Estimating Latent Positions and Vertex Classification for Random Dot Product Graphs

Joint Statistical Meetings 2012
Stochastic Blockmodels: New Uses and Directions

Daniel Sussman, Minh Tang, Donniell Fishkind, Joshua Vogelstein, Carey Priebe

Johns Hopkins University, Applied Math and Statistics Department

July 29, 2012
Outline

1. Latent Position Models
   - Random Dot Product Graphs

2. Latent Position Estimation

3. Pattern Recognition

4. Empirical Results
Outline

1 Latent Position Models
   - Random Dot Product Graphs

2 Latent Position Estimation

3 Pattern Recognition

4 Empirical Results
Beyond the Stochastic Blockmodel

For \( n \in \mathbb{N} \), \# vertices, and \( b \in \mathbb{N} \), \# blocks, define

- \( \tau \in [b]^n \) — block membership vector, \( \tau_i \sim iid \text{ Cat}([b]) \)
- \( B \in [0, 1]^{b \times b} \) — edge probability matrix

\[
P[i \sim j] = P[A_{ij} = 1] = B_{\tau_i; \tau_j}.
\]
Beyond the Stochastic Blockmodel

**Stochastic Blockmodel**
- For $n \in \mathbb{N}$, \# vertices, and $b \in \mathbb{N}$, \# blocks, define
  - $\tau \in [b]^n$ — block membership vector, $\tau_i \overset{iid}{\sim} \text{Cat}([b])$
  - $B \in [0, 1]^{b \times b}$ — edge probability matrix
  - $\mathbb{P}[i \sim j] = \mathbb{P}[A_{ij} = 1] = B_{\tau_i, \tau_j}$

**First Start: Needs Extension**
- Only $b < \infty$ “kinds” of vertices.
- Two vertices of the same “kind” are exactly the same; ie, stochastically equivalently.
- Variation within blocks needed to make model more realistic.
Each vertex associated with $Z_i^{iid} \sim \text{Unif}(0, 1)$.

$\mathbb{P}[i \sim j] = g(Z_i, Z_j); \text{ for } g : [0, 1]^2 \mapsto [0, 1]$.

$g$ is the *link function*.
Each vertex associated with $Z_i \overset{iid}{\sim} \text{Unif}(0, 1)$.

- $\mathbb{P}[i \sim j] = g(Z_i, Z_j)$; for $g : [0, 1]^2 \mapsto [0, 1]$.
- $g$ is the link function.

**Note:** Link function $g$ for exch. random graphs is thought of as an operator $G : L^2[0, 1] \mapsto L^2[0, 1]$.

$$Gf(x) = \int_0^1 g(x, y)f(y)dy$$

Dimension of image of $G$ is rank of the link function.
Example Exchangable Graph

Figure: Link function for random threshold graphs, \( g(x, y) = \mathbb{I}\{x + y > 1\} \)
Graph Parameters

- Each vertex $i \in [n]$ associated with vector $Z_i \overset{iid}{\sim} F$ for some distribution $F$ on $\Omega \subset B(0, 1) \subset \mathbb{R}^d$
  - $\langle x, y \rangle \in [0, 1]$ for all $x, y \in \Omega$
  - No further assumptions on $F$.

- Edge probabilities: $\mathbb{P}[i \sim j] = \mathbb{P}[A_{ij} = 1] = \langle Z_i, Z_j \rangle$

Advantage over just looking at general exchangeable graphs is added structure of $\mathbb{R}^d$ and the inner product.
Why RDPG?

Random dot product graphs are the “finite-dimensional” exchangeable random graphs.
Why RDPG?

Random dot product graphs are the “finite-dimensional” exchangeable random graphs.

