

# Robust LLG

March 5, 2017

## 1 Background and Overview

Network analysis is becoming more and more widely used recently. In a general parametric framework,  $G \sim f \in \mathcal{F} = \{f_\theta : \theta \in \Theta\}$ , selecting a reasonable estimator  $\hat{\theta}(G)$  for the unknown  $\theta$  given a finite graph sample  $\{G^{(1)}, \dots, G^{(m)}\}$  is one of the most important tasks.

Consider the most basic setting, i.e. undirected and unweighted graphs with each edge independently distributed from a Bernoulli distribution, with the parameters to be the probabilities of the existence of edges between all pair of vertices. Then the maximum likelihood estimate, which happens to be the entry-wise sample mean in this situation, is the uniformly minimum-variance unbiased estimator under the independent edge graph model (IEM) [Bollobás et al., 2007] without taking any graph structure into account. However, in a high dimensional situation, unbiased estimators often leads to inaccurate estimates with very high variance when the sample size is small. And for this basic setting we considered, the graphs are high dimensional object with  $n^2$  parameters since  $\Theta = [0, 1]^{n \times n}$ , where  $n$  denotes the number of vertices. Thus the MLE does not perform very well especially when there are only a few observations available, which is likely the case in real world.

Generally, by biasing towards low-rank structures carefully, the estimators will have a greatly reduced variances and win the bias-variance trade-off [Trunk, 1979]. Such improvement is not only important for the estimation itself, but also can help with other statistical inference procedures. For example, Ginestet et al. [2014] proposed a method to test if there is a difference between the networks of two groups of subjects. While hypothesis testing is the final goal, the estimation procedure is a key intermediate step and can be improved.

In order to improve the MLE considered above, the underlying graph structures are taken into account for exploiting the low-rank structure when constructing the estimator. One of the most important structures about the networks is the community structure in which vertices are clustered into different communities such that vertices within the same community behave similarly. The stochastic blockmodel (SBM) [Holland et al., 1983] captures such structural property strongly by assuming that vertices from the same block behave exactly the same.

As a generalization of the SBM, the latent position model (LPM) [Hoff et al., 2002] allows vertices to behave differently. Generally, the adjacencies between vertices depend on unobserved properties of the corresponding vertices. Thus under LPM, each vertex is associated with a latent position which influences

the adjacencies for that vertex based on a link function. In this paper, we consider a special case of LPM, the random dot product graph (RDPG) [Young and Scheinerman, 2007, Nickel, 2007], which has the dot product as the link function.

Recently, Tang et al. [2016] considers an estimator based on a low-rank approximation of the sample mean graph motivated by the RDPG model and proves that in the basic Bernoulli setting under SBM, the new estimator outperforms the element-wise sample mean since it decreases the overall asymptotic variance dramatically by smoothing towards the low-rank structure.

While Tang et al. [2016] demonstrate the advantage of the low-rank estimator for the unweighted graphs with Bernoulli distribution, the results are based on the assumption that graphs are observed without contaminations. However in practice there will be noise in the observed graphs. In this situation, there is no guarantee that the performance of the low-rank estimator is still better.

In this work, we are going to improve the estimation procedure in a contaminated scenario. Before we introduce the contaminations, it is helpful to extend the unweighted graphs with Bernoulli distribution to weighted graphs with a general distribution  $f$ . All the models we mentioned above (IEM, RDPG, and SBM) could be extended naturally and are discussed in details in Section 2.1, Section 2.2, and Section 2.3 respectively.

One of the most popular contamination model is the gross error model [Mah and Tamhane, 1982, Bickel and Doksum, 2001]. In a gross error model, we observe good measurement  $G^* \sim f_P \in \mathcal{F}$  most of the time, while there are a few wild values  $G^{**} \sim h_C \in \mathcal{H}$  when the gross errors occur. As to the graphs, one way to generalize from the gross error model is to contaminate the entire graph with some small probability  $\epsilon \in (0, 1)$ , that is  $G \sim (1 - \epsilon)f_P + \epsilon h_C$ . However, since all the models we consider are subsets of the IEM, it is more natural to consider the contaminations with respect to each edge, i.e. for  $1 \leq i, j \leq n$ ,  $G_{ij} \sim (1 - \epsilon)f_{P_{ij}} + \epsilon h_{C_{ij}}$  with  $f \in \mathcal{F}$  and  $h \in \mathcal{H}$ , where both  $\mathcal{F}$  and  $\mathcal{H}$  are one-parameter distribution families.

Under the contamination model, although we observe  $G$  instead of  $G^*$ , estimating the parameters  $P_{ij}$  ( $1 \leq i, j \leq n$ ) of  $f_{P_{ij}}$  in  $\mathcal{F}$  is still our goal. We first prove that the low-rank estimator ( $\tilde{P}^{(1)}$  in Figure 1) proposed in [Tang et al., 2016] is still much better than the entry-wise MLE ( $\hat{P}^{(1)}$  in Figure 1) in terms of mean squared error when the observations are contaminated under proper conditions.

Furthermore, with contaminations, it is more preferable to use robust methods, like MLqE [Ferrari and Yang, 2010, Qin and Priebe, 2013] considered in this paper. We prove that entry-wise MLqE ( $\hat{P}^{(q)}$  in Figure 1) improves the performance compared to entry-wise MLE ( $\hat{P}^{(1)}$  in Figure 1) whenever contamination is relatively large.

Similarly, in order to take advantage of the low-rank structure, we enforce a low-rank approximation on the entry-wise MLqE. We prove that, under proper assumptions, the new estimator ( $\tilde{P}^{(q)}$  in Figure 1) not only inherits the robust property from MLqE ( $\hat{P}^{(q)}$  in Figure 1), but also wins the bias-variance trade-off by taking advantage of the low-rank structure.

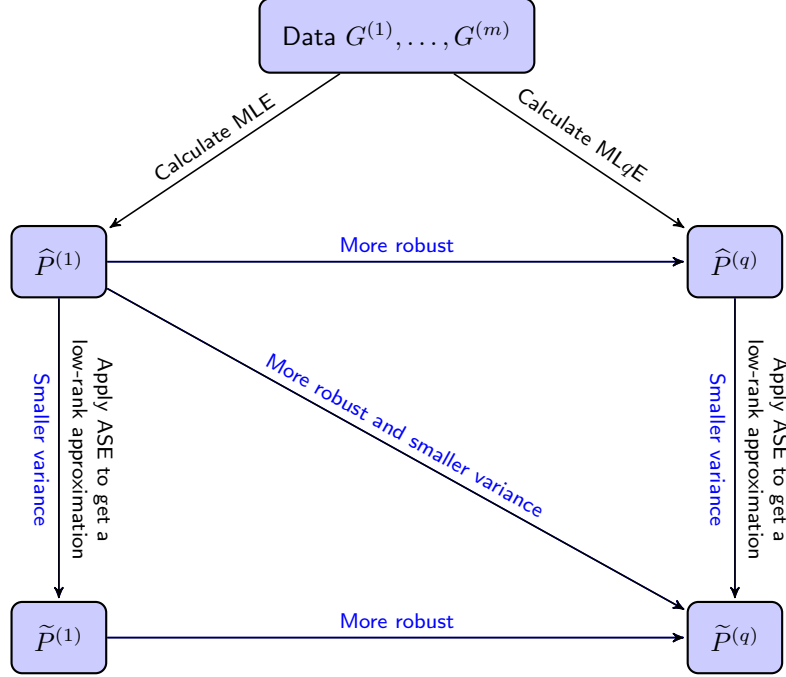


Figure 1: Roadmap among the data and four estimators.

## 2 Models

- For this work, we are in the scenario where  $m$  weighted graphs each with  $n$  vertices are given in the adjacency matrices form  $\{A^{(t)}\}(t = 1, \dots, m)$ . The graphs we consider in this work are undirected without self-loop, i.e. each  $A^{(t)}$  is symmetric with zeros along the diagonal. Moreover, we assume the vertex correspondence is known throughout different graphs, so that vertex  $i$  of graph  $t_1$  corresponds to vertex  $i$  of graph  $t_2$  for any  $i, t_1, t_2$ .
- In this section, we present three nested models, the weighted independent edge model in Section 2.1, the weighted random dot product graph model in Section 2.2, and the weighted stochastic blockmodel (WSBM) as a WRDPG in Section 2.3. Moreover, we introduce a contaminated model based on Section 2.3 in Section 2.4.

### 2.1 Weighted Independent Edge Model

- We first extend the definition of independent edge model (IEM) Bollobás et al. [2007] to the weighted independent edge model (WIEM) corresponding to a one-parameter family  $\mathcal{F} = \{f_\theta : \theta \in \Theta \subset \mathbb{R}\}$ . Denote the graph parameter as a matrix  $P \in \Theta^{n \times n} \subset \mathbb{R}^{n \times n}$ , then under a WIEM, each edge between vertex  $i$  and vertex  $j$  ( $i < j$  because of symmetry) is distributed from  $f_{P_{ij}}$  independently.

