# Learning Multifractal Structure in Large Networks

Austin Benson, Carlos Riquelme, Sven Schmit

Stanford University
{ arbenson, rikel, schmit } @ stanford.edu

Knowledge Discovery and Data Mining (KDD)
August 26, 2014

# Setting

We want a simple, scalable method to model networks and generate random (undirected) graphs

- Looking for random graph generators that can mimic real world graph structure
  - Power law degree distribution,
  - High clustering coefficient, etc.
- Many models have been proposed, starting with Erdos-Renyi graphs
- Relatively recent models: SKG [Leskovec et al. 2010], BTER [Seshadhri et al. 2012], TCL [Pfeiffer et al. 2012]
- In 2011 Palla et al. introduce multifractal network generators, 'generalizing' SKG

# Our contributions

We propose methods to make MFNG a feasible framework to model large networks

# Our contributions

We propose methods to make MFNG a feasible framework to model large networks

- First, we give an intuitive theoretical result that opens the door to scalable estimation
- We show how we can fit MFNG to graphs using mehod of moments estimation, with runtime independent of the size of the graph
- We develop a fast heuristic for sampling MFNG
- We demonstrate the effictiveness of our approach in synthetic and real world settings.

# An introduction to Multifractal Network Generators (MFNG)

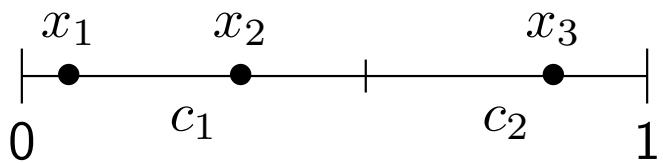
Ingredients

- Number of nodes: $n$
- Number of categories: $m$ with specified lenghts $l_i$
- Number of recursive levels: $k \approx \log_m(n)$
- Probabilities of edges between nodes, based on categories, stored in matrix $P \in [0,1]^{m \times m}$

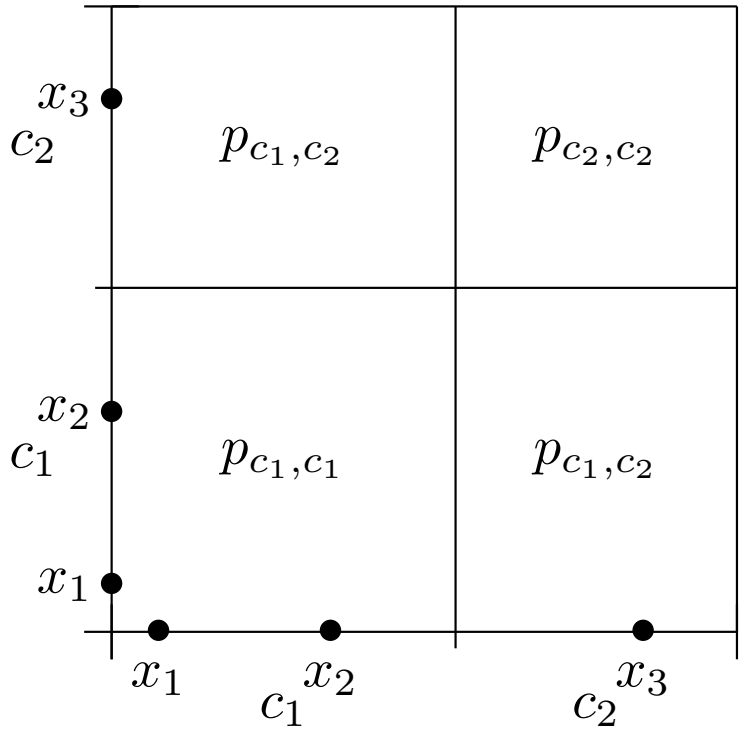
# Generating a graph with no recursion

Let's consider the simple case first: $k = 1$

- Begin with a line: $[0, 1]$
- Divide the line in $m$ intervals (or categories) with lengths $l_1, l_2, \ldots, l_m$
- Sample nodes on the line according to a uniform distribution: this gives every node a category



