

Variable Selection in High-Dimensional Clustering

Sijian Wang¹, Ji Zhu^{2*}

¹ Department of Biostatistics

² Department of Statistics

University of Michigan, Ann Arbor, MI 48109

Abstract

Variable selection in high-dimensional clustering analysis is an important yet challenging problem. In this paper, we propose a method that simultaneously separate data points into similar clusters and select informative variables that contribute to the clustering. Our methods are in the framework of penalized model-based clustering. Unlike the classical L_1 -norm penalization, the penalty term that we propose makes use of the fact that parameters belonging to one variable should be treated as a natural group. Numerical results indicate that the new methods tends to remove non-informative variables more effectively and provide better clustering results than the L_1 -norm approach.

1 Introduction

Clustering data into similar clusters is an important practical problem in a wide variety of fields, including statistics, bioinformatics, artificial intelligence, and data mining. With the recent advent of technologies, good clustering algorithms are very much desired for analyzing high-dimensional data where the number of variables is considerably larger than the number of observations.

Pan and Shen (2006) proposed an approach for variable selection in clustering through penalized model-based clustering. They parameterized the mean in cluster k for variable x_j as $\mu_{kj} = \phi_j + \delta_{kj}$, where ϕ_j is the global mean for variable x_j . If for different k , all δ_{kj} are 0, then the variable x_j is not informative for clustering, at least in terms of the mean. In their method, the L_1 -norm penalty was employed to shrink the cluster-specific means μ_{kj} towards the global mean ϕ_j , and this effectively realizes the variable selection.

In this paper, we focus on the clustering for high-dimensional data characterized by high dimension and low sample size. Enlightened by their method, we propose a new approach that is also in the framework of penalized model-based clustering. Noticing that cluster-specific mean parameters associated with the same variable can be naturally “grouped” together, and intuitively should be treated as a group, we propose a novel penalty function, different from the one in Pan and Shen (2006), to make use of such natural grouping information within the data. As we will see in the numerical study, the new method tends to remove non-informative variables more effectively and provide better clustering results.

The rest of the paper is organized as follows. In Section 2, we propose our new model: the adaptive L_∞ -norm penalized Gaussian mixture model (ALP-GMM). In Section 3, we derive algorithm to estimate the parameters in the model. Numerical results are in Section 4 and Section 5. We conclude the paper with Section 6.

*Address for correspondence: Ji Zhu, 439 West Hall, 1085 South University, Ann Arbor, MI 48109-1107. E-mail: jizhu@umich.edu.

2 The Adaptive L_∞ -norm Penalized Gaussian Mixture Model (ALP-GMM)

We observe n p -dimensional samples $\mathbf{x}_i = (x_{i1}, \dots, x_{ij}, \dots, x_{ip})$, $i = 1, \dots, n$, and without loss of generality, we assume that the data are centered in each dimension (variable), i.e., $\sum_{i=1}^n x_{ij} = 0$, $j = 1, \dots, p$. Our aim is to separate the data into K clusters.

The Gaussian mixture model (GMM) is a standard tool for this purpose (Fraley and Raftery, 2002; McLachlan and Peel, 2002). We assume that each observation \mathbf{x}_i is drawn from a finite Gaussian mixture distribution

$$f(\mathbf{x}_i) = \sum_{k=1}^K \pi_k f_k(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

where π_k 's are the mixing proportions satisfying $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^K \pi_k = 1$. $\boldsymbol{\mu}_k = (\mu_{k1}, \dots, \mu_{kj}, \dots, \mu_{kp})$ is the mean vector of the Gaussian distribution characterizing the k th cluster, and $\boldsymbol{\Sigma}_k$ is the corresponding covariance matrix. In this paper, we focus on high-dimensional data and assume $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \dots, \sigma_j^2, \dots, \sigma_p^2)$, i.e., the covariance matrices are the same across different clusters and are diagonal. This is a common assumption when one works with high dimension and low sample size data. Some theoretical justification for this assumption can be found in Bickel and Levina (2004).

