The Latent Structure Block Model

John Koo
Department of YYY, University of XXX

September 30, 2023

Abstract

The text of your abstract. 200 or fewer words.

Keywords: block models, community detection, coordinate descent, latent structure models, manifold clustering, random dot product graph

1 Introduction

We define a *Bernoulli graph* as a random graph model for which edge probabilities are contained in an edge probability matrix $P \in [0,1]^{n \times n}$, and an edge occurs between vertices i and j with probability P_{ij}. Common random graph models then impose structure on P, based on various assumptions about the way in which the data are generated, or to allow P to be estimated. One example is the Erdös-Rényi model, in which all edge probabilities are fixed, i.e., $P_{ij} = p$.

One common analysis task for graph and network data is community detection, which assumes that each vertex of a graph has a hidden community label. The goal of the analysis is then to estimate these labels upon observing a graph. One approach is to define a random graph model that includes community labels. First, each vertex is assigned a label $z_1, ..., z_n \in \{1, 2, ..., K\}$ where $K \ll n$. Then the edge probabilities are restricted such that they depend on the labels of the vertices, and possibly some other parameters specific to each vertex, i.e., $P_{ij} = g(z_i, z_j, \phi_i, \phi_j)$, where ϕ_i is a set of parameters corresponding to vertex i. We call such models *block models*. The simplest and best known block model is the stochastic block model (SBM), which sets a fixed edge probability for each pair of communities, i.e., $P_{ij} = \theta_{z_i, z_j}$. The degree-corrected block model (DCBM) assigns an additional parameter ω_i to each vertex by which edge probabilities are scaled, i.e., $P_{ij} = \omega_i \omega_j \theta_{z_i, z_j}$. The popularity adjusted block model (PABM) assigns K parameters to each vertex $\lambda_{i1}, \lambda_{i2}, ..., \lambda_{iK}$ that describe that vertex's affinity toward each community; the edge probability between vertices i and j is then defined as the product of vertex i's affinity toward vertex j's community and vertex j's affinity toward vertex i's community, i.e., $P_{ij} = \lambda_{i z_j} \lambda_{j z_i}$. The three block model types, as well as the Erdös-Rényi model, impose structure on P, including on the rank of P. P has rank 1 for the Erdös-Rényi model, rank K for the SBM and DCBM, and rank K^2 for the PABM. This provides the intuition behind another family of Bernoulli graphs called the *random dot product graph* (RDPG) and *generalized random dot product graph* (GRDPG). In the RDPG, each vertex has a corresponding latent vector in d-dimensional Euclidean space, where d is the rank of P and P is positive semidefinite. Then the edge probability between each pair of vertices is defined as the inner product between the corresponding latent vectors, i.e., $P_{ij} = x_i^\top x_j$. If the latent vectors are collected in a data matrix $X = \begin{bmatrix} x_1 \mid \cdots \mid x_n \end{bmatrix}^\top$, then the edge probability matrix for the RDPG is $P = XX^\top$. Similarly, the edge probability between each pair of vertices for the GRDPG is defined as the indefinite inner product between the corresponding latent vectors, i.e., $P_{ij} = x_i^\top I_{p,q} x_j$, where $I_{p,q} = \text{blockdiag}(I_p, -I_q)$ and $p + q = d$. Then the edge probability matrix for the GRDPG is $P = XI_{p,q}X^\top$. This allows for a model similar to the RDPG for non-positive semidefinite P. While the RDPG and GRDPG do not necessarily have community structure, it has been shown that block models are specific cases of the RDPG or GRDPG in which latent vectors are organized by community. This includes the SBM, in which communities correspond to point masses, DCBM, in which communities correspond to line segments, and PABM, in which communities correspond to orthogonal subspaces. In this work, we extend this idea to communities organized into more general latent structures. In particular, we assume that each community corresponds to a manifold in the latent space.

2 Generalized Random Dot Product Graphs with Community Structure

All Bernoulli graphs are generalized random dot product graphs. For a specific example, consider the two-community DCBM and its GRDPG latent space representation. In the DCBM, the edge probability between each pair of vertices is the product of the baseline community-wise edge probability and the degree correction factors of the vertices, or $P_{ij} = \theta_{z_i,z_j}\omega_i\omega_j$.

To motivate this, consider a generalization of the Erdös-Rényi model. Recall that when viewed as an RDPG, the latent space of an Erdös-Rényi model consists of one point in Euclidean space. In the following example, instead of fixing the edge probability, it is sampled from a distribution in such a way that when viewed as an RDPG, the latent space consists of a curve.

Example 1 (Hierarchical Erdös-Rényi model)**.** In the Erdös-Rényi model, the edge probability matrix has a fixed value $[P_{ij}] \equiv p \in [0,1]$.

Suppose that we have a random dot product graph in which the latent space is $\mathbb{R}^2$ and latent vectors are drawn uniformly from the quarter circle defined by $g(t) = \begin{bmatrix} \cos(\frac{\pi}{2}t) & \sin(\frac{\pi}{2}t) \end{bmatrix}^\top$, $0 \leq t \leq 1$. Then it can be shown that in this model, instead of a fixed $P_{ij} = p$, the edge probabilities are distributed with density $f(p) = \frac{2}{\pi-2}\Big(\frac{1}{\sqrt{1-p^2}} - 1\Big)$.

