MAS 557 - KIJUNG JEON

A spectral algorithm for learning mixture models

1 Introduction

1.1 Notation & Question

1. Mixture distribution : $F = w_1F_1 + W_2F_2 + ... + w_kF_k$
2. mixing weight : $0 \leq w_i \leq 1$ satisfying $\sum_{i=1}^{k} w_i = 1$ Note : $w_{min} := min(w_1, ..., w_k)$
3. component distribution : $F_i \sim MN(\mu_i, \sigma_i^2 I_n)$ for all i
4. Sample matrix : $A \in \mathbb{R}^{m \times n}$, where m : sample size, n : dimension of sample
5. Separation between F_i and $F_j := ||\mu_i - \mu_j||$
6. Radius of $F_i := \sigma_i \sqrt{n}$

Note : Let $\mathbb{X} \sim MN(0, \sigma^2 I_n)$, then $E[X_1^2 + ... + X_n^2] = n\sigma^2$
$\Rightarrow$ Radius of spherical Gaussian represents expected distance from center

Question : How well can we cluster samples from F without knowing μ_i, σ_i in a efficient way?

1.2 Proposed Spectral Algorithm

Algorithm 1 Spectral algorithm in a mixture spherical Gaussian model

```
STEP 1. Get top $k$ right singular vectors of sample matrix $A$ using SVD
STEP 2. Project each samples to the rank $k$ subspace spanned by the top $k$ right singular vectors (same as PCA)
STEP 3. Perform distance-based classification (clustering) in the $k$-dimensional space.
```

2 Performance guarantee : PCA

On STEP2, we project each samples to the rank k subspace spanned by the top k right singular vectors. Although there is no guarantee that this projection keeps the samples still separated (in general), Author claims the followings:

1. Subspace spanned by $\mu_1, ..., \mu_k$ is expected best subspace. [Corollary 2.1]

2. Subspace spanned by top k right singular vectors is a good approximation of subspace spanned by $\mu_1, ..., \mu_k$. [Corollary 2.2]

3. So the separation between mean vectors will be preserved, and the radius is decreased by $\sqrt{\frac{k}{n}}$ [Observation 6.1]. Therefore, we can use this subspace to cluster.

Fact : On random projection on to k dimension subspace, the intercenter distances and the radii decrease at the same rate, namely $\sqrt{\frac{k}{n}}$, which makes the separation condition gets worse. However, the projection using PCA does not worsen separation condition.

Corollary 2.1. (*Expected best subspace*) [*3*]
Let $V \subset \mathbb{R}^n$ be the k-dimensional subspace spanned by the top k right singular vectors and let U be the subspace spanned by the mean vectors $\mu_1, ..., \mu_k$. Then

$$E[||proj_U A||^2] \geq E[||proj_V A||^2]$$

Note : $||proj_U A||^2 := \sum_{i=1}^{m} ||proj_U A_i||^2$, where A_i = ith row of A

Corollary 2.2. (*Likely Best Subspace*) [*3*]
Let $\mu_1', ..., \mu_k'$ be projections of $\mu_1, ..., \mu_k$ onto subspace spanned by the top k right singular vectors of the sample matrix A. With a sample size $m > \frac{1000}{\epsilon^2 w_{min}}(n \log(\frac{n}{\epsilon} + \max_i \frac{||\mu_i^2||}{\epsilon \sigma_i^2}) + \frac{1}{n-k} \log \frac{1}{\delta})$, we have, with high probability

