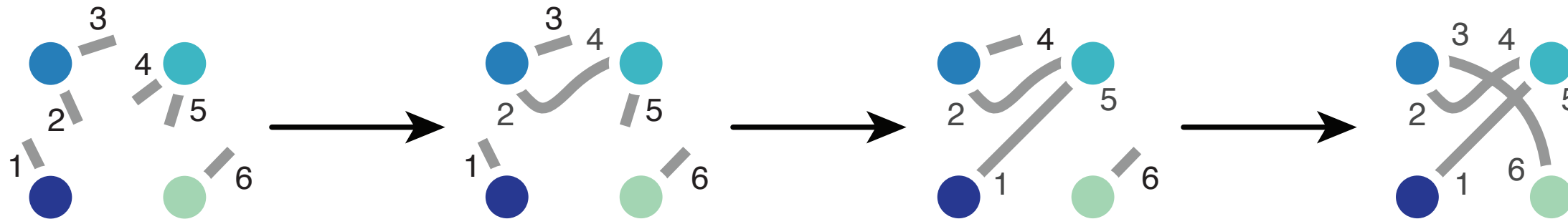
1. initialize each node  $n$  with  $k_n$  half-edges or stubs.
2. choose two stubs uniformly at random and join to form an edge.

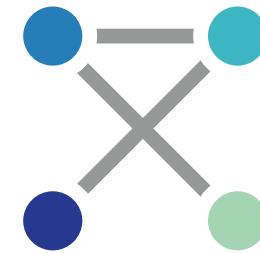
# Stub Matching to draw from the config. model

draw #1



draw #2





Are these two different networks? or the same network?

Are stubs distinguishable or not?

The rest of this talk: the answer matters.



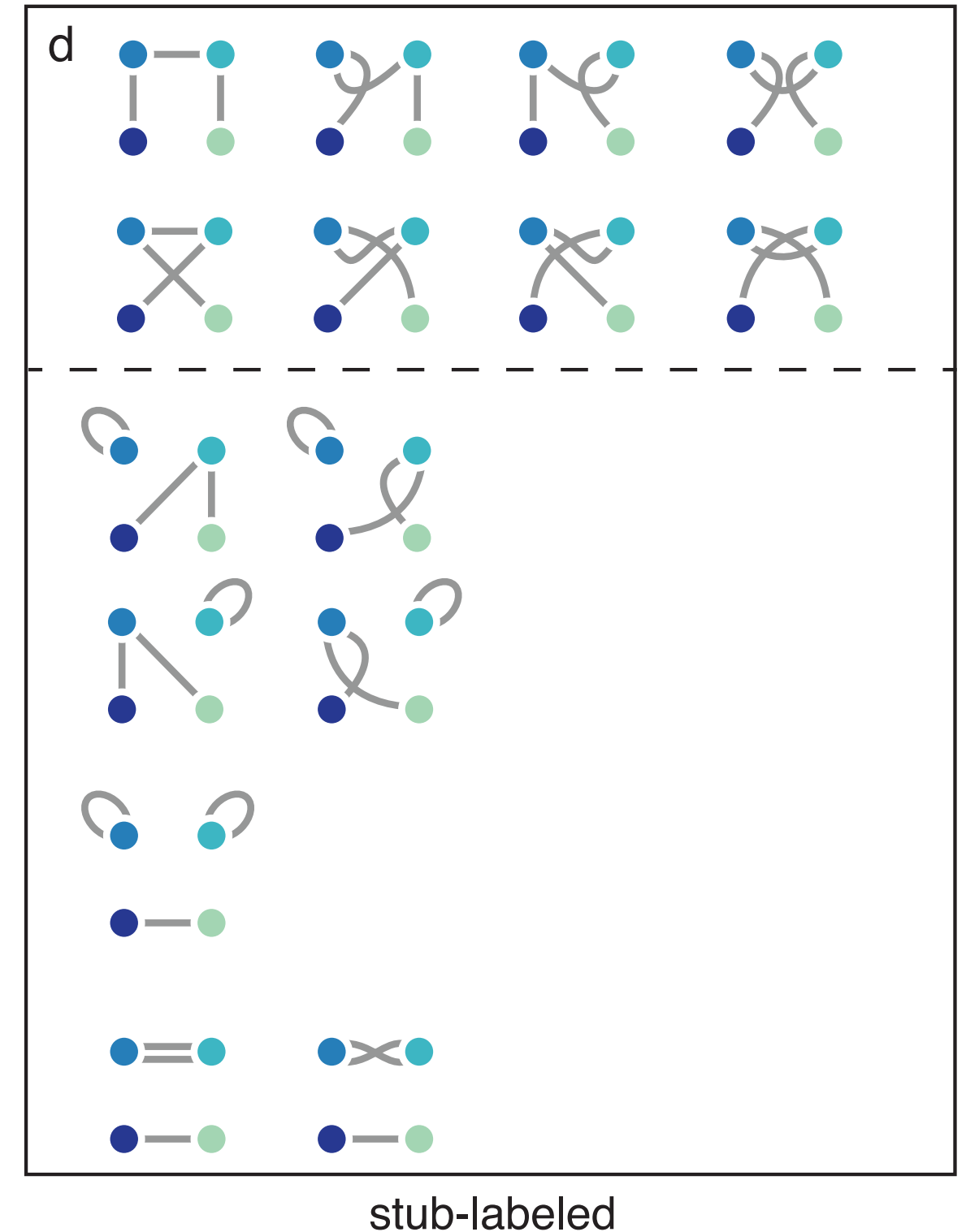
# The distribution according to stub-matching

When we draw a graph using stub matching, this is the set of graphs that we uniformly sample.