By changing the latent structure from a point mass to a curve, we are able to come up with more flexible Bernoulli graph models in which edge probabilities follow more general probability distributions. Community structure then can be added by sampling latent vectors from multiple curves. Then the adjacency spectral embedding of the resulting graph allows us to recover that community structure. This is illustrated in the following example.

Example 2. Define two one-dimensional manifolds in $\mathbb{R}^2$ by $f_1(t) = \begin{bmatrix} \cos(\frac{\pi}{3}t) & \sin(\frac{\pi}{3}t) \end{bmatrix}^\top$ and $f_2(t) = \begin{bmatrix} 1 - \cos(\frac{\pi}{3}t) & 1 - \sin(\frac{\pi}{3}t) \end{bmatrix}^\top$. Draw $t_1, ..., t_n \overset{\text{iid}}{\sim} \text{Uniform}(0, 1)$ and $z_1, ..., z_n \overset{\text{iid}}{\sim} \text{Multinomial}(\frac{1}{2}, \frac{1}{2})$, and compute latent vectors $x_i = f_{z_i}(t_i)$, which are collected in data matrix $X = \begin{bmatrix} x_1 & \dots & x_n \end{bmatrix}^\top$. Finally, let $A \sim \text{RDPG}(X)$. Fig. 1 shows the latent configuration drawn from this latent distribution, a random dot product graph drawn from the latent configuration, and the adjacency spectral embedding of the graph. Although the community structure is not obvious from the graph, the embedding shows a clear separation between the two communities.

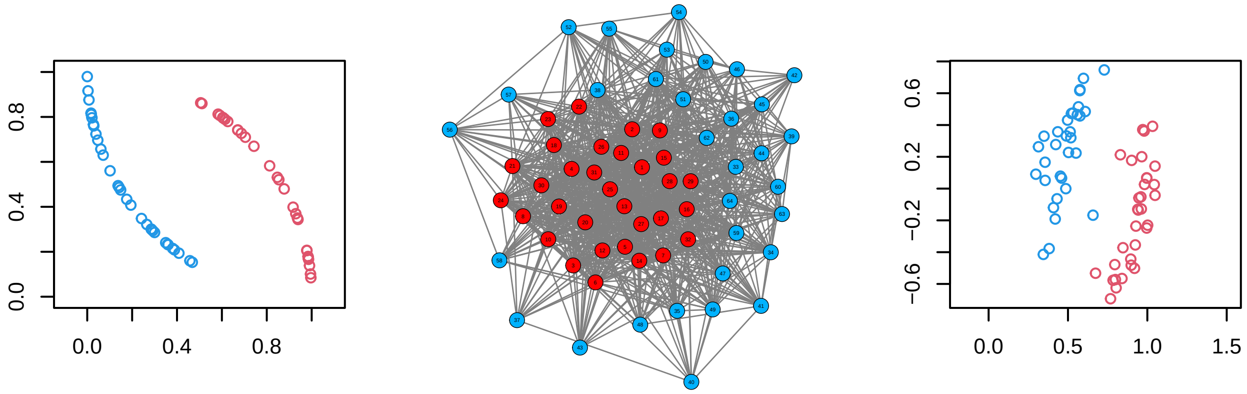


Figure 1: Manifold block model described in Example 1. The latent configuration is on the left, a random dot product graph drawn from the latent configuration is on the middle, and the ASE is on the right.

We now formally define the manifold block model.

Definition 1 (Manifold block model)**.** Let $p, q \geq 0$, $d = p + q \geq 1$, $1 \leq r < d$, $K \geq 2$, and $n \geq 1$ be integers. Define manifolds $\mathcal{M}_1, ..., \mathcal{M}_K \subset \mathcal{X}$ for $\mathcal{X} = \{x, y \in \mathbb{R}^d : x^\top I_{p,q} y \in [0, 1]\}$ each by continuous function $g_k : [0, 1] \to \mathcal{X}$, and probability distributions $F_1, ..., F_K$ each with support $[0, 1]^r$. Then the following mixture model is a manifold block model:

1. Draw labels $z_1, ..., z_n \overset{\text{iid}}{\sim} \text{Multinomial}(\alpha_1, ..., \alpha_K)$.

2. Draw latent vectors by first drawing each $t_i \overset{\text{ind}}{\sim} F_{z_i}$ and then compute each $x_i = g_{z_i}(t_i)$.

3. Let $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\top$, and draw $A \sim \text{RDPG}(X; \rho_n)$ or $A \sim \text{GRDPG}_{p,q}(X; \rho_n)$.

3 Methods

We provide two approaches to community detection for the manifold block model. First, we consider the case in which communities correspond to manifolds in the latent space that do not intersect and are separated by some finite distance. In this scenario, we use the convergence of the ASE to show that single linkage clustering on the latent space produces a clustering such that the total number of misclustered vertices goes to zero, with high probability.

