TEMPLE UNIVERSITY

A CONCAVE PAIRWISE FUSION APPROACH TO CLUSTERING OF MULTI-RESPONSE REGRESSION AND ITS ROBUST EXTENSIONS

A Dissertation
Submitted to
the Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree of
DOCTOR OF PHILOSOPHY

---

by
Chen Chen
Diploma Date May 2022

Examining Committee Members:

Dr. Yuexiao Dong, Advisory Chair, Department of Statistical Science
Dr. Pallavi Chitturi, Department of Statistical Science
Dr. Cheng Yong Tang, Department of Statistical Science
Dr. CenCheng Shen, External Member, Department of Applied Economics and Statistics, University of Delaware

ABSTRACT

Solution-path convex clustering is combined with concave penalties by Ma and Huang (2017) to reduce clustering bias. Their method was introduced in the setting of single-response regression to handle heterogeneity. Such heterogeneity may come from either the regression intercepts or the regression slopes. The procedure, realized by the alternating direction method of multipliers (ADMM) algorithm, can simultaneously identify the grouping structure of observations and estimate regression coefficients.

In the first part of our work, we extend this procedure to multi-response regression. We propose models to solve cases with heterogeneity in either the regression intercepts or the regression slopes. We combine the existing gadgets of the ADMM algorithm and group-wise concave penalties to find solutions for the model. Our work improves model performance in both clustering accuracy and estimation accuracy. We also demonstrate the necessity of such extension through the fact that by utilizing information in multi-dimensional space, the performance can be greatly improved.

In the second part, we introduce robust solutions to our proposed work. We introduce two approaches to handle outliers or long-tail distributions. The first is to replace the squared loss with robust loss, among which are absolute loss and Huber loss. The second is to characterize and remove outliers' effects by a mean-shift vector. We demonstrate that these robust solutions outperform the squared loss based method when outliers are present, or the underlying distribution is long-tailed.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisory chair, Dr. Yuexiao Dong. You have helped shape my research from a vague interest in computational statistics into this dissertation. You continued to have faith in me, brought me back when I deviated from the goal, and motivated me to move forward when I got lost amid uncertainty. Thank you for being a scientifically rigorous mentor and a caring one. I am also grateful to the committee members, Dr. Pallavi Chitturi, Dr. Cheng Yong Tang, and Dr. CenCheng Shen. Thank you, Dr. Chitturi, not only for your encouraging comments on my research work but, more importantly, for helping me with the first job that led me onto the journey as an applied statistician. Thank you to Dr. Tang for your invaluable advice that led to the addition of the mean-shift solution to my work. Your excellent introduction course on statistical elements in machine learning sparked my desire to pursue research in this field. Thank you, Dr. Shen, for your insightful comments on my research work and directions on how to improve.

I am forever grateful to my former employer and mentor, Dror Rom. You took me under your wing and guided me to become a useful statistician. You saw value in me and led me onto the path of research. I aspire to become a statistician like you, who can originate from first principles and transit smoothly between complex theories and real-life applications. I learned the how of statistics from school. And you taught me the why. Most importantly, you gave me a job, financed my postdoctoral education, and supported me through the difficult years when I had to juggle immigration complexities, continuing school, and staying employed.

I am immensely fortunate to have had comradeship during my days juggling life, school, and work. I am grateful to my previous and current colleagues at Prosoft Clinical and Verily. Collaborating with you all has sharpened my communication and enhanced my understanding of the utility of my research work. Special thanks to my previous manager, Diane Tipping. Thank you for the encouragement and support that enabled me to balance school and work. You taught me how to work with cross-functional teams and how not to compromise statistical principles. Also, to Johnny Ho, thank you for the encouragement, proofreading my work, and for lifting the work-related burdens off my shoulders during the period when I raced

across the finish line.

I am also blessed with great friends over the years. Yulan, Yiyun, and Meng, you were always among the first to cheer when I achieved milestones and encourage me when I encountered disappointments. Big thanks to Meng for your insightful and sharp questions. They enhanced my understanding of the ADMM algorithm.

Last but not least, I am beyond grateful to have had my family alongside me during this process. Thank you to my grandparents and parents, who have raised me to value education and hard work. Thank you, especially to my grandpa, for bringing me to the wonderland of mathematics and showing me the beauty of numbers. Finally, it is to my life partner, Jayant. You have been my pillar of courage and support during these years. Because of your encouragement, I have become more confident. You have dragged me out of my several emotional downturns and when I was at edges of giving up. You have encouraged me to explore life outside my professional world. If life is an optimization problem, you have continuously bounced me off the local optimum that I tried to settle in and rebooted me for improved searches. I can't wait to embark on the next adventure with you now that I have sailed to the end of this doctoral program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AAD	Average absolute deviation
ADMM	Alternating direction method of multipliers
BIC	Bayesian Information Criteria
DB-index	Davies-Boulding Index
DPI	Duration of pause intervals
FN	False negative
FP	False positive
H-LAD	Huber's approxmation to least absoluate devianc
IRLS	Iterative reweighted least squares
LASSO	least absolute shrinkage and selection operator
LLA	Local linear approximation
MAD	Median absolute deviation
MCP	mini-max concave penalty
MLE	Maximum likelihood estimation
MSE	Mean squared error
OLS	Ordinary least square
PD	Parkinson's disease
RBD	rapid eye movement sleep behavior disorder
RI	Rand Index
RMSE	squared root of mean squared error
RST	Rate of speech timing
SCAD	smoothly clipped absolute deviations penalty
SE	Standard error
SON	Sum of norms
TN	True negative
TP	True positive

To my family

CHAPTER 1

INTRODUCTION AND LITERATURE REVIEW

1.1 K-means Clustering

Being the most well-known clustering method, k-means clustering gained its fame for the simplicity to implement and being intuitive to interpret. Using notation from Lidsten, Ohlsson, and Ljung (2011), the k-means problem is given as

$$\begin{aligned} &\min_{S} \sum_{i=1}^{k} \sum_{j \in S_i} \left\| x_j - c_i \right\|^2 \\ &\text{s.t. } c_i = \frac{1}{\text{card } S_i} \sum_{j \in S_i} x_j, \\ &\bigcup_{i=1}^{k} S_i = \{1, ..., n\}, \end{aligned} \tag{1.1}$$

where $\{x_j\}_{j=1}^{n}$ is a set of observations from $\mathbb{R}^p$ space, c_i is the centroid for cluster i, and the number of clusters k needs to be pre-defined for the problem. K-means clustering is to minimize the within-cluster sum-of-squares error, for a fixed number of clusters k. This problem in (1.1) has been shown to be NP hard, and the algorithm proposed by Lloyd 1982 is commonly used to obtain an approximate solution. The simplicity of k-means comes at a cost. It is well-known that the k-means clustering can be unstable and sensitive to initialization, due to the nonconvex optimization problem underlying the method. In practice, one usually runs the k-means clustering with multiple initial values and select the best solution. Moreover, similar to other clustering method, such as Gaussian mixture, such methods require the number of clusters k to be determined in advance.