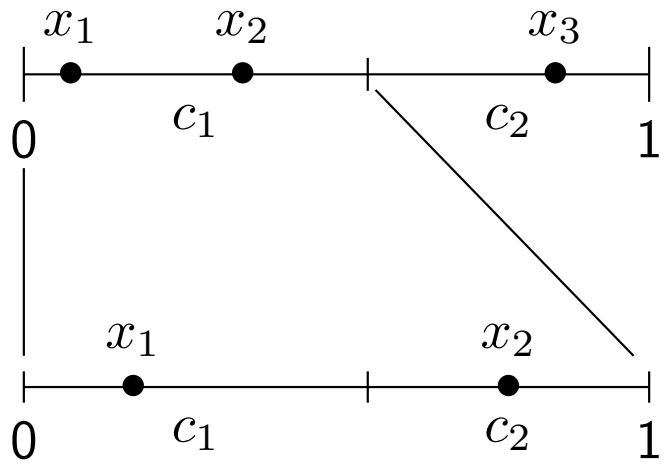
# From line to square




- For any two nodes $u \in c_i, v \in c_j$, add an edge with probability according to $p_{c_i,c_j}$

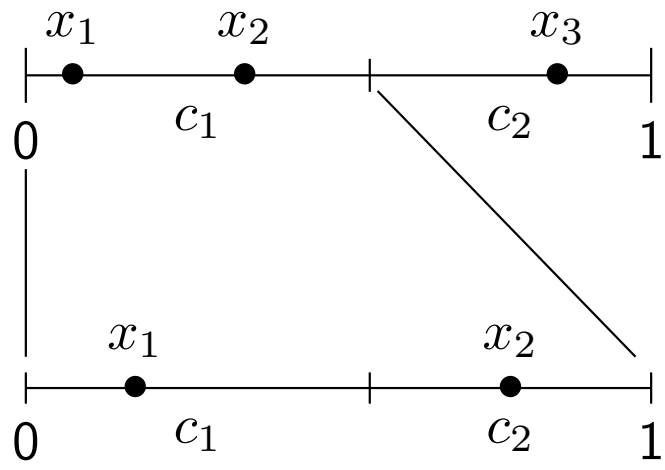
# Adding recursion

For subsequent levels, we subdivide the intervals again to find categories of nodes in the next layer



# Adding recursion

For subsequent levels, we subdivide the intervals again to find categories of nodes in the next layer




Now add an edge between nodes by multiplying probabilities corresponding to categories in each layer.
In the above two layer example

- node $x_1$ has categories $(c_1, c_1)$,
- node $x_2$ has categories $(c_1, c_2)$, and
- node $x_3$ has categories $(c_2, c_2)$

And hence, we add edge $(x_1, x_2)$ with probability $p_{c_1,c_1} p_{c_1,c_2}$.