Given an observation $\mathbf{x}^* = (x_1^*, \dots, x_p^*)$, one can compute the probability that \mathbf{x}^* is from the k th cluster

$$p_k = \frac{\pi_k}{\sqrt{2\pi} \prod_{j=1}^p \sigma_j} \exp\left(-\sum_{j=1}^p \frac{(x_j^* - \mu_{kj})^2}{2\sigma_j^2}\right), \quad k = 1, \dots, K \quad (1)$$

and \mathbf{x}^* will be assigned to the cluster with the largest p_k .

We denote $\Theta = \{\sigma_j^2, \pi_k, \mu_{kj}, k = 1, \dots, K; j = 1, \dots, p\}$ as the set containing all the parameters. Given the data $\mathbf{x}_1, \dots, \mathbf{x}_n$, the log-likelihood function is

$$\ell_0(\Theta) = \sum_{i=1}^n \log\left(\sum_{k=1}^K \pi_k f_k(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma})\right). \quad (2)$$

Maximization of the above objective function with respect to Θ is often difficult, and it is common to use the EM algorithm (Dempster et al., 1977) via the framework of missing data. Let τ_{ik} be the indicator of whether \mathbf{x}_i is from cluster k , i.e., $\tau_{ik} = 1$ if \mathbf{x}_i belongs to cluster k , and $\tau_{ik} = 0$ otherwise. If the missing data τ_{ik} were observed, the log-likelihood function for the complete data is

$$\ell(\Theta) = \sum_{i=1}^n \sum_{k=1}^K \tau_{ik} (\log \pi_k + \log f_k(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma})). \quad (3)$$

For the purpose of variable selection, Pan and Shen (2006) proposed the regularized log-likelihood function

$$\ell_P(\Theta) = \ell(\Theta) + \lambda \sum_{k=1}^K \sum_{j=1}^p |\mu_{kj}|, \quad (4)$$

where the penalty function is the L_1 -norm of the mean vectors, and we refer this model as the L_1 -norm Gaussian mixture model (L_1 -GMM). The L_1 -norm penalty shrinks some of the fitted means μ_{kj} to be *exactly* zero when making λ sufficiently large. As we can see from (1), if for the j th variable, all the cluster-specific means μ_{kj} , $k = 1, \dots, K$, are shrunk to zero, $\exp(-(x_j^* - \mu_{kj})^2 / (2\sigma_j^2))$ becomes a common factor $\exp(-x_j^{*2} / (2\sigma_j^2))$ that does not depend on k ; hence the j th variable does not contribute to the clustering score (1), and it can be removed.

In order to remove the j th variable, we require all μ_{kj} , $k = 1, \dots, K$, to be zero. However, we can see from (4) that the L_1 -norm penalty treats all the μ_{kj} the same, i.e., it does not use the information that

μ_{kj} and $\mu_{k'j}$ are associated with the same variable x_j , and intuitively, they belong to one “group” and they should be treated differently from $\mu_{kj'}$, which is associated with a different variable $x_{j'}$. We propose a different penalty function, i.e., the L_∞ -norm penalty that incorporates this information into the modeling procedure. Specifically, we consider the penalized log-likelihood function:

$$\ell_P(\Theta) = \ell(\Theta) + \lambda \sum_{j=1}^p \max_k (|\mu_{1j}|, \dots, |\mu_{kj}|, \dots, |\mu_{Kj}|), \quad (5)$$

where $\max(|\mu_{1j}|, \dots, |\mu_{Kj}|) = \|(\mu_{1j}, \dots, \mu_{Kj})\|_\infty$. Different from penalizing every μ_{kj} individually, the L_∞ -norm penalizes the maximum absolute value of μ_{kj} , $k = 1, \dots, K$, for the j th variable. If the maximum of $|\mu_{kj}|$, $k = 1, \dots, K$, is shrunken to zero, all μ_{kj} are automatically shrunken to zero. The L_∞ -norm penalty has also been used in Zhao, Rocha, and Yu (2006) for regression problems.

