MATH 690: Topics in Probablity Theory September 14, 19, 2017

Dimension Reduction

Lecturer: Xiuyuan Cheng *Scribe: Fred Zhang*

1 Introduction

In this lecture, we continue to discuss covariance estimation. We observe the Stein's phenomenon when MLE is suboptimal and develop the methods of *spectral shrinkage* and covariance thresholding. We then turn to dimension reduction and consider the *Multidimensional Scaling* (MDS) problem, which is equivalent of PCA.

2 Stein's Phenomenon

Let us wrap up our previous discussion on the Stein's phenomenon. Suppose we a one dimensional random variable $y \sim \mathcal{N}(\mu, 1)$; that is $p = 1$, $n = 1$. Given only one observation, we want to estimate the mean μ. It is easy to see that the Maximum Likelihood Estimator is

$$\widehat{\mu}^{MLE}(y) = y.$$

Can we construct better ones, and how do we evaluate an estimator? For that purpose we consider the *Mean Squared Error* (MSE) of an estimator, defined as

$$\mathbb{E}\,(\widehat{\mu} - \mu)^2 = \underbrace{\mathbb{E}\,(\widehat{\mu} - \mu)^2}_{\text{variance}} + \underbrace{(\mathbb{E}\,\widehat{\mu} - \mu)^2}_{\text{bias}^2}.$$

Notice that the first term is the variance and the second is the bias squared (of the estimator). Now based on this definition, we see that the MSE of the MLE is

$$\mathbb{E}\left(\widehat{\mu}^{MLE} - \mu\right)^2 = 1 + 0 = 1.$$

Can we have another estimator that has less MSE? Let us consider $\widehat{\mu}(u) = \rho \cdot y$ for some ρ to be determined. Then its MSE is

$$\mathbb{E}\,(\widehat{\mu} - \mu)^2 = \rho^2 + (\rho - 1)^2\mu^2.$$

Here, we calculate that the minimizer of the RHS, a quadratic polynomial in ρ, is

$$\rho^* = \frac{\mu^2}{1 + \mu^2}.$$

Hence, MLE is suboptimal in terms of the MSE. To construct a better estimator, we should, in this case, *shrink* it by some factor ρ, although its optimal choice can be unknown to us. The question is how we should do this in general setting.

2.1 Spectral Shrinkage

In covariance estimation of high dimensions, what we shall consider is the *spectral shrinkage* on the eigenvectors. Last time we saw the BBP phase transition of the spiked model in rank-1 case, where if R is the signal-noise ratio, then the largest eigenvalue converges to $1 + R$. The transition threshold occurs at $\sqrt{\gamma}$, where $\gamma = p/n$.

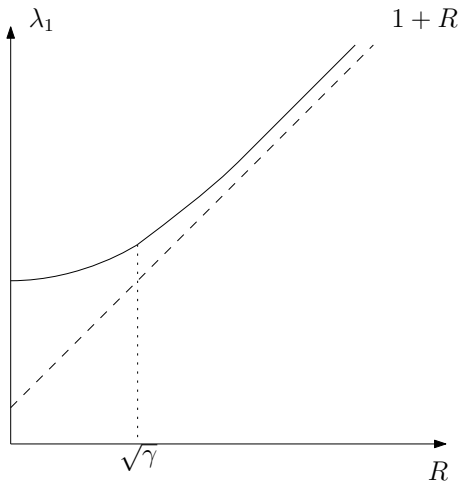

Figure 1: The largest eigenvalue starts to approach to $1 + R$ at $R = \sqrt{\gamma}$.

In general, eigenvectors corresponding to eigenvalues below certain threshold come from noise. Now consider the eigendecomposition of the covariance matrix $\mathbf{S_y} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. We intend to estimate the population covariance matrix by applying a *shrinkage function* f:

$$\widehat{\mathbf{\Sigma}} = \mathbf{V}f(\mathbf{\Lambda})\mathbf{V}^T = \sum_{i=1}^{p} f(\mathbf{\Lambda_i})\mathbf{v_i}\mathbf{v_i}^T.$$

Here, we would choose some thresholding function f that filters out small eigenvalues; *e.g.*,

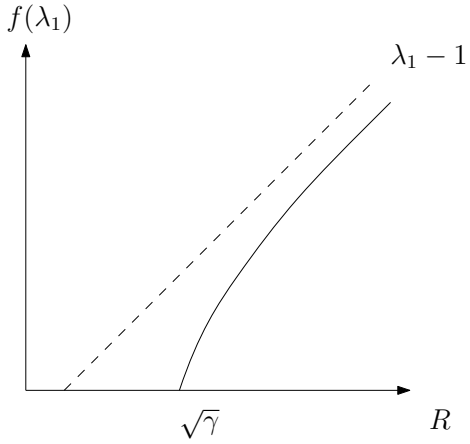

Figure 2: Shrinkage function.