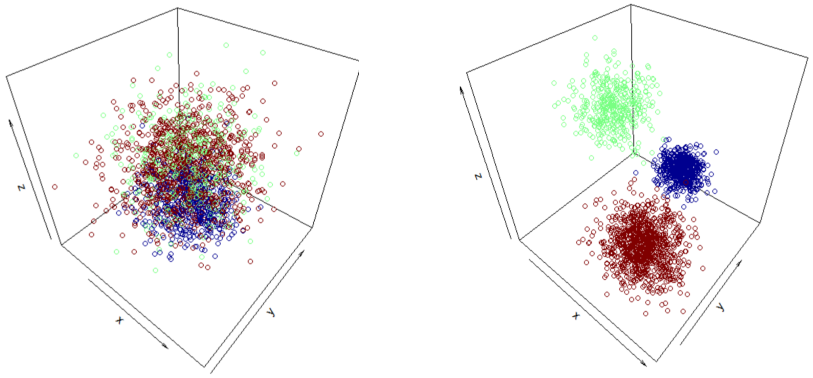
$$\sum_{i=1}^{k} w_i(||\mu_i||^2 - ||\mu_i'||^2) \leq \epsilon(n-k) \sum_{i=1}^{k} w_i \sigma_i^2$$

By corollary 2.2, With high probability the subspace spanned by the top k right singular vectors lies very close to subspace spanned by the mean vectors $(\mu_1, ..., \mu_k)$ using the fact that $||\mu_i - \mu_i'||^2 = ||\mu_i||^2 - ||\mu_i'||^2$. Also, by Observation 6.1, the radius is decreased by $\sqrt{\frac{k}{n}}$, since we use k-dimensional subspace. Therefore, We can guarantees performance projection in mixture of spherical Gaussian model.

Question: Does the proposed algorithm really works well in real worlds?

3 Empirical verification : PCA

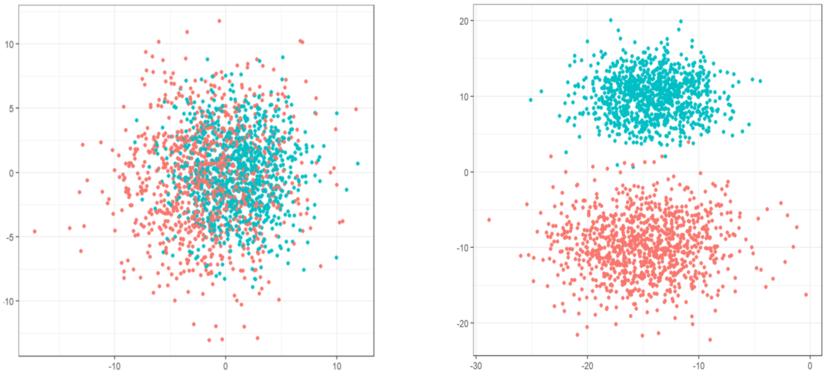
Empirical setting :
1. Basic parameter setting: n = 100, k = 3 or 2, m = 2000
2. Means and variances are randomly sampled from specific distributions and fixed
(In this experiment : $\mu_1, ..., \mu_k \sim MN(0, I_n), \quad \sigma_1, ..., \sigma_k \sim possion(3) + 1$)
3. Weights : $(w_1, w_2, w_3) = (0.3, 0.2, 0.5)$ $[k = 3]$ or $(w_1, w_2) = (0.5, 0.5)$ $[k = 2]$
4. Samples are randomly generated from $F = w_1 F_1 + ... + W_k F_k$, where $F_i \sim MN(\mu_i, \sigma_i^2 I_n)$



(a) Best random projection

(b) PCA

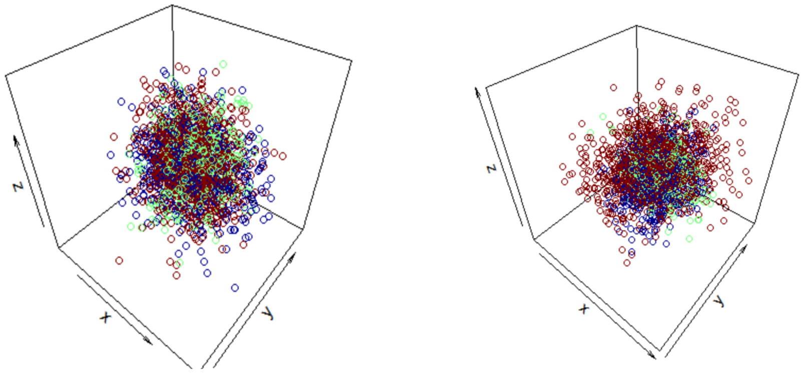
Figure 1: Projection comparison ($k = 3$)