To further improve the model (5), we borrow the adaptive idea from Zou (2006), i.e., to penalize different variables differently. We consider

$$\ell_P(\Theta) = \ell(\Theta) + \lambda \sum_{j=1}^p w_j \cdot \max_k (|\mu_{1j}|, \dots, |\mu_{kj}|, \dots, |\mu_{Kj}|), \quad (6)$$

where w_j are pre-specified weights. The intuition is that if the j th variable is informative for clustering, we would like the corresponding w_j to be small, hence the j th variable is lightly penalized, while if the j th variable is non-informative for clustering, we would like the corresponding w_j to be large, hence the j th variable is heavily penalized. How to pre-specify w_j from the data will be discussed in the numerical study section.

3 Algorithms

We consider

$$\ell_P(\Theta) = \ell(\Theta) + J(\Omega) \quad (7)$$

$$= \sum_{i=1}^n \sum_{k=1}^K \tau_{ik} (\log \pi_k + \log f_k(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma})) + J(\Omega), \quad (8)$$

where $\Omega = \{\mu_{kj}, k = 1, \dots, K; j = 1, \dots, p\}$, $J(\Omega) = \lambda \sum_{j=1}^p w_j \max(|\mu_{1j}|, \dots, |\mu_{Kj}|)$ for our ALP-GMM. The indicators τ_{ik} are not observed, and the EM algorithm can be used to maximize the above penalized log-likelihood with respect to Θ , and it follows closely to the EM algorithm for the standard non-penalized GMM model (McLachlan and Peel, 2002). The only difference exists in estimating μ_{kj} in the M-step. Due to lack of space, we omit the description for estimating τ_{ik} , π_k , and σ_k^2 , and only describe how to update μ_{ik} in each iterative step of the EM algorithm. Consider the minimization problem:

$$\min_{\mu_{kj}} \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \sum_{j=1}^p \tau_{ik} (x_{ij} - \mu_{kj})^2 / \sigma_j^2 + \lambda \sum_{j=1}^p w_j \cdot \max_k (|\mu_{1j}|, \dots, |\mu_{Kj}|).$$

This can be decomposed into p separate minimization problems

$$\min_{\mu_{kj}} \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \tau_{ik} (x_{ij} - \mu_{kj})^2 / \sigma_j^2 + \lambda \cdot w_j \cdot \max_k (|\mu_{1j}|, \dots, |\mu_{Kj}|), \quad 1 \leq j \leq p. \quad (9)$$

For each j , (9) can be transformed into a quadratic programming problem:

$$\min_{\mu_{kj}, M_j} \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \tau_{ik} (x_{ij} - \mu_{kj})^2 / \sigma_j^2 + \lambda \cdot w_j \cdot M_j \quad (10)$$

$$\text{subject to} \quad -M_j \leq \mu_{kj} \leq M_j, \quad k = 1, \dots, K \quad (11)$$

$$M_j \geq 0 \quad (12)$$

Hence most commercially available packages can be used to solve it.

We have also explored explicit forms for the solutions to (9), which help us gain more insights into the nature of the L_∞ -norm penalty. Let $\mu_{kj}^0 = \sum_{i=1}^n \tau_{ik} x_{ij} / \sum_{i=1}^n \tau_{ik}$, for $j = 1, \dots, p$ and $k = 1, \dots, K$, which are the solutions when there is no penalty (or $\lambda = 0$). We can show that $\hat{\mu}_{kj}$, the solution to the minimization problem (9), can be achieved by shrinking a weighted average of several μ_{kj}^0 .

Theorem 1 *For the j th minimization problem (9), if there exist k_1, \dots, k_r , such that*

$$|\hat{\mu}_{k_1 j}| = \dots = |\hat{\mu}_{k_r j}| > |\hat{\mu}_{k j}|, \text{ for } k \notin \{k_1, \dots, k_r\} \quad (13)$$

then

$$\hat{\mu}_{kj} = \begin{cases} \mu_{kj}^0 & k \notin \{k_1, \dots, k_r\} \\ \text{sgn}(\mu_{kj}^0) \left(\sum_{s=1}^r \frac{\tau_{k_s j}}{\sum_{s=1}^r \tau_{k_s}} |\mu_{k_s j}^0| - \frac{\lambda w_j \sigma_j^2}{\sum_{s=1}^r \tau_{k_s}} \right)_+ & k \in \{k_1, \dots, k_r\} \end{cases} \quad (14)$$

where $\tau_{k_s} = \sum_{i=1}^n \tau_{ik_s}$; $(\cdot)_+$ is the positive part of the argument.