- A simple example here is a WIEM for binary graphs with respect to  $\mathcal{F}$  to be Bernoulli with a symmetric and hollow parameter matrix  $P \in [0, 1]^{n \times n}$ .
- Note that the graphs considered in this paper are undirected without self-loop, thus the parameter matrix  $P$  needs to be symmetric and hollow. However, for convenience, we still define the parameters to be an  $n$ -by- $n$  matrix while only  $\binom{n}{2}$  of them are effective.

## 2.2 Weighted Random Dot Product Graph

- The connectivity between two vertices in a graph generally depends on some hidden properties of the corresponding vertices. The latent position model proposed by Hoff et al. [2002] captures such properties by assigning each vertex  $i$  with a corresponding latent vector  $X_i \in \mathbb{R}^d$ . Conditioned on the latent vectors  $X_i$  and  $X_j$ , the edge between vertex  $i$  and vertex  $j$  is independent of all other edges and depends only on  $X_i$  and  $X_j$  through a link function. This can easily be generalized to the weighted version easily.
- A special case of the latent position model is the random dot product graph model (RDPG) in which the link function is the inner product Young and Scheinerman [2007], Nickel [2007]. In the following, we give a definition of the weighted random dot product graph (WRDPG) as a generalization of the weighted latent position model.

**Definition 2.1** *Consider a collection of one-parameter distributions  $\mathcal{F} = \{f_\theta, \theta \in \Theta \subset \mathbb{R}\}$ . Then the weighted random dot product graph model (WRDPG) with respect to  $\mathcal{F}$  is defined as following: Let  $\mathbf{X} \in \mathbb{R}^{n \times d}$  be such that  $\mathbf{X} = [X_1, X_2, \dots, X_n]^T$ , where  $X_i \in \mathbb{R}^d$  for all  $i \in [n]$ . The matrix  $\mathbf{X}$  is random and satisfies  $\mathbb{P}[\langle X_i, X_j \rangle \in \Theta] = 1$  for all  $i, j \in [n]$ . Conditioned on  $\mathbf{X}$ , the entries of the adjacency matrix  $\mathbf{A}$  are independent and  $A_{ij}$  is a random variable with distribution  $f \in \mathcal{F}$  with parameter  $\langle X_i, X_j \rangle$  for all  $i \neq j \in [n]$ .*

Under the WRDPG defined above, the parameter matrix  $P = \mathbf{X}\mathbf{X}^T \in \mathbb{R}^{n \times n}$  is automatically symmetric because the link function is inner product. Moreover, to have symmetric graphs without self-loop, only  $A_{ij}$  ( $i < j$ ) are sampled while leaving the diagonals of  $A$  to be all zeros.

## 2.3 Weighted Stochastic Blockmodel as a Weighted Random Dot Product Graph

- Community structure is an important property of graphs under which vertices are clustered into different communities such that vertices within the same community behave similarly. The stochastic blockmodel (SBM) Holland et al. [1983] captures such property, where each vertex is assigned to one block and the connectivity between two vertices depends only on their respective block memberships.
- Formally, the SBM is determined by the number of blocks  $K$  (generally much smaller than the number of vertices  $n$ ), block probability matrix

$B \in \Theta^{K \times K} \subset \mathbb{R}^{K \times K}$ , and the block assignment vector  $\tau \in [K]^n$ , where  $\tau_i = k$  represents vertex  $i$  belongs to block  $k$ . Conditioned on the block membership  $\tau$ , the connectivity between vertex  $i$  and vertex  $j$  follows a Bernoulli distribution with parameter  $B_{\tau_i, \tau_j}$ . This can easily be generalized to the weighted stochastic blockmodel (WSBM), with the Bernoulli distribution replaced by a general distribution family  $\mathcal{F}$ .

- In order to consider WSBM as a WRDPG, the matrix  $B$  needs to be positive semi-definite. From now on, we will note the sub-model of WSBM with positive semi-definite  $B$  as the WSBM.
- Now consider the WSBM as a WRDPG with respect to  $\mathcal{F} = \{f_\theta, \theta \in \Theta \subset \mathbb{R}\}$ . Let  $d = \text{rank}(B)$ , then all vertices in block  $k$  have shared latent position  $\nu_k \in \mathbb{R}^d$ , where  $B = \nu\nu^\top$  and  $\nu = [\nu_1, \dots, \nu_K]^\top \in \mathbb{R}^{K \times d}$ . That is to say,  $X_i = \nu_{\tau_i}$  and  $A_{ij}$  ( $i < j$ ) is distributed from  $f$  with parameter  $B_{\tau_i, \tau_j} = \nu_{\tau_i}^\top \nu_{\tau_j}$ . Here the parameter matrix  $P \in \mathbb{R}^{n \times n}$  is symmetric, hollow, and satisfies  $P_{ij} = X_i^\top X_j = \nu_{\tau_i}^\top \nu_{\tau_j} = B_{\tau_i, \tau_j}$ .
- In order to generate  $m$  graphs under this model with known vertex correspondence, we first sample  $\tau$  from the categorical distribution with parameter  $\rho$  and keep it fixed for all  $m$  graphs. Then  $m$  symmetric and hollow graphs  $G^{(1)}, \dots, G^{(m)}$  are sampled such that conditioning on  $\tau$ , the adjacency matrices are distributed entry-wise independently as  $A_{ij}^{(t)} \stackrel{\text{ind}}{\sim} f_{B_{\tau_i, \tau_j}} = f_{P_{ij}}$  for each  $1 \leq t \leq m$ ,  $1 \leq i < j \leq n$ . Here  $P \in \mathbb{R}^{n \times n}$  is the actual parameter matrix for each pair of vertices with the same block structure as  $B$ .
- Here is an example of the SBM. (Figure for simulation)

## 2.4 Stochastic Blockmodel as a Weighted Random Dot Product Graph with Contaminations

- In practice, we always get noise in the observations, which deviates from our general model assumptions. In order to incorporate this effect, a contamination model, e.g. gross error model Mah and Tamhane [1982], Bickel and Doksum [2001] considered in this work, is always preferred.
- Generally in a gross error model, we observe good measurement  $G^* \sim f_P \in \mathcal{F}$  most of the time, while there are a few wild values  $G^{**} \sim h_C \in \mathcal{H}$  when the gross errors occur. Here  $P$  and  $C$  represent the respective parameter matrices for the two distribution families. As to the graphs, one way is to contaminate the entire graph with some small probability  $\epsilon$ , that is  $G \sim (1 - \epsilon)f_P + \epsilon h_C$ . However, since we are under the independent edge model, it is more natural to consider the contaminations on each edge, i.e. for  $1 \leq i < j \leq n$ ,  $G_{ij} \sim (1 - \epsilon)f_{P_{ij}} + \epsilon h_{C_{ij}}$  with  $f \in \mathcal{F}$  and  $h \in \mathcal{H}$ , where both  $\mathcal{F}$  and  $\mathcal{H}$  are one-parameter distribution families.
- In this paper, we assume that when gross errors occur, the connectivity also follows the WSBM as a WRDPG. That is, the contamination distributions  $h_{C_{ij}}$  are also from the same one-parameter family  $\mathcal{F}$  as  $f_{P_{ij}}$  do. Moreover, the parameter matrix  $C$  for the contamination has the same block structure. (Note: not necessarily the same block structure)

- To generate  $m$  graphs under this contamination model with known vertex correspondence, we first sample  $\tau$  from the categorical distribution with parameter  $\rho$  and keep it fixed for all  $m$  graphs as in Section 2.3. Then  $m$  symmetric and hollow graphs  $G^{(1)}, \dots, G^{(m)}$  are sampled such that conditioning on  $\tau$ , the adjacency matrices are distributed entry-wise independently as  $A_{ij}^{(t)} \stackrel{\text{ind}}{\sim} (1 - \epsilon)f_{P_{ij}} + \epsilon f_{C_{ij}}$  for each  $1 \leq t \leq m, 1 \leq i < j \leq n$ . Here  $\epsilon$  is the probability of an edge to be contaminated,  $P$  is the parameter matrix as in Section 2.3, and  $C$  is the parameter matrix for contaminations.

### 3 Estimators

Under each of the models mentioned in Section 2, our goal is to estimate the parameter matrix  $P$  based on the  $m$  observations  $A^{(1)}, \dots, A^{(m)}$ . In this section, we present four estimators as in Figure 1, the standard entry-wise MLE  $\hat{P}^{(1)}$ , the low-rank approximation of the entry-wise MLE  $\tilde{P}^{(1)}$ , the entry-wise robust estimator MLqE  $\hat{P}^{(q)}$ , and the low-rank approximation of the entry-wise MLqE  $\tilde{P}^{(q)}$ . Since the observed graphs are symmetric and hollow with a symmetric parameter matrix of the model, we don't really care about the estimate of the diagonal of  $P$ , but the estimate itself should be at least symmetric.