# Expanding the recursion

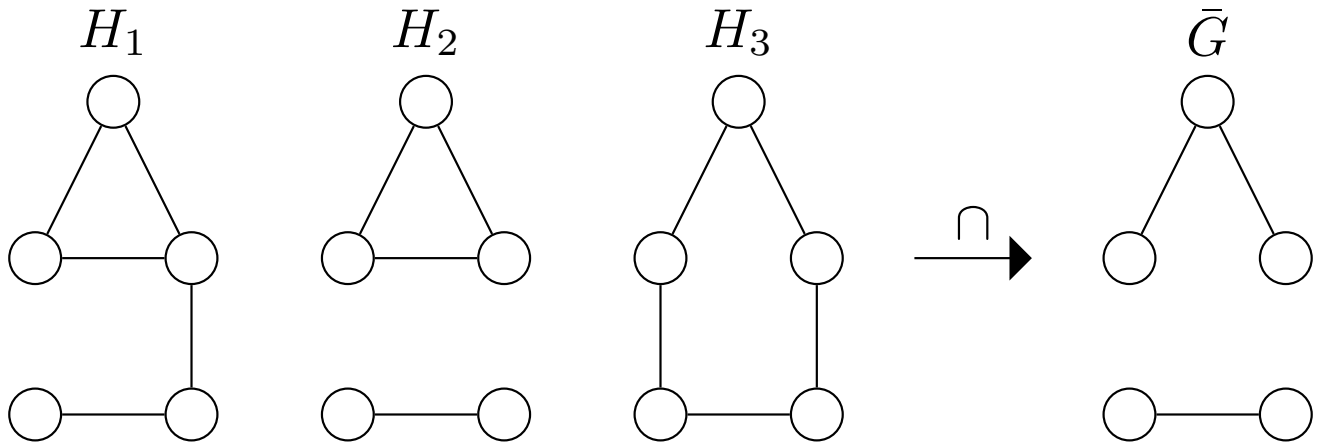
So we can get a full probabilistic adjacency matrix $Q \in [0,1]^{m^k \times m^k}$ by expanding all recursive levels

Problem: $Q$ grows fast with $k$. Difficult to do inference.

Intuitively, we should not have to do this.

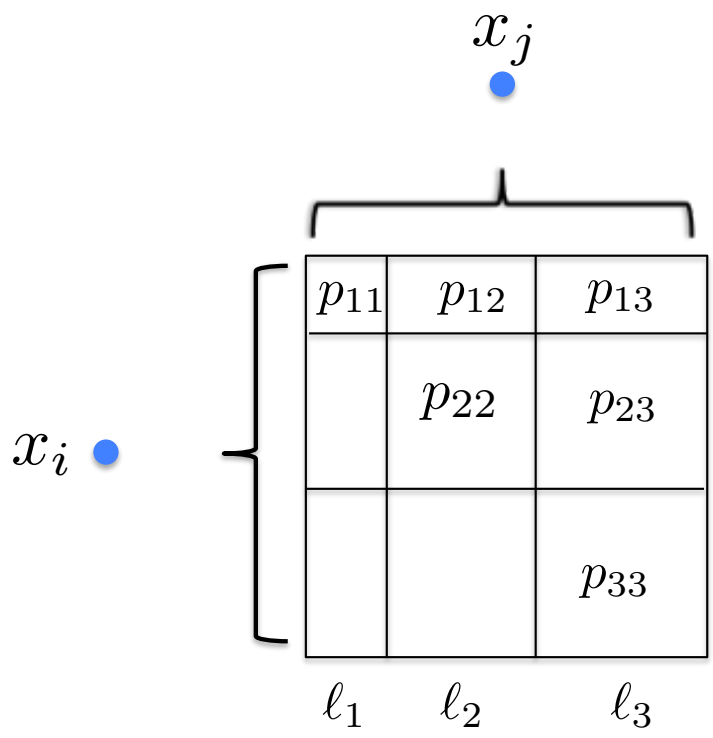
# Main theoretical result

Consider sampling $k$ graphs from a MFNG with $1$ recursive level, and construct a new graph $\bar{G}$ by taking the intersection over graphs:




Then $\bar{G}$ has same distribution as a graph $G$ generated from a MFNG with $k$ recursive levels.