8 of the graphs are simple, while the other 7 have self-loops or multiedges.

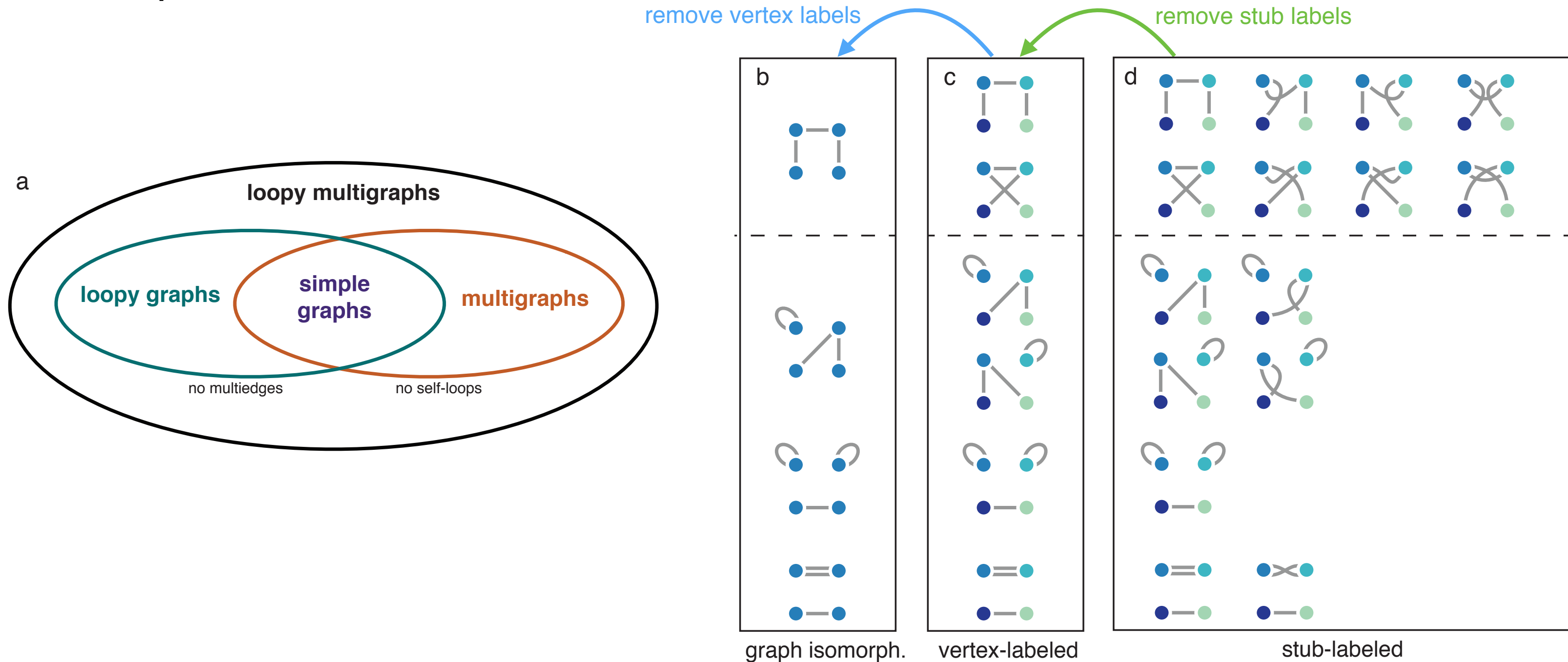
We therefore say that stub matching uniformly samples space of **stub-labeled loopy multigraphs**.

Note, however, that this is not a uniform sample over **adjacency matrices** (rows).