Proposition

An exchangeable random graph is rank \( d < \infty \) and p.s.d.
if and only if
the random graph is distributed according to an RDPG model for some distribution \( F \) of latent positions on \( \Omega \subset \mathcal{B}(0, 1) \subset \mathbb{R}^d \)

Recall: \( \mathcal{G} : L^2[0, 1] \mapsto L^2[0, 1] \).

\[
\mathcal{G} f(x) = \int_0^1 g(x, y) f(y) dy
\]

Dimension of image of \( \mathcal{G} \) is rank of the link function.
RDPG Example

\[ \mathbf{X} = [Z_1, \ldots, Z_n]^\top \in \mathbb{R}^{n \times d} \]

Original “Data” Matrix

\[ \mathbf{P} = \mathbf{X} \mathbf{X}^\top \in [0, 1]^{n \times n} \]

Probability Matrix

- Both are rank($d$) and eigenvalues are $O(n)$
- $Z_i$ are distributed according to mixture of Dirichlets with $d = 2$. 
\[ P = \left[ \langle Z_i, Z_j \rangle \right]_{i,j=1}^{n} \in [0, 1]^{n \times n} \]

- **Probability Matrix**

\[ A \in \{0, 1\}^{n \times n} \]

- **Adjacency Matrix**

- **Key:** \( A \) “looks like” \( P \)! At least on coarse scales.
Outline

1. Latent Position Models
   - Random Dot Product Graphs

2. Latent Position Estimation

3. Pattern Recognition

4. Empirical Results
Goal: Estimate $X$ from $A$

- Let $P = U_P S_P U_P^T$ be a truncated eigen-decomposition of $P$.
- $U_P \in \mathbb{R}^{n \times d}, S_P \in \mathbb{R}^{d \times d}$
Goal: Estimate $X$ from $A$

- Let $P = U_P S_P U_P^T$ be a truncated eigen-decomposition of $P$.
  - $U_P \in \mathbb{R}^{n \times d}$, $S_P \in \mathbb{R}^{d \times d}$
- $X$ is a rotation of $U_P S_P^{1/2}$
  - Note: $X$ is identifiable only up to rotation.
Goal: Estimate $X$ from $A$

- Let $P = U_P S_P U_P^T$ be a truncated eigen-decomposition of $P$.
  - $U_P \in \mathbb{R}^{n \times d}$, $S_P \in \mathbb{R}^{d \times d}$
- $X$ is a *rotation* of $U_P S_P^{1/2}$
- **Note**: $X$ is identifiable only up to rotation.
- Let $U_A S_A U_A^T$ be a truncated eigen-decomposition of $A$.
- Since $A$ looks like $P$, hopefully $U_A S_A^{1/2}$ and $U_P S_P^{1/2}$ are close.
Main Results

**Theorem ([Sussman et al., 2012b])**

> With probability greater than $1 - \frac{2(d^2+1)}{n^2}$, there exists an orthogonal matrix $W \in \mathbb{R}^{d \times d}$ such that

$$\|U_A S_A^{1/2} W - X\|_F \leq (d + 1) \sqrt{\frac{3 \log n}{\delta^3}}.$$  \hspace{1cm} (1)

- $0 < \delta < 1$ is the eigengap of $\mathbb{E}[ZZ^\top]$, the second moment matrix for $Z \sim F$
- Can also bound the individual latent position estimates rather then the above global bound.
Proof Sketch

Proposition ([Sussman et al., 2012a] (see also [Rohe et al., 2011]))

For $\mathbf{A}$ and $\mathbf{P}$ as above, it holds with probability greater than $1 - \frac{2}{n^2}$ that

$$\|\mathbf{A}^2 - \mathbf{P}^2\|_F \leq \sqrt{3n^3 \log n}.$$  \hfill (2)

Proposition

For $i \leq d$, it holds with probability greater than $1 - \frac{2d^2}{n^2}$ that

$$|\lambda_i(\mathbf{P}) - n\lambda_i(\mathbb{E}[\mathbf{ZZ}^\top])| \leq 2d^2 \sqrt{n \log n}.$$  \hfill (3)