Next, we consider the case in which communities correspond to one-dimensional manifolds in the latent space and may or may not intersect. In this scenario, we propose an alternating coordinate descent algorithm that alternates between estimating the structure of the manifolds and the community labels, which we call K-curves clustering. We again use the convergence of the ASE to show that under certain conditions, K-curves clustering produces a clustering such that the proportion of misclustered vertices goes to zero, with high probability.

3.1 Nonintersecting Manifolds

In this section, we consider the following scenario: Suppose that each community is represented by a closed manifold $\mathcal{M}_k$, $k \in \{1, ..., K\}$ in the latent space of a RDPG or GDRPG. Define $\delta = \min_{k \neq \ell} \min_{x \in \mathcal{M}_k, y \in \mathcal{M}_\ell} \|x - y\|$, the minimum distance between two manifolds. We assume that $\delta > 0$, i.e., the manifolds do not intersect.

In the noiseless setting, if the subsample on each manifold is sufficiently dense, it is possible to construct for each manifold an η_k-neighborhood graph for each manifold for some $\eta_k > 0$ such that the graph is connected. Then if $\max_k \eta_k = \eta < \delta$, an η-neighborhood graph for the entire sample will consist of K disconnected subgraphs that map onto each manifold. Equivalently, we can apply single-linkage clustering. The remainder of this section explores under which conditions these criteria are met for the latent configuration, in which latent vectors lie exactly on manifolds, as well as the ASE, which introduces noise.

```
Algorithm 1: ASE clustering for nonintersecting communities.
  Data: Adjacency matrix $A$, number of communities $K$, embedding dimensions $p$
        and $q$.
  Result: Community assignments $z_1, ..., z_n \in \{1, ..., K\}$.
1 Compute $\hat{X}$, the ASE of $A$ using the $p$ most positive and $q$ most negative
    eigenvalues and their corresponding eigenvectors.
2 Apply single linkage clustering with $K$ communities on $\hat{X}$.
```

Let F_k be a probability distribution with support $\mathcal{M}_k$. Then we define a mixture model as follows:

1. Draw labels $z_1, ..., z_n \overset{\text{iid}}{\sim} \text{Multinomial}(\alpha_1, ..., \alpha_K)$.
2. Draw latent vectors each as $x_i \overset{\text{ind}}{\sim} F_{z_i}$ for distributions $F_1, ..., F_K$ with respective supports $\mathcal{M}_1, ..., \mathcal{M}_K$.
3. Let $X = \begin{bmatrix} x_1 & \dots & x_n \end{bmatrix}^\top$, and draw $A \sim \text{RDPG}(X; \rho_n)$ or $A \sim \text{GRDPG}_{p,q}(X; \rho_n)$.

Note that here, we redefine the model to ignore $g_1, ..., g_K$, the parameterizations of each manifold. Instead, we sample points directly on the manifolds themselves. We will return to the parameterizations in Section 3.2.

Theorem 1 (Community detection for the GRDPG for which the communities come from nonintersecting manifolds)**.** *Let $x_1, ..., x_n$ be points sampled on K compact, connected manifolds $\mathcal{M}_1, ..., \mathcal{M}_K \subset \mathbb{R}^d$ each with probability measures $F_1, ..., F_K$, and the manifolds are separated by distance at least $\delta = \min_{k \neq \ell} \min_{x_i \in \mathcal{M}_k, x_j \in \mathcal{M}_\ell} \|x_i - x_j\| > 0$. Let $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\top$ and $A \sim \mathrm{GRDPG}_{p,q}(X; \rho_n)$ for some $p, q \in \mathbb{N}_0$ such that $p + q = d$ and sparsity parameter ρ_n that satisfies $n\rho_n = \omega(\log^c n)$ for some $c > 1$. Define $A_n(\eta)$ as the event that an η-neighborhood graph constructed from the ASE of A consists of exactly K disconnected subgraphs that map exactly to each manifold. Then for some $C > 0$ and any $\eta \in (0, C\delta)$, $\lim_{n \to \infty} P(A_n(\eta)) = 1$.*

If the manifolds are one-dimensional, then a more precise rate of convergence can be derived.

3.2 Intersecting Manifolds

In this section, we again consider the setting for the RDPG or GRDPG in which each community lies on a manifold in the latent space. However, this time, we do not assume that the manifolds are nonintersecting. We also restrict this case to one-dimensional manifolds which are each described by $g_k : [0, 1] \to \mathcal{X}$. Then we define a mixture model as follows:

1. Draw $t_1, ..., t_n \overset{\text{iid}}{\sim} F$ for probability distribution F with support $[0, 1]$.
2. Draw $z_1, ..., z_n \overset{\text{iid}}{\sim} \text{Multinomial}(\alpha_1, ..., \alpha_K)$, the community labels.
3. Let each $x_i = g_{z_i}(t_i)$ be the latent vector for vertex v_i, and collect the latent vectors into matrix $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\top$.
4. Draw $A \sim \text{RDPG}(X)$ or $A \sim \text{GRDPG}_{p,q}(X)$.