#### 3.1 Entry-wise Maximum Likelihood Estimator $\hat{P}^{(1)}$

- Under the WIEM, the most natural estimator is the element-wise MLE  $\hat{P}^{(1)}$  based on the adjacency matrices  $A^{(1)}, \dots, A^{(m)}$ .
- In some cases, for instance Bernoulli, Exponential, and Poisson, the entry-wise MLE happens to be the mean graph  $\bar{A}$ , which is the UMVUE under WIEG with no other constraints. So it has the smallest variance among all unbiased estimators. In addition, as being an MLE, it satisfies many good asymptotic properties as the number of graphs  $m \rightarrow \infty$ . But the performance will not get any better when the number of vertices in each graph  $n$  increases. In addition, it doesn't exploit any graph structure. If the graphs are actually distributed under a WRDPG or a WSBM, then the entry-wise MLE is no longer the MLE any more. As a result, the performance can be very poor, especially when  $m$  is small.

#### 3.2 Estimator $\tilde{P}^{(1)}$ Based on Adjacency Spectral Embedding of $\hat{P}^{(1)}$

- Motivated by the low-rank structure of the parameter matrix  $P$  in WRDPG, we consider the estimator  $\tilde{P}^{(1)}$  proposed by Tang et al. Tang et al. [2016] based on the spectral decomposition of  $\hat{P}^{(1)}$ . The estimator  $\tilde{P}^{(1)}$  is similar to the estimator proposed by Chatterjee Chatterjee et al. [2015] with adjustment for the specific estimation task, including a different dimension selection technique and a diagonal augmentation procedure.

### 3.2.1 Rank- $d$ Approximation

- Given the dimension  $d$ , we consider  $\text{lowrank}_d(\hat{P}^{(1)})$ , the best rank- $d$  positive semi-definite approximation of  $\hat{P}^{(1)}$ , denoted as  $\tilde{P}^{(1)}$ . First calculate the eigen-decomposition of the symmetric matrix  $\hat{P}^{(1)} = \hat{U}\hat{S}\hat{U}^\top + \tilde{U}\tilde{S}\tilde{U}^\top$ , where  $\hat{S}$  is the diagonal matrix with the largest  $d$  eigenvalues of  $\hat{P}^{(1)}$ , and  $\hat{U}$  has the corresponding eigenvectors as each column. Similarly,  $\tilde{S}$  is the diagonal matrix with non-increasing entries along the diagonal corresponding to the rest  $n - d$  eigenvalues of  $\hat{P}^{(1)}$ , and  $\tilde{U}$  has the columns given by the corresponding eigenvectors. The  $d$ -dimensional adjacency spectral embedding (ASE) of  $\hat{P}^{(1)}$  is given by  $\hat{X} = \hat{U}\hat{S}^{1/2} \in \mathbb{R}^{n \times d}$ . Using the adjacency spectral embedding, we have the best rank- $d$  positive semi-definite approximation of  $\hat{P}^{(1)}$  to be  $\tilde{P}^{(1)} = \hat{X}\hat{X}^\top = \hat{U}\hat{S}\hat{U}^\top$ . In the RDPG setting, Sussman et al. [2014] proved that each row of  $\hat{X}$  can accurately estimate the latent position for each vertex up to an orthogonal transformation. We will analyze its performance under the WRDPG setting in Section 4.
- We restate the algorithm in Tang et al. [2016] to give the steps of computing this low-rank approximation of a general  $n$ -by- $n$  symmetric matrix  $A$  in Algorithm 1.

---

**Algorithm 1** Algorithm to compute the rank- $d$  approximation of a matrix.

---

**Input:** Symmetric matrix  $A \in \mathbb{R}^{n \times n}$  and dimension  $d \leq n$ .

**Output:**  $\text{lowrank}_d(A) \in \mathbb{R}^{n \times n}$

- 1: Compute the algebraically largest  $d$  eigenvalues of  $A$ ,  $s_1 \geq s_2 \geq \dots \geq s_d$  and corresponding unit-norm eigenvectors  $u_1, u_2, \dots, u_d \in \mathbb{R}^n$ ;
  - 2: Set  $\hat{S}$  to the  $d \times d$  diagonal matrix  $\text{diag}(s_1, \dots, s_d)$ ;
  - 3: Set  $\hat{U} = [u_1, \dots, u_d] \in \mathbb{R}^{n \times d}$ ;
  - 4: Set  $\text{lowrank}_d(A)$  to  $\hat{U}\hat{S}\hat{U}^\top$ ;
- 

### 3.2.2 Dimension Selection

- A general way to choose the dimension  $d$  in a dimension reduction setting is based on analyzing the ordered eigenvalues and looking for the “gap” or “elbow” in the scree-plot.
- In 2006, Zhu and Ghodsi proposed an automatic method for finding the gap in the scree-plot by only looking at the eigenvalues based on a Gaussian mixture model Zhu and Ghodsi [2006]. Generally, the method provides multiple results based on different “elbow”. In this paper, to avoid under-estimating the dimension, which is often much more harmful than over-estimating it, we always choose the 3rd elbow.
- Although it is always challenge to choose a proper dimension, based on the (try simulation?) and real data experiment, a wide range of dimensions will lead to results nearly optimal. Thus a proper dimension selection method can be applied directly without carefully tuning the parameter, which is much more practical.

### 3.2.3 Diagonal Augmentation

- Since the graphs considered in this paper have no self-loops, all the adjacency matrices  $A^{(t)}$  ( $1 \leq t \leq m$ ) are hollow, i.e. all diagonal entries are zeros. And thus the diagonal of the parameter matrix  $P$  doesn't matter since all off-diagonal entries are independent of them conditioned on the off-diagonal entries of  $P$ .
- However, unlike the entry-wise estimators, e.g.  $\hat{P}^{(1)}$ , the ones which take advantage of the graph structure need the information from the diagonals. As a result, the zero diagonals of the observed graphs will lead to biased estimates.
- To compensate for such inaccurate estimate, Marchette et al. [2011] suggested to use the average of the non-diagonal entries of the corresponding row as the diagonal entry before embedding. Also, Scheinerman and Tucker [2010] proposed an iterative method, which gives a different approach to resolve such issue.
- As suggested in Tang et al. [2016], in this work we are going to combine both ideas by first using Marchette's row-averaging method (see Step 3 of Algorithm 2) and then another one-step Scheinerman's iterative method (see Step 6 of Algorithm 2).

---

#### Algorithm 2 Algorithm to compute $\tilde{P}^{(1)}$

---

**Input:** Symmetric adjacency matrices  $A^{(1)}, A^{(2)}, \dots, A^{(m)}$ , with each  $A^{(t)} \in \mathbb{R}^{n \times n}$

**Output:** Estimate  $\tilde{P}^{(1)} \in \mathbb{R}^{n \times n}$

- 1: Calculate the entry-wise MLE  $\hat{P}^{(1)}$ ;
  - 2: Calculate the scaled degree matrix  $D = \text{diag}(\hat{P}^{(1)}\mathbf{1})/(n-1)$ ;
  - 3: Select the dimension  $d$  based on the eigenvalues of  $\hat{P}^{(1)} + D$ ; (see Section 3.2.2)
  - 4: Set  $Q$  to  $\text{lowrank}_d(\hat{P}^{(1)} + D)$ ; (see Algorithm 1)
  - 5: Set  $D'$  to  $\text{diag}(Q)$ , the diagonal matrix with diagonal matching  $Q$ ;
  - 6: Set  $Q'$  to  $\text{lowrank}_d(Q + D')$ ; (see Algorithm 1)
  - 7: Set  $\tilde{P}^{(1)}$  with each entry  $\tilde{P}_{ij}^{(1)} = \min(\hat{P}_{ij}^{(1)}, \max(Q'_{ij}, 0))$ .
- 

- Truncation procedure. Need reasonable explanation here.
- Detailed description of the estimator  $\tilde{P}^{(1)}$  with dimension selection method and diagonal augmentation procedure are given in Algorithm 2.

### 3.3 Entry-wise Maximum $L_q$ -likelihood Estimator $\hat{P}^{(q)}$

- The MLE is asymptotically efficient, i.e. when sample size is large enough, the MLE is at least as accurate as any other estimator. However, when the sample size is moderate, robust estimators always outperforms MLE in terms of mean squared error by winning the bias-variance tradeoff. Moreover, under contamination models, robust estimators can even beat



MLE asymptotically since they are designed to be not unduly affected by the outliers. And we are going to consider the maximum  $L_q$ -likelihood estimator (ML $q$ E), proposed by Ferrari and Young in 2010 Ferrari and Yang [2010], in this work.