(a) Best random projection

(b) PCA

Figure 2: Projection comparison ($k = 2$)



(a) Best random projection

(b) PCA

Figure 3: Non-spherical Gaussian model projection comparison ($k = 3$)

By observing Figure 1,2, We can find PCA works well in a mixture of spherical Gaussian model. However, PCA sometimes dos not work well when F_i is not spherical Gaussian (e.g : Multivariate normal with covariance $\neq c \times I_n$, where $c = constant > 0$)
Note : In Figure 3, $F_i \sim MN(\mu_i, \Sigma_i)$ is used for experiment, where Σ_i is randomly sampled from $Wishart(\nu = n + 4, \Psi = I_n)$

4 Performance Guarantee : Distance-based classification

In this paper [3] , there are some problems to realize the algorithm. The problems are followings :

1. Description of algorithm assumes when performance is guaranteed. Thus, setting parameters on given algorithm is not trivial on practical circumstance, which is a difficult part to realize the algorithm. (From theorem 6.2, when $n = 100$, $w_{min} = 0.2$, it requires at least $m = 1.5 \times 10^8$ for correctly classify with probability 0.5, so we need another description of this algorithm in a practical manner)

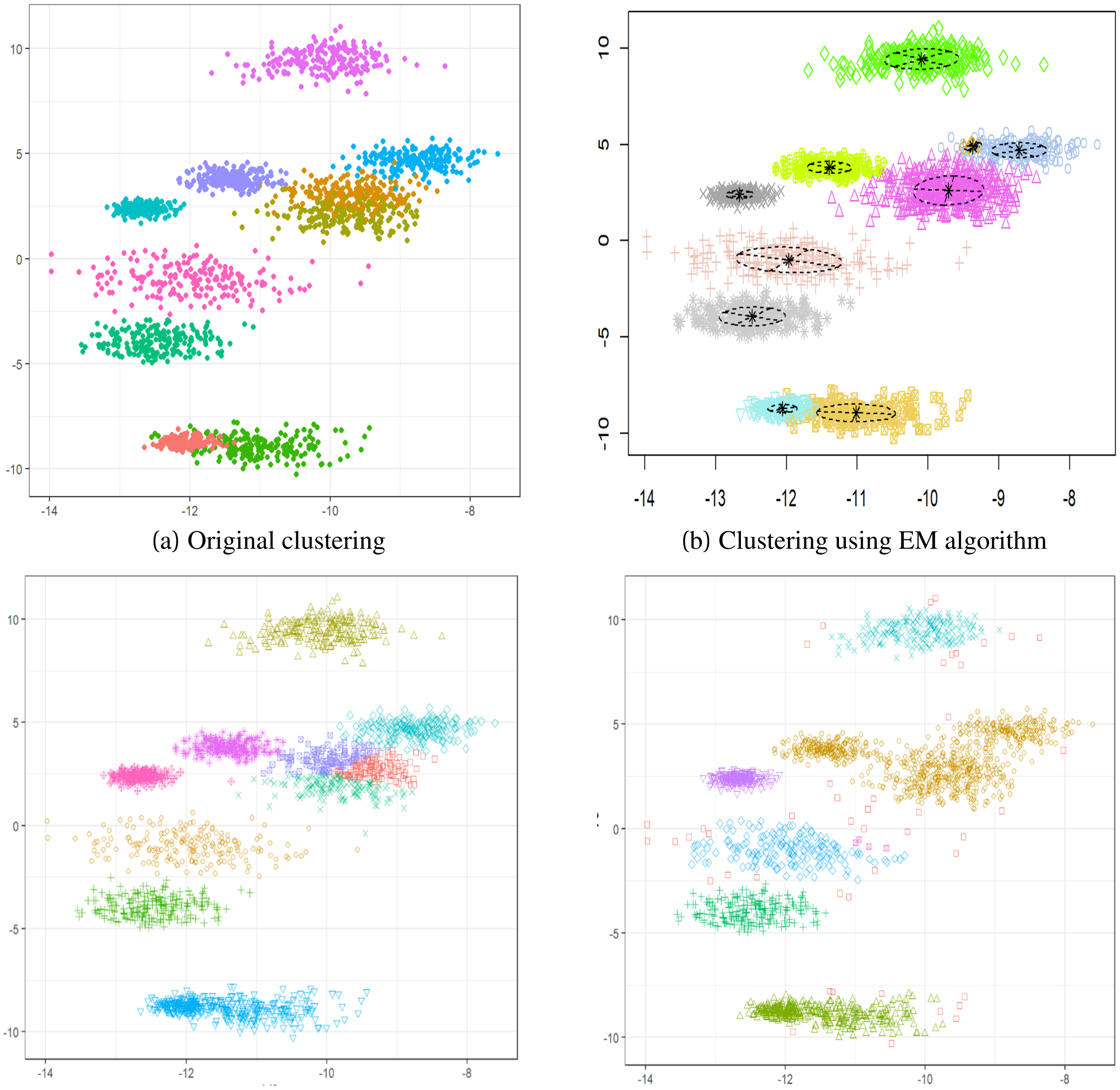
2. For a small sample size case (ex: $n = 100$, $m < 5000$), the algorithm seems to fail, as it does not work as intended in paper.