2.2 Thresholding Sample Covariance

Now we introduce another method for denoising the sample covariance and estimate the population covariance. Let $\mathbf{S} = \frac{1}{n}\sum_i \mathbf{y_i}\mathbf{y_i}^T$ be the sample covariance. Then simply apply a thresholding function:

$$\widehat{\Sigma}_{ij} = Tr(S_{ij}).$$

Here, the function $Tr(\cdot)$ is

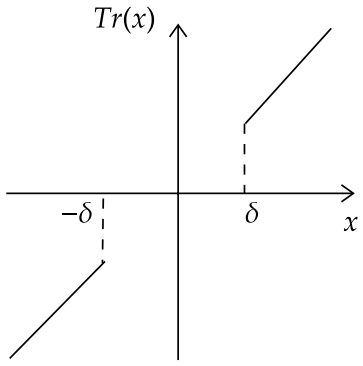

Figure 3: Thresholding function $Tr(x)$.

That is, we filter out entries of small absolute values.

Excercise 1. Let $y_i \sim \mathcal{N}(0,1)$ be i.i.d. Gaussian samples. After applying the thresholding function with $\delta = \frac{2\log n}{\sqrt{n}}$ on the sample covariance, non-diagonal entries converge to $\mathcal{N}(0, c/n)$ as $n \to \infty$, for some constant c. Roughly speaking, only diagonal entries remain.

For more on the thresholding method and its theoretical justification, see [BL08]. This is also related to the whole area of Sparse PCA; see [JL09].

3 Introduction to Dimension Reduction

Given $\{\mathbf{x_i} \in \mathbb{R}^D\}_{i=1}^n$. For the purpose of faster algorithms, visualization, *etc.*, we would like to to have another set of points $\{\mathbf{y_i} \in \mathbb{R}^d\}_{i=1}^n$ with $d \ll D$. We want that the "geometry" of $\{\mathbf{x_i}\}$ are as much preserved as possible in $\{\mathbf{y_i}\}$. Can we do it in general, say, to preserve distances between $\{\mathbf{x_i}\}$?

Example 1. Suppose $D = 3$, $d = 2$ and $n = 3$. Take the hyperplane the three points lie on. It is easy to see that we can embed them in $\mathbb{R}^2$ without distorting pairwise distances.

Example 2. Suppose $D = 3$, $d = 2$ and $n = 4$. If the four points do not lie on the same hyperplane, then it is not possible to preserve the pairwise distances in $\mathbb{R}^2$.

Hence, it not possible to always preserve metrics in reducing dimensions. However, we also observed that if the data lie on a subspace, then finding the subspace can give us a low dimension embedding. Next, we will discuss how to do this under Euclidean distances.

4 Multidimensional Scaling (MDS)

Given $\mathbf{X} = [\mathbf{x_1}, \ldots, \mathbf{x_n}]_{D\times n}$, we would like to find $\mathbf{Y} = [\mathbf{y_1}, \ldots, \mathbf{y_n}]_{d\times n}$ that minimize the discrepancies $L(Y)$

$$\min L(\mathbf{Y}) = \sum_{i,j} \left(\left\|\mathbf{y_i} - \mathbf{y_j}\right\|^2 - \underbrace{\left\|\mathbf{x_i} - \mathbf{x_j}\right\|^2}_{D_{ij}} \right)^2 . \tag{1}$$

Without loss of generality, assume that input data are centered at origin (*i.e.*, $\sum \mathbf{x_i} = \mathbf{0}$ and $\mathbf{X} \cdot \mathbf{1} = \mathbf{0}$). Let

$$D_{ij} = \left\|\mathbf{x_i} - \mathbf{x_j}\right\|^2 = \|\mathbf{x_i}\|^2 + \|\mathbf{x}\|_j^2 - 2\mathbf{x_i}^T\mathbf{x_j}.$$

Then it can be expressed as

$$\mathbf{D} = \boldsymbol{\xi}\mathbf{1}^T + \mathbf{1}\boldsymbol{\xi}^T - 2\boldsymbol{G},$$

where $\mathbf{G} = \mathbf{X}^T\mathbf{X}$ is the Gramian and $\boldsymbol{\xi} = \left[\|\mathbf{x_i}\|^2, \ldots, \|\mathbf{x_n}\|^2\right]^T$. In general, the input is not necessarily $\mathbf{X}$ but only the distance matrix $\mathbf{D}$. This problem is in fact equivalent of PCA. However, one can still compute the Gramian without not knowing the data coordinates $\mathbf{X}$ by *double centering*.

