Using Social Space Models for Inference on Missing Nodes

David J. Marchette*

Abstract
Social network analysis generally assumes a model for the network giving the probabilistic edge structure of the graph. One class of models posits a "social space", defined by latent variables, where the position of the vertices in the space determines the edge probabilities. This has been used for various inferences including predicting missing edges and assigning labels to unlabeled vertices. Often one assumes that all the vertices have been observed, or that the only vertices about which information is sought are those that have been observed. It may seem that the very nature of these models make it impossible to make inferences about missing vertices. We discuss a method for making these inferences, under certain assumptions about the social space, and give some examples of this inference using a simple model which can be fit using a least squares algorithm.

Key Words: Random graph, random dot product graph, latent position model

1. Introduction

Social network analysis has experienced a resurgence among the statistical community as more sophisticated models have become available, and the number of interesting data sets has increased dramatically. For the purposes of this paper, we will consider the simplest of social networks: a simple graph describing the relationships between individuals. Thus, the network consists of a set of vertices, also referred to as nodes or actors, and a set of edges, also referred to as links, or arcs or arrows in the case of directed edges. Early work focused on small networks, because of the difficulty of collecting and processing more complex networks. Both of these issues have been mitigated by modern computers and data methods, and it is not uncommon to come across very large networks, with tens to hundreds of thousands of nodes. One rationale for considering the simple model which we will describe below is that it can be fit to very large graphs, using quite robust sparse matrix methods from linear algebra.

A graph is a pair (V, E), with V a set of vertices, and E a set of edges: pairs of vertices (in a directed graph these are ordered pairs). All graphs will be simple: we do not allow self loops, or edges from a vertex to itself, and we assume there is at most one edge between any two vertices. We will write $u \sim v$ to indicate that $\{u, v\} \in E$ and will denote the edge from u to v as uv. We will concern ourselves with (undirected) graphs, although it is possible to extend the ideas described here to directed graphs. A graph G on n vertices $V(G) = \{v_1, \ldots, v_n\}$ can be represented as an $n \times n$ adjacency matrix. This is a binary matrix $A(G) = (a_{ij})$ with $a_{ij} = 1 \iff v_i \sim v_j$.

Given such a graph, one would like to be able to perform inference on it. One might construct descriptive statistics, such as average path length, clique number, or various measures of centrality of nodes, and compare these with values expected under a given null hypotheses. Alternatively, one might use a model fit to the data to ask questions about the graph or the underlying population, such as: whether a given edge is really there or not, or what a future graph might look like in a dynamic graph. In this paper we consider the question of whether there are nodes missing: can we tell that we have missed important subgroups within the population we are sampling?

We consider a simple model, the random dot product graph (RDPG) model ([2]). This is related to the latent position model of [1]. The idea is to consider the graph an instantiation of a graph-valued random variable, where the edge probabilities are dependent on the actors positions in a *s*ocial space. A space $X \subset \mathbb{R}^d$ is posited, such that to each actor is associated a point (vector) in X, and the probability of an edge between vertices u and v in the graph is some function of their points: $P[u \sim v] = f(x_u, x_v)$. In the latent position model, this is a function of the distance between the points, in our model it is the dot product:

$$P[u \sim v] = f(x_u' x_v).$$

In our work, we assume the function f is a simple threshold to ensure the result is in $[0, 1]$.

$$f(x) = \begin{cases} 0 & x < 0 \\ x & 0 \leq x \leq 1 \\ 1 & x > 1 \end{cases}$$

*Naval Surface Warfare Center, 18444 Frontage Rd., Suite 327, Dahlgren, VA, 22448

We assume that, conditional on the vectors, the edges are independent. Thus, the problem of fitting the model to a given graph reduces to finding vectors whose dot products best "match" the adjacency matrix of the graph.

We use the Frobenius norm to compare matrices.

$$||A - B||_F = \frac{1}{2}\sum(a_ij - b_ij)^2.$$

At first, it would seem that we are looking for vectors $\{x_1, \ldots, x_n\} \in \mathbb{R}^d$ such that $||A(G) - (x_i'x_j)||_F$ is minimized, and hence spectral methods from linear algebra provide the solution. However, note that the diagonal of the adjacency matrix has no meaning, and thus we don't care about the terms in which $i = j$ in the Frobenius norm. We will deal with this issue in the following section.