- Let  $X_1, \dots, X_m$  be sampled from  $f_{\theta_0} \in \mathcal{F} = \{f_\theta, \theta \in \Theta\}$ ,  $\theta_0 \in \Theta$ . Then the maximum  $L_q$ -likelihood estimate ( $q > 0$ ) of  $\theta_0$  based on the parametric model  $\mathcal{F}$  is defined as

$$\hat{\theta}_{\text{ML}q\text{E}} = \arg \max_{\theta \in \Theta} \sum_{i=1}^m L_q[f_\theta(X_i)],$$

where  $L_q(u) = (u^{1-q} - 1)/(1 - q)$ . Note that  $L_q(u) \rightarrow \log(u)$  when  $q \rightarrow 1$ . Thus ML $q$ E is a generalization of MLE. Moreover, define

$$U_\theta(x) = \nabla_\theta \log f_\theta(x)$$

and

$$U_\theta^*(x; q) = U_\theta(x) f_\theta(x)^{1-q}.$$

Then the ML $q$ E  $\hat{\theta}_{\text{ML}q\text{E}}$  can also be seen as a solution to the equation

$$\sum_{i=1}^m U_{\hat{\theta}}^*(X_i; q) = 0.$$

This form interprets  $\hat{\theta}_{\text{ML}q\text{E}}$  as a solution to the weighted likelihood equation. The weights  $f_\theta(x)^{1-q}$  are proportional to the  $(1 - q)$ th power of the corresponding probability. Specifically, when  $0 < q < 1$ , the ML $q$ E puts less weight on the data points which don't fit the current distribution well. Equal weights happen when  $q = 1$  and lead to the MLE.

- Under the WIEM, to have robustness, we can calculate the entry-wise ML $q$ E  $\hat{P}^{(q)}$  based on the adjacency matrices  $A^{(1)}, \dots, A^{(m)}$ . Note that  $\hat{P}^{(1)}$ , the entry-wise MLE, is a special case of entry-wise ML $q$ E  $\hat{P}^{(q)}$  when  $q = 1$ . That is what the superscriptions  $q$  and 1 mean.

### 3.4 Estimator $\tilde{P}^{(q)}$ Based on Adjacency Spectral Embedding $\hat{P}^{(q)}$

- Intuitively, the low-rank structure of the parameter matrix  $P$  in WRDPG should be preserved more or less in the entry-wise ML $q$ E  $\hat{P}^{(q)}$ . Thus, in order to take advantage of such low-rank structure as well as the robustness, we apply the similar idea here as in building  $\tilde{P}^{(1)}$ , i.e. enforce a low-rank approximation on the entry-wise ML $q$ E matrix  $\hat{P}^{(q)}$  to get  $\tilde{P}^{(q)}$ . As in Algorithm 2, we apply the same dimension selection method and diagonal augmentation procedure. The only change is to substitute  $\hat{P}^{(1)}$  by  $\hat{P}^{(q)}$ . The details of the algorithm is shown in Algorithm 3. (Maybe adding a procedure for selecting  $q$ ?)

---

**Algorithm 3** Algorithm to compute  $\tilde{P}^{(q)}$ 

---

**Input:** Symmetric adjacency matrices  $A^{(1)}, A^{(2)}, \dots, A^{(m)}$ , with each  $A^{(t)} \in \mathbb{R}^{n \times n}$

**Output:** Estimate  $\tilde{P}^{(q)} \in \mathbb{R}^{n \times n}$

- 1: Calculate the entry-wise MLqE  $\hat{P}^{(q)}$ ;
  - 2: Calculate the scaled degree matrix  $D = \text{diag}(\hat{P}^{(q)}\mathbf{1})/(n-1)$ ;
  - 3: Select the dimension  $d$  based on the eigenvalues of  $\hat{P}^{(q)} + D$ ; (see Section 3.2.2)
  - 4: Set  $Q$  to  $\text{lowrank}_d(\hat{P}^{(q)} + D)$ ; (see Algorithm 1)
  - 5: Set  $D'$  to  $\text{diag}(Q)$ , the diagonal matrix with diagonal matching  $Q$ ;
  - 6: Set  $Q'$  to  $\text{lowrank}_d(Q + D')$ ; (see Algorithm 1)
  - 7: Set  $\tilde{P}^{(q)}$  to  $Q'$  with values  $< 0$  set to 0.
- 

## 4 Theoretical Results

In this section, for illustrative purpose, we are going to present theoretical results when the contaminated model introduced in 2.4 is based on exponential distributions, i.e.  $\mathcal{F} = \{f_\theta(x) = \frac{1}{\theta}e^{-x/\theta}, \theta \in [0, R] \subset \mathbb{R}\}$ , where  $R > 0$  is a constant. The results can be extended to a general situation with proper assumptions, which will be discussed in Section 5.

More specifically, consider the SBM with parameter  $B$  and  $\rho$ . First sample the block membership  $\tau$  from the categorical distribution with parameter  $\rho$  and keep it fixed for all  $m$  graphs. Conditioned on this  $\tau$  we sampled, the probability matrix  $P$  then satisfies  $P_{ij} = B_{\tau_i, \tau_j}$ . In this section, we assume the contamination has the same block membership  $\tau$ , thus the contamination matrix  $C \in \mathbb{R}^{n \times n}$  has the same block structure as  $P$ . Note that this is not necessary for the result. Different block structure can lead to the same result since the rank is still finite. Denote  $\epsilon$  as the probability of an edge to be contaminated. Then  $m$  symmetric graphs without loops  $G^{(1)}, \dots, G^{(m)}$  are sampled such that conditioning on  $\tau$ , the adjacency matrices are distributed entry-wise independently as  $A_{ij}^{(t)} \stackrel{\text{ind}}{\sim} (1 - \epsilon)f_{P_{ij}} + \epsilon f_{C_{ij}}$  for each  $1 \leq t \leq m, 1 \leq i < j \leq n$ .

Under such setting, we analyze the performance of all four estimators based on  $m$  adjacency matrices for estimating the probability matrix  $P$  with respect to the mean squared error. When comparing two estimators, we always consider about both the asymptotic bias and the asymptotic variance. Note that all the results in this section are entry-wise, which can easily lead to a result of the total MSE for the entire matrix.

We only present the most important results in this section. The proofs with more results are in the Appendix.

### 4.1 $\hat{P}^{(1)}$ vs. $\hat{P}^{(q)}$

We first compare the performance between the entry-wise MLE  $\hat{P}^{(1)}$  and the entry-wise MLqE  $\hat{P}^{(q)}$ . Without using the graphs structure, the asymptotic results for these two estimators are in terms of the number of graphs  $m$ , not the number of vertices  $n$  within each graph.

**Lemma 4.1** *For any  $0 < q < 1$ , there exists  $C_0(P_{ij}, \epsilon, q) > 0$  such that under the contaminated model with  $C > C_0(P_{ij}, \epsilon, q)$ ,*

$$\lim_{m \rightarrow \infty} \left| E[\hat{P}_{ij}^{(q)}] - P_{ij} \right| < \lim_{m \rightarrow \infty} \left| E[\hat{P}_{ij}^{(1)}] - P_{ij} \right|,$$

for  $1 \leq i, j \leq n$  and  $i \neq j$ .

Lemma 4.1 shows that the entry-wise MLqE  $\hat{P}^{(q)}$  has smaller bias for estimating  $P$  asymptotically compared to the entry-wise MLE  $\hat{P}^{(1)}$  under proper conditions. Although we put restrictions on the parameter matrix  $C$  as an assumption, to have the current result about asymptotic bias, we only need to satisfy the inequality  $\epsilon(C_{ij} - P_{ij}) > (1 - q)P_{ij}$ . This condition actually requires the contamination of the model is large enough (either large contamination parameter matrix, or more likely to encounter an outlier). From a different perspective, it also requires  $\hat{P}^{(q)}$  to be robust enough with respect to the contamination. Thus besides the current condition for  $C$ , equivalently, we can also replace it by the assumption of a large enough  $\epsilon$  or small enough  $q$ .

**Lemma 4.2**

$$\lim_{m \rightarrow \infty} \text{Var}(\hat{P}_{ij}^{(1)}) = \lim_{m \rightarrow \infty} \text{Var}(\hat{P}_{ij}^{(q)}) = 0,$$

for  $1 \leq i, j \leq n$ .

By this lemma, both estimators have asymptotic variance to be zero as the number of graphs  $m \rightarrow \infty$  according to Lemma 4.2. Under the mixture model, the result for MLE follows the central limit theorem immediately, while the MLqE result is not trivial. Our proof is based on the minimum contrast estimates (more details later).

As a result,  $\hat{P}^{(q)}$  reduces the bias while keeping variance the same asymptotically compared to  $\hat{P}^{(1)}$ . Thus in terms of MSE,  $\hat{P}^{(q)}$  is a better estimator than  $\hat{P}^{(1)}$  when the number of graphs  $m$  is large with enough contaminations.

## 4.2 $\hat{P}^{(1)}$ vs. $\tilde{P}^{(1)}$

Then we are going to analyze the effect of the ASE procedure applied to the entry-wise MLE  $\hat{P}^{(1)}$  under the contamination model.

