Vertex Nomination via Seeded Graph Matching

Carey E. Priebe, Youngser Park, Heather Patsolic, Vince Lyzinski
Johns Hopkins University

July 29, 2016

We are given two graphs, $G = (V,E)$ and $G' = (V',E')$, with $V = \{x\} \cup S \cup W \cup J$ and $V' = \{x'\} \cup S' \cup W' \cup J'$. We denote by $x \in V$ the vertex of interest (VOI) in G, and $x' \in V'$ denotes its corresponding vertex in G'. (NB: We are considering just one VOI, for the nonce, for simplicity.) S and S' are known seeds between the two graphs, with $|S| = |S'| = s$, and the one-to-one correspondence $S \leftrightarrow S'$ between these s vertices in G and their corresponding s vertices in G' is known to us a priori. W and W' are the remaining shared vertices, with $|W| = |W'| = n$, and the one-to-one correspondence between these n vertices in G and their corresponding n vertices in G' is unknown to us. J and J' are the remaining unshared vertices, with $|J| = m$ and $|J'| = m'$, and there is no correspondence between these m vertices in G and m' vertices in G'. Thus $|V| = 1+s+n+m$ and $|V'| = 1+s+n+m'$. Note that we do not know which vertices are in W vs J in G, or which vertices are in W' vs J' in G', or even the values n, m, and m'; all we know is which vertex in G is the VOI x and the correspondence $S \leftrightarrow S'$ ... and we assume that the VOI's match, x', is somewhere in G' (for otherwise there is no point in carrying on and our time can be better spent reading Proust). The identification of the VOI's match x' is the goal of our exercise – specifically, we will nominate candidate vertices in G' ranked in order of our confidence that they are indeed the x' we seek. To reiterate: we assume in this development that the VOI x does indeed have its corresponding x'; if no such x' exists in G', our search will go forward mutatis mutandis, but is doomed to abject failure. (Gam zu l'tovah, eh?)

Let $U = \{x\} \cup S \cup W$ and $U' = \{x'\} \cup S' \cup W'$; $|U| = |U'| = 1+s+n$. Consider induced subgraphs $H = \Omega(U)$ in G and $H' = \Omega(U')$ in G'; Since U and U' are all the shared vertices – all the vertices with one-to-one correspondence between the graphs – our model assumes that H and H' have some formal relationship. Thus we consider $(H,H') \sim \rho\text{-}RDPG(X)$ for $(1+s+n) \times d$ latent position matrix X. That is, the ρ-*RDPG* model defines what we mean by "shared vertices."

In order to understand ρ-*RDPG*, we first recall the definition of an RDPG (Athreya et al. (2016)) For a graph G, we say $G \sim RDPG(X)$ if the following hold. Let $X_1, \ldots, X_n$ be independent random variables and define

$$X = [X_1, \ldots, X_n]^\top \in \mathbb{R}^{n \times d} \text{ and } P = XX^\top \in [0,1]^{n \times n}. \tag{1}$$

The X_i are the latent positions for the random graph. The matrix $A \in \{0,1\}^{n \times n}$, the adjacency

matrix for G, is a symmetric matrix with all 0's on the diagonal and so that for all $i < j$, conditioned on X_i, X_j, the $A_{i,j}$ are independent and

$$A_{i,j} \sim Bernoulli(X_i^\top X_j), \tag{2}$$

namely,

$$P(A|X) = \prod_{i<j} (X_i^\top X_j)^{A_{i,j}} (1 - X_i^\top X_j)^{1-A_{i,j}}. \tag{3}$$

To be clear, this means that edge presence between pairs of vertices is independent across vertex sets and for any pair of vertices, the probability of an edge between them is determined by their latent position vectors (in X).

Letting A and A' denote the adjacency matrices for H and H', respectively, H and H' are said to be generated from a ρ-$RDPG(X)$ if the following hold. First, $H, H' \sim RDPG(X)$. Furthermore, $A_{i,j}$ and $A'_{k,l}$ ($i < j$ and $k < l$) are independent whenever $i \neq k$ or $j \neq l$; otherwise, $A_{i,j}$ and $A'_{i,j}$ have correlation ρ. That is, edge presence between vertices i and j in graphs H and H' has correlation ρ, and otherwise edge presence is independent across the two graphs. Note that ρ-ER and ρ-SBM are special cases of the ρ-$RDPG(X)$ for specific types of matrices X.

Now that we have H and H', our model for G and G' adds the "junk" vertices J and J' to arrive at $(G, G') \sim \rho$-$RDPG(X, Y, Y')$ using $m \times d$ latent position matrix Y and $m' \times d$ latent position matrix Y'. In this case, $G \sim \rho$-$RDPG([X, Y])$ and $G' \sim \rho$-$RDPG([X, Y'])$, so that the induced subgraphs $H, H' \sim \rho$-$RDPG(X)$, and otherwise edge presence is independent across networks.