rves clustering.} \end{algorithm}

Algorithm 2: K-curves clustering.

```
Data: Adjacency matrix $A$, number of communities $K$, embedding dimensions $p$,
      $q$, stopping criterion $\epsilon$
Result: Community assignments $1, ..., K$, curves $g_1, ..., g_K$
Compute $X$, the ASE of $A$ using the $p$ most positive and $q$ most negative
  eigenvalues and their corresponding eigenvectors.
Initialize community labels $z_1, ..., z_n$.
repeat
    for $k = 1, ..., K$ do
        Define $X_k$ as the rows of $X$ for which $z_i = k$.
        Fit curve $g_k$ and positions $t_{k_i}$ to $X_k$ by minimizing $\sum_{k_i} \|x_{k_i} - g_k(t_{k_i})\|^2$.
    end
    for $k = 1, ..., K$ do
        Assign $z_i \leftarrow \arg\min_\ell \|x_i - g_\ell(t_i)\|^2$.
    end
until the change in $\sum_k \sum_{i \in C_k} \|x_i - g_k(t_i)\|^2$ is less than $\epsilon$
```

K-curves clustering assumes that the functional form of g_k is known. The choice of g_k affects the difficulty of the algorithm. As a balance between flexibility and ease of estimation, we consider the case where each g_k is a Bezier polynomial of degree R with coefficients p_k. Then we have $g_k(t) = g(t; p_k) = \sum_{r=0}^R p_k^{(r)} \binom{R}{r} (1-t)^{R-r} t^r$.

Given $\{t_i\}$ and $\{z_i\}$, it is straightforward to obtain $\hat{p}_k = \arg\min_p \sum_{k_i} \|x_{k_i} - g_k(t_{k_i}; p)\|^2$

$$\hat{p}_k = (T_k^\top T_k)^{-1} T_k^\top X_k,$$

where T_k is an $n_k \times (R+1)$ matrix with rows $\begin{bmatrix} (1-t_{k_i})^R & (1-t_{k_i})^{R-1} t_{k_i} & \cdots & (1-t_{k_i}) t_{k_i}^{R-1} & t_{k_i}^R \end{bmatrix}$.

Estimation of $\{t_i\}$ given $\{z_i\}$ and $\{p_k\}$ is more difficult. Each t_i can be estimated separately:

$$\hat{t}_i = \arg\min_t \|x_i - g(t; p_{z_i})\|^2. \tag{1}$$

This is equivalent to solving $0 = (x_i - g(t; p_{z_i}))^\top(\dot{g}(t; p_{z_i}))$. Setting $c^{(s)} = \sum_{r=0}^{s}(-1)^{s-r}\binom{R}{r}p_{z_i}^{(r)}$ for $s \neq 0$ and $c^{(0)} = p_{z_i}^{(0)} - x_i$, let $c = \begin{bmatrix} c^{(0)} & \dots & c^{(R)} \end{bmatrix}^\top$. Then solving Eq. 1 is equivalent to finding the real roots of a polynomial with coefficients that are the sums of the reverse diagonals of $CD^\top$, where $C_{ij} = c_{ij}(-1)^i\binom{R}{i}$ and $D_{ij} = c_{i-1,j}(-1)^{i-1}\binom{R-1}{i-1}$.

Algorithm 3: Semi-supervised K-curves clustering.

```
Data: Adjacency matrix $A$, number of communities $K$, embedding dimensions $p$,
      $q$, stopping criterion $\epsilon$, $m_k \leq n_k$ known community assignments for each
      community
Result: Community assignments $1, ..., K$, curves $g_1, ..., g_K$
Compute $X$, the ASE of $A$ using the $p$ most positive and $q$ most negative
  eigenvalues and their corresponding eigenvectors.
Fit curves $g_1, ..., g_K$ using each of the $m_1, ..., m_K$ points with known community
  labels by minimizing $\sum_{j=1}^{m_i} \|x_j - g_k(t_j)\|^2$.
Assign labels $z_1, ..., z_n$ to each $x_1, ..., x_n$ by minimizing $\|x_i - g_k(t_i)\|^2$ for $k$, holding
  the initial known labels constant.
repeat
    for $k = 1, ..., K$ do
        Define $X_k$ as the rows of $X$ for which $z_i = k$.
        Fit curve $g_k$ and positions $t_{k_i}$ to $X_k$ by minimizing $\sum_{k_i} \|x_{k_i} - g_k(t_{k_i})\|^2$.
    end
    for $k = 1, ..., K$ do
        Assign $z_i \leftarrow \arg\min_\ell \|x_i - g_\ell(t_i)\|^2$, holding the known initial labels
          constant.
    end
until the change in $\sum_k \sum_{i \in C_k} \|x_i - g_k(t_i)\|^2$ is less than $\epsilon$
```

upervised K-curves clustering.} \end{algorithm}

Theorem 2. *Let each $g(\cdot; p_k)$ be a nonintersecting Bezier polynomial of order R, and a GRDPG is drawn from vectors that lie on the curves. Suppose we observe the true labels of m_k vertices from each community, and each $m_k > R + 1$. Suppose further that latent* *vectors $x_j = g(t_i; p_{z_j})$ that correspond to vertices with observed labels are such that Then as $n \to \infty$, the proportion of misclustered vertices from K-curves clustering approaches* 0 *with probability* 1.