**Corollary 4.3** *Assuming that  $m = O(n^b)$  for any  $b > 0$ , then the estimator based on ASE of MLE has the same entry-wise asymptotic bias as MLE, i.e.*

$$\lim_{n \rightarrow \infty} \text{Bias}(\tilde{P}_{ij}^{(1)}) = \lim_{n \rightarrow \infty} E[\tilde{P}_{ij}^{(1)}] - P_{ij} = \lim_{n \rightarrow \infty} E[\hat{P}_{ij}^{(1)}] - P_{ij} = \lim_{n \rightarrow \infty} \text{Bias}(\hat{P}_{ij}^{(1)}).$$

To prove the the corollary, we first give a bound on the 2-norm of  $\hat{P}^{(1)} - E[\hat{P}^{(1)}]$  by the matrix Bernstein inequality Tropp [2012]. Since we are under the RDGP setting,  $E[\hat{P}^{(1)}]$  still has finite rank, which makes the embedding reasonable. Then we approximate  $U^T \hat{U}$  by an orthogonal matrix  $W^*$  based on Davis-Kahan theorem Davis and Kahan [1970], where  $U$  and  $\hat{U}$  are the eigenspaces for  $E[\hat{P}^{(1)}]$  and  $\hat{P}^{(1)}$  respectively. Then we can analyze the bounds much more easily by interchanging  $U^T \hat{U}$  in the matrix multiplications. As a result, we can get the bound between the estimates of the latent positions and the true

latent positions up to an orthogonal transformation in terms of  $2 \rightarrow \infty$  norm. Corollary 4.3 then follows by selecting the parameter carefully in the analysis.

Corollary 4.3 says that when  $m = O(n^b)$  for any  $b > 0$ , the ASE procedure applied to  $\hat{P}^{(1)}$  will not affect the asymptotic bias for estimating  $P$ . In this case, the asymptotic relative efficiency (ARE) Serfling [2011] provides a good way to compare these two estimators. The definition proposed by Serfling in 2009 is based on unbiased estimators, here we extend it a little bit such that it can be measure two estimators which have the same asymptotic bias.

**Definition 4.4** For any parameter  $\theta$  of a distribution  $f$ , and for estimators  $\hat{\theta}^{(1)}$  and  $\hat{\theta}^{(2)}$  approximately  $N(\theta', V_1(f)/n)$  and  $N(\theta', V_2(f)/n)$  respectively, the ARE of  $\hat{\theta}^{(2)}$  to  $\hat{\theta}^{(1)}$  is given by

$$\text{ARE}(\hat{\theta}^{(2)}, \hat{\theta}^{(1)}) = \frac{V_1(f)}{V_2(f)}.$$

In our situation, we can compare the performance between  $\tilde{P}^{(1)}$  and  $\hat{P}^{(1)}$  entry-wise based on the ARE, which can be written as  $\text{ARE}(\tilde{P}^{(1)}, \hat{P}^{(1)}) = \lim_{n \rightarrow \infty} \text{Var}(\tilde{P}_{ij}^{(1)})/\text{Var}(\hat{P}_{ij}^{(1)})$ . First, we analyze the order of  $\text{Var}(\tilde{P}_{ij}^{(1)})$  in the following theorem.

**Theorem 4.5** Assuming that  $m = O(n^b)$  for any  $b > 0$ , then  $\text{Var}((\hat{Z}_i^T \hat{Z}_j)_{\text{tr}}) = O(m^{-1}n^{-1}(\log n)^3)$ .

The tools we use to prove this theorem is the same as for Corollary 4.3. Combined with Lemma 4.2, we directly have the ARE results as following:

**Theorem 4.6** For fixed  $m$ ,  $1 \leq i, j \leq n$  and  $i \neq j$ ,

$$\frac{\text{Var}(\tilde{P}_{ij}^{(1)})}{\text{Var}(\hat{P}_{ij}^{(1)})} = O(n^{-1}(\log n)^3).$$

Thus

$$\text{ARE}(\hat{P}_{ij}^{(1)}, \tilde{P}_{ij}^{(1)}) = 0.$$

Furthermore, as long as  $m$  goes to infinity of order  $O(n^b)$  for any  $b > 0$ ,

$$\text{ARE}(\hat{P}_{ij}^{(1)}, \tilde{P}_{ij}^{(1)}) = 0.$$

Theorem 4.6 tells us that whenever  $m = O(n^b)$  for any  $b > 0$ , i.e.  $m$  is fixed or it grows not faster than polynomial with respect to  $n$ , the order of the ARE is  $O(n^{-1}(\log n)^3)$ , which converges to 0 when  $n \rightarrow \infty$ . An interesting fact here is that this bound of the ARE does not depend on  $m$ .

As a result, the ASE procedure applied to the entry-wise MLE  $\hat{P}^{(1)}$  helps reduce the variance while keeping the bias asymptotically, leading to a better estimate  $\tilde{P}^{(1)}$  for  $P$  in terms of MSE.

### 4.3 $\hat{P}^{(q)}$ vs. $\tilde{P}^{(q)}$

Similarly, we now analyze the effect of the ASE procedure applied to the entry-wise MLqE  $\hat{P}^{(q)}$  under the contamination model.

**Corollary 4.7** *Assuming that  $m = O(n^b)$  for any  $b > 0$ , then the estimator based on ASE of MLqE has the same entry-wise asymptotic bias as MLqE, i.e.*

$$\lim_{n \rightarrow \infty} \text{Bias}(\tilde{P}_{ij}^{(q)}) = \lim_{n \rightarrow \infty} E[\tilde{P}_{ij}^{(q)}] - P_{ij} = \lim_{n \rightarrow \infty} E[\hat{P}_{ij}^{(q)}] - P_{ij} = \lim_{n \rightarrow \infty} \text{Bias}(\hat{P}_{ij}^{(q)}).$$

The proof for Corollary 4.7 is almost the same as the proof for Corollary 4.3. But unlike results for MLE, we cannot get the factor  $m^{-1/2}$  in the results due to the structure of MLq equation. Although it does not affect the result for asymptotic bias as in Corollary 4.7, the order of the variance of MLqE is missing  $m^{-1}$  compared to MLE as following. Moreover, we will see later that this will cause a slight difference in the comparison.

**Theorem 4.8** *Assuming that  $m = O(n^b)$  for any  $b > 0$ , then  $\text{Var}((\hat{Z}_i^T \hat{Z}_j)_{\text{tr}}) = O(n^{-1}(\log n)^3)$ .*

Combined with Lemma 4.2, we directly have the ARE results as following:

**Theorem 4.9** *For fixed  $m$ ,  $1 \leq i, j \leq n$ ,*

$$\frac{\text{Var}(\tilde{P}_{ij}^{(q)})}{\text{Var}(\hat{P}_{ij}^{(q)})} = O(mn^{-1}(\log n)^3).$$

Thus

$$\text{ARE}(\hat{P}_{ij}^{(q)}, \tilde{P}_{ij}^{(q)}) = 0.$$

Furthermore, as long as  $m$  goes to infinity of order  $o(n^2(\log n)^{-6})$ ,

$$\text{ARE}(\hat{P}_{ij}^{(q)}, \tilde{P}_{ij}^{(q)}) = 0.$$

Theorem 4.9 tells us that whenever  $m = O(n^b)$  for any  $b > 0$ , i.e.  $m$  is fixed or it grows not faster than polynomial with respect to  $n$ , the order of the ARE is  $O(mn^{-1}(\log n)^3)$ . When  $m$  is fixed, the order of the ARE is  $O(n^{-1}(\log n)^3)$ , which will goes to 0 as  $n \rightarrow \infty$ . Even if  $m$  also increases, as long as it grows in the order of  $o(n(\log n)^{-3})$ , the ARE still goes to 0.

Thus the ASE procedure applied to the entry-wise MLqE  $\hat{P}^{(q)}$  also helps reduce the variance while keeping the bias asymptotically, leading to a better estimate  $\tilde{P}^{(q)}$  for  $P$  in terms of MSE.

### 4.4 $\tilde{P}^{(1)}$ vs. $\tilde{P}^{(q)}$

To finish the last piece, we compare the performance between  $\tilde{P}^{(1)}$  and  $\tilde{P}^{(q)}$  without doing any extra work.