2. Fitting the Model

If we want to find the vectors U which best "match" the adjacency matrix A (best in Frobenius norm), then the spectral decomposition almost works: the problem, as mentioned above, is the diagonal. We don't care what values are on the diagonal, and we certainly don't want to insist that the vectors have approximately 0 norm (corresponding to the diagonal entries). The iterative algorithm fits the vectors by using prior fits to estimate the diagonal values. Given d:

```
1. Set $D = (0)$, an $n \times n$ matrix of zeros.
    (a) $s = \text{eigen}(A + D, d)$.
    (b) $X = U\sqrt{\Lambda}$.
    (c) $D = \text{diag}(XX')$.
2. Repeat (a)–(c) until convergence.
3. Return $X$.
```

In the algorithm, the function diag() returns a diagonal matrix whose diagonal matches that of its argument. The function eigen() returns the top d eigenvectors U (in the columns) and eigenvalues Λ. We set all negative eigenvalues to 0 prior to setting X in step (b) above.

For directed graphs we are looking for vectors X and Y such that $||A - XY'||_F$ is small, (again ignoring the diagonal). Here, we can think of the model as having an "in" vector and an "out" vector, with the probability of an edge $u \rightarrow v$ being the dot product of the "out" vector of u and the "in" vector of v, corresponding to the left and right singular vectors (scaled by the square root of the singular values) obtained from the singular value decomposition. We will not consider directed graphs in this work, however the approach can be easily modified to handle them.

At first blush, one might question whether our model is capable of detecting missing nodes. Consider the graphs G and $H = G \backslash \{v_i\}$. How can one tell that a model fit to H is "not good enough" and hence a vertex (v_i) is missing? In fact, if we had a perfect fit to the adjacency matrix of G, the corresponding vectors provide a perfect fit to the adjacency matrix of H, and vice versa. Therefore, we cannot use the model (the vectors plus their measured error in accounting for the entries in the adjacency matrix) alone to infer the existence of missing nodes.

The solution, is to use assumptions on the social space. We will assume that the distribution of vectors within the social space is known. This may be accomplished either *a priori* or by observations of similar networks. By making this assumption, we can posit an underlying distribution of vectors in the social space, fit the vectors associated to our graph, then test to see if they have the posited distribution, and if not, make an inference about missing nodes.

In particular, we are looking for "gaps" in the distribution: low density regions (or voids) in areas that are supposed to be high density. Note that this approach does not allow us to infer that a single node is missing: we are looking for changes in distribution. Also, we are assuming a structure to the missing data: they are not missing at random, but rather actors from a particular region of social space are missing. Thus we can detect that certain "groups" are missing or are under-represented in the graph.

To this end, we use the Delaunay triangularization of the points in social space to define a statistic to compare distributions. Given points $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, the Voronoi cell for a point x_i is defined to be the (convex, polygonal) set of all points in $\mathbb{R}^d$ that are closer to x_i than to any other point $x_j \in X$:

$$\mathcal{V}(x_i) = \{x \in \mathbb{R}^d | d(x, x_i) \leq d(x, X \setminus \{x_i\})\}.$$