# Computing expected number of edges is easy



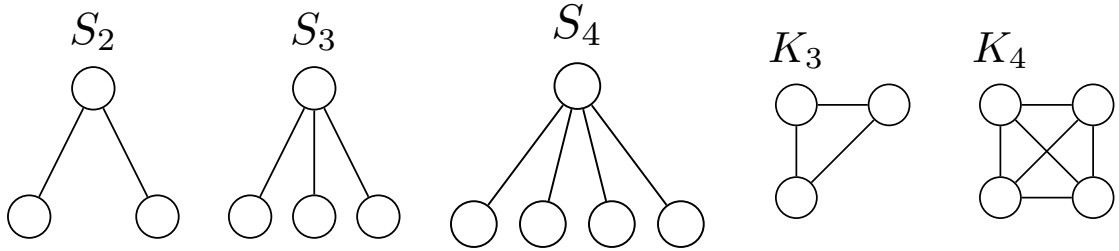

$$\mathbf{Prob}((u,v) \in E) = p = \sum_{i=1}^{3}\sum_{j=1}^{3} \ell_i \ell_j p_{ij}, \quad \mathbb{E}\{|E|\} = \binom{n}{2} p^k$$

So computing $p$ is $O(m^2)$ instead of $O(m^{2k})$.

# Computing moments of certain subgraphs is easy

With above theory, we can easily compute the expected number of...

- edges, wedges, $3$-stars, $4$-stars ...
- triangles, $4$-cliques...



# We can learn multifractal structure quickly

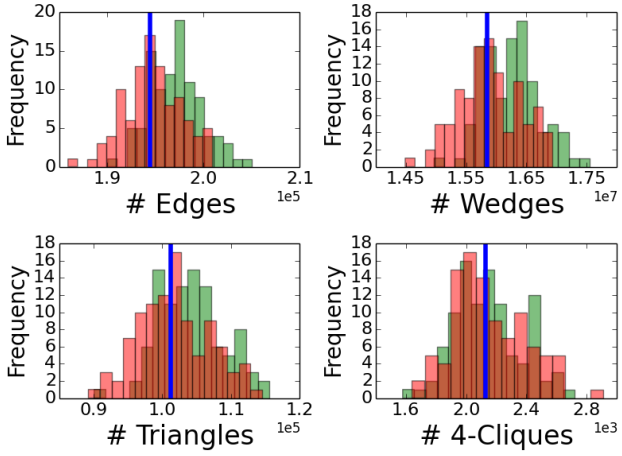
Method of moments:

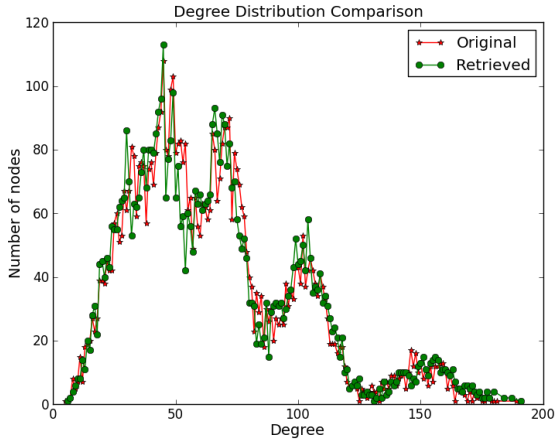
1. Count number of wedges, 3-stars, triangles, 4-cliques, etc. in network of interest
2. Try to find parameters such that the expected values, $\mathbb{E}[F_i]$, match the empirical counts, $f_i$

$$\begin{aligned}
\underset{P,\ell,r}{\text{minimize}} \quad & \sum_i \frac{|f_i - \mathbb{E}[F_i]|}{f_i} \\
\text{subject to} \quad & 0 \le p_{ij} = p_{ji} \le 1, \quad 1 \le i \le j \le c \\
& 0 \le \ell_i \le 1, \qquad\qquad 1 \le i \le c \\
& \sum_{i=1}^{m} \ell_i = 1
\end{aligned}$$

Key idea: Once we have the counts $(f_i)$, this optimization routine is independent of the size of the graph.