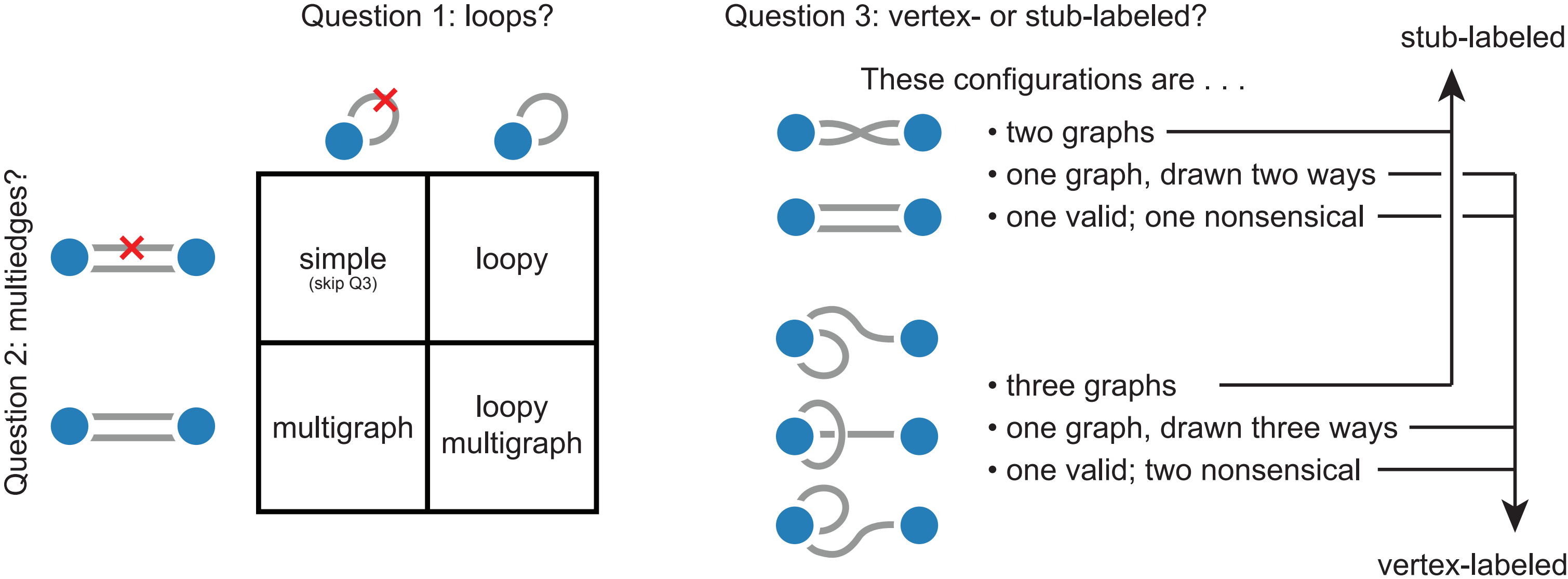


# The importance of uniform distributions



goal: provably uniform sampling for all eight spaces:  
 $\text{loopy}\{0, 1\} \times \text{multigraph}\{0, 1\} \times \{\text{stub-}, \text{vertex-}\}$

# Choosing a space for your configuration model



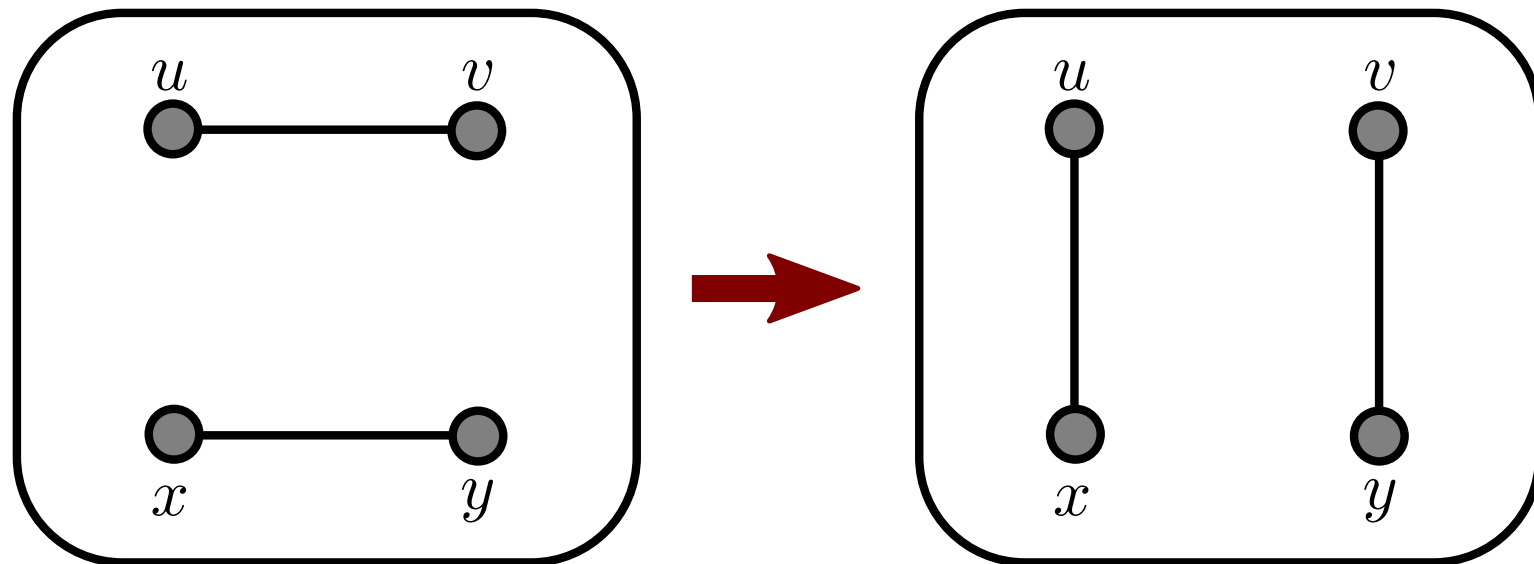
example: Are loops reasonable? Would a loop make sense? [tennis matches: no | author citations: yes]