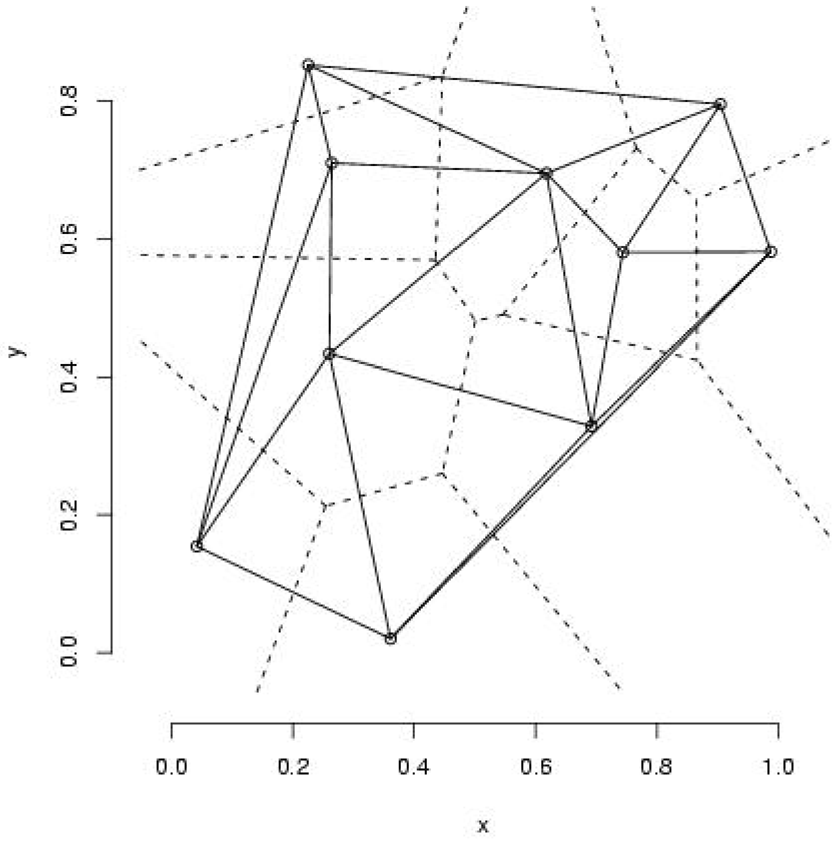

Figure 1: Delaunay triangularization of a set of points in $\mathbb{R}^2$.

Here the distance from a point to a set is defined to be the minimum of the distances from the point to each element in the set. The Delaunay triangularization is the dual to the Voronoi regions: it is the graph with vertices $V = X$ and an edge $x_i \sim x_j$ if and only if the boundaries of their Voronoi regions intersect in an interval. See Figure 1. For the rest of the paper, we will assume $d = 2$. The Delaunay graph has associated with the vertices the points in $\mathbb{R}^2$, and so we may make various calculations, such as the area of triangles in the graph. The graph is (in the 2-d case) planar, and so areas of triangles that are larger than expected represent areas of voids in the distribution of the points.

From the Delaunay triangularization we construct a test statistic, designed to find voids, as follows:

1. Fit the vectors, as above, in $\mathbb{R}_+^2 \cap \mathcal{D}$, where $\mathbb{R}_+$ corresponds to the non-negative reals and $\mathcal{D}$ is the unit disc.

2. Compute the Delaunay triangularization for the fitted points.

3. Compute the test statistic $\tau = \max \text{Area}(\Delta(x_i, x_j, x_k))$ for triangles $\Delta(x_i, x_j, x_k)$ in the Delaunay graph: that is, triples of vertices $\{x_i, x_j, x_k\}$ such that $x_i \sim x_j, x_j \sim x_k$ and $x_i \sim x_k$.

4. Reject the hypothesis that the vectors are uniformly distributed in $\mathbb{R}^2 \cap \mathcal{D}$ for large values of τ.

The rejection tells us there are large empty triangles in the triangularization, which in turn tells us that there are gaps in the distribution. These gaps may be because we have made an incorrect assumption about the underlying "social space" distribution, or that we have missed vertices (actors) in our sample. Furthermore, in the latter case, it gives us a way of characterizing those missed, either in terms of their position in social space, or in terms of their probabilities of association (edges).