4 Examples

Example 3. Here, $K = 2$ with $g_1(t) = \begin{bmatrix} t^2 & 2t(1-t) \end{bmatrix}^\top$ and $g_2(t) = \begin{bmatrix} 2t(1-t) & (1-t)^2 \end{bmatrix}^\top$. We draw $n_1 = n_2 = 2^8$ points uniformly from each curve.

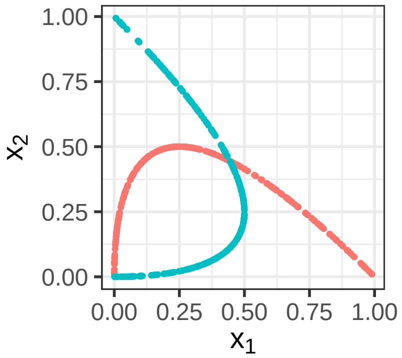


Figure 2: Latent positions, labeled by curve/community.

We draw $A \sim \text{RDPG}(X)$ and obtain the following ASE:

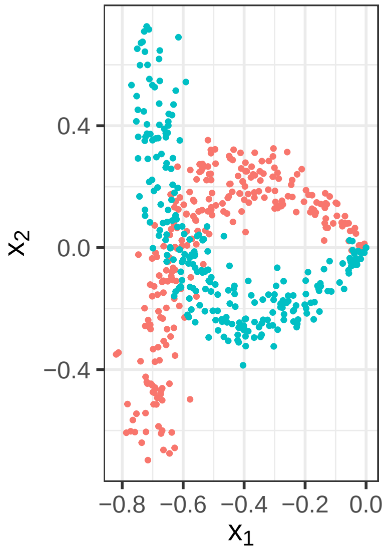


Figure 3: ASE of an RDPG drawn from the latent positions, labeled by curve/community.

We then try applying K-curves clustering to this graph. The first three are with random initial labels, forcing the intercept to be zero. The fourth initializes the labels randomly but allows the intercept to be nonzero. The fifth initializes the labels by spectral clustering with the normalized Laplacian, again forcing the intercept to be zero. The sixth also initializes via spectral clustering but allows the intercept to be nonzero.

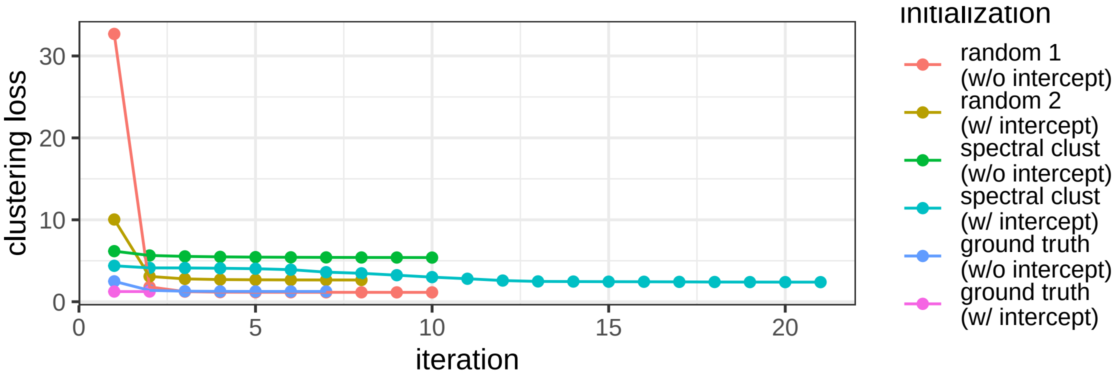


Figure 4: Clustering loss vs. iteration for each run of K-curve clustering.