# Method of moments recovers small synthetic graphs

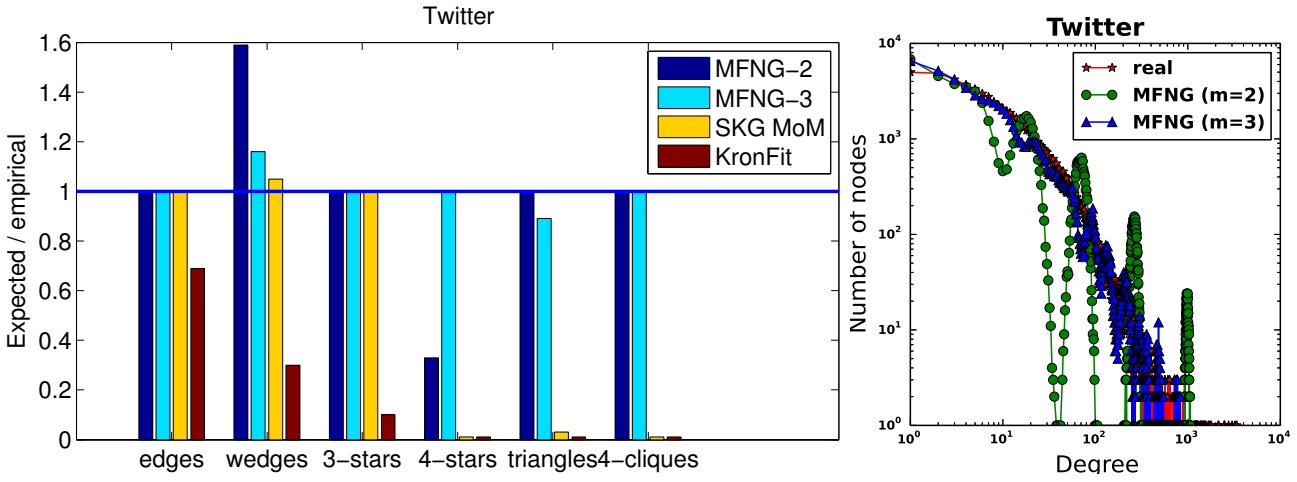



Original MFNG, Single sample, Recovered MFNG

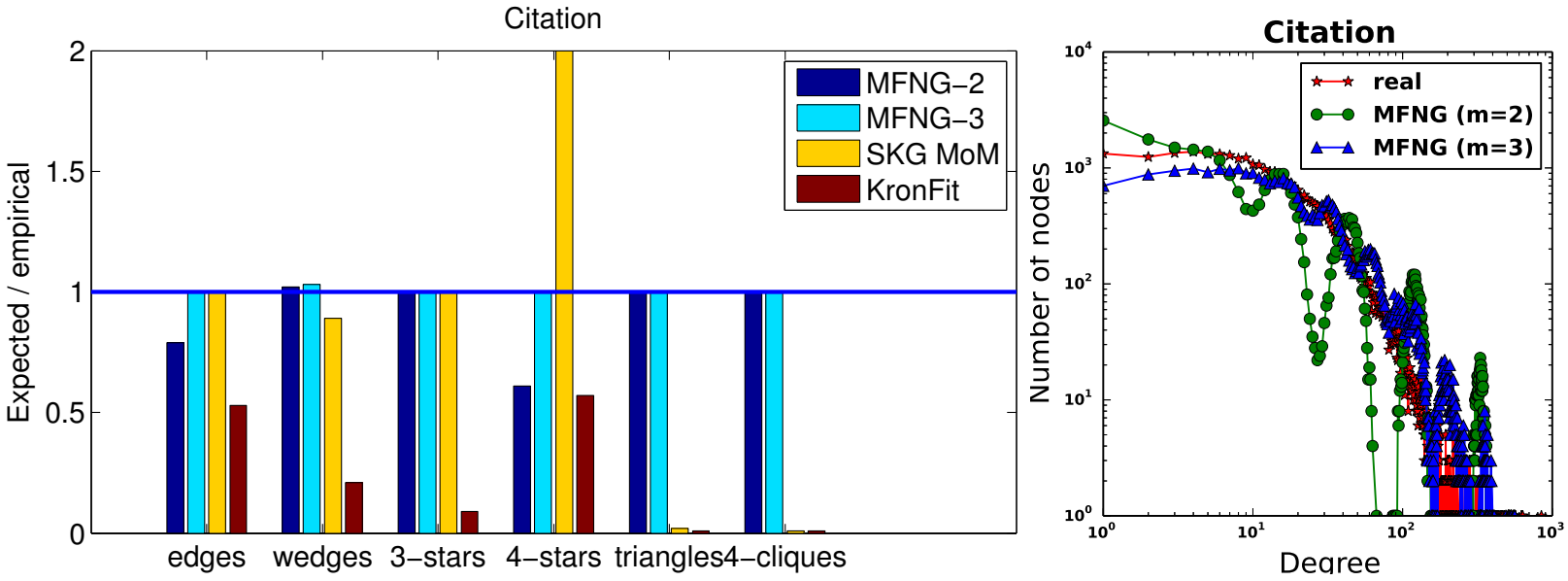
| | $\lvert V\rvert$ | $m$ | $k$ | $\ell_1$ | $\ell_2$ | $p_{11}$ | $p_{12}$ | $p_{22}$ |
|---|---|---|---|---|---|---|---|---|
| Original | 6,000 | 2 | 10 | 0.25 | 0.75 | 0.59 | 0.43 | 0.78 |
| Recovered | 6,000 | 2 | 9 | 0.2728 | 0.7272 | 0.5431 | 0.4101 | 0.7593 |