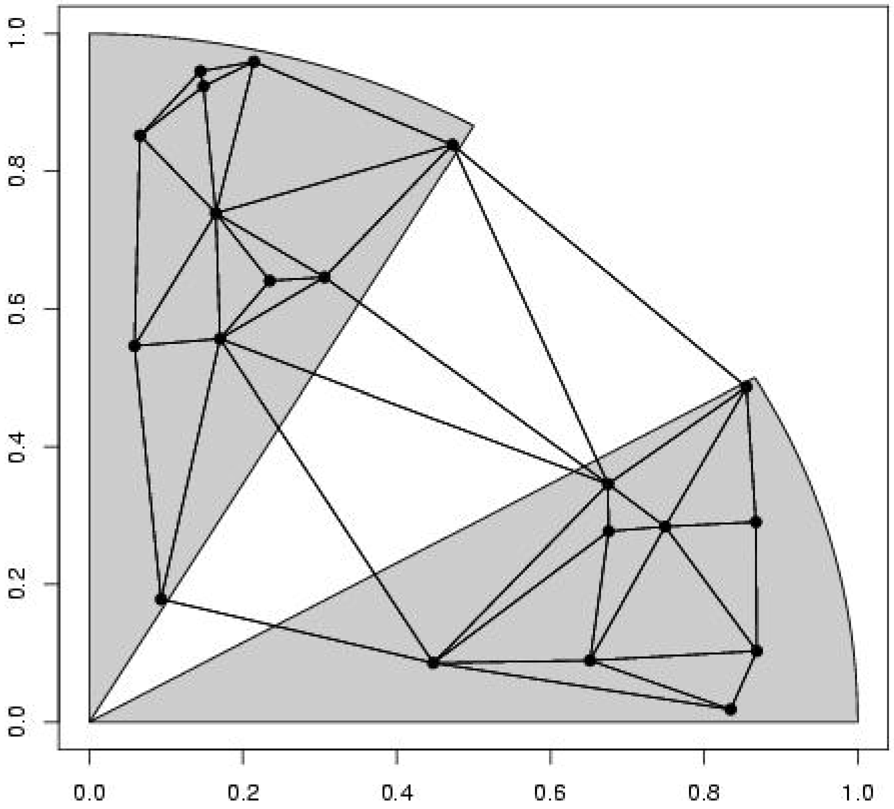

Figure 2: Setup for the simulations. The vectors are sampled uniformly from the gray regions. The angle of the white "void" region varies from 10° to 70°. The Delaunay triangularization is shown for a representative sample, demonstrating that the areas of the triangles which cross the void tend to be larger than those that do not.

3. Monte Carlo Experiments

Several moments of the distributions of the areas of triangles in the Delaunay triangularization of Poisson point clouds are known, so one might consider using these to define our critical value. However, the least squares fit introduces bias in the vector estimates, and so the estimated vectors tend not to have the same distribution as the true vectors. Thus, we rely on simulations to provide critical values. For a given n and *a priori* null distribution F, we repeat the following many times to obtain an estimate of the distribution of our test statistic under the null:

1. Sample n vectors $X = \{x_1, \ldots, x_n\}$ from F.

2. Sample m random dot product graphs using X.

3. Use the least squares algorithm to fit the vectors, resulting in $\widehat{X}_1, \ldots, \widehat{X}_m$.

4. Compute the value of the statistic on the fitted vectors.

Given a critical value for the test statistic (we use an alpha of 0.05 throughout), we perform 1000 Monte Carlo replicates of the experiment in which a void exists in the distribution. Figure 2 shows the experimental setup. Note that we could design a much more powerful test for this specific alternative by looking at angles between points. However, we won't take into account this specialized knowledge of the alternative, and instead use our statistic defined by the Delaunay triangles. We computer the power of the test against this alternative as a function of $\theta = \theta = \theta_2 - \theta_1$, the angle between the two lines defining the wedge-shaped void. The idea is that the void corresponds to vertices that were not observed, and are hence missing from our graph.