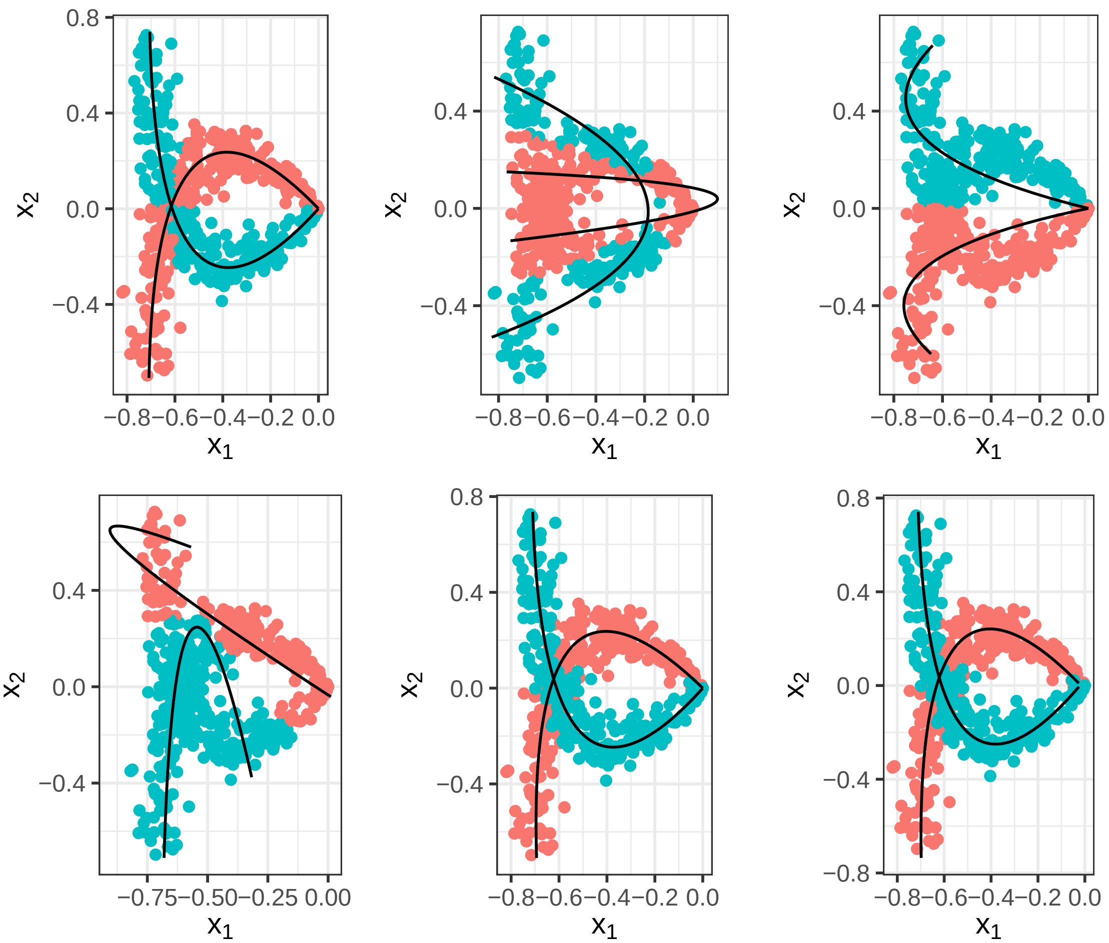
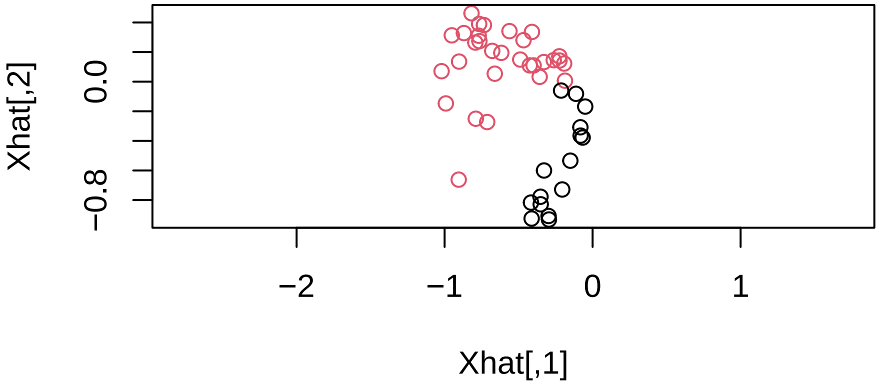
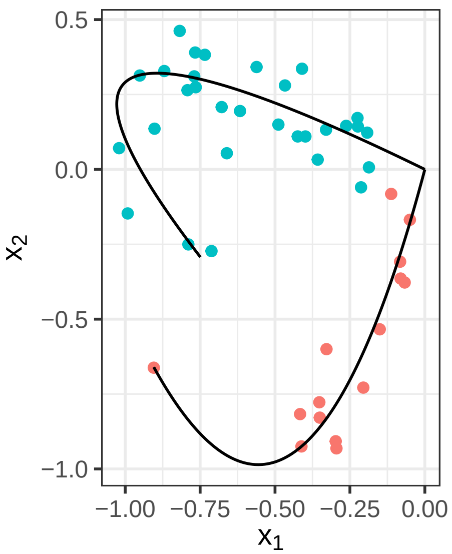
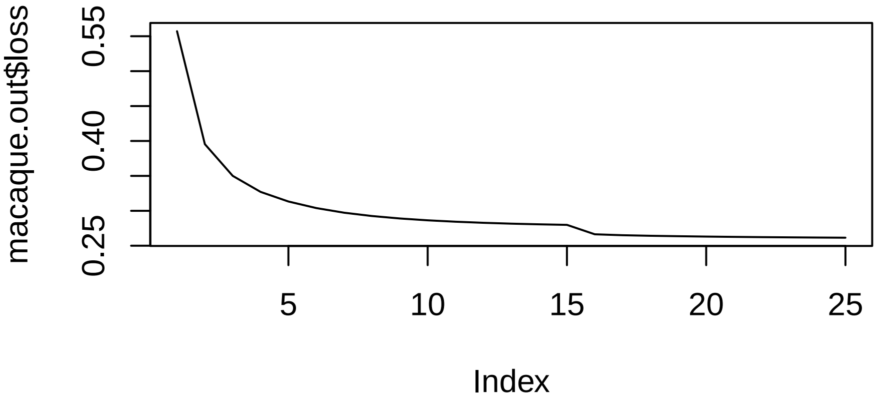


Figure 5: ASE labeled by estimated community labels for each initialization strategy.