Let $\boldsymbol{H} = \boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$. We notice that since $\boldsymbol{H}\cdot\mathbf{1} = \mathbf{0}$, $\boldsymbol{HGH} = \mathbf{H}(\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{H} = G$. Then

$$-\frac{1}{2}\boldsymbol{HDH} = \boldsymbol{G}.$$

We obtain $\boldsymbol{Y}$, we compute $\widetilde{\mathbf{G}}$ by applying PCA on $\boldsymbol{G}$ using the top k eigenvectors. It turns out that setting $\boldsymbol{Y} = \boldsymbol{S}_{1:d,1:d}\boldsymbol{V}_{:,1:d}$ minimizes the objective (1).

5 Isomap

Given $\mathbf{X} = [\mathbf{x_i}]_{D\times n}$, we want to embed them in lower dimension by finding $\mathbf{Y} = [\mathbf{y_i}]_{d\times n}$.

Step 1: Graph construction. Construct a k-NN graph such that $A_{ij} = 1$ if $i \in N(j)$ and (or) $j \in N(i)$ and $A_{ij} = 0$ otherwise. Here, $N(i)$ is a list of k nearest neighbors in Euclidean distance. As an alternative, one constructs ϵ-graph where $A_{ij} = 1$ iff $\left\|\boldsymbol{x_i} - \boldsymbol{x_j}\right\| < \epsilon$.

Step 2: Compute graph distance. Compute shortest-path distance d_{ij} for all pairs i, j in the graph

Step 3: Classical MDS. Compute $D_{ij} = d_{ij}^2$, the Gramian $\boldsymbol{G} = -\frac{1}{2}\boldsymbol{HDH}$, then compute the top d eigenvectors of G.

Remark 1. If the data sit on some low-dimensional manifold, under certain assumptions (*e.g.*, on sampling of the points), the shortest path distance we compute approximates the geodesic distance on the manifold.

Remark 2. The decay of the eigenvalues of G can indicate the intrinsic dimension of the data.

6 Locally Linear Embedding (LLE)

Step 1: Construct the $N(i)$ list of k-NN for all points.

Step 2: Solve least-square optimization problem

$$\min_{d_j^{(i)}} \left\| x_i - \sum_{j \in N(i)} d_j^{(i)} x_j \right\|^2$$

such that $\sum_{j \in N(i)} d_j^{(i)} = 1$.

Step 3: Compute

$$W_{ij} = \begin{cases} d_j^{(i)} & j \in N(i) \\ 0 & \text{otherwise} \end{cases}$$

and

$$\min_{\boldsymbol{Y}} \sum_i \left\| y_i - \sum_j W_{ij} y_j \right\|^2$$

such that $\boldsymbol{Y}\mathbf{1} = \mathbf{0}$ and $\boldsymbol{Y}^T\boldsymbol{Y} = \boldsymbol{I}_d$. This turns out to be an eigvenvalue problem on $\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{W})^T(\boldsymbol{I} - \boldsymbol{W})$.

Remark 3. The matrix $\boldsymbol{M}$ approximates the graph Laplacian $\boldsymbol{L}$ of the k-NN graph.

7 Laplacian Eigenmap

Step 1: Construct the affinity matrix $\boldsymbol{W}$ where

$$W_{ij} = e^{\frac{-\left\|x_i - x_j\right\|^2}{\epsilon}}.$$

As a simpler alternative, let W be the adjacency matrix by the k-NN graph or ϵ-graph

Step 2: Compute the graph Laplacian $\boldsymbol{L} = \mathbf{D} - \boldsymbol{W}$.

Step 3: Compute the eigendecomposition of $\boldsymbol{L_{rw}} = \boldsymbol{D}^{-1}\boldsymbol{L}$:

$$\boldsymbol{L\Phi} = \boldsymbol{\lambda D\Phi}$$

Then $\mathbf{Y} = [\Phi_2, \ldots, \Phi_{d+1}]$, eigvenvectors corresponding to the 2nd and $(d+1)$st smallest eigvenvalue.

Note

$$\mathbf{L_{rw}} = \mathbf{D^{-1}}(\mathbf{D} - \mathbf{W}) = \boldsymbol{I} - \underbrace{\boldsymbol{D}^{-1}\boldsymbol{W}}_{\boldsymbol{P}}.$$

Here $P_{ij} = W_{ij}/d_i$. We notice that $\mathbf{P}$ is a probablisitic matrix, and this defines a random walk on the graph. The eigenvalues of $\boldsymbol{P}$ are in $[-1, 1]$. The largest eigenvalue is $\lambda_1 = 1$ is of multiplicity 1 if the graph is connected and -1 is not an eigenvalue if the random walk has not cylces.

Excercise 2. What is the stationary distribution of $\boldsymbol{P}$?

References

[BL08] Peter J Bickel and Elizaveta Levina. Covariance regularization by thresholding. *The Annals of Statistics*, pages 2577–2604, 2008.

[JL09] Iain M Johnstone and Arthur Yu Lu. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 104(486):682–693, 2009.