# Sampling from configuration models

stub matching samples uniformly from stub-labeled loopy multigraphs

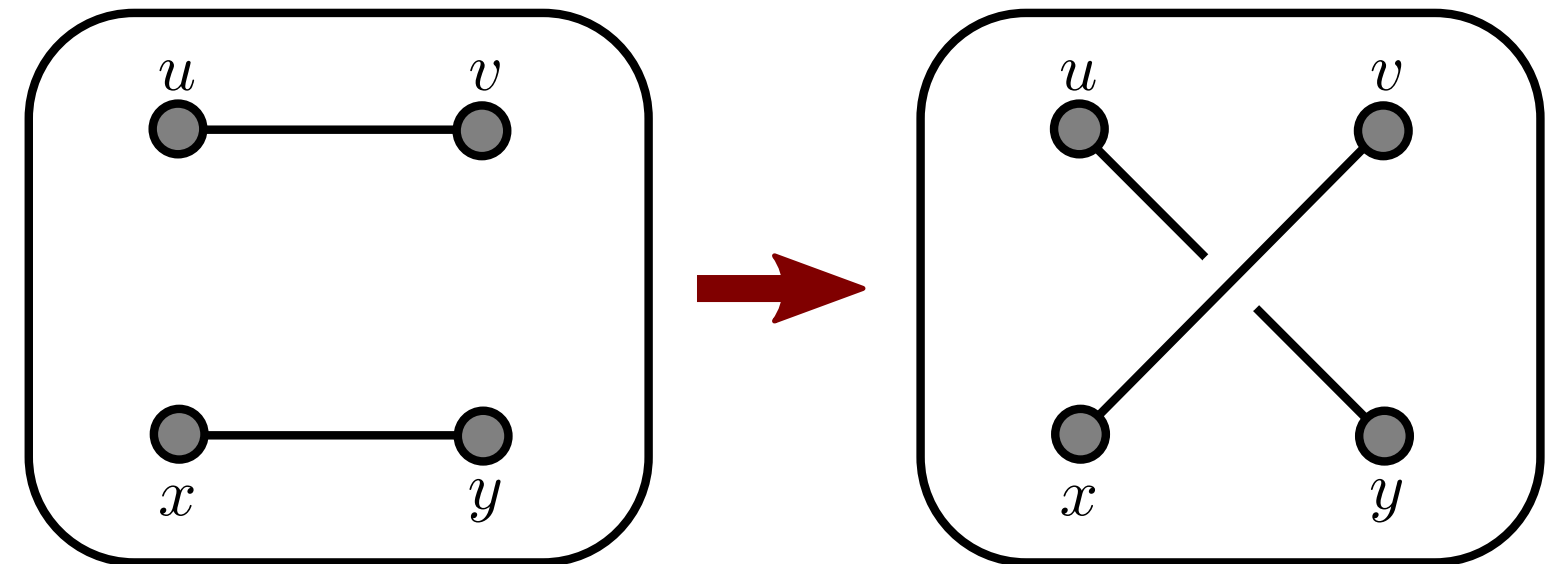
for other spaces, define a Markov chain over the “graph of graphs”  $\mathbf{G}$   
→ each vertex is a graph, and directed edges are “double-edge swaps”

$$(u, v), (x, y) \rightsquigarrow (u, x), (v, y)$$



swap this way

$$(u, v), (y, x) \rightsquigarrow (u, y), (v, x)$$



or the other way

# Markov chains for uniform sampling

Prove that:

- the transition matrix is doubly stochastic (**G** is regular)
- the chain is irreducible (**G** is strongly connected)
- the chain is aperiodic (**G** is aperiodic; gcd of all cycles is one)

Straightforward for **stub-labeled loopy multigraphs**.

Choose two edges uniformly at random and swap them. Accept all swaps and treat each resulting graph as a sample from the U distribution. (Each node in **G** has degree  $m$ -choose-2.)

Easy for **stub-labeled multigraphs**.

Choose two edges uniformly at random and swap them. Reject swaps that create a self-loop and *resample* the current graph. (Think of any “rejected swap” as a self-loop in **G**.)