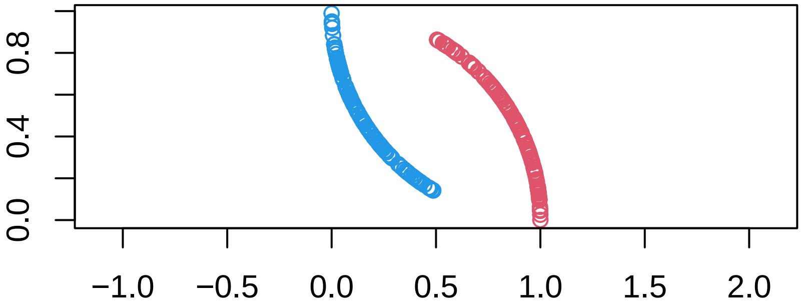
Example 4 (Macaque visuotactile brain areas and connections [1]).

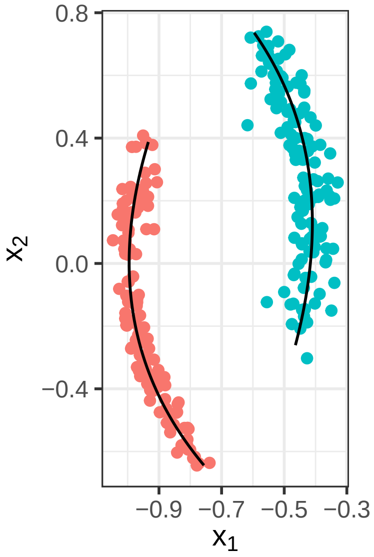




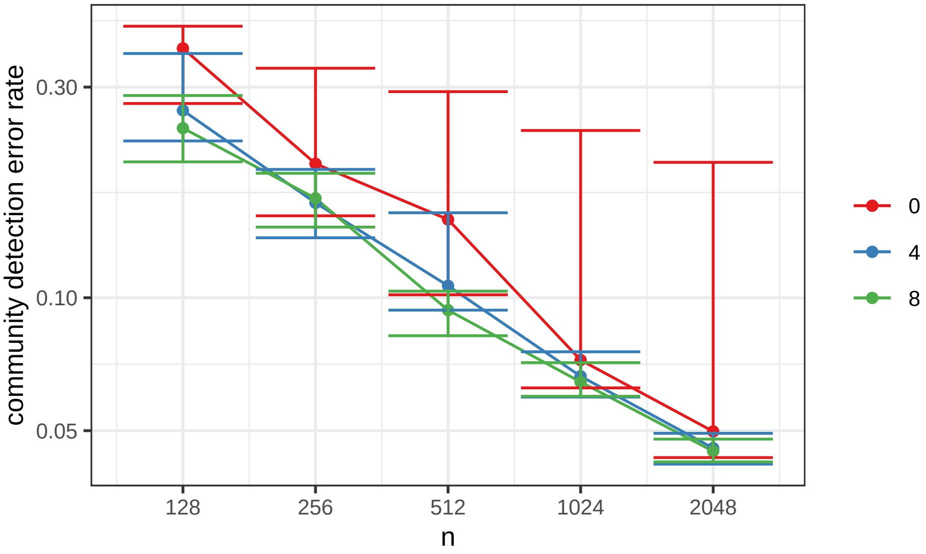


Example 5 (Non-intersecting curves).





5 Simulation Study



6 Discussion

A Proofs of Theorems

Lemma 1. *Let $x_1, ..., x_n$ be drawn from K compact, connected manifolds $\mathcal{M}_1$, ..., $\mathcal{M}_K$ each with probability measures F_1, ..., F_K, and the manifolds are separated by distance at least*

$\delta > 0$. *Suppose that for any* $\epsilon > 0$ *and* x *drawn from each* F_k*, on* $\mathcal{M}_k$*,* $F(B(x, \epsilon)) > 0$ *where* $B(x, \epsilon)$ *is the open ball of radius* ϵ *centered at* x*. Let* $E_n(\eta)$ *denote the event that an* η*-neighborhood graph constructed from* $x_1, ..., x_n$ *is comprised of exactly* K *disjoint subgraphs that map to each of the* K *manifolds. Then if each* $n_k \to \infty$ *as* $n \to \infty$*,* $\lim\limits_{n\to\infty} P(E_n(\eta)) = 1$ *for each* $\eta \in (0, \delta)$*.*

Proof. It is clear that if $\eta \in (0, \delta)$, an η-neighborhood graph constructed from the sample will always consist of at least K disjoint subgraphs for which no subgraph contains vertices belonging to points from two different manifolds. Then it is sufficient to show that for a sufficiently large n, any η-neighborhood graph (where $\eta \in (0, \delta)$) will achieve E_n.