Now use Davis-Kahan theorem.
Outline

1. Latent Position Models
   - Random Dot Product Graphs

2. Latent Position Estimation

3. Pattern Recognition

4. Empirical Results
Classification

Classical Setting

- Let \((Z, Y), (Z_1, Y_1), \ldots, (Z_n, Y_n) \sim F_{Z,Y}\)
- Observe \(D = (Z_1, Y_1), \ldots, (Z_n, Y_n)\) and \(Z\).
- Learn function \(h(\cdot; D)\) to minimize \(P[h(Z; D) \neq Y]\).
Classification

Classical Setting

- Let \((Z, Y), (Z_1, Y_1), \ldots, (Z_n, Y_n) \sim F_{Z,Y}\)
- Observe \(D = (Z_1, Y_1), \ldots, (Z_n, Y_n)\) and \(Z\).
- Learn function \(h(\cdot; D)\) to minimize \(\mathbb{P}[h(Z; D) \neq Y]\).

Theorem (Stone 1977)

Let \(h_n\) be the k-nearest-neighbor rule trained on \(\{(Z_i, Y_i)\}_{i=1}^n\). If \(k \to \infty\) and \(k/n \to 0\) as \(n \to \infty\), then

\[
\mathbb{E}[\mathbb{P}[h_n(Z) \neq Y|\{(Z_i, Y_i)\}_{i=1}^n]] = \mathbb{E}[L_n] \to L^*_Z \quad (4)
\]

for all distributions \(F_{Z,Y}\).
Goal: Classify vertex $n + 1$ based on observing class labels for vertices $1, \ldots, n$.

Do not observe $\{Z_i\}$.
Only $\{Y_i\}$ and the adjacency matrix $A$.

Distribution of $A$ is determined by $Z, Z_1, \ldots, Z_n$.

(Training and test data are inherently tangled together)

Can we still have universal consistency for vertex classification?
Theorem ([Sussman et al., 2012b])

Let $h_n$ be the k-nearest-neighbor rule trained on $\{(\hat{Z}_i, Y_i)\}_{i=1}^n$. If $k \to \infty$ and $k/n \to 0$ as $n \to \infty$, then

$$\mathbb{E}[\mathbb{P}[h_n(\hat{Z}) \neq Y|A, \{Y_i\}_{i=1}^n]] = \mathbb{E}[L_n] \to L^*$$

for all distributions $F_{Z,Y}$ on $\Omega$.

Proof.

Follows mutatis mutandis from [Devroye et al., 1996]. Need that $\{(\hat{Z}_i, Z_i)\}$ are exchangeable plus our bound from above.
Outline

1 Latent Position Models
   • Random Dot Product Graphs

2 Latent Position Estimation

3 Pattern Recognition

4 Empirical Results
Each vertex in the graph corresponds to a Wikipedia page and the edges correspond to the presence of a hyperlink between two pages (in either direction).

Every article within two hyperlinks of the article “Algebraic Geometry” was included as a node in the graph.

Each vertex manually labeled as one of the following:

<table>
<thead>
<tr>
<th>Person</th>
<th>Location</th>
<th>Date</th>
<th>Category</th>
<th>Math</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>372</td>
<td>270</td>
<td>191</td>
<td>119</td>
<td>430</td>
<td>1382</td>
</tr>
</tbody>
</table>
Classification error by \# Vertices

\( k = 9, d = 10 \)

\[ n \text{ — number of vertices in subgraph} \]

\( n \) — number of vertices in subgraph

\text{Chance} \approx 0.689
Classification error for varying $k$ and $d$

Chance $\approx 0.689$
Conclusion

- Random dot product graphs are low dimensional exchangeable random graphs
- Eigen-decomposition of adjacency matrix gives consistent estimation of latent positions if latent positions are i.i.d.
- $k$-nearest-neighbors is universally consistent for random dot product graphs.

Future Directions

- How does this work extend to $\infty$-dimensional exchangeable graphs and other kernels?
- How can we do local versions?
References

Springer Verlag.

Spectral clustering and the high-dimensional stochastic blockmodel.  

A consistent adjacency spectral embedding for stochastic blockmodel graphs (in press).  

Universally consistent latent position estimation and vertex classification for random dot product graphs.  
Arxiv pre-print.