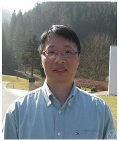
Sparse Clustering of High-Dimensional Gaussian Mixtures

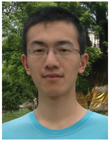
Jing Ma

Department of Statistics, The Wharton School
University of Pennsylvania

JSM, August 4 2016



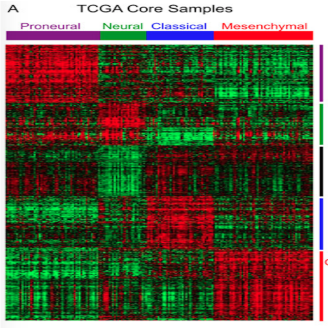
Tony Cai



Linjun Zhang

Clustering

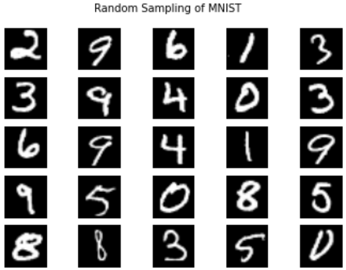
- Disease diagnosis



Verhaak et al. Cancer Cell, '10

Clustering

- Disease diagnosis
- Pattern recognition
- ...



Clustering

Existing algorithms

- K-means/ K-median
- Hierarchical clustering
- Expectation-Maximization (EM) algorithm
- ...

Clustering

Existing algorithms

- K-means/ K-median
- Hierarchical clustering
- Expectation-Maximization (EM) algorithm
- ...

However, theoretical performance of the clustering algorithm is not fully understood.

Gaussian mixture model

General form (2-class)

- Model:

$$y^{(1)}, \dots y^{(n)} \text{ i.i.d } \sim \begin{cases} 1, & \text{with probability } 1-\omega; \\ 2, & \text{with probability } \omega. \end{cases}$$

$$\boldsymbol{z}^{(i)} \mid y^{(i)} = k \ i.i.d. \sim N_p(\boldsymbol{\mu}_k, \Sigma); \quad k = 1, 2. \tag{1}$$

- Observations: $\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}, \dots, \boldsymbol{z}^{(n)}$.
- Goal: Cluster $\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(n)}$ into two groups with statistical guarantees.

Gaussian mixture model

- When p is small, we solve for MLE to maximize

$$L(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \log\left\{ f(\boldsymbol{z}^{(i)}|\boldsymbol{\mu}_1, \Sigma)P(y^{(i)} = 1) + f(\boldsymbol{z}^{(i)}|\boldsymbol{\mu}_2, \Sigma)P(y^{(i)} = 2)\right\}.$$

- Drawbacks: $L(\boldsymbol{\theta})$ is not convex; MLE is challenging for large p.
- Solution: Expectation-Maximization (EM) algorithm (Dempster et al. '77).

If we know the true parameters $\omega, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2$ and Σ, and denote the discriminating direction $\boldsymbol{\beta} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$, then the following classification rule yields the minimal mis-classification error:

$$C_{opt}(\boldsymbol{z}) = \begin{cases} 1, & \{\boldsymbol{z} - (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)/2\}'\boldsymbol{\beta} \geq \log(\frac{\omega}{1-\omega}) \\ 2, & \{\boldsymbol{z} - (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)/2\}'\boldsymbol{\beta} < \log(\frac{\omega}{1-\omega}). \end{cases}$$

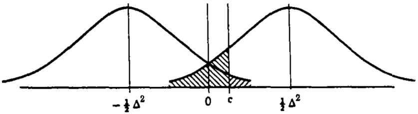


Figure: Mis-classification error of LDA

Linear programming discriminant

If we know the sample labels $y^{(1)}, ...y^{(n)}$, then one can estimate $\boldsymbol{\mu}_k$ by

$$\hat{\boldsymbol{\mu}}_k = \frac{1}{n_k} \sum_{i=1}^{n} \boldsymbol{z}^{(i)} I(y^{(i)} = k), \quad k = 1, 2,$$

and

$$\hat{\Sigma} = \frac{1}{n}(n_1 \hat{\Sigma}_1 + n_2 \hat{\Sigma}_2).$$

Assuming sparse $\boldsymbol{\beta}$, one can apply the LPD (Cai and Liu '11) to get

$$\hat{\boldsymbol{\beta}} = \arg\min\{\|\boldsymbol{\beta}\|_1 : \|\hat{\Sigma}\boldsymbol{\beta} - (\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)\|_\infty \leq \lambda_n\}.$$