1.2 Solution Path Convex Clustering

As a solution to the nonconvex feature of k-means clustering, Lindsten, Ohlsson, and Ljung (2011) re-formulated the problem (1.1) into a convex fashion:

$$\begin{aligned} &\min_{c} \sum_{j=1}^{n} \left\| x_j - c_j \right\|^2, \\ &\text{s.t.} \{c_1, ..., c_n\} \text{ contains k unique vectors.} \end{aligned} \tag{1.2}$$

Here each datapoint x_j has its own center c_j. Lindsten et. al. (2011) showed that the formulation in (1.2) could be mathmatically formulated into below format, by counting duplicates among vectors $\{c_j\}_{j=1}^{n}$ and utilizing what is termed as sum-of-norms (SON) clustering:

$$\begin{aligned} &\min_{c} \sum_{j=1}^{n} \left\| x_j - c_j \right\|^2, \\ &\text{s.t.} \sum_{1 \le i < j \le n} p(c_i, c_j) = \frac{3n - n^2}{2} - k, \end{aligned} \tag{1.3}$$

and $p(\cdot, \cdot) : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ is any nonnegative and symmetric function with property:

$$p(c_i, c_j) = 0 \Leftrightarrow c_i = c_j.$$

The intuition from formulation (1.1) to (1.3) is that x_j is allowed to have its own center c_j, under the constraint that $\{c_j\}_{j=1}^{n}$ contains k unique vectors. When $p(\cdot, \cdot)$ in (1.3) is chosen with care, above formulation becomes a convex problem. Common choices that lead (1.3) to become a convex problem is $p(x, y) = \|x - y\|_q$.

Note that exchanging penalties that address local weighting ($\lambda \to \lambda_{ij}$), the objective function of above clustering problem takes form,

$$L(c) = \frac{1}{2} \sum_{i=1}^{n} \|x_i - c_i\|_2^2 + \sum_{1 \le i < j \le n} \lambda_{ij} \left\| c_i - c_j \right\|_q, \tag{1.4}$$

where c is a set of c_i, i.e., $c = \{c_i\}_{i=1}^{n}$, c_i is center of the cluster that x_i belongs to, and λ_{ij} is a tuning parameter which can be driven by the pair that is being compared. $\|\cdot\|_q$ is the l_q norm, and q is usually taken from $\{1, 2, \infty\}$. As $\left\| c_i - c_j \right\|_q$ shrinks to zero, it indicates that x_i and x_j are clustered to the same group. This formulation

partitions the space into k groups, where k is not pre-defined. This formulation also simultaneously estimates clusters' centers, and assigns membership for each data point.

A solution path is constructed by tuning the value of λ_{ij}. Values of λ_{ij} can be either taken as a constant for all pairs, i.e. $\lambda_{ij} = \lambda$, or taken differently based on differently pairs. In the case where different values of λ_{ij} is chosen, λ_{ij} is usually broken down as $\lambda_{ij} = \lambda w_{ij}$, where w_{ij} is a non-negative weight. Hocking (2011) suggested Gaussian weights:

$$w_{ij} = \exp\{-\phi \left\|x_i - x_j\right\|_2^2\}, \tag{1.5}$$

where $\phi > 0$, and ϕ controls the speed of shrinkage. As shown in Figure 3.1, the larger ϕ is, the slower the values of c_i is shrunk to a common center, in other words a longer solution path. When $\phi = 0$, the penalty reduces to least absolute shrinkage and selection operator (LASSO).

With this breakdown of λ_{ij}, solution path of λ can also be constructed. When $\lambda = 0$, (1.4) is minimized when $c_i = x_i$, which indicates that each data point is their own center. As $\lambda \to \infty$, the objective function is minimized when all data points come to a common center.

1.3 Convex Clustering in Regression Setting and Unbiased Concave Penalties

Ma and Huang (2017) extended above model of convex clustering into regression setting

$$y_i = \mu_i + x_i^T \beta + \epsilon_i,\ i = 1, ..., n. \tag{1.6}$$

This model assumes that the underlying structure of the data is such that the data comes from different populations and they differ in their means after taking out confounding effect of covariates. Solutions can be formulated in the light of Chi

and Lange (2015) as following.

$$L(\mu, \beta; \lambda) = \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \mu_i - x_i^T \beta\right)^2 + \sum_{1 \le i < j \le n} p\left(|\mu_i - \mu_j|, \lambda\right), \tag{1.7}$$

where μ is a set of intercept from each observation, i.e., $\mu = \{\mu_i\}_{i=1}^n$. Here Ma and Huang (2017) introduced $p(\cdot, \lambda)$ as a concave penalty function, rather than convex as in previous applications. It is a setup with tuning parameter $\lambda \geq 0$, and μ and β are to be estimated.

One can solve above model (1.7) following Chi and Lange (2015) by setting $p(\cdot, \lambda) = \lambda \|\cdot\|_q$ with a convex penalty. However, this model setup comes at a cost. The cost emerges because that the convex penalty $\|\cdot\|_q$ brings bias. Note that q has to be greater or equal to 1 for the objective function in (1.4) and (1.7) to remain convex.

As Ma and Huang (2017) noted, with l_q penalty the clustering method tends to end up with an either small or large number of clusters. Noteworthy is that l_q penalty ensures that the objective function in (1.7) has a global minimum, and bias can be mitigated with introduction of local weights such as Gaussian weights w_{ij} as displayed in (1.5). These Gaussian weights are set up such that the closer two points are, the more penalty is enforced. Different from traditional penalties which shrink distances of all pairs of points equally and thus over-shrinks distant pairs, the weighted l_q penalty penalizes pairs of data according to their distance. As a result, observations converge to one common center more slowly over a good range of λ, and the convergence rate of clustering can be controlled by the parameter ϕ. However, this solution may not be ideal as the choice of weights can dramatically affect clustering results. For example, with selection of Gaussian weights, the question on how to select a good value of ϕ is subjective. And as pointed out by Ma and Huang (2017), in the regression settings, selection of weights can be much more challenging.

To improve the quality of clustering results, Ma and Huang (2017) was motivated to use concave penalties such as smoothly clipped absolute deviations penalty (SCAD, Fan and Li 2001) and mini-max concave penalty (MCP, Zhang 2010), which are asymptotically unbiased and more aggressive in enforcing a sparser solution. In

the univariate case, the MCP penalty takes form

$$p_\gamma(t,\lambda) = \lambda \int_0^t (1 - \frac{x}{\gamma\lambda})_+ dx, \gamma > 1,$$

and the SCAD penalty takes form

$$p_\gamma(t,\lambda) = \lambda \int_0^t \min\{1, \frac{(\gamma - x/\lambda)_+}{\gamma - 1}\} dx, \gamma > 2.$$

SCAD and MCP behave similarly and yield similar results. To simplify presentation for this paper, we will focus our discussion and present examples with MCP.

