SAND2011-3396C

Constructing and Sampling Graphs with a Given Joint Degree Distribution

Isabelle Stanton
isabelle@eecs.berkeley.edu
Computer Science Dept. UC Berkeley

Ali Pinar
apinar@sandia.gov
Sandia National Laboratories

MOTIVATION:

- Many real graphs have power law or lognormal degree distributions
- Many generative models have the right degree distribution, but fail in metrics like conductance
- The joint degree distribution is the distribution over edges
- The joint degree distribution contains all of the information about the degree distribution
- Graphs with the same degree distribution can have very different joint degree distributions
- Maybe looking at the joint degree distribution can help us create better generative models

RELATED WORK: For degree distribution, Sequential Importance Sampling: Bayati, Kim and Saberi, 2007, and Markov Chain Methods: Gkantsidis, Mihail and Zegura and Kannan, Tetali, and Vempala.

Joint degree distribution (or similar quantities) have previously been studied by both Newman and Mahadevan, Krioukov, Fall, and Vahdat. Both give a Markov Chain for sampling random graphs. Newman's method converges to a random graph with a fixed assortativity value, rather than a fixed JDD. Mahadevan et al. fix the JDD but do not provide a transition function that connects the state space. We improve on both of these methods by giving a method for deciding if a JDD is graphical and constructing one if it is, and a provably correct Markov Chain for sampling from the space of graphs.

DEFINITIONS: The joint degree distribution (JDD) of a graph is the probability of a randomly sampled *edge* being between vertices of degree *k* and *l*. This is denoted by P[*k*,*l*]. The joint degree matrix (JDM) is J[*k*,*l*]=*m*P[*k*,*l*], the count of the number of edges a graph has between degrees *k* and *l*.

The degree distribution of a graph can be found from the JDD. It is exactly that

$$p_k = \frac{\mu}{k}\left(P[k,k] + \sum_l P[k,l]\right)$$

where μ is the average degree in the graph. Similarly, the degree sequence can be obtained as the unnormalized version from the JDM.

$$kP[k] = J[k,k] + \sum J[k,l]$$

A nearly identical distribution is the *joint probability distribution of the remaining degrees*, as is used in the definition of assortativity by Newman.

Constructing a Graph with a Given Joint Degree Distribution

Necessary and Sufficient Conditions: A JDM *J* is realizable as a simple graph if:

$$\forall k,l,k \neq l, J[k,l] \leq P[k]P[l] \qquad \forall k, J[k,k] \leq \binom{P[k]}{2}$$

The necessity comes from the fact that the LHS is the maximum number of edges of that type possible in any simple graph. The sufficiency is due to the following constructive algorithm that succeeds exactly when these are satisfied. The proof of correctness of the algorithm relies on the fact that this method maximizes the number of vertices with non-zero residual degree at all times.

Additionally, these conditions provably imply the Erdos-Gallai condition for a degree sequence to be graphically realizable. The condition is that a decreasing sorted degree sequence, $\{d_1,d_2,\ldots d_n\}$ is graphical if and only if