Due to lack of space, we skip the proof in this paper. From Theorem 1, we can see when there are r maximums among $|\hat{\mu}_{kj}|$, only the corresponding μ_{kj}^0 will be shrunken by the L_∞ -norm penalty, and they are shrunken to the same absolute value. This value is based on a weighted average of μ_{kj}^0 of the corresponding r clusters, and the weights are proportional to τ_{k_s} . We can also see that if the j th variable is non-informative and all $|\mu_{kj}^0|$ are close to zero, then the L_∞ -norm penalty tends to shrink all of them to zero (with an appropriately chosen λw_j).

To implement Theorem 1, we need to decide r , the number of maximums among $\hat{\mu}_{kj}$, and the set $\{k_1, \dots, k_r\}$, which indicates which r μ_{kj}^0 should be shrunken. When K is not very large, say $K \leq 10$, we can use an exhaustive search to find r and $\{k_1, \dots, k_r\}$, i.e., for each $1 \leq r \leq K$, we search over all possible sets $\{k_1, \dots, k_r\}$. For each possible set, we estimate $\hat{\mu}_{kj}$ using (14), then check whether the estimate satisfies the assumption (13). If the assumption is satisfied, we compute the corresponding value for the objective function (9). Finally, we choose $\hat{\mu}_{kj}$ that give the smallest value for the objective function. When K is large, we will resort to the quadratic programming (10)–(12).

4 Simulation Study

In this section, we use simulation data to demonstrate our method ALP-GMM, and compare the results with that of the L_1 -GMM.

We considered two three-cluster scenarios. In both scenarios, there were a total of $p = 402$ variables with the first 2 informative and the other 400 non-informative in forming three clusters. The first 2 variables were i.i.d. $N(0, 1)$ for the first cluster, i.i.d. $N(2.5, 1)$ for the second cluster, and i.i.d. $N(5, 1)$ for the third cluster, whereas the remaining 400 variables were all i.i.d. $N(0, 1)$ for all three clusters. In the first scenario, we generated 20 observations for each of the first cluster and the third cluster, and 100 for the second cluster. We denote it as “20-100-20”. In the second scenario, we generated 50 observations for each of the first cluster and the third cluster, and 20 for the second cluster. We denote it as “50-20-50”. Similar to Breiman (1995) and Zou (2006), we computed the weights w_j in (6) using the un-penalized estimates $\mu_{kj}^0 = \sum_{i=1}^n \tau_{ik} x_{ij} / \sum_{i=1}^n \tau_{ik}$. Specifically:

$$M_j^0 = \max_k (|\mu_{1j}^0|, \dots, |\mu_{Kj}^0|), \text{ and } w_j = 1/M_j^0. \quad (15)$$

We chose the tuning parameters and the number of clusters using the Bayesian Information Criterion (BIC) (Schwarz, 1978)

$$\text{BIC} = -2 \sum_{i=1}^n \log \left(\sum_{k=1}^K \hat{\pi}_k f_k(\mathbf{x}_i; \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}) \right) + P \log n, \quad (16)$$

where P is the total number of non-zero estimates in $\hat{\mu}_{kj}$, $\hat{\sigma}_j^2$ and $\hat{\pi}_k$.

We repeated this 50 times, and computed the average number of identified clusters, the average balanced error rate, the average number of selected informative variables, the average number of non-informative variables that were kept and their corresponding standard errors. The balanced error rate (BER) is defined as:

$$\text{BER} = \frac{1}{K} \sum_{k=1}^K \frac{\# \text{ of data points in cluster } k \text{ that were clustered otherwise}}{\# \text{ of data points in cluster } k},$$

which takes into account that different clusters may have very different numbers of data points. The results are summarized in Table 1.