In the penalty setup, γ controls the rate of penalization, i.e., how quickly observations are compressed to one cluster, and λ controls the degree of penalization. Noteworthy is that as $\gamma \to \infty$, both MCP and SCAD penalties converge to the l_1 penalty, and as γ approaches its minimal value (in the case of MCP the minimal value is 1), the bias is minimized. Here bias is in the sense that the penalty overshrinks nonzero values. In the case of convex clustering, the nonzero values come from pairs that are distant.

The drawback with introduction of concave penalties is that the objective function is no longer convex, and the solution may be a local minimum. Ma and Huang (2017) shows that the oracle estimator of the objective function is a local minimizer with high probability. The oracle estimator is the minimizer obtained when assuming true cluster memberships are known.

Ma and Huang (2017) also extended their model to the case in which heterogeneity comes how the response variable correlates with covariates, i.e.,

$$y_i = \mu + z_i^T \theta_i + x_i^T \beta + \epsilon_i,\ i = 1, ..., n, \tag{1.8}$$

where θ_i is regression coefficients that are heterogeneous, and and z_i are covariates from which the heterogeneity comes. Similar to (1.6), μ is the common mean, β is common regression coefficients among clusters, and accordingly x_i are covariates with common effect on the response variable among clusters.

Since Ma and Huang (2017), robust subgroup identification (Zhang et al., 2019),

subgroup detection with grouped predictors (Liang et al., 2020), and subgroup identification for longitudinal data with dropouts (Lu et al., 2021) have been proposed for regression clustering under a similar framework.

1.4 ADMM Algorithm

Chi and Lange (2015) introduced a series of methods to solve the convex clustering problem shown in (1.4). Among these methods, the alternating direction method of multipliers (ADMM, Boyd et. al. 2011) is most commonly applied. ADMM solves the following equality-constrained convex optimization problem

$$\begin{aligned} &\text{minimize } f(u_1) + g(u_2), \\ &\text{subject to } Au_1 + Bu_2 = c, \end{aligned} \tag{1.9}$$

where $f(\cdot)$ and $g(\cdot)$ are convex functions. Chi and Lange (2015) employed a variable splitting method, with which minimizing of (1.4) can be turned into a similar formulation:

$$\begin{aligned} &\text{minimize } L(c) = \frac{1}{2}\sum_{i=1}^{n} \|x_i - c_i\|_2^2 + \sum_{1\le i<j\le n} \lambda_{ij} \left\|\eta_{ij}\right\|_q, \\ &\text{subject to } c_i - c_j - \eta_{ij} = 0, \end{aligned} \tag{1.10}$$

where c is a set of c_i, i.e., $c = \{c_i\}_{i=1}^n$. With the reparameterization, above formulation (1.10) is nothing but a special case of the ADMM problem (1.9). ADMM then invokes the augmented Lagrangian,

$$L_\upsilon(u_1, u_2, \nu) = f(u_1) + g(u_2) + \langle \nu, Au_1 + Bu_2 - c\rangle + \frac{\upsilon}{2}\|Au_1 + Bu_2 - c\|_2^2, \tag{1.11}$$

where dual variables ν and υ are utilized. ν is a vector (or matrix) of Lagrangian multipliers and υ is a non-negative tuning parameter. When $\upsilon = 0$, the augmented Lagrangian reduces to ordinary Lagrangian.

If we set

$$f(c) = \frac{1}{2}\sum_{i=1}^{n} \|x_i - c_i\|_2^2$$

and

$$g(\eta) = \sum_{1 \le i < j \le n} \lambda_{ij} \left\| \eta_{ij} \right\|_q ,$$

where $\eta = \{\eta_{ij}\}_{1 \le i < j \le n}$. Model in (1.10) can be formulated as

$$\begin{aligned} L_\upsilon(c, \eta, \nu; \lambda) = & \frac{1}{2} \sum_{i=1}^{n} \|x_i - c_i\|_2^2 + \sum_{1 \le i < j \le n} \lambda_{ij} \left\| \eta_{ij} \right\|_q \\ & + \sum_{1 \le i < j \le n} \langle \nu_{ij}, c_i - c_j - \eta_{ij} \rangle \\ & + \frac{\upsilon}{2} \sum_{1 \le i < j \le n} (c_i - c_j - \eta_{ij})^2 . \end{aligned}$$

ADMM then minimizes the augmented Lagrangian one block of variables at a time before updating the dual variable ν. This yields the following iterative algorithm.

$$\begin{aligned} c^{(m+1)} &= \text{argmin}_c L_\upsilon(c, \eta^{(m)}, \nu^{(m)}) \\ \eta^{(m+1)} &= \text{argmin}_\eta L_\upsilon(c^{(m+1)}, \eta, \nu^{(m)}) \\ \nu_{ij}^{(m+1)} &= \nu_{ij}^{(m)} + \upsilon(c_i^{(m+1)} - c_j^{(m+1)} + \eta_{ij}^{(m+1)}), \text{ for } 1 \le i < j \le n. \end{aligned} \tag{1.12}$$

The iteration will stop when a pre-defined convergence criterion is met. Two residuals are generally used to judge such convergence, the dual and primal residuals. Below we demonstrate the primal and dual residuals for (1.11):

$$\begin{aligned} \text{primal residual: } r &= \sum_{1 \le i < j \le n} \left\| c_i^{(m+1)} - c_j^{(m+1)} - \eta_{ij}^{(m+1)} \right\| , \\ \text{dual residual: } s &= \upsilon \sum_{1 \le i < j \le n} \left\| \eta_{ij}^{(m+1)} - \eta_{ij}^{(m)} \right\| . \end{aligned}$$

If these residuals are small enough, the algorithm is considered converged, and values in (1.12) at the given iteration will be considered solutions to the model in (1.10). As explained in Chi and Lange (2015), the purpose of splitting variables (reparameterization) is to simplify the optimization with respect to the penalty terms, which are otherwise non-separable in terms of c.

The ADMM algorithm in the setting of clustering, despite its favorable properties and intuitive interpretability, suffers from computational difficulty. The pain comes from the fact that the realization of clustering is done by a fused penalty. Given

n data points, the number of comparisons needed to be evaluated is $C_n^2 = n(n-1)/2$. The number of comparisons grows much faster than the sample size. It is computationally challenging as n scales to a large number. Potential solutions may include random sampling pairs for comparison.