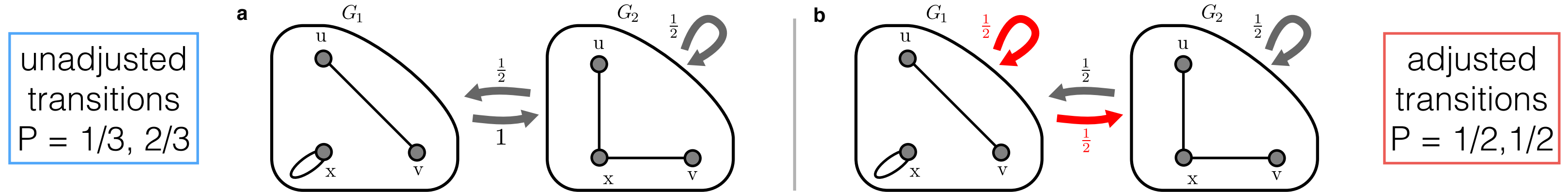
Easy for **simple graphs**.

Choose two edges uniformly at random and swap them. Reject swaps that create a self-loop or multiedge and *resample* the current graph. (Again, treat “rejected swaps” as a self loops in **G**.)

# Markov chains for uniform sampling

For **vertex-labeled** graphs, we inherit the strong connectedness of **G** as well as its aperiodicity.

However, ensuring that the Markov chain has a uniform distribution as its stationary distribution requires that we adjust transition probabilities.



These asymmetric modifications to transition probabilities depend on the number of self-loops and multiedges in the current state.

**Intuition:** decrease outflow (and increase resampling)  
of graphs with multiedges or self-loops.

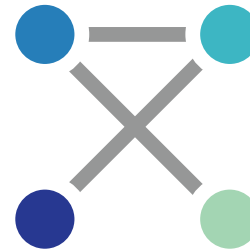
# Stub-labeled loopy graphs: not connected



**counterexample:** no double-edge swap connects these two graphs!

Do {**stub labels**, **self-loops**, **multiedges**} matter for how we sample CMs? **yes**

showed that these spaces are far from equivalent, even in thermodynamic lim.



introduced (and just outlined) provably uniform sampling methods.

Do {**stub labels**, **self-loops**, **multiedges**} matter in applications of CMs? **next...**

→ hypothesis testing



→ null model for modularity



# Hypothesis testing

Do barn swallows tend to associate with other swallows of similar color?

Data: bird interactions, bird colors.

Compute color assortativity  
[correlation over edges]





# Choose a graph space for barn swallows

Q: 1: loops?

**Nonsensical**



## 2: multiedges?

Reasonable

Reasonable  
[in our data, in fact!]

simple (skip Q3)	loopy
multigraph	loopy multigraph

Question 3: vertex- or star-labeled?

These conversations are . . .

vertex-labeled



- two graphs —————
- one graph, drawn two ways
- one valid; one nonsensical -

[Why? If we interacted today and yesterday, a randomization in which my today interacts with your yesterday is nonsensical!]