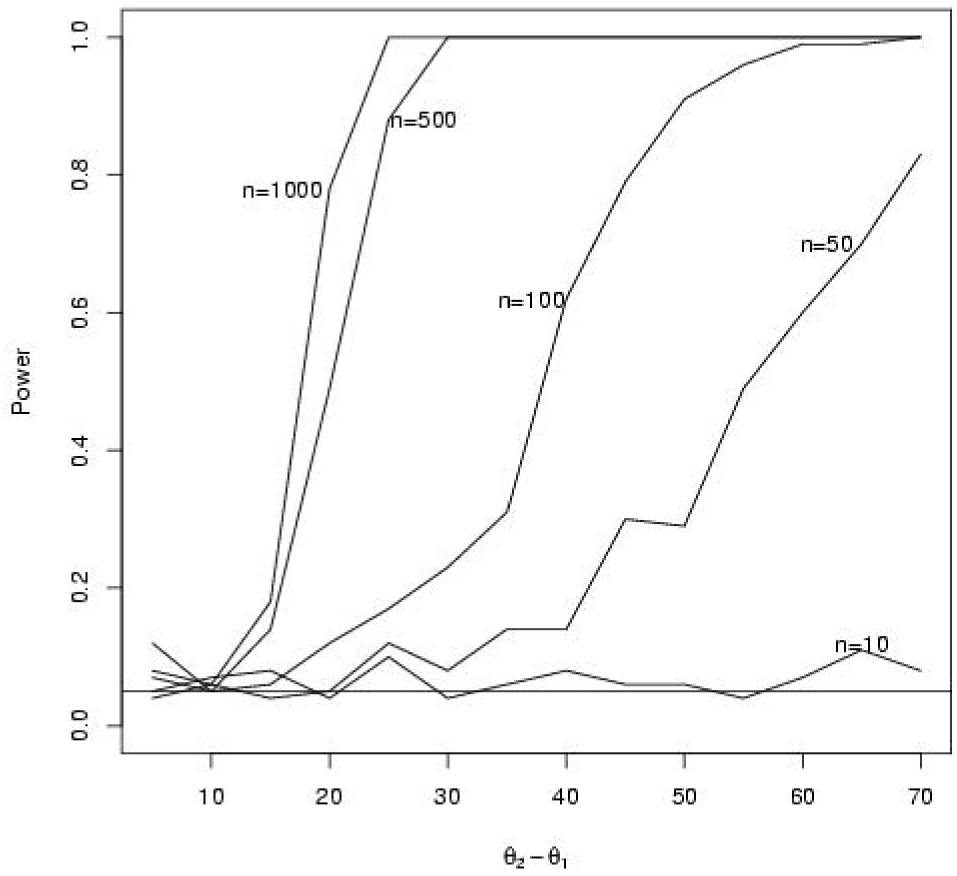

Figure 3: Power curves, as a function of void angle, for various values of n, the order of the observed graph.

Figure 3 depicts the power curves as a function of n, the number of vertices in the observed graph. Note that for $n = 10$ there is essentially no hope of detecting the void, and hence the fact that vertices are missing. However, as n increases, the power curves show that it quickly becomes possible to detect voids, and that for larger n the size of the detected void decreases. For $n = 1000$ we have maximal power for detecting a void of 25°.

The power curves give an indication of the amount of data needed to determine the existence of a "void" corresponding to missing nodes. We can get an estimate of the number of missing nodes by considering the size of the void, and using the *a priori* distribution. Thus we use the area to determine what proportion of the density is missing, and using the number of nodes observed, determine an estimate of the number of nodes missing.

This model also provides a natural way of associating properties to the missing nodes. If one has a way of inverting the "social space" in the sense of relating position in social space to position in some measured attributes, then the fact that the detected missing nodes fall in particular regions of social space provide information about what types of attributes they may have. In the absence of this inversion (or representation in attributes), one has the edge probabilities defined by the dot products, which can be used to guide further data collection. For example, if the problem were one of a sexually transmitted disease, with edges representing sexual partners, edge probabilities between the void and the observed nodes could suggest which actors are most likely to have relationships with the missing nodes, which in turn could be used to focus further data collection.

4. Conclusions

The methodology described in this work assumes two things: an *a priori* assumption of the distribution of the latent vectors in social space, and that missing nodes will be clustered in a region of social space. In some situations these might not be reasonable assumptions, and the work will not be applicable. Although this work is preliminary, and we have not yet applied it to actual data, we believe there are situations where these assumptions can be made, and the methodology can be used to infer missing nodes.

One of the most important arguments for an approach of this type is that in covert networks, in which certain actors have a vested interest in being unobserved, these actors tend to share attributes that make it reasonable to assume that they would be close in "social space". In an application where one may be able to collect data from many similar networks, such as those related to terrorist activity, criminal activity such as drug trafficking, or sexually transmitted diseases, the assumption that an *a priori* distribution exists and can be estimated may be reasonable. It may also be that in these and other applications, the further assumption that missing nodes have some relationships in social space may also be reasonable. Future work will involve investigating such applications.

References

[1] Peter D. Hoff, Adrien E. Raftery, and Mark S. Handcock. Latent space approaches to social network analysis. *JASA*, 97:1090–1098, 2002.

[2] David J. Marchette and Carey E. Priebe. Predicting unobserved links in incompletely observed networks. *Computational Statistics and Data Analysis*, 52:1373–1386, 2008.