Table 1: Simulation results for the “20-100-20” example and the “50-20-50” example: the upper part is for the “20-100-20” example, and the lower part is for the “50-20-50” example. “# $K = 3$ ” is the number of times (out of 50) that 3 was identified as the number of clusters. “Error Rate” is the average balanced proportion of wrongly clustered data points. “# Info” is the average number of selected informative variables (out of 2). “# Non-Info” is the average number of non-informative variables (out of 400) that were kept. The numbers in the parentheses are the corresponding standard errors. “GMM without noise” is to apply the standard GMM method on the dataset with only the first 2 informative variables, and its “Error Rate” can be considered as a benchmark. “GMM” is the standard GMM method using all 402 variables.

Method	# $K = 3$	Error Rate	# Info	# Non-Info
The “20-100-20” Example				
GMM without noise	50	0.065 (0.034)	—	—
GMM	0	—	—	—
L_1 -GMM	45	0.376 (0.092)	1.30 (0.94)	105.0 (22.8)
ALP-GMM	50	0.070 (0.032)	2 (0)	0 (0)
The “50-20-50” Example				
GMM without noise	49	0.056 (0.029)	—	—
GMM	0	—	—	—
L_1 -GMM	44	0.115 (0.097)	2 (0)	38.1 (7.4)
ALP-GMM	48	0.065 (0.036)	2 (0)	0 (0)

As we can see, the ALP-GMM method discovered the three-cluster data structure for almost every repetition (out of 50), and the clustering error rates were just slightly higher than that of the GMM method without using any of the non-informative variables, i.e., the “oracle”. The ALP-GMM method removed all 400 non-informative variables for every repetition. In contrast, the L_1 -GMM method tended to have a higher error rate and keep more non-informative variables.

5 Real Data Analysis

In this section, we apply the ALP-GMM method to a gene microarray dataset, which consists of microarray experiments of small round blue cell tumors (SRBCT) of childhood cancer (Khan et al., 2001). The tumors are classified as Burkitt lymphoma (BL), Ewing sarcoma (EWS), neuroblastoma (NB), or rhabdomyosarcoma (RMS). A total of 63 training samples and 20 test samples were provided. Each sample consists of expression measurements on $p = 2,308$ genes. We applied the ALP-GMM method to the training data, ignoring the class labels. The tuning parameter and the number of clusters were chosen using BIC, and the chosen model was evaluated on the test data. The results are summarized in Table 2. The

ALP-GMM selected $K = 4$ as the number of clusters. The training errors and the test errors are all zero. The ALP-GMM selected 44 genes.

Table 2: Results on the SRBCT dataset

Method	# K	# of Genes	Training Error	Testing Error
Kahn et al. (2001)	—	96	0/63	0/20
ALP-GMM	4	44	0/63	0/20

6 Conclusion

In this paper, we have proposed a new method for simultaneously clustering high-dimensional data and selecting informative variables. Our method is in the framework of penalized model-based clustering. Unlike the L_1 -norm penalization, the penalty term that we propose makes use of the fact that parameters belonging to one variable should be treated as a natural group. We have presented some evidence that the new method tends to remove non-informative variables more effectively and provide better clustering results than the L_1 -norm approach.

References

- Bickel, P. J. and Levina, L. (2004). Some theory for fisher’s linear discriminant function, “naive bayes”, and some alternatives when there are many more variables than observations. *Bernoulli* **10**, 989-1010.
- Breiman, L. (1995). Better subset regression using the non-negative garrote. *Technometrics* **37**, 373-384.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society, Series B* **39**, 1-38.
- Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* **97**, 611-631.
- Khan, J., Wei, J. S., Ringner, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C. R., Peterson, C., and Meltzer, P. S. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature medicine* **7**, 673-679.
- McLachlan, G. and Peel, D. (2002). *Finite Mixture Models*. John Wiley & Sons.
- Pan, W., and Shen, X. (2006). Penalized model-based clustering with application to variable selection. Technical Report, Division of Biostatistics, University of Minnesota.
- Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics* **6**, 461-464.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**, 267-288.
- Zhao, P. Rocha, G., and Yu, B. (2006). Grouped and hierarchical model selection through composite absolute penalties. Technical Report, Department of Statistics, University of California at Berkeley.
- Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, To Appear.