**Theorem 4.10** *For sufficiently large  $n$  and  $C$ , any  $1 \leq i, j \leq n$ ,*

$$\lim_{m \rightarrow \infty} \text{Bias}(\tilde{P}_{ij}^{(1)}) > \lim_{m \rightarrow \infty} \text{Bias}(\tilde{P}_{ij}^{(q)})$$

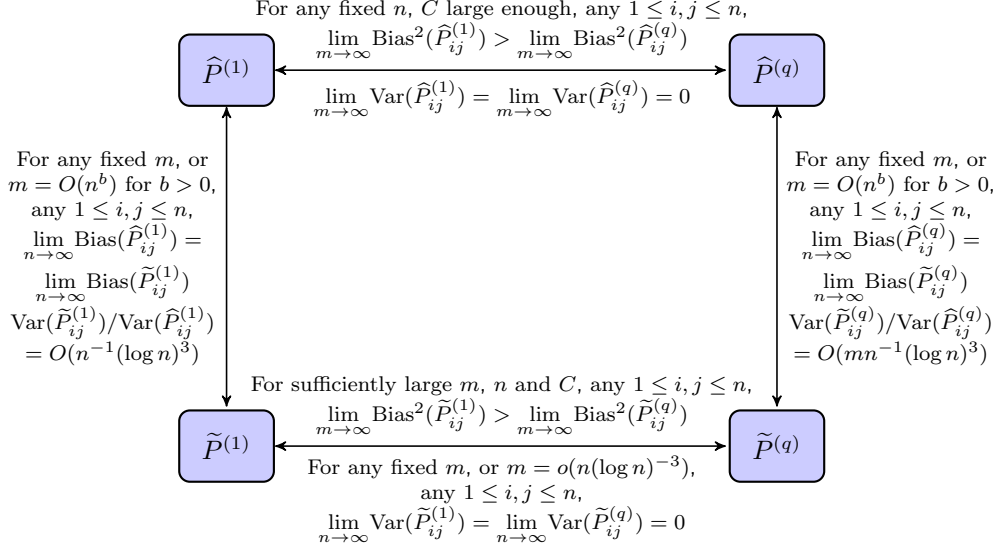


Figure 2: Relationship among four estimators.

**Theorem 4.11** For any fixed  $m$ , any  $1 \leq i, j \leq n$ ,

$$\lim_{n \rightarrow \infty} \text{Var}(\tilde{P}_{ij}^{(1)}) = \lim_{n \rightarrow \infty} \text{Var}(\tilde{P}_{ij}^{(q)}) = 0.$$

Furthermore, as long as  $m$  goes to infinity of order  $o(n^2(\log n)^{-6})$ , any  $1 \leq i, j \leq n$ ,

$$\lim_{n \rightarrow \infty} \text{Var}(\tilde{P}_{ij}^{(1)}) = \lim_{n \rightarrow \infty} \text{Var}(\tilde{P}_{ij}^{(q)}) = 0$$

Theorem 4.10 is a direct result of Lemma 4.1, Corollary 4.3, and Corollary 4.7, while Theorem 4.11 is based on Lemma 4.2, Theorem 4.6, and Theorem 4.9.

So  $\tilde{P}^{(q)}$  inherits the robustness from the entry-wise MLqE  $\hat{P}^{(q)}$  and has a smaller asymptotic bias compared to  $\tilde{P}^{(1)}$  while both estimates have variance goes to 0 as  $m \rightarrow \infty$ .

## 4.5 Summary

We summarize all the comparison in this section and we plot the relationship among four estimators in Figure 2. In conclusion, when contamination is relatively large and the number of graphs  $m$  is not growing too fast,  $\tilde{P}^{(q)}$  is the best among the four estimators.

## 5 Extensions

Although in Section 4, we only present the results under exponential distributions, the results can be generalized to a broader class of distribution families, and even a different entry-wise robust estimator other than MLqE with the following conditions:

1. Let  $A_{ij} \stackrel{ind}{\sim} (1-\epsilon)f_{P_{ij}} + \epsilon f_{C_{ij}}$ , then  $E[(A_{ij} - E[\hat{P}_{ij}^{(1)}])^k] \leq \text{const} \cdot k!$ , where  $\hat{P}^{(1)}$  is the entry-wise MLE as defined before;  
This is to ensure the that observations will not deviate from the expectation too far away, such that the concentration inequality can apply.
2. There exists  $C_0(P_{ij}, \epsilon) > 0$  such that under the contaminated model with  $C > C_0(P_{ij}, \epsilon)$ ,

$$\lim_{m \rightarrow \infty} |E[\hat{P}_{ij}] - P_{ij}| < \lim_{m \rightarrow \infty} |E[\hat{P}_{ij}^{(1)}] - P_{ij}|;$$

This condition is discussed in Section 4.1. It requires the contamination of the model to be large enough (a restriction on the distribution) and  $\hat{P}$  to be robust enough with respect to the contamination (a condition on the estimator).

3.  $\hat{P}_{ij} \leq \text{const} \cdot \hat{P}_{ij}^{(1)}$ ; (This might be generalized to with high probability later)  
Since we use the results of  $\hat{P}^{(1)}$  to bound  $\hat{P}^{(q)}$ , the proof can apply directly with this condition for an arbitrary  $\hat{P}$ .
4.  $\text{Var}(\hat{P}_{ij}) = O(m^{-1})$ , where  $m$  is the number of observations.  
We will get exactly the same results as in Section 4. However, even if the variance of the new estimator is not of order  $O(m^{-1})$ , we will get similar results with a different term related to  $m$ .

## 6 Empirical Results

### 6.1 Simulation Results

- In this section, we will illustrate the theoretical results of four estimators in Section 4 via various Monte Carlo simulation experiments.

#### 6.1.1 Simulation Setting

- Here we consider the 3-block SBM with respect to the exponential distributions parameterized by

$$B = \begin{bmatrix} 4.2 & 2 \\ 2 & 7 \end{bmatrix}, \quad \rho = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}.$$

- The contamination is also a 3-block SBM with respect to the exponential distributions parameterized by

$$B' = \begin{bmatrix} 20 & 18 \\ 18 & 25 \end{bmatrix}, \quad \rho = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}.$$

- And we embed the graphs into the dimension  $d = \text{rank}(B) = 2$ .

### 6.1.2 Simulation Results

- Figure 3 plots the mean squared error in average by varying contamination ratio  $\epsilon$  with fixed  $n = 100$  and  $m = 10$  based on 1000 Monte Carlo replicates. And we use  $q = 0.8$  when applying MLqE. Different colors represent the simulated MSE associated with four different estimators. **1. MLE  $\hat{P}^{(1)}$  vs MLqE  $\hat{P}^{(q)}$ :** MLE outperforms a little bit when there is no contamination (i.e.  $\epsilon = 0$ ), but it degrades dramatically when contamination increases; **2. MLE  $\hat{P}^{(1)}$  vs ASE o MLE  $\tilde{P}^{(1)}$ :** ASE procedure takes the low rank structure into account and  $\tilde{P}^{(1)}$  wins the bias-variance tradeoff; **3. MLqE  $\hat{P}^{(q)}$  vs ASE o MLqE  $\tilde{P}^{(q)}$ :** MLqE preserves the low rank structure of the original graph more or less, so ASE procedure still helps and  $\tilde{P}^{(q)}$  wins the bias-variance tradeoff; **4. ASE o MLqE  $\tilde{P}^{(q)}$  vs ASE o MLE  $\tilde{P}^{(1)}$ :** When contamination is large enough,  $\tilde{P}^{(q)}$  based on MLqE is better, since it inherits the robustness from MLqE.
- Figure 4 show the mean squared error in average by varying the parameter  $q$  in MLqE with fixed  $n = 100$ ,  $m = 10$  and  $\epsilon = 0.2$  based on 1000 Monte Carlo replicates. Different colors represent the simulated MSE associated with four different estimators. 1. ASE procedure takes advantage of the graph structure and improves the performance of the corresponding estimators; 2. MLqE shows the robustness property compare to the MLE. And as  $q$  goes to 1, MLqE goes to the MLE as expected.
- By comparing the performance of the four estimators based on different setting, we demonstrate the theoretical results in Section 4.

## 6.2 CoRR Graphs

- In practice, the graphs may not perfectly follow an RDPG, or even not IEM. But we are still interested in the performance of the four estimators. In this work, we examine the structural connectomic data. The graphs are based on diffusion tensor MR images. It contains 114 different brain scans, each of which was processed to yield an undirected, weighted graph with no self-loops, using the m2g pipeline described in Kiar et al. [2016]. The vertices of the graphs represent different regions in the brain defined according to an atlas. We used the desikan atlas with 70 vertices. The weight of an edge between two vertices represents the number of white-matter tract connecting the corresponding two regions of the brain.
- We first check if the dataset has the low-rank property. In the left panel of Fig. 5, we plot the eigenvalues of the mean graph of all 114 graphs (with diagonal augmentation) in decreasing algebraic order for the Desikan atlases based on the m2g pipeline. The eigenvalues first decrease dramatically and then stay around 0 for a large range of dimensions. In addition, we also plot the histograms in the right panel of Fig. 5. From the figures we can see many eigenvalues are around zero, with a few large eigenvalues. So the information is mostly contained in the first few dimensions. Such quasi low-rank property could be used.