1.5 Multi-response Regression

In classical regression literature, multi-response regression was not given much attention for the fact that to estimate regression coefficients of a multi-response regression of dimension q, for example, equals running q regressions for every of the q responses separately. That is, if we let $\boldsymbol{Y}$ be an $n \times q$ matrix, which contains q response variables, and n observations. m is an q-dimension vector that contains intercepts, and $\boldsymbol{X}$ is an $n \times p$ matrix for p predictors. $\boldsymbol{B}$ is a $p \times q$ matrix with regression coefficients. Lastly, $\boldsymbol{E}$ is the independent random error matrix with dimension $n \times q$, and it has zero mean and covariance $\boldsymbol{\Sigma}$. We have model:

$$\boldsymbol{Y} = \mathbb{1}_n m^T + \boldsymbol{X}\boldsymbol{B} + \boldsymbol{E}, \tag{1.13}$$

where $\mathbb{1}_n$ is a n-dimension vector. Above multi-response regression problem can be simultaneously modeled by q single-response regression models:

$$y_{(j)} = \mathbb{1}_n \mu_{(j)} + \boldsymbol{X}\beta_{(j)} + \epsilon_{(j)}, \text{ and } j = 1, ..., q, \tag{1.14}$$

where $y_{(j)}$ is the jth column of $\boldsymbol{Y}$, $\mu_{(j)}$ is the jth element of μ, $\beta_{(j)}$ is the jth column of $\boldsymbol{B}$, and $\epsilon_{(j)}$ is the jth column of $\boldsymbol{E}$. The ordinary least square (OLS) estimate of $\boldsymbol{B}$ in (1.13) is the combined OLS estimates of $\beta_{(j)}$ in (1.14). However, this easy translation between separate OLS estimates and simultaneous estimate of multi-response coefficients no longer holds in inference, in which case the correlation between different response variables emerges in the off-diagonal cells of the estimated covariance matrix $\hat{\boldsymbol{\Sigma}}$. The OLS approach to estimate coefficients in the multi-response regression, despite being simple to implement, makes no use of the association between response variables.

Recent advancements in tackling the multivariate multiple regression problem, which can be multi-response and/or multi-covariates regression, consider the setup (1.13) as a high-dimensional problem. Note that the naive OLS approach to estimate $\boldsymbol{B}$ is not a high-dimensional problem. Common directions include regularized estimation and reduced-rank methods, and both approaches assume the underlying connectivity between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is sparse.

The former approach focuses mostly on feature selection with single response variable, i.e., $q = 1$. Huang et. al (2012) has a comprehensive review for this line of methods. The later ones consider the coefficient matrix $\boldsymbol{B}$ as whole, and by casting a low-rank constraint it translates the problem to dimension reduction for both $\boldsymbol{X}$ and $\boldsymbol{Y}$. It is more commonly used in cases where both p and q are greater than 1. Li, Liu, and Chen (2019) bridged these two approaches by introducing an integrative reduced-rank regression. This method aimed to enhance multi-response regression performance by conducting dimension reduction through casting reduced-rank constraint and sparse selection of feature groups. The reduced-rank nature of the method gives convenience for it to perform well in $q > 1$ cases.

While traditional high-dimensional regression problems deal with estimation for a set of coefficient parameters that are either a vector or a matrix, recent development of the problem explored possibilities of higher-order tensor structures for these parameters. Raskutti, Yuan, and Chen (2019) framed the high-dimensional multi-response tensor regression problem as a convex regularization problem. The application of higher-order of multi-response regression gained its attention from applications such as image analysis, text mining, and audio classification.

1.6 Robust Solutions

1.6.1 The Problem

It is well known that the presence of high-leverage points (e.g. outlier) poses challenges for modeling techniques that base on least square formulation (Euclidean distance). For example in the clustering setting using K-means, the Euclidean based loss entails that the average of the coordinates of data points in a cluster is the centroid of the cluster. This is problematic when there are outliers and outliers can dramatically

shift the location of the centroid, thus impair results of K means. Raykov et. al. 2016 demonstrated such examples with a visualization, see Figure 1.1.

In the regression setting the presence of such outliers may increase the sum square of residuals sharply and overweigh such observations in the estimation of regression parameter. As a result, these observations may tilt the estimate regression coefficients in favor of them. See an example in Figure 1.2.

The above problem is worsened when the underlying distribution deviates from Normal, such as a distribution with fat tail. It implies the the frequency of such data points increase and it will lead to an increased influence.

1.6.2 The Robust Loss Approach

The Robust Estimators

As mentioned above, the culprit of the problem is the least-square based loss. In theory one can use absolute loss in replacement. However, in practice, absolute loss is hard to work with because it is not differentiable. An elegant compromise was introduced by Huber in 1964. Another well-known robust loss is Tukey biweight 1974. These methods fall in the big family of M-estimator, M comes from **M**inimizing the loss function. Maximum Likelihood Estimators (MLEs) also belong to the M-estimator family. See Figure 1.3 for a visualization of how the objective functions and weight functions compare between least-square, huber, and tukey loss.

To solve such loss functions, the problem is typically turned into a weighted least square and there are closed form solutions to weighted least squares. Conceptually, these multiplicative weights can help downweight bad observations. Solving for these weights depends on the residuals, the residuals depend on the estimated coefficients, and the estimated coefficients depend on the weights. An iterative solution is therefore required, it is termed the iteratively reweighted least-squares (IRLS).

Robust solutions in the regression setting has been widely used in model fitting for real-world applications, such as image processing, ecnometrics, social sciences, and etc. Diego et. al. (2021) provides a comprehensive literature review for a collection of M-estimator based robust models. Relevant methods and extensions include least absolute deviation (LAD) regression, quantile regression, and etc.

In the clustering setting, a natural transition for quantifying distance using Euclidean distance to Manhattan distance in the context of K-means modeling is referred to as K-medians clustering. Various methods have been proposed to solve clustering problem with deviation from Normal. Dohan et. al. (2015) provides a review of existing theoretical and empirical work to solve such formulation. Note that median in multi-dimensional space is known as geometric median. Instead of using sum of l_1 norm across dimensions, one measures distance by l_2 norm.

Combining Robust Estimators with Solution Path Clustering

In the context of solution path clustering, Schuberg (2019) proposed introducing robust loss functions for the goodness-of-fit term in (1.4). Schuberg's work was established on the convex-fied K means and the model as specified in (1.4) can be formulated as

$$L(c) = \frac{1}{2}\sum_{i=1}^{n} h(x_i - c_i) + \sum_{1\le i<j\le n} \lambda_{ij} \left\| c_i - c_j \right\|_q , \tag{1.15}$$

where $h(\cdot)$ is a robust loss function, such as absolute loss (l_1 norm, Manhattan distance), which leads to that the median, instead of the mean, becomes the solution. And medians are in principle robust to outliers when compared with the mean.