# Twitter network




Comparison against a method of moments for Stochastic Kronecker Graphs [Gleich and Owen 2012] and KronFit [Leskovec et al. 2010]

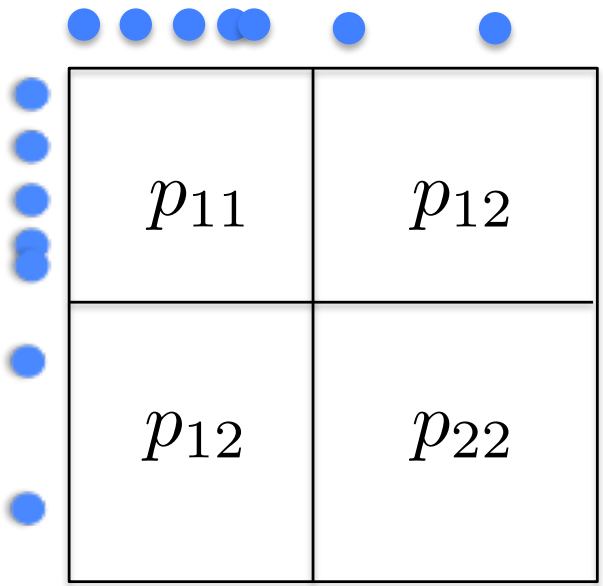
# Citation network




- Triangles and $4$-cliques are again matched in expectation.
- We employ "noisy SKG" strategy [Seshadhri et al. 2013] to dampen the degree distribution.

# Fast sampling is challenging

- Naive way: flip a coin for each $O(n^2)$, while we would like $O(|E|)$
- Idea from SKG: fix number of edges and then use 'ball-dropping'
- Problem for MFNG: many nodes can fall into a single box, we have to ensure we still sample enough edges from that box



# Conclusion

We proposed methods to make MFNG a feasible framework to model large networks

- First, we give an intuitive theoretical result that opens the door to scalable estimation
- We show how we can fit MFNG parameters to arbitrarily largh graphs using mehod of moments estimation
- We develop a fast heuristic for sampling MFNG
- We demonstrate the effictiveness of our approach in synthetic and real world settings

# Learning Multifractal Structure in Large Networks

Questions?


- Austin Benson: arbenson@stanford.edu
- Carlos Riquelme: rikel@stanford.edu
- Sven Schmit: schmit@stanford.edu