$$\forall k \leq n, \sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min(d_i,k)$$

```
Input: $\mathcal{J}, n, m, \mathcal{D}$
for $k = n \cdots 1$ do
    for $l = k \cdots 1$ do
        if $\mathcal{J}_{k,l} \neq 0$ then
            if $k \neq l$ then
                Let $a = \mathcal{J}_{k,l} \mod \mathcal{D}_k$ and $b = \mathcal{J}_{k,l} \mod \mathcal{D}_l$
                Let $x_1 \cdots x_a = \lfloor \frac{\mathcal{J}_{k,l}}{\mathcal{D}_k} \rfloor + 1$ and $x_{a+1} \cdots x_{\mathcal{D}_k} = \lfloor \frac{\mathcal{J}_{k,l}}{\mathcal{D}_k} \rfloor$
                Let $y_1 \cdots y_b = \lfloor \frac{\mathcal{J}_{k,l}}{\mathcal{D}_l} \rfloor + 1$ and $y_{b+1} \cdots y_{\mathcal{D}_l} = \lfloor \frac{\mathcal{J}_{k,l}}{\mathcal{D}_l} \rfloor$
                Construct a bipartite graph $B$ with degree sequence $x_1 \cdots x_{\mathcal{D}_k}, y_1 \cdots y_{\mathcal{D}_l}$
            else
                Let $c = 2\mathcal{J}_{k,k} \mod \mathcal{D}_k$
                Let $x_1 \cdots x_c = \lfloor \frac{2\mathcal{J}_{k,k}}{\mathcal{D}_k} \rfloor + 1$ and $x_{c+1} \cdots x_{\mathcal{D}_k} = \lfloor \frac{2\mathcal{J}_{k,k}}{\mathcal{D}_k} \rfloor$
                Construct a simple graph $B$ with the degree sequence $x_1 \cdots x_{\mathcal{D}_k}$
            end if
            Place $B$ into $G$ such that $x_1 \cdots x_a$ and $y_1 \cdots y_b$ are matched with the appropriate degree
            nodes of higher residual degree.
            Update the residual degrees of each $k$ and $l$ degree node.
```

Monte Carlo Markov Chain Sampling: Given a fixed JDM, we are able to sample a random graph with that JDM using a Markov Chain. We either construct or are given a graphs the starting configuration, and then convert it into a configuration in the endpoint model. The transition function is then:

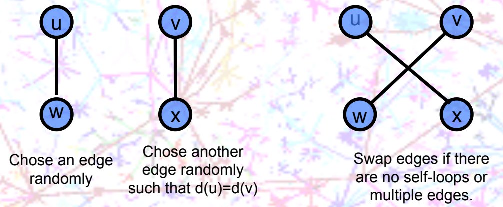
1) With probability 0.5, stay at this configuration. Else:

2) Select any endpoint *e* at uniformly at random. It is currently matched to vertex *u*

3) Select any edge *f* that has the same degree requirement as *e* uniformly at random. It neighbors vertex *v*

4) If the swap (*v*,*e*) (*u*,*f*) does not create a self-loop or multiple edge, then swap. Return to Step 1

If we disregard the if statement in Step 4, the Markov Chain samples uniformly over all pseudographs with a fixed JDD in polynomial time. This can be easily shown by adapting the work of Kannan, Tetali and Vempala. However, we are primarily interested in the problem of sampling simple graphs.



Chose an edge randomly

Chose another edge randomly such that d(u)=d(v)

Swap edges if there are no self-loops or multiple edges.

CORRECTNESS of MARKOV CHAIN: This Markov Chain has the uniform distribution over graphs with the given JDM as its stationary distribution. This is due to a combination of standard facts about Markov Chains i.e. the chain is ergodic and symmetric and an inductive proof that endpoint switches form a connected space in the tradition of Taylor's 1972 result about the same for edge switching.

MIXING TIME of MARKOV CHAIN: Proving the mixing time (time to approximately converge to the stationary distribution) is always challenging. For our Markov Chain, existing approaches like canonical paths aren't easily applied because we reject transitions that create self-loops and multiple edges. This makes it difficult to isolate local changes. Instead, we experimented with the autocorrelation time.

AUTOCORRELATION: Given an independent random sampler from a space with mean μ and variance σ, the autocorrelation of a set of samples adjusted by μ and σ will be 0. If we know the mean and variance then we can use the autocorrelation values of samples generated by the Markov Chain to judge when the chain has converged.

The autocorrelation function is

$$\tau_{\text{int},X} = \frac{1}{2}\sum_{t=-\infty}^{\infty} R_X(t)$$

The integrated autocorrelation time is defined as

$$R_X(\tau) = \frac{E[(X_t - \mu)(X_{t-\tau} - \mu)]}{\sigma^2}$$

Given a length *t* series of sampled graph data, these values can all be easily calculated in O(n^2t log *t*) time.