The solution to the new formulation with robust loss differs in the step to obtain $c^{(m+1)}$ as in (1.12). With robust loss added, solving for $c^{(m+1)}$ may not be analytically tractable and iterative methods should be used. Iterative reweighted least square (IRLS) in combination of ADMM algorithm was developed to solve the objective function (1.15) in Schuberg (2019). ADMM was used to tackle the non-separability of the penalty term, while IRLS method was used to handle robust loss which turns solving a non-differentiable problem to a weighted least square problem, i.e. turning (1.15) into below form.

$$L(c, W) = \frac{1}{2}\sum_{i=1}^{n} w_i(x_i - c_i) + \sum_{1\le i<j\le n} \lambda_{ij} \left\| c_i - c_j \right\|_q , \tag{1.16}$$

where $W = \text{diag}\{w_i, i = 1, \ldots, n\} \in \mathbb{R}^n$.

Specifically, as described by Schuberg (2019), the IRLS based solution comes from the concept of majorization-minimization (Lange 2004). It states that one can find a

surrogate function which bounds the original loss function and is easy to optimize. Minimizing the surrogate function will yield an update that also minimize in the original loss function. In the given problem (1.15), the surrogate function is chosen to bound $h(x_i - c_i)$. If we let $e_i = (x_i - c_i)^2$ and $e_i^{(t)} = (x_i - c_i^{(t)})^2$, it is important to keep in mind that most robust loss functions $h(\sqrt{e_i})$ are concave for $e_i \geq 0$. A concave function $f(\cdot)$ satisfies

$$f(y) \leq f(x) + f'(x)(y - x), \forall x, y.$$

Given concavity, it is easy to see that

$$h(\sqrt{e_i}) \leq h(\sqrt{e_i^{(t)}}) + \frac{h'(\sqrt{e_i^{(t)}})}{2\sqrt{e_i^{(t)}}}(e_i - e_i^{(t)}).$$

This yields

$$\begin{aligned}
\sum_{i=1}^{n} h(x_i - c_i) &= \sum_{i=1}^{n} h(\sqrt{e_i}) \\
&\leq \sum_{i=1}^{n} \Big[h(\sqrt{e_i^{(t)}}) + \frac{h'(\sqrt{e_i^{(t)}})}{2\sqrt{e_i^{(t)}}}(e_i - e_i^{(t)}) \Big] \\
&= \sum_{i=1}^{n} h(x_i - c_i^{(t)}) + \sum_{i=1}^{n} \frac{h'(x_i - c_i^{(t)})}{2(x_i - c_i^{(t)})} \big[(x_i - c_i)^2 - (x_i - c_i^{(t)})^2 \big] \\
&:= u_o(c, c^{(t)}).
\end{aligned}$$

Solution to the surrogate function $u_0(c, c^{(t)})$ will lead to solution for $\sum_{i=1}^{n} h(x_i - c_i)$. Minimum of $\sum_{i=1}^{n} h(x_i - c_i)$ over c is equivalent to minimum of $\sum_{i=1}^{n} h(\sqrt{e_i})$ over e, when $x^T = \{x_i, i = 1, \ldots, n\}$ is fixed. As a result, we have

$$\begin{aligned}
\min_e \sum_{i=1}^{n} \frac{h'(\sqrt{e_i^{(t)}})}{2\sqrt{e_i^{(t)}}} e_i &= \min_c \sum_{i=1}^{n} \frac{h'(x_i - c_i^{(t)})}{2(x_i - c_i^{(t)})} (x_i - c_i)^2 \\
&= \min_c \sum_{i=1}^{n} w_i (x_i - c_i)^2,
\end{aligned}$$

where w_i is defined as

$$w_i := w(x_i, c_i^{(t)}) = \frac{h'(x_i - c_i^{(t)})}{2(x_i - c_i^{(t)})}, \tag{1.17}$$

Through above derivation minimizing (1.15) is turned into a problem of minimizing (1.16), a weighted least square problem. There is close form solution to a weighted least square problem and the ADMM algorithm can be deployed with an added step to compute the weights. Schuberg (2019) also extended their work to handle multi-dimensional clustering.

Zhang (2019) also established their work based on the same robust formulation, specifically with absolute loss and concave penalties. Instead of solving the formulation with the ADMM algorithm the authors proposed their solutions using local linear approximation (LLA, Zou and Li 2008). Despite unclear advantage in clustering and estimation results over Ma and Huang (2017), their method is much faster in convergence than the algorithms based on ADMM.

1.6.3 The Mean Shift Approach

Another school of thoughts handling outliers follows the idea that the location of outliers is shifted from the data center. This philosophy was popularized by She and Owen (2011). They established their work in the regression setting. The model adds a mean shift vector in the goodness-of-fit part of the objective function to represent the location shifts from outliers, while a penalty is used to ensure sparsity of the shifted-mean vector. Interestingly, She and Owen (2011) showed that with l_1 penalty the minizer of their mean-shift formulation matches those from minimizing Huber's formulation of the regression problem.

Witten (2013) utilizes the mean-shift philosophy in the K-means clustering. Witten's work was based on the original formulation of K-means as specified in (1.1),

and the extension is spelt as

$$\begin{aligned} &\min_{S,\boldsymbol{E}} \sum_{i=1}^{K} \sum_{j \in S_i} \left\| x_j - e_j - c_i \right\|^2 + p(\left\| e_j \right\|, \lambda) \\ &\text{s.t. } c_i = \frac{1}{\text{card } S_i} \sum_{j \in S_i} x_j, \\ &\bigcup_{i=1}^{K} S_i = \{1, ..., n\}. \end{aligned} \tag{1.18}$$

This formulation is intuitive. If an observation x_j does not belong to any cluster S_i, then e_j will neutralize the nonzero value so that $x_j - e_j$ can be included to a cluster. The regularization component $p(||e_j||, \lambda)$ encourages sparsity for the outlier neutralizing / shifted-mean component e_j.

Witten proposed an algorithm that is combined with transitional K means to solve the above minimization. The algorithm iterates between K means and solving for the shifted-mean vector $||e_j||$. In order to solve $||e_j||$ in each iteration, Witten used group LASSO approach, $p(||e_j||; \lambda) = \lambda ||e_j||$, and the solution follows as

$$e_j = (x_j - c_i) \max\{0, 1 - \frac{\lambda}{||x_j - c_i||}\}.$$

After convergence, i.e. after outliers are identified, a new K means is exercised among non-outlier data pool, i.e., observations in the set $\{j : ||e_j|| = 0\}$. Witten also demonstrated the close connection between the mean-shift K means formulation and Huber's estimator from the robust statistics framework.

Witten's work has left K as known and proposed a simple solution to tune the hyper-parameter λ. Specifically, λ is selected at the largest value such that no observation with $e_j = 0$ has $||x_j - c_i(j)||$ larger than $m(\lambda) + 3s(\lambda)$, where $m(\lambda)$ and $s(\lambda)$ are the mean and standard deviation of $||x_j - c_i(j)||$ for all observations with $e_j = 0$.

1.7 Outline of the Work

The following chapters are divided into two parts.