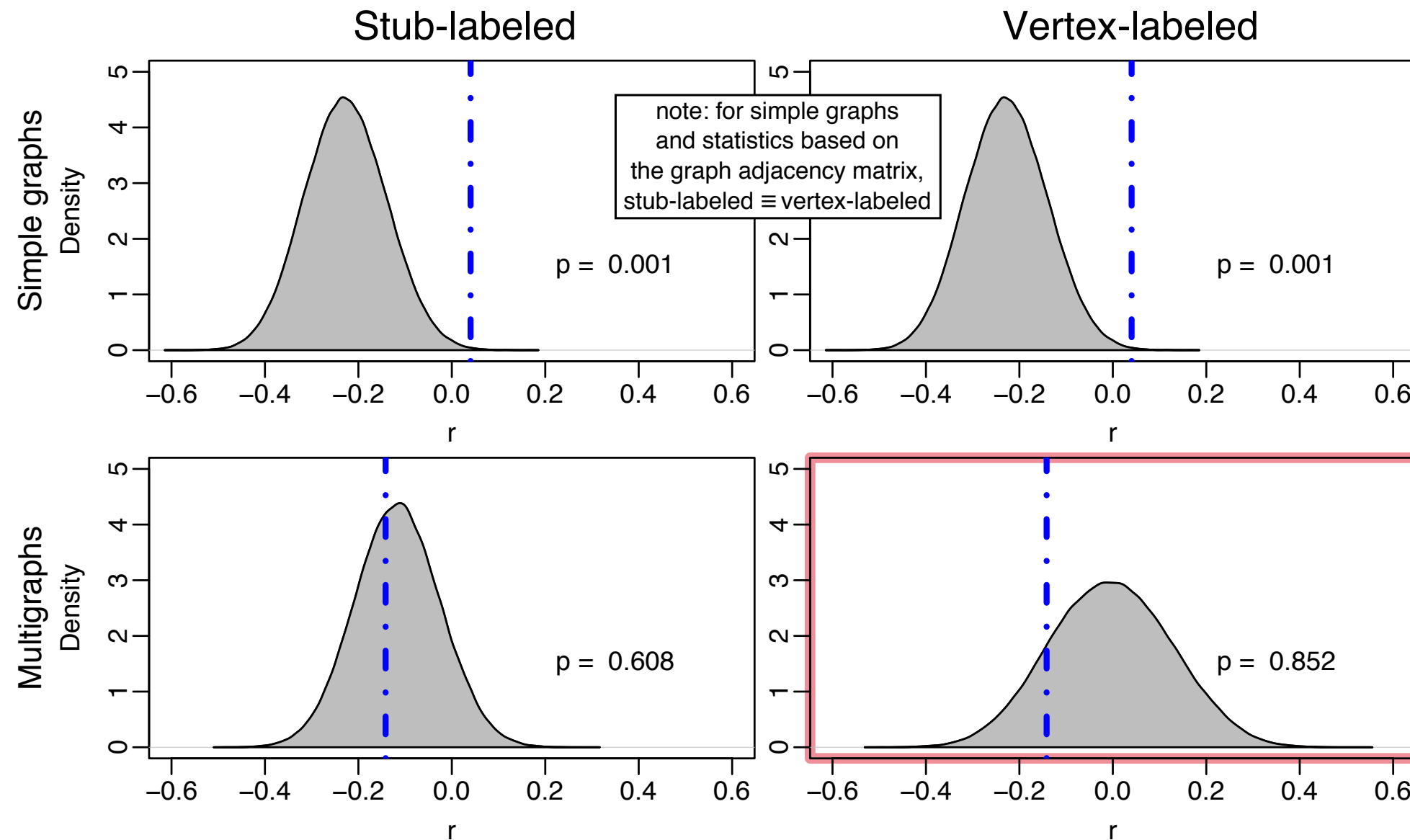
- one valid; two nonsensical

## stub-labeled

vertex-labeled

This should be modeled as a **vertex-labeled multigraph**.

# Assortative pairing of barn swallows



Sanity check:  
should be = for simple

NONE of these is centered at zero.  
Correct space is *meaningfully* different.

Uniform sampling means we can compare empirical value  
to null distribution to draw scientific conclusions.

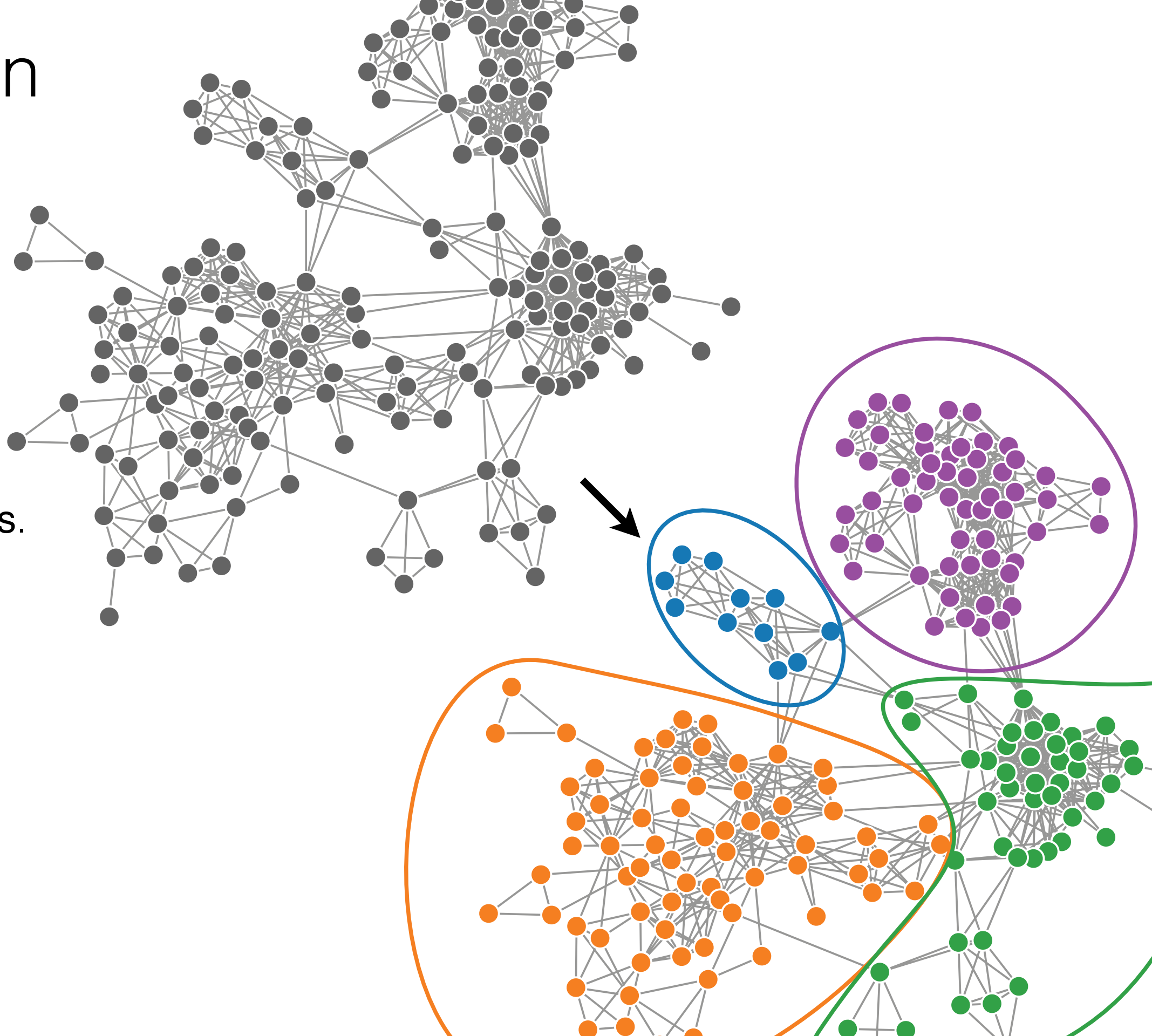
The choice of graph space matters—careful choice & sampling can flip conclusions!