The EM algorithm

We combine the above ideas to iteratively estimate $\boldsymbol{\theta} = (\omega, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\beta})$. The conditional log-likelihood

$$\begin{aligned} Q_n(\boldsymbol{\theta} \mid \tilde{\boldsymbol{\theta}}) &= \mathbb{E}_n[\log L(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}, \boldsymbol{z})] \\ &= \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{2} P(y^{(i)} = k \mid \tilde{\boldsymbol{\theta}}) \log f(\boldsymbol{z}^{(i)} | \boldsymbol{\mu}_k, \Sigma) \end{aligned}$$

The EM algorithm

Unsupervised Linear Programming Discriminant (ULPD)

- Initialization $\boldsymbol{\theta}^{(0)} = \{\omega^{(0)}, \boldsymbol{\mu}_k^{(0)}, \boldsymbol{\beta}^{(0)}\}; \kappa \in [1/2, 3/4]; \lambda_n^{(0)}; T_{stop}$.
- E-step: Evaluate $Q_n(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$.
- M-step:

$$
\begin{aligned}
(\omega^{(t+1)}, \boldsymbol{\mu}_k^{(t+1)}, \Sigma^{(t+1)}) &= \arg\max Q_n(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) \\
\lambda_n^{(t+1)} &= \kappa \lambda_n^{(t)} + C\sqrt{\log p/n} \\
\boldsymbol{\beta}^{(t+1)} &= \arg\min\{\|\boldsymbol{\beta}\|_1 : \|\Sigma^{(t+1)}\boldsymbol{\beta} - (\boldsymbol{\mu}_1^{(t+1)} - \boldsymbol{\mu}_2^{(t+1)})\|_\infty \le \lambda_n^{(t+1)}\}.
\end{aligned}
$$

- Upon convergence, output $\hat{\omega}, \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\beta}} \leftarrow \omega^{(T_{stop})}, \boldsymbol{\mu}_k^{(T_{stop})}, \boldsymbol{\beta}^{(T_{stop})}$.

Upper bound

Theorem 1

Assume $\|\boldsymbol{\beta}\|_0 \leq s$*. Under certain technical conditions, the output* $\boldsymbol{\beta}^{(T_{stop})}$ *satisfies with high probability*

$$\|\boldsymbol{\beta}^{(T_{stop})} - \boldsymbol{\beta}\|_2 \lesssim \kappa^{T_{stop}} \|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}\|_2 + \sqrt{\frac{s \log p}{n}}.$$

Consequently, if $T_{stop} \gtrsim \log n$*, then*

$$\|\boldsymbol{\beta}^{(T_{stop})} - \boldsymbol{\beta}\|_2 \lesssim \sqrt{\frac{s \log p}{n}}.$$

Remarks

The results in Wang et al. ('15) ($\Sigma = \sigma^2 \mathbf{I}_p$) show

$$\|\boldsymbol{\beta}^{(T_{stop})} - \boldsymbol{\beta}\|_2 \lesssim \sqrt{\frac{s \log p \cdot \log n}{n}}.$$

The proposed classifier

Given the estimated $\hat{\omega}, \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\beta}}$, the sample $\boldsymbol{z}$ is classified as

$$\hat{C}(\boldsymbol{z}) = \begin{cases} 1, & \{\boldsymbol{z} - (\hat{\boldsymbol{\mu}}_1 + \hat{\boldsymbol{\mu}}_2)/2\}'\hat{\boldsymbol{\beta}} \geq \log(\frac{\hat{\omega}}{1-\hat{\omega}}) \\ 2, & \{\boldsymbol{z} - (\hat{\boldsymbol{\mu}}_1 + \hat{\boldsymbol{\mu}}_2)/2\}'\hat{\boldsymbol{\beta}} < \log(\frac{\hat{\omega}}{1-\hat{\omega}}). \end{cases}$$

The mis-clustering error is defined as

$$R(\hat{C}) = \min_{\pi \in \mathbb{P}_2} \mathbb{E}[I(\hat{C}(\boldsymbol{z}) \neq \pi(y))],$$

where $\mathbb{P}_2 = \{\pi : [1,2] \to [1,2]\}$ is a set of permutation function.