In Part I, we extend the solution path clustering work in regression setting from a

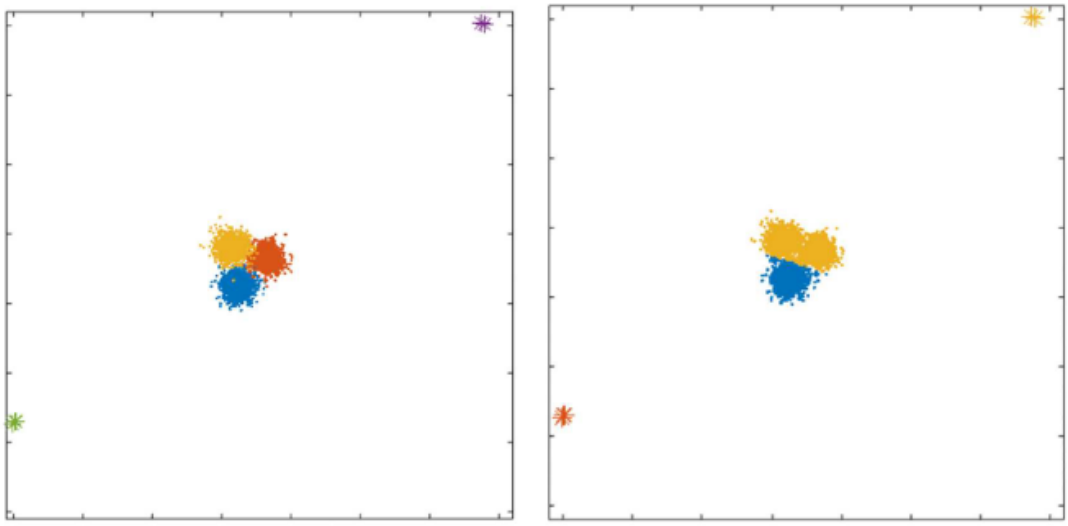


FIGURE 1.1: K Means clustering results for Gaussian data with outliers (taken from Rakov et. al. 2016)

uni-dimension problem to multi-dimension. This part consists 3 chapters. In Chapter 2, we lay out the mathematical formulation of our extension and we update the ADMM algorithm used to solve the formulation. In Chapter 3 and Chapter 4 we test the performance of the extension with simulated and empirical data sets.

In Part II, we propose two robust solutions to clustering in regression setting. This part is made up by 4 chapters. In Chapter 5 we lay out the addition of robust loss to solution path clustering of regression discussed in Part I and update the algorithm to solve such formulation. In Chapter 6 we extend the mean-shift based K-means into regression setting. And in Chapter 7 and 8 we study the performance of these two robust solutions with simulated and empirical examples.

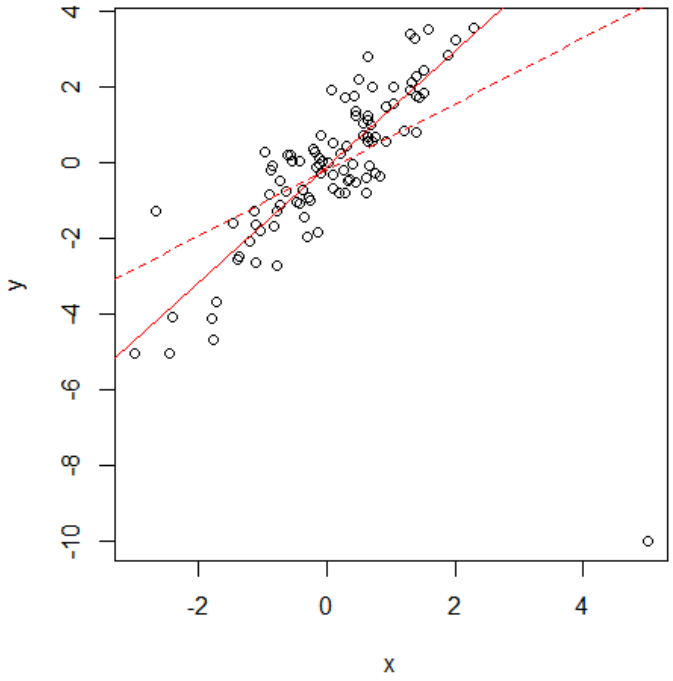


FIGURE 1.2: Least square regression estimate with outlier. Solid line is the OLS estimate without outlier and the dashed line is the OLS estimate with all data.

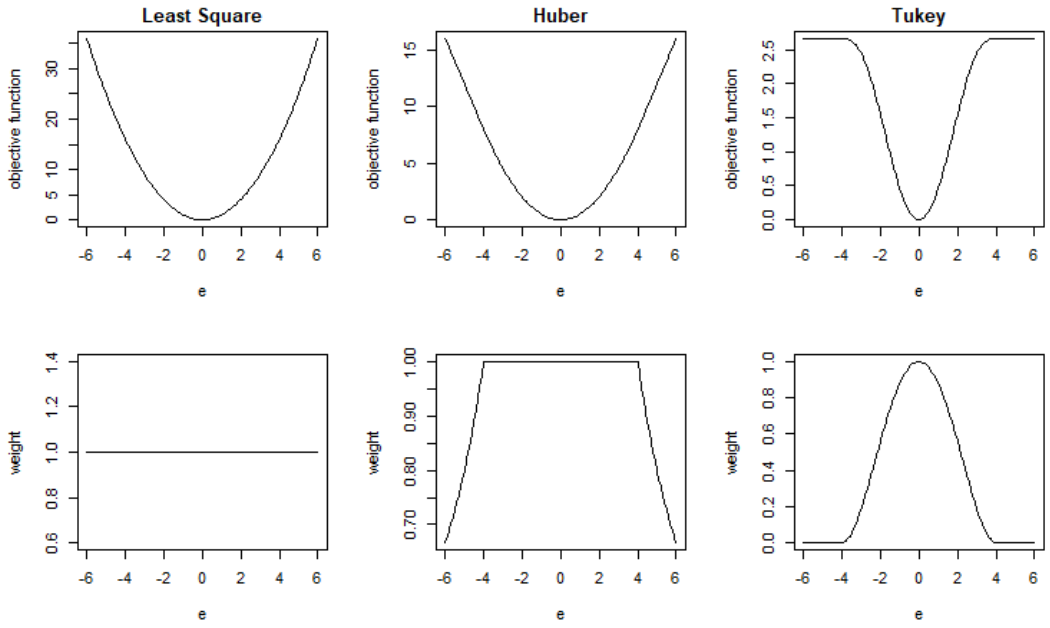


FIGURE 1.3: Objective (top) and weight (bottom) functions for the least-squares (left), Huber (middle), and tukey (right) estimators.