# Community Detection

Are there groups of vertices that  
**tend to associate** with each other  
more than **we expect by chance**?

Data: collaborations among geometers.

Maximize modularity, e.g.



# Coauthorship communities (vertex-labeled multigraph)

## Modularity

$$Q = \frac{1}{2M} \sum_{i,j} \left( A_{ij} - \frac{k_i k_j}{2M} \right) \delta(g_i, g_j)$$

expected number of edges in a random degree-preserving null model

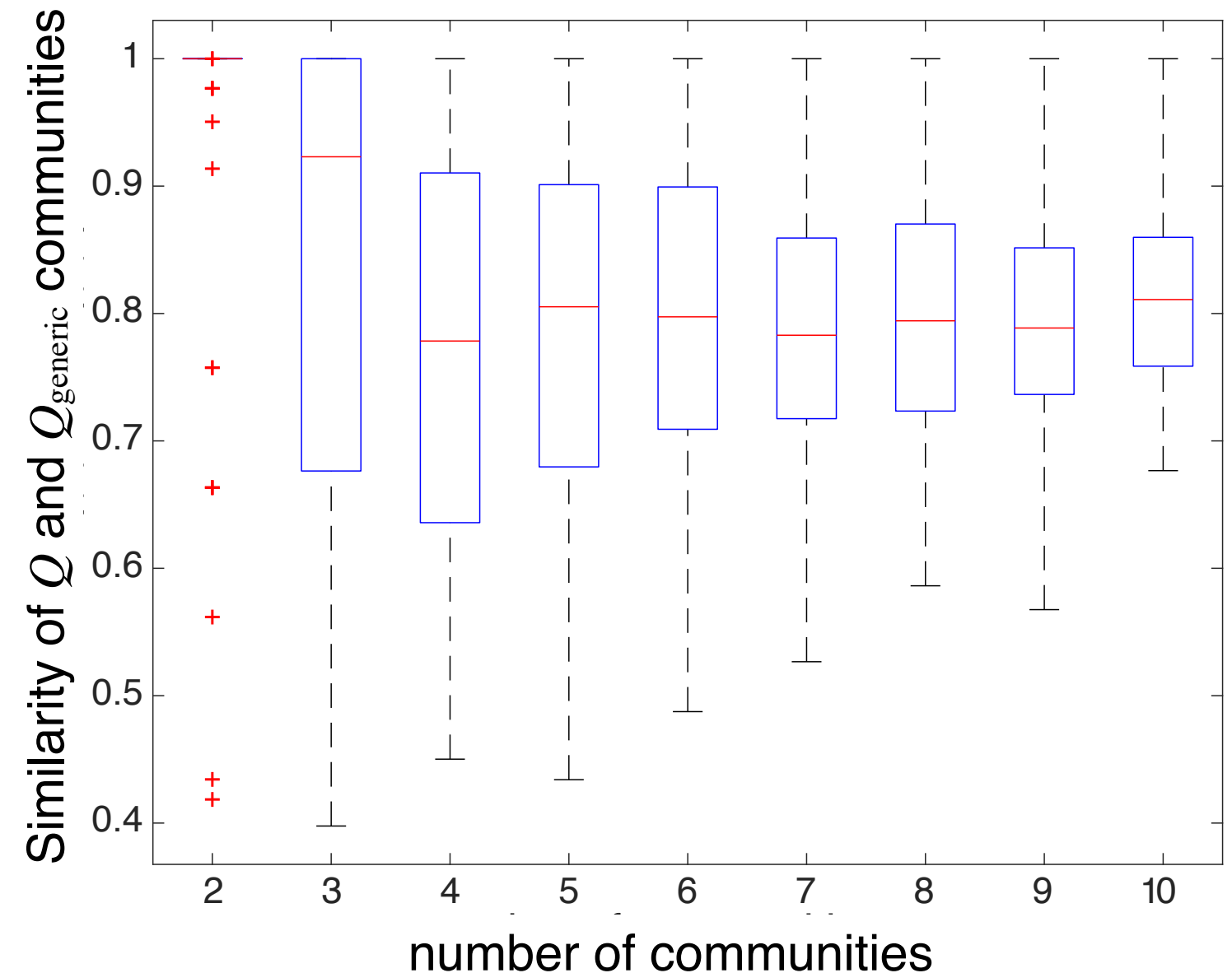
specifically, in the stub-labeled loopy multigraph CM