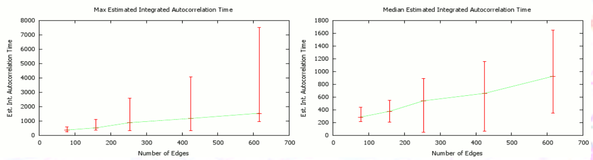
	Word Adjacencies	Dolphins	Football	Karate	Les Miserables
|Vertices|	112	62	115	34	77
|Edges|	425	159	616	78	254
|JDD|	159	61	18	40	99
Median Est. Int. Autocorrelation Time	2589	868	3052	492	1897

Initial results published in Proc. Alenex 11, journal version submitted to ACM JEA.

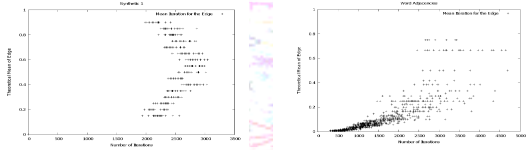
EXPERIMENTS: We used 5 data sets to test the autocorrelation time of our Markov Chain: Word Adjacencies, the Dolphin Social Network, Football Conference Games, the Karate Graph and the Les Miserables dataset. For each dataset, we ran the Markov Chain 15 times and recorded the output for 100,000 steps. For each edge in each run, we calculated the autocorrelation value for lags in step size of 100 from 100 to 15000. We used this to estimate the time for the autocorrelation to drop under a threshold for all edges, and the estimated integrated autocorrelation time.

Given this estimate, we further experimented by taking samples at varying time steps up to the estimated integrated autocorrelation time. For each time step, we then computed the sample mean for each edge and compared it with the theoretical mean obtained from the JDM of the dataset. The mean for an edge between degrees *k* and *l* is *J[k,l]/P[k]P[l].* We show the total variational distance of the sample means from the theoretical means decreases with both the gap and the number of samples.

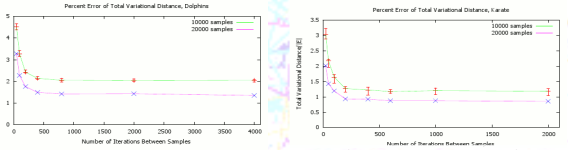
Our experiments were designed based on Sokal's survey on Monte Carlo Methods in Statistical Mechanics and Raftery and Lewis's The Number of Iterations, Convergence Diagnostics and Generic Metropolis Algorithms.

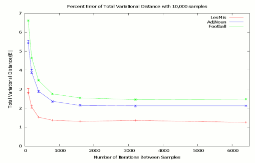


The above graphs have the number of edges in each our 5 datasets as the *x* axis. For each edge of each dataset and each of the 15 runs, we calculated the estimated integrated autocorrelation time. We then took the max and median of each edge respectively, and graphed the median, min and max values of these for each dataset.



We graph the time for the autocorrelation for each edge to drop under 0.001 vs its theoretical mean. The left graph is the Word Adjacency dataset and is the mean over 15 runs, while the right hand diagram is for a synthetic graph that we designed to have many edges over a range of theoretical means. It is the mean of 200 runs for the graph. These graphs suggest that the 'slowest' edges in the system are those with mean 0.5. We are repeating the experiments for larger graphs by sampling edges with means between 0.4 and 0.6 and observing their performance on our metrics.





For each of our datasets, we ran the Markov Chain 10 times and took samples at varying gaps until we had 10,000 samples. These graph the percent error from of the total variational distance between the sample mean for each edge and the theoretical mean from the JDM. We demonstrate that the residual error is due to too few samples by increasing the sample rate to 20,000 samples for 1 run for each graph. The error bars represent that maximum and minimum error for each graph size over the 10 runs. The graph to the left presents the results for the three larger datasets with only 10,000 samples for 1 run.

Funding Statement: This work is supported by Applied Mathematics program of DOE Advanced Scientific Computing Research. Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy, National Nuclear Security Administration under contract DE-AC04-94AL85000.