Part I

MULTI-RESPONSE EXTENSION TO CONCAVE PAIRWISE FUSION APPROACH IN REGRESSION CLUSTERING

CHAPTER 2

MODEL CONSTRUCTION AND COMPUTATIONAL ALGORITHM

2.1 Model 1: Multi-Response Model with Subject-specific Intercepts

Given model

$$y_i = \mu_i + \boldsymbol{B}^T x_i + \epsilon_i, \tag{2.1}$$

where $y_i, \mu_i, \epsilon_i \in \mathbb{R}^q$, $x_i \in \mathbb{R}^p$, and $\boldsymbol{B} \in \mathbb{R}^{p \times q}$. In this model y_i is a multivariate response variable of subject i, and μ_i is the subject-specific intercept. We assume the heterogeneity is driven by these subject-specific intercepts and after extracting common factors $\boldsymbol{B}$ from pre-specified covariates x_i, the heterogeneity can be modeled. Error term ϵ_i is independent of x_i with mean 0 and variance σ^2.

We will estimate $\boldsymbol{B}$ and μ_i simultaneously, with the understanding that good estimation of $\boldsymbol{B}$ amplifies the chance of efficient subgroup identification, and thus good estimation of μ_i. Subgroups in this setup is identified by μ_i, and the i^{th} and j^{th} observations with $\mu_i - \mu_j = 0$ are considered to be from the same group. It is assumed that the underlying data to be analyzed are from K groups with $K \geq 1$. Let $G = (G_1, ..., G_K)$ be a partition of $1, ..., n$, we have $\mu_i = \alpha_k$ for all $i \in G_k$, where α_k is the common value for μ_i of all the subjects in the same group.

Above estimation problem is solved via a least square loss setup,

$$S(\mu, \boldsymbol{B}) = \frac{1}{2} \sum_{i=1}^{n} \left\| y_i - \mu_i - \boldsymbol{B}^T x_i \right\|^2 + \sum_{1 \leq i < j \leq n} p(\mu_i - \mu_j, \lambda), \tag{2.2}$$

with penalty element $\sum_{1\le i<j\le n} p(\mu_i-\mu_j,\lambda)$ and $\lambda>0$, is used to estimate parameters in above loss function. Here $\mu=\{\mu_1^T,\mu_2^T,...,\mu_n^T\}^T$ and $\|x\|=(\sum_{i=1}^n x_i^2)^{1/2}$. Parameters μ_i are non separable in the loss function, and the augmented Lagrangian method is used by reparameterizing $\mu_i-\mu_j$.

Let $\eta_{ij}=\mu_i-\mu_j$, we can rewrite above objective function as

$$\begin{aligned} S(\mu,\boldsymbol{B},\eta)&=\frac{1}{2}\sum_{i=1}^{n}\left\|y_i-\mu_i-\boldsymbol{B}^T x_i\right\|^2+\sum_{1\le i<j\le n} p(\eta_{ij},\lambda),\\ &\text{subject to } \mu_i-\mu_j=\eta_{ij}. \end{aligned} \tag{2.3}$$

Here $\eta=\{\eta_{ij}^T, i<j\}^T$. To obtain the solutions, we start by introducing a set of dual variables $\nu=\{\nu_{ij}^T, i<j\}^T$ as the Lagrange multipliers and υ as the penalty parameter. The loss function is turned into

$$L(\mu,\boldsymbol{B},\eta,\nu)=S(\mu,\boldsymbol{B},\eta)+\sum_{1\le i<j\le n}\nu_{ij}^T(\mu_i-\mu_j-\eta_{ij})+\frac{\upsilon}{2}\sum_{1\le i<j\le n}\left\|\mu_i-\mu_j-\eta_{ij}\right\|^2. \tag{2.4}$$

Parameters μ, $\boldsymbol{B}$, η, and ν are solved iteratively through the ADMM algorithm.

Step 1 Given (η,ν), update $(\mu,\boldsymbol{B})$. In this step, we estimate μ and $\boldsymbol{B}$ by minimizing the objective function $L(\mu,\boldsymbol{B},\eta,\nu)$ as in (2.4) in relation to μ and $\boldsymbol{B}$, assuming that η and ν are known. We rewrite the objective function, such that

$$L(\mu,\boldsymbol{B};\eta,\nu)=\frac{1}{2}\sum_{i}^{n}\left\|y_i-\mu_i-\boldsymbol{B}^T x_i\right\|^2+\frac{\upsilon}{2}\left\|\boldsymbol{A}\mu-\eta+\upsilon^{-1}\nu\right\|^2+C,$$

here we have

$$\boldsymbol{A}=\boldsymbol{\Delta}\otimes\boldsymbol{I}_q\in\mathbb{R}^{dq\times nq},\ \text{and}\ d=\frac{n(n-1)}{2},$$

where $\boldsymbol{\Delta}=\left\{(e_i-e_j), i<j\right\}^T\in\mathbb{R}^{d\times n}$, e_i is a column vector with the i^{th} element as 1 and the rest as 0, and I_q is an identity matrix with q dimension. And again $\mu=\{\mu_1^T,\mu_2^T,...,\mu_n^T\}^T\in\mathbb{R}^{nq}$, $\eta=\{\eta_{ij}^T, i<j\}^T\in\mathbb{R}^{dq}$, and $\nu=\{\nu_{ij}^T, i<j\}^T\in\mathbb{R}^{dq}$.

Next, we let

$$\begin{aligned}\boldsymbol{M} &= (\mu_1, \mu_2, ..., \mu_n)^T \in \mathbb{R}^{n\times q}\\ \boldsymbol{Y} &= (y_1, y_2, ..., y_n)^T \in \mathbb{R}^{n\times q}\\ \boldsymbol{X} &= (x_1, x_2, ..., x_n)^T \in \mathbb{R}^{n\times p}\\ \boldsymbol{E} &= (\epsilon_1, \epsilon_2, ..., \epsilon_n)^T \in \mathbb{R}^{n\times q}\end{aligned}$$

then we have,

$$\begin{aligned}\sum_{i=1}^{n}\left\|y_i - \mu_i - \boldsymbol{B}^T x_i\right\|^2 &= \sum_{i=1}^{n}\epsilon_i^T\epsilon_i\\ &= \text{trace}(\boldsymbol{E}\boldsymbol{E}^T)\\ &= \text{trace}[(\boldsymbol{Y}^* - \boldsymbol{X}\boldsymbol{B})(\boldsymbol{Y}^* - \boldsymbol{X}\boldsymbol{B})^T].\end{aligned}$$

Here $\boldsymbol{Y}^* = \boldsymbol{Y} - \boldsymbol{M} \in \mathbb{R}^{n\times q}$. The minimizer of $\boldsymbol{B}$ in this case is

$$\hat{\boldsymbol{B}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{M}). \tag{2.5}$$