## Generic Modularity

$$Q_{\text{generic}} = \frac{1}{2M} \sum_{i,j} \left( A_{ij} - \frac{\mathbb{E}[C_{k_i, k_j}]}{n_{k_i} n_{k_j}} \right) \delta(g_i, g_j)$$

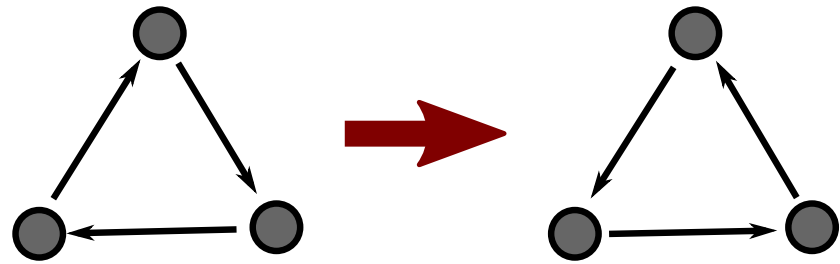
expected number of edges in *any* random degree-preserving null model

same community detection algorithm, same initial state, different results



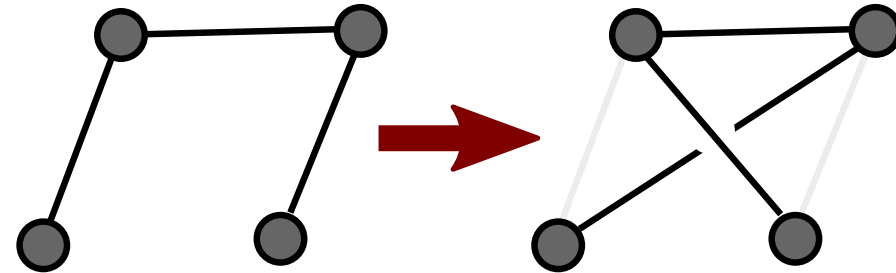
# Advanced edge swaps

**a** reversing a directed triangle



required for graph-of-graphs  
irreducibility in directed networks

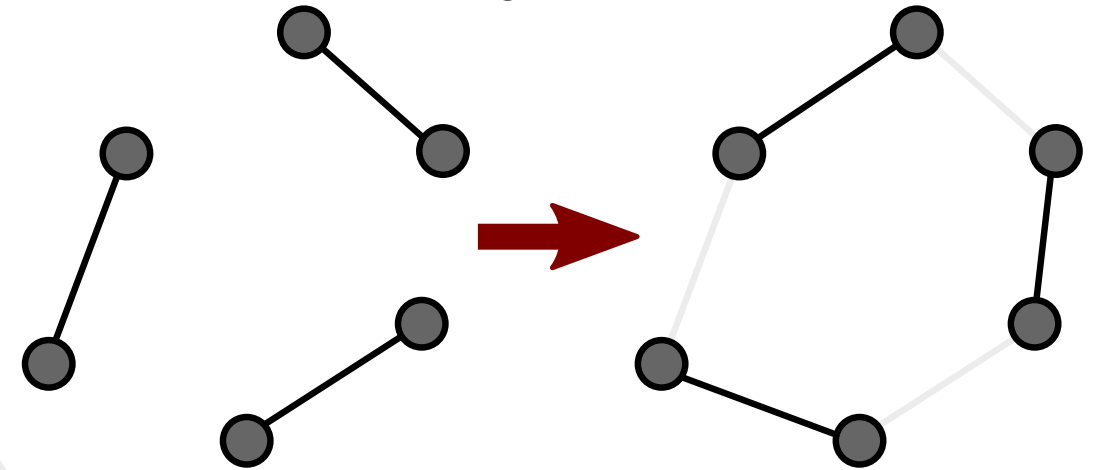
**b** connectivity preserving edge swap



useful if you wish to sample *only*  
networks that have a fixed number  
of connected components

**c**

3 edge swap



other swaps have been proposed,  
e.g. to improve mixing time

Proofs, samplers, the history of the configuration model, and applications in the paper

The point: graph spaces & stub labels *matter*, in theory and in practice.

Recognizing this exposes a number of unrecognized & unsolved problems.

Provably uniform sampling methods exist—some have existed for decades!





Bailey Fosdick  




Joel Nishimura  




Johan Ugander  


Configuring Random Graph Models with Fixed Degree Sequences  
Fosdick, Larremore, Nishimura, Ugander. To Appear in *SIAM Review*.  
[arxiv.org/abs/1608.00607](https://arxiv.org/abs/1608.00607)



[danlarremore.com/configuration\\_models](http://danlarremore.com/configuration_models)



# Thank you

@danlarremore  
larremore@santafe.edu