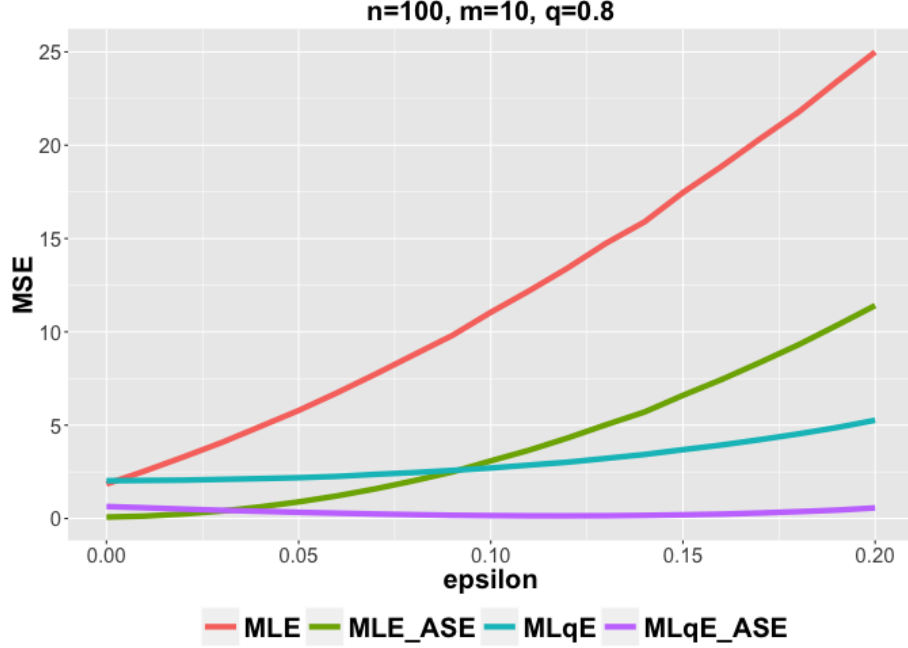


Figure 3: Mean squared error in average by varying contamination ratio  $\epsilon$  with fixed  $n = 100$  and  $m = 10$  based on 1000 Monte Carlo replicates. And we use  $q = 0.8$  when applying MLqE. Different colors represent the simulated MSE associated with four different estimators. **1.  $\text{MLE } \hat{P}^{(1)}$  vs  $\text{MLqE } \hat{P}^{(q)}$ :** MLE outperforms a little bit when there is no contamination (i.e.  $\epsilon = 0$ ), but it degrades dramatically when contamination increases; **2.  $\text{MLE } \hat{P}^{(1)}$  vs ASE  $\circ$   $\text{MLE } \tilde{P}^{(1)}$ :** ASE procedure takes the low rank structure into account and  $\tilde{P}^{(1)}$  wins the bias-variance tradeoff; **3.  $\text{MLqE } \hat{P}^{(q)}$  vs ASE  $\circ$   $\text{MLqE } \tilde{P}^{(q)}$ :** MLqE preserves the low rank structure of the original graph more or less, so ASE procedure still helps and  $\tilde{P}^{(q)}$  wins the bias-variance tradeoff; **4. ASE  $\circ$   $\text{MLqE } \tilde{P}^{(q)}$  vs ASE  $\circ$   $\text{MLE } \tilde{P}^{(1)}$ :** When contamination is large enough,  $\tilde{P}^{(q)}$  based on MLqE is better, since it inherits the robustness from MLqE.

- To compare the four estimators, we need to know the true parameter matrix  $P$ . However,  $P$  is definitely not obtainable in practice since the 114 graphs are also a sample from the population. In this work, we are going to use two different approaches to approximate the probability matrix  $P$ .
- First, we assume that the true probability matrix  $P$  has the low-rank property. In this case, the low-rank approximation of the mean graph of all 114 graphs is a good estimate of the probability matrix  $P$ . We admit that this setting is preferable by the low-rank estimators. However, it is illustrative to see how the estimators perform in this case.
- In the above setting, we run 100 simulations on the dataset for each sample size  $m = 2$ ,  $m = 5$ , and  $m = 10$ . Specifically, for each Monte Carlo replicate, we sample  $m$  graphs out of the 114 from the m2g dataset and

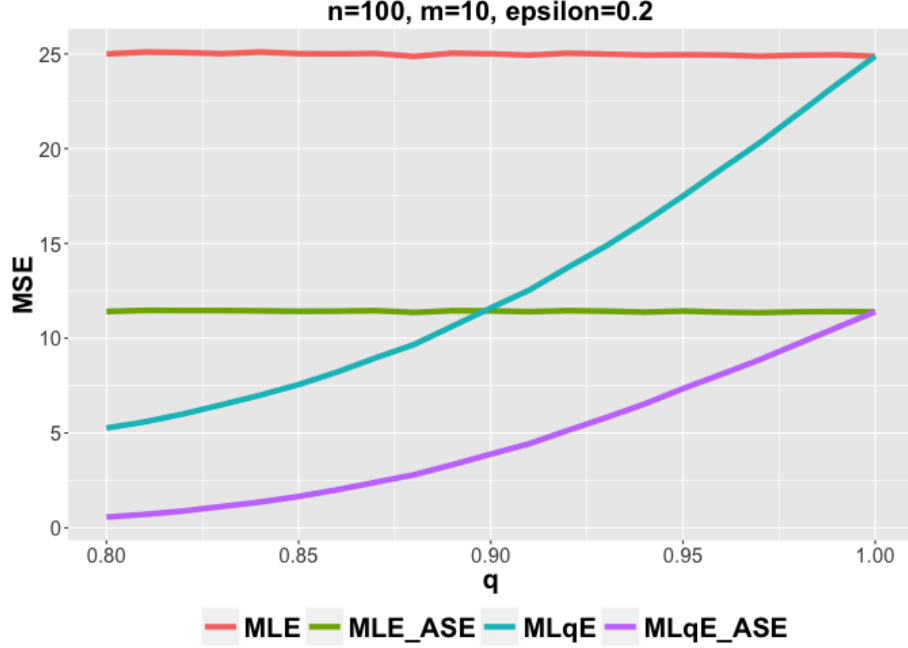


Figure 4: Mean squared error in average by varying the parameter  $q$  in  $MLqE$  with fixed  $n = 100$ ,  $m = 10$  and  $\epsilon = 0.2$  based on 1000 Monte Carlo replicates. Different colors represent the simulated MSE associated with four different estimators. 1. ASE procedure takes advantage of the graph structure and improves the performance of the corresponding estimators; 2.  $MLqE$  shows the robustness property compare to the MLE. And as  $q$  goes to 1,  $MLqE$  goes to the MLE as expected.

computing the four estimates based on the  $m$  sampled graphs. We then compared these estimates to the low-rank approximation of the mean of all 114 graphs from the m2g dataset. For those two low-rank estimators, we apply ASE for all possible dimensions, i.e.  $d$  ranges from 1 to  $n$ . The result is shown in Figure 6.

- Since the approximated  $P$  contains outliers, we don't expect  $MLqE$  helps much as shown in the figure. However, since the estimated  $P$  has the low-rank property, ASE procedure helps improving the performance. The dimensions chosen by the 3d elbow of Zhu and Ghodsi are denoted in triangle and square. We can see it does a good job. More importantly, a wide range of dimensions could lead to an improvement.
- Also, we consider a setting which is favorable to the MLE, i.e. assuming that the probability matrix  $P$  has a high-rank property. Since we know ndmg2 is a better pipeline compared to m2g, the mean graph derived from ndmg2 should be a relative accurate estimate of the actual population mean graph. Since the mean of 114 graphs has full rank, it breaks the low-rank assumptions and are favorable to the MLE. In order to evaluate the performance of the four estimators, we build estimates based on the

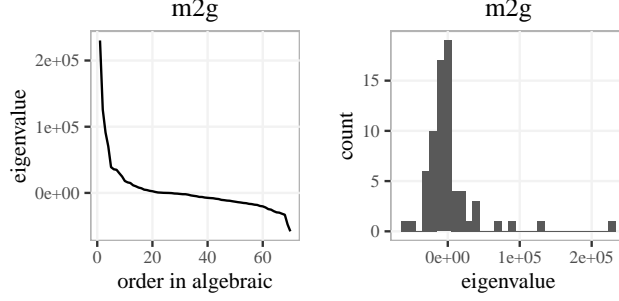


Figure 5: **Screeplot and the histogram of the eigenvalues of the mean of 114 graphs based on m2g pipeline.** The screeplot in the left panel shows the eigenvalues of the mean graph of all 114 graphs with diagonal augmentation in decreasing algebraic order for the Desikan atlas. The right panel shows the histogram of the eigenvalues of the mean graph of all 114 graphs with diagonal augmentation. Many eigenvalues are around zero, which lead to a quasi low-rank structure.