Define each $E_{n_k}^{(k)}(\eta)$ as the event that if a sub-sample of size n_k drawn from manifold $\mathcal{M}_k$, every $x \in \mathcal{M}_k$ is within distance η of some x_j of the sub-sample. Then if $E_{n_k}^{(k)}(\eta)$ is true, the η-neighborhood graph results in a connected subgraph for points within the k^{th} manifold. By lemma 2 of **(author?)** [3], $P((E_{n_k}^{(k)}(\eta))^c) \leq \ell_k(1 - b_k)^{n_k}$ for some $\ell_k \in \mathbb{N}$ and $b_k \in (0, 1]$. If each $E_{n_k}^{(k)}(\eta)$ is true, then E_n is achieved, so $E_n(\eta) = \bigcap_k E_{n_k}^{(k)}(\eta))$. $\bigcap_k E_{n_k}^{(k)}(\eta) = \Big(\bigcup_k (E_{n_k}^{(k)}(\eta))^c\Big)^c$, so it is sufficient to show $\lim\limits_{n\to\infty} P\big(\bigcup_k (E_{n_k}^{(k)}(\eta))^c\big) \to 0$.

$$\begin{aligned} P\Big(\bigcup_k (E_{n_k}^{(k)})^c\Big) &\leq \sum_k P\big((E_{n_k}^{(k)})^c\big) \\ &\leq \sum_k \ell_k(1 - b_k)^{n_k} \\ &\leq K\ell_{\max}(1 - b_{\min})^{n_{\min}}, \end{aligned}$$

which tends to 0 as $n \to \infty$. □

Corollary 1. *Let* $x_1, ..., x_n$ *be drawn from* K *compact, connected, one-dimensional manifolds* $\mathcal{M}_1, ..., \mathcal{M}_K$*, each with probability measures* $F_1, ..., F_K$*, and the manifolds are spearated by distance at least* $\delta > 0$*.*

Proof of theorem 1. Define $E_n(\eta)$ as in lemma 1 for manifolds $Q_n(\mathcal{M}_1), ..., Q_n(\mathcal{M}_K)$ and each $e_i = \hat{x}_i - Q_n x_i$ where $\hat{x}_i$ is the i^{th} embedding vector and Q_n is some indefinite orthogonal transformation as in **(author?)** [2]. Since Q_n is a linear map, for any $\eta \in (0, \|Q_n\|\delta)$, $P(E_n(\eta)) \to 1$ as $n \to \infty$. Let $\epsilon_i = \|e_i\|$, $\epsilon = \max_i \epsilon_i$, and $C_n = \|Q_n\|$.

$A_n(\eta)$ is true if $\eta < \min\limits_{k,\ell} \min\limits_{x_i \in \mathcal{M}_k, x_j \in \mathcal{M}_\ell} \|\hat{x}_i - \hat{x}_j\|$, which is defined as event $D_n(\eta)$, and $\eta \geq \max\limits_{k} \max\limits_{x_i, x_j \in \mathcal{M}_k} \|\hat{x}_i - \hat{x}_j\|$, which is defined as event $\hat{E}_n(\eta)$.

For any x_i, x_j from different manifolds, $\|\hat{x}_i - \hat{x}_j\| \geq C_n\delta - 2\epsilon$ if $2\epsilon \leq C_n\delta$. By theorem 3 of **(author?)** [2], for some finite $M > 0$, $P\big(\epsilon < M\frac{\log^c n}{\sqrt{n}}\big) \to 1$ as $n \to \infty$, so $P(C_n\delta < 2\epsilon) \leq P(C_n\delta < 2Mn^{1/2}\log^c n) \to 0$ since $C_n\delta > 0$. Then since $P(C_n\delta - 2\epsilon > 0) \to 1$, there is an $\epsilon \in (0, C_n\delta - 2\epsilon)$ with probability 1. Thus, $P(D_n(\eta)) \to 1$.

Then to show that $P(A_n) \to 1$:

$$\begin{aligned}
P(A_n) &= P(\hat{E}_n(\eta) \cap D_n) \\
&= P((\hat{E}_n^c(\eta) \cup D_n^c)^c) \\
&= 1 - P(\hat{E}_n^c(\eta) \cup D_n^c) \\
&\geq 1 - P(\hat{E}_n^c(\eta)) - P(D_n^c) \\
&= P(\hat{E}_n(\eta)) + P(D_n) - 1,
\end{aligned}$$

which tends toward 1 as $n \to \infty$ since both $P(E_n(\eta))$ and $P(D_n)$ tend toward 1 as $n \to \infty$. □

B Details on Fitting Bezier Curves with Noise

References

[1] László Négyessy, Tamás Nepusz, László Kocsis, and Fülöp Bazsó. Prediction of the main cortical areas and connections involved in the tactile function of the visual cortex by network analysis. *European Journal of Neuroscience*, 23(7):1919–1930, 2006.

[2] Patrick Rubin-Delanchy, Joshua Cape, Minh Tang, and Carey E. Priebe. A statistical interpretation of spectral embedding: The generalised random dot product graph. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2022.

[3] Michael W. Trosset and Gokcen Buyukbas. Rehabilitating isomap: Euclidean representation of geodesic structure, 2020.