Theorem 2

Under the same conditions of Theorem 1 and with $T_{stop} \gtrsim \log n$, the classifier $\hat{C}$ with mis-clustering error $R(\hat{C})$, satisfies

$$R(\hat{C}) - R_{opt} \lesssim \frac{s \log p}{n} + \sqrt{\frac{s \log p}{n}} \cdot \left| \log \left(\frac{\omega}{1 - \omega} \right) \right|.$$

Simulation Example

Competing methods

- KM: *k*-means
- SKM: sparse *k*-means (Witten and Tibshirani '12)
- SHP: sparse clustering with HARDT-PRICE (Azizyan et al. '14)
- PCCM: penalized clustering with common covariances (Zhou et al. '09)

Benchmark

- LPD: supervised linear program discriminant rule (Cai and Liu '11)
- Oracle: Fisher's LDA with true parameters

Simulation: Erdős-Rényi Random Graph

- $\omega = 0.5, \boldsymbol{\mu}_1 = \mathbf{0}, \boldsymbol{\beta} = (\underbrace{1, \ldots, 1}_{s=10}, 0, \ldots, 0)'$.
- Ω is generated from an Erdős-Rényi random graph with adjacency matrix as follows:

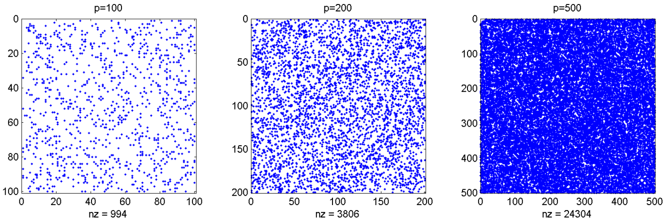


Figure: Sparsity patterns of Ω

Simulation: Erdős-Rényi Random Graph

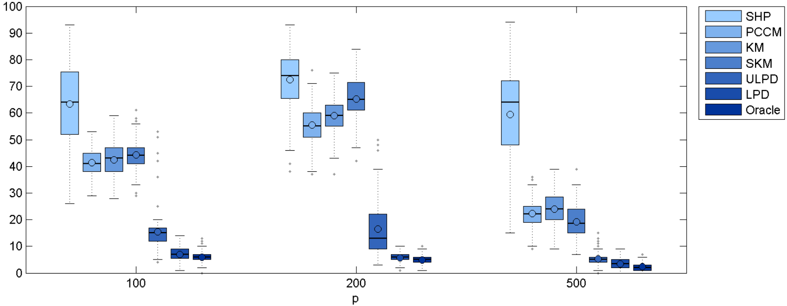


Figure: Clustering errors based on $n = 200$ test samples and 100 replications.
Circle: mean. Line: median

Handwritten digits data

We use the digits ‘0’ and ‘9’ as an example ($n=200$, $p=256$).

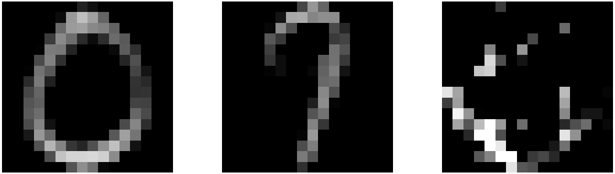


Figure: Group mean based on ULPD identified labels (left, middle) and discriminative pixels selected by ULPD (right)

Contributions

Summary

- Knowing labels doesn't improve the convergence rate of estimation and classification.