So $(G, G') \sim \rho$-$RDPG(X, Y, Y')$ returns two graphs: G on $1 + s + n + m$ vertices $\{x\} \cup S \cup W \cup J$ and G' on $1 + s + n + m'$ vertices $\{x'\} \cup S' \cup W' \cup J'$ (where the (observed) VOI x and its (unobserved) match x' are distinguished by the vertex nomination problem specification from among the $1 + n + m'$ matched non-seeds) as well as an observed one-to-one correspondence σ_S twixt S and S' and an unobserved one-to-one correspondence σ twixt $\{x\} \cup W$ and $\{x'\} \cup W'$. (Recall that there is no correspondence between J and J'.) The unknowns here are (1) n and m (we know $n + m$ when we observe the graph G and the seeds S and are given the VOI x) and which vertices in $W \cup J$ are which, (2) n and m' (we know $n + m'$ when we observe the graph G' and the seeds S' and assume the VOI's match x' is in G') and which vertices in $\{x'\} \cup W' \cup J'$ are which, and (3) the one-to-one correspondence twixt $\{x\} \cup W$ and $\{x'\} \cup W'$ – and in particular, twixt x and x'. Note that all of these unknowns are nuisance parameters except the parameter of interest: which vertex in $\{x'\} \cup W' \cup J'$ corresponds to the VOI x.

In the absence of junk vertices, we only have H and H', and the original seeded graph matching algorithm introduced by Fishkind et al. (2012), which minimizes the number of edge disagreements between the two graphs while fixing the bijection for the seeded vertices, can be employed, since the vertex sets U and U' have the same size. In the presence of J and J', however, we aren't guaranteed $|V| = |V'|$, so SGM can't be used directly. Therefore, we pad the smaller graph with "phantom" vertices so that the vertex sets are the same size. The output of SGM (with padding) outputs $\widehat{\sigma}_F$, a one-to-one correspondence between the padded vertex sets of G and G', which appropriately assigns $S \leftrightarrow S'$ and otherwise minimizes the number of edge disagreements between

the padded graphs while weighing a discrepency solely involving vertices in V and V' more heavily than a discrepency involving "phantom" vertices. The estimate for σ will be contained in $\widehat{\sigma}_F$, but is inseparable out-right, since it is uncertain which vertices in G belong to W and J and which vertices in G' belong to W' and J'. Rather than a hard one-to-one correspondence, since the output of SGM changes based on initialization, we run the algorithm several times and then create a matrix P so that $P[i,j]$ is the proportion of times vertex j in G' maps to vertex i in G. Note that we also get proportions involving the "phantom" vertices, but these vertices don't actually exist, so we ignore the rows (or columns) corresponding to these vertices.

To be fair, we are only interested in finding the vertex $x' \in V'$, so we only consider $P[x,:]$, the row in P corresponding to $x \in V$. We can then obtain a ranked nomination list for x by ordering these vertices in decreasing order of likelihood of being x', and our estimate for x' is the vertex in $W' \cup J'$ which was most often mapped to $x \in V$; that is, $\widehat{x}' = \arg\max_{v \in C'_x} P[x,v]$.

So $sgm(G, G', S \leftrightarrow S')$ solves our problem. But this SGMP might be too big, since the SGM algorithm we use is an inexact method which approximates σ in $O(n^3)$ time, where n is the number of vertices in the largest graph.

Instead, we proceed as follows.
Given $h \in \mathbb{N}$ we see that $S_x = S \cap N_h(x)$ are the seeds within an h-path, i.e. in the h-neighborhood, of VOI x in G. S'_x are the corresponding seeds in G', and we write $|S_x| = s_x = |S'_x|$. (Notationally, we say that $h \to \infty$ yields $N_h(x)$ to be the vertices in the connected component of G in which x lives, and we say that $h = \infty$ yields $N_h(x)$ to be the the entire vertex set V of G.) If $s_x = 0$ ($S_x = \emptyset$), then doom ensues, as we have no seeds with which to proceed; thus we assume $s_x > 0$ henceforth, and otherwise we must increase h. (If there are no seeds in the connected component of x, we give up and go back to Proust.) Now, for $\ell \geq h$, we define $G_x = \Omega(N_\ell[S_x])$ and $G'_x = \Omega(N_\ell[S'_x])$. We say that $C'_x = N_\ell(S'_x)$ are the *candidates* for the match x' to the VOI x – observe that x is an h-neighbor of S_x, and C'_x are the ℓ-neighbors of S'_x. These candidates are the vertices in G'_x that are potential matches for x. (If $x \notin C'_x$, then we will not succeed ... but so be it; we will still nominate.) To check the reader's math: G_x is connected, and x and S_x are vertices therein; G'_x is not necessarily connected, and S'_x are vertices therein ... and we hope x' is in there, too. In any event, we now have a vertex nomination problem on a smaller SGMP.
...
So now we do $sgm(G_x, G'_x, S_x \leftrightarrow S'_x)$ – a smaller SGMP. (NB: the original SGMP is this with $h = \infty$.)

Simulations, Illustrative Experiments, and Real Application

Here we describe our approach to both the simulations and the illustrative experiments, which allows the same code to be used to address a real problem in anger (when we don't know any truth except the VOI x and some seeds $S \leftrightarrow S'$).
If it's a simulation or illustrative experiment: (1) generate G and G' with some shared vertices and some unshared vertices, or start with real data G and G' with a collection of known shared vertices

and some unknown or unshared vertices, and (2) randomly pick VOI x and some number of seeds S from amongst the shared vertices; then embark on our procedure described above. If it's a real problem, with given VOI x and some seeds $S \leftrightarrow S'$, then embark immediately on our procedure described above.

References

Athreya, A., Priebe, C. E., Tang, M., Lyzinski, V., Marchette, D. J., and Sussman, D. L. (2016). A limit theorem for scaled eigenvectors of random dot product graphs. *Sankhya: The Indian Journal of Statistics*, 78-A(1):1–18.

Fishkind, D., Adali, S., and Priebe, C. (2012). Seeded graph matching. *arXiv:1209.0367*.