However, It turns out that EM algorithm is an alternative way to cluster these projected samples in previous paper [2], although it does not converge sometimes.

5 Empirical test : Distance-based classification

Empirical setting :
1. Basic parameter setting: n = 100, k = 10, m = 2000, $w_i = 0.1$ for all i
2. Means and variances are samples in the same manner as in section 3 (except each variance is decreased by $\frac{1}{10}$ for reasonable separation)

Since we projected samples onto subspace spanned by top 2 right singular vectors for visualization, the performance would be much better if we used projection onto subspace spanned by top k right singular vectors. From observation Figure 4, it seems clustering using EM algorithm is the most probable way to cluster in mixture of Gaussian model.
(Note : In this experiment, the EM algorithm for Gaussian Mixtures [1] is used to implement below clusterings.)



(a) Original clustering

(b) Clustering using EM algorithm

(c) Llyod K-means clustering

(d) DBSCAN clustering

Figure 4: Comparison of various clustering on mixture of spherical Gaussian model ($k = 10$)

5.1 Open problem

1. How to figure out n-dimensional samples follows mixture of spherical Gaussians?

2. How to cluster projected samples when we don't know the number of component distributions $(= k)$?

References

[1] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4, pages 438–455. Springer, 2006.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B* (*Methodological*), 39(1):1–38, 1977.

[3] S. Vempala and G. Wang. A spectral algorithm for learning mixture models. *Journal of Computer and System Sciences*, 68(4):841–860, 2004.

6 Appendix

Observation 6.1. (*Projection of spherical Gaussians*)
Let $F \sim MN(\mu, \sigma^2 I_n) = (N(\mu_1, \sigma^2), ..., N(\mu_n, \sigma^2))$ and let $v_1, ..., v_k$ be a set of orthonormal vectors spanning subspace $V \subset \mathbb{R}^n$, then projection of F onto V is following :

$$F|_V = (N(\nu_1^T \mu, \sigma^2), ..., N(\nu_k^T \mu, \sigma^2))$$

In other words, projection of F onto V is a spherical Gaussian with radius $\sigma\sqrt{k}$

Theorem 6.2. [*3*] (*Performance guarantee for spectral algorithm*)
With a sample of size

$$m = \Omega(\frac{n^3}{w_{min}^2}(\log \frac{n}{w_{min}}(1 + \max_{i=1}^{k} \frac{|\mu_i|^2}{\sigma_i^2})) + n \log \frac{4k}{\delta})$$

and initial seperation,

$$||\mu_i - \mu_j|| \geq 85 \max\{\sigma_i, \sigma_j\}(k \log \frac{m}{\delta})^{\frac{1}{4}}$$

the algorithm correctly classifies all Gaussians with probability at least 1 -δ