Not covered in this talk

Contributions

Summary

- Knowing labels doesn't improve the convergence rate of estimation and classification.
- Imbalanced data is harder to classify than balanced data.

Not covered in this talk

Contributions

Summary

- Knowing labels doesn't improve the convergence rate of estimation and classification.
- Imbalanced data is harder to classify than balanced data.

Not covered in this talk

- Lower bound of estimation and clustering error is in the same order of the respective upper bound.

Contributions

Summary

- Knowing labels doesn't improve the convergence rate of estimation and classification.
- Imbalanced data is harder to classify than balanced data.

Not covered in this talk

- Lower bound of estimation and clustering error is in the same order of the respective upper bound.
- Extensions to multi-class GMM and/or unequal covariance matrices are available.

Contributions

Summary

- Knowing labels doesn't improve the convergence rate of estimation and classification.
- Imbalanced data is harder to classify than balanced data.

Not covered in this talk

- Lower bound of estimation and clustering error is in the same order of the respective upper bound.
- Extensions to multi-class GMM and/or unequal covariance matrices are available.
- Tuning parameter λ_n can be chosen via adaptive estimation.

Thanks!

Lower bound

Theorem 3

Let $\Theta = \{\boldsymbol{\theta} = (\omega, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma) : \|\boldsymbol{\beta}\|_0 \leq s, \omega \in (c_1, 1-c_1), (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \geq \eta > 0, 1/c_2 < \phi_{\min}(\Sigma) \leq \phi_{\max}(\Sigma) < c_2\}$ *and* $\mathcal{C}$ *contain all classifiers. Then with probability at least* $1 - p^{-1} - n^{-1}$*, the estimated* $\hat{\boldsymbol{\beta}}$ *of any algorithm satisfies*

$$\inf_{\hat{\boldsymbol{\beta}}} \sup_{\Theta} \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2 \geq C_1 \sqrt{\frac{s \log p}{n}},$$

and the mis-clustering error rate satisfies

$$\inf_{\hat{C} \in \mathcal{C}} \sup_{\Theta} \mathbb{E}_{\boldsymbol{\theta}}[R(\hat{C}) - R_{opt}] \geq C_2 \min\left\{\frac{s \log p}{n} + \sqrt{\frac{s \log p}{n}} \cdot \left|\log\left(\frac{\omega}{1-\omega}\right)\right|, 1\right\},$$

for some constant $C_1, C_2 > 0$.

Remarks

- Theorems 1, 2, and 3 jointly show the minimax optimality of the proposed algorithm.

Initialization

- Initialization Condition 1 (IC-1):

$$\max_{k=1,2} \{\|\boldsymbol{\mu}_k - \hat{\boldsymbol{\mu}}_{\pi(k)}\|_\infty\} \lesssim \frac{1}{\|\boldsymbol{\beta}\|_1}\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_\infty,$$

$$\|\Sigma - \hat{\Sigma}\|_\infty \lesssim \frac{1}{\|\boldsymbol{\beta}\|_1 \min\{\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_1, \sqrt{s}\}}\|\Sigma\|_\infty.$$

- Initialization Condition 2 (IC-2):

$$\max_{k=1,2} \{\|\boldsymbol{\mu}_k - \hat{\boldsymbol{\mu}}_{\pi(k)}\|_2\} \leq \frac{1}{4}\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_2,$$

$$\|\Sigma - \hat{\Sigma}\|_2 \leq \frac{1}{4}\phi_{\min}(\Sigma).$$

Initialization

When $\|\boldsymbol{\beta}\|_1 \cdot \min\{\sqrt{s}, \|\boldsymbol{\beta}\|_1, \|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|\} \lesssim n^{1/6}$, the initialization given by HARDT-PRICE algorithm satisfies IC-1.