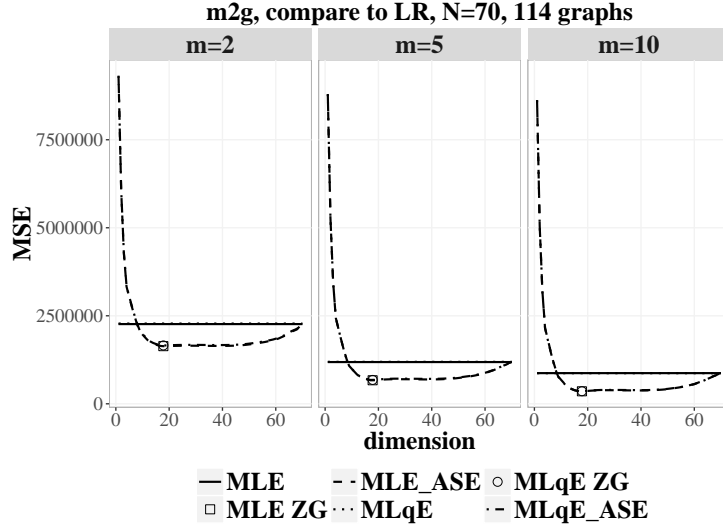


Figure 6: Comparison of mean squared error in average among four estimators while embedding the graphs into different dimensions with different size  $m$  of the subsamples by using the low-rank approximation of the mean of the rest  $114 - m$  graphs as the approximation of  $P$ . Four estimators are represented by different line styles as explained in the legend. Since the approximated  $P$  contains outliers, we don't expect MLqE helps much as shown in the figure. However, since the estimated  $P$  has the low-rank property, ASE procedure helps improving the performance. The dimensions chosen by the 3rd elbow of Zhu and Ghodsi are denoted in triangle and square. We can see it does a good job. More importantly, a wide range of dimensions could lead to an improvement.

samples with size  $m$  from the m2g dataset, while using the sample mean of all 114 graphs from the ndmg2 dataset as an estimate of the probability matrix  $P$ .

- We run 100 simulations on the dataset for each sample size  $m = 2$ ,  $m = 5$ , and  $m = 10$ . Specifically, for each Monte Carlo replicate, we sample  $m$  graphs out of the 114 from the m2g dataset and computing the four estimates based on the  $m$  sampled graphs. We then compared these estimates to the mean of all 114 graphs in the ndmg2 dataset. For those two low-rank estimators, we apply ASE for all possible dimensions, i.e.  $d$  ranges from 1 to  $n$ . The result is shown in Figure 7.
- Since it is real data, MLqE outperforms MLE because of the robustness property. Moreover, as suggested in the previous theorems, such property is kept after the ASE procedure.
- When  $d$  is small, ASE procedure underestimates the dimension and fails to get important information, which leads to poor performance. In practice, we use algorithms like Zhu and Ghodsi’s method to select the dimension  $d$ . We can see Zhu and Ghodsi’s algorithm does a pretty good job for selecting the dimension to embed.
- For all three sample sizes ( $m = 2$ ,  $m = 5$ ,  $m = 10$ ), MLE and MLqE have large variances which lead to large MSE. Meanwhile, the ASE procedure reduces the variance by taking advantages of the graph structure.

## 7 Discussion

## 8 Appendix

All proofs here.

## References

- P.J. Bickel and K.A. Doksum. *Mathematical Statistics: Basic Ideas and Selected Topics*. Number v. 1 in Holden-Day series in probability and statistics. Prentice Hall, 2001. ISBN 9780138503635. URL <https://books.google.co.uk/books?id=8poZAQAIAAJ>.
- Béla Bollobás, Svante Janson, and Oliver Riordan. The phase transition in inhomogeneous random graphs. *Random Structures & Algorithms*, 31(1):3–122, 2007.
- Sourav Chatterjee et al. Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214, 2015.
- Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

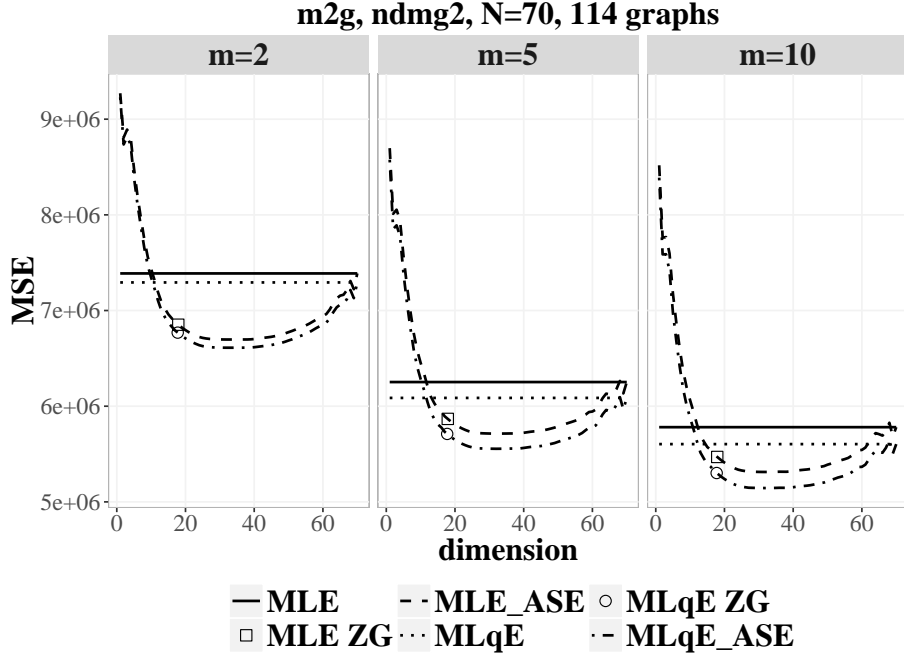


Figure 7: **Comparison of MSE of the four estimators for the desikan atlases at three sample sizes based on m2g and ndmg2 pipelines.** 1. **MLE (horizontal solid line) vs MLqE (horizontal dotted line):** MLqE outperforms MLE since robust estimators are always preferred in practice; 2. **MLE (horizontal solid line) vs MLE\_ASE (dashed line):** MLE\_ASE wins the bias-variance tradeoff when embedded into a proper dimension; 3. **MLqE (horizontal dotted line) vs MLqE\_ASE (dashed dotted line):** MLqE\_ASE wins the bias-variance tradeoff when embedded into a proper dimension; 4. **MLqE\_ASE (dashed dotted line) vs MLE\_ASE (dashed line):** MLqE\_ASE is better, since it inherits the robustness from MLqE. And the square and circle represent the dimensions selected by the Zhu and Ghodsi method. We can see it does a pretty good job. And more importantly, a wide range of dimensions could lead to an improvement.

- Davide Ferrari and Yuhong Yang. Maximum lq-likelihood estimation. *Ann. Statist.*, 38(2):753–783, 04 2010. doi: 10.1214/09-AOS687. URL <http://dx.doi.org/10.1214/09-AOS687>.
- Cedric E Ginestet, Prakash Balanchandran, Steven Rosenberg, and Eric D Kolarczyk. Hypothesis testing for network data in functional neuroimaging. *arXiv preprint arXiv:1407.5525*, 2014.
- Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the american Statistical association*, 97(460):1090–1098, 2002.
- Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.
- G Kiar, W Gray Roncal, D Mhembere, E Bridgeford, R Burns, and J Vogelstein. ndmg: Neurodata’s mri graphs pipeline, 2016.
- R. S. H. Mah and A. C. Tamhane. Detection of gross errors in process data. *AICHE Journal*, 28(5):828–830, 1982. ISSN 1547-5905. doi: 10.1002/aic.690280519. URL <http://dx.doi.org/10.1002/aic.690280519>.
- David Marchette, Carey Priebe, and Glen Coppersmith. Vertex nomination via attributed random dot product graphs. In *Proceedings of the 57th ISI World Statistics Congress*, volume 6, page 16, 2011.
- Christine Leigh Myers Nickel. *Random dot product graphs: A model for social networks*, volume 68. 2007.
- Yichen Qin and Carey E Priebe. Maximum l q-likelihood estimation via the expectation-maximization algorithm: A robust estimation of mixture models. *Journal of the American Statistical Association*, 108(503):914–928, 2013.
- Edward R Scheinerman and Kimberly Tucker. Modeling graphs using dot product representations. *Computational Statistics*, 25(1):1–16, 2010.
- Robert Serfling. Asymptotic relative efficiency in estimation. In *International encyclopedia of statistical science*, pages 68–72. Springer, 2011.
- Daniel L Sussman, Minh Tang, and Carey E Priebe. Consistent latent position estimation and vertex classification for random dot product graphs. *IEEE transactions on pattern analysis and machine intelligence*, 36(1):48–57, 2014.
- Runze Tang, Michael Ketcha, Joshua T Vogelstein, Carey E Priebe, and Daniel L Sussman. Law of large graphs. *arXiv preprint arXiv:1609.01672*, 2016.
- Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12(4):389–434, 2012.
- G. V. Trunk. A problem of dimensionality: A simple example. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (3):306–307, 1979.
- Stephen J Young and Edward R Scheinerman. Random dot product graph models for social networks. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 138–149. Springer, 2007.

Mu Zhu and Ali Ghodsi. Automatic dimensionality selection from the scree plot via the use of profile likelihood. *Computational Statistics & Data Analysis*, 51(2):918–930, 2006.