Theorem 4 (Hardt and Price '15)

Given $\epsilon, \delta > 0$ and n samples from the GMM, if $n = O(\frac{1}{\epsilon^6} \log(\frac{d}{\delta} \log(\frac{1}{\epsilon})))$, then with probability at least $1 - \delta$, HARDT-PRICE algorithm produces estimates $\hat{\boldsymbol{\mu}}_1$, $\hat{\boldsymbol{\mu}}_2$ and $\hat{\Sigma}$ such that for some permutation $\pi : \{1, 2\} \to \{1, 2\}$,

$$\max\big(\max_{k=1,2}\{\|\boldsymbol{\mu}_k - \hat{\boldsymbol{\mu}}_{\pi(k)}\|_\infty^2\}, \|\Sigma - \hat{\Sigma}\|_\infty\big) \leq \epsilon(\frac{1}{4}\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_\infty^2 + \|\Sigma\|_\infty).$$

Adaptive estimation

Update $\boldsymbol{\beta}^{(t+1)}$:

1: **Parameters: $\boldsymbol{\theta}^{(t)}$**

2: **Step 1:**

$$\begin{aligned}\tilde{\boldsymbol{\beta}}^{(t+1)} =& \arg\min_{\boldsymbol{\beta}} \Big\{ \|\boldsymbol{\beta}\|_1 : |\Sigma^{(t)}\boldsymbol{\beta} - (\boldsymbol{\mu}_1^{(t)} - \boldsymbol{\mu}_2^{(t)})|_j \\ &\leq 2\sqrt{\frac{\log p}{n}}(8|\boldsymbol{\mu}_{1j}^{(0)}|^2 + 8|\boldsymbol{\mu}_{2j}^{(0)}|^2 + 16\Sigma_{jj}^{(0)} + 16|\boldsymbol{\beta}'\boldsymbol{\mu}_1^{(t)}| + 16|\boldsymbol{\beta}'\boldsymbol{\mu}_2^{(t)}| + \kappa^t\|\boldsymbol{\theta}^{(0)}\|_2) \\ &\quad + 4\{1 + (|\boldsymbol{\beta}'\boldsymbol{\mu}_1^{(t)}| + \boldsymbol{\beta}'\boldsymbol{\mu}_2^{(t)}|) \cdot (|\boldsymbol{\mu}_{1j}^{(0)}| + |\boldsymbol{\mu}_{2j}^{(0)}|)\}\kappa^t\|\boldsymbol{\theta}^{(0)}\|_2 \Big\}\end{aligned}$$

3: **Step 2:**

$$\begin{aligned}\boldsymbol{\beta}^{(t+1)} =& \arg\min_{\boldsymbol{\beta}} \Big\{ \|\boldsymbol{\beta}\|_1 : |\Sigma^{(t)}\boldsymbol{\beta} - (\boldsymbol{\mu}_1^{(t)} - \boldsymbol{\mu}_2^{(t)})|_j \\ &\leq \sqrt{\Sigma_{jj}^{(t)}\frac{\log p}{n}\Big\{2(\boldsymbol{\mu}_1^{(t)} - \boldsymbol{\mu}_2^{(t)})'\tilde{\boldsymbol{\beta}}^{(t+1)} + \frac{1}{\omega^{(t)}(1-\omega^{(t)})}\Big\}} + \kappa^t\|\boldsymbol{\theta}^{(0)}\|_2. \Big\}\end{aligned}$$

Extension

If we consider the multi-class Gaussian Mixture Model

$$P(Y=k)=\omega_k, \quad Z|Y=k \sim N(\mu_k, \Sigma), \quad k \in \{1,2,...,K\},$$

then the Bayes rule is

$$Y_{Bayes}(Z) = \arg\max\{(Z-\mu_k)'\boldsymbol{\beta}_k + \log \omega_k\},$$

where $\boldsymbol{\beta}_k = \Sigma^{-1}\boldsymbol{\mu}_k$ for $k=1,...,K$.

If sample labels are known and $\boldsymbol{\beta}_k$ is sparse, one can estimate $\boldsymbol{\beta}_k$ by

$$\hat{\boldsymbol{\beta}}_k = \arg\min_{\boldsymbol{\beta}}\{\|\boldsymbol{\beta}\|_1 : \|\hat{\Sigma}\boldsymbol{\beta} - \hat{\mu}_k\| \le C\sqrt{\frac{\log p}{n}}\}.$$