If we define $\boldsymbol{Q} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$, the squared loss can be written as

$$\sum_{i=1}^{n}\left\|y_i - \mu_i - \boldsymbol{B}^T x_i\right\|^2 = \text{trace}\Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M})(\boldsymbol{Y} - \boldsymbol{M})^T(\boldsymbol{I}_n - \boldsymbol{Q})^T\Big]. \tag{2.6}$$

Next, we want to rewrite the above in terms of μ (not $\boldsymbol{M}$), so that we can solve for μ. Note that

$$\begin{aligned}\text{trace}(\boldsymbol{A}\boldsymbol{A}^T) &= \text{trace}\Big[\text{vec}(\boldsymbol{A})\text{vec}^T(\boldsymbol{A})\Big]\\ &= \text{vec}^T(\boldsymbol{A})\text{vec}(\boldsymbol{A}),\end{aligned} \tag{2.7}$$

and $\text{vec}(\boldsymbol{A}\boldsymbol{X}\boldsymbol{B}) = (\boldsymbol{B}^T \otimes \boldsymbol{A})\text{vec}(\boldsymbol{X})$ for any arbitrary matrix $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{X}$. Knowing that $\mu = (\mu_1^T, ..., \mu_n^T)^T$ and $y = (y_1^T, ..., y_n^T)^T$, we have

$$\begin{aligned}\text{vec}\Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M})\Big] &= \text{vec}\Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M})\boldsymbol{I}_q\Big]\\ &= \Big[\boldsymbol{I}_q \otimes (\boldsymbol{I}_n - \boldsymbol{Q})\text{vec}(\boldsymbol{Y} - \boldsymbol{M})\Big]\\ &= \boldsymbol{D}(y - \mu),\end{aligned} \tag{2.8}$$

where $\boldsymbol{D} = \boldsymbol{I}_q \otimes (\boldsymbol{I}_n - \boldsymbol{Q})$. Combining results (2.6), (2.7), and (2.8) together, it leads to

$$\begin{aligned}\sum_{i=1}^{n} \left\| y_i - \mu_i - \boldsymbol{B}^T x_i \right\|^2 &= \text{trace}\Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M})(\boldsymbol{Y} - \boldsymbol{M})^T (\boldsymbol{I}_n - \boldsymbol{Q})^T \Big] \\ &= \text{vec}^T \Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M}) \Big] \text{vec} \Big[(\boldsymbol{I}_n - \boldsymbol{Q})(\boldsymbol{Y} - \boldsymbol{M}) \Big] \\ &= \| \boldsymbol{D}(y - \mu) \|^2 \end{aligned} \tag{2.9}$$

Finally, the loss function becomes

$$L(\mu, \boldsymbol{B}; \eta, \nu) = \frac{1}{2} \| \boldsymbol{D}(y - \mu) \|^2 + \frac{\upsilon}{2} \left\| \boldsymbol{A}\mu - \eta + \upsilon^{-1}\nu \right\|^2,$$

and we can take derivative with respect to μ:

$$\begin{aligned} &\boldsymbol{D}^T\boldsymbol{D}(\mu - y) + \upsilon \boldsymbol{A}^T(\boldsymbol{A}\mu - \eta + \upsilon^{-1}\nu) = 0 \\ &\boldsymbol{D}^T\boldsymbol{D}\mu - \boldsymbol{D}^T\boldsymbol{D}y + \upsilon \boldsymbol{A}^T\boldsymbol{A}\mu - \upsilon \boldsymbol{A}^T\boldsymbol{A}(\eta - \upsilon^{-1}\nu) = 0 \\ &(\boldsymbol{D}^T\boldsymbol{D} + \upsilon \boldsymbol{A}^T\boldsymbol{A})\mu = \boldsymbol{D}^T\boldsymbol{D}y + \upsilon \boldsymbol{A}^T(\eta - \upsilon^{-1}\nu). \end{aligned}$$

The solution for μ is

$$\hat{\mu} = (\boldsymbol{D}^T\boldsymbol{D} + \upsilon \boldsymbol{A}^T\boldsymbol{A})^{-1} \Big[\boldsymbol{D}^T\boldsymbol{D}y + \upsilon \boldsymbol{A}^T(\eta - \upsilon^{-1}\nu) \Big]. \tag{2.10}$$

As in the iterations, solutions of μ and $\boldsymbol{B}$ depends on estimates from the previous step. Thus, in-iteration solutions in (2.5) and (2.10) can be expressed as

$$\begin{aligned} \boldsymbol{B}^{(m+1)} &= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{M}^{(m)}), \\ \mu^{(m+1)} &= (\boldsymbol{D}^T\boldsymbol{D} + \upsilon \boldsymbol{A}^T\boldsymbol{A})^{-1} \Big[\boldsymbol{D}^T\boldsymbol{D}y + \upsilon \boldsymbol{A}^T(\eta^{(m)} - \upsilon^{-1}\nu^{(m)}) \Big]. \end{aligned}$$

Step 2 Update η given $(\boldsymbol{B}, \mu, \nu)$. Details will be provided in Section 2.2.

Step 3 Update ν given $(\boldsymbol{B}, \mu, \eta)$.

$$\nu_{ij}^{(m+1)} = \nu_{ij}^{(m)} + \upsilon(\mu_i^{(m+1)} - \mu_j^{(m+1)} - \eta_{ij}^{(m+1)}) \tag{2.11}$$

Iterations will stop when the pre-defined convergence criteria are met.

$$\text{primal residual: } r = \sum_{1\le i<j\le n} \left\| \mu_i^{(m+1)} - \mu_j^{(m+1)} - \eta_{ij}^{(m+1)} \right\| < \epsilon,$$
$$\text{dual residual: } s = \upsilon \sum_{1\le i<j\le n} \left\| \eta_{ij}^{(m+1)} - \eta_{ij}^{(m)} \right\| < \epsilon,$$

for any $\epsilon > 0$.

2.2 ADMM with component-wise and group-wise MCP

Going back to the objective function as in (2.4),

$$L(\mu, \boldsymbol{B}, \eta, \nu) = S(\mu, \boldsymbol{B}, \eta) + \sum_{1\le i<j\le n} \nu_{ij}^T(\mu_i - \mu_j - \eta_{ij}) + \frac{\upsilon}{2} \sum_{1\le i<j\le n} \left\| \mu_i - \mu_j - \eta_{ij} \right\|^2,$$

we will minimize in relation to η. Note that minimize $L(\eta; \boldsymbol{B}, \mu, \nu)$ with respect to η is a piece-wise minimization problem. In other words,

$$\begin{aligned} \text{argmin}_\eta L(\eta; \mu, \boldsymbol{B}, \nu) &= \text{argmin}_{\eta_{ij}} L_{ij}(\eta_{ij}; \mu, \boldsymbol{B}, \nu) \\ &= \text{argmin}_{\eta_{ij}} \left\{ \frac{\upsilon}{2} \left\| (\mu_i - \mu_j + \upsilon^{-1}\nu_{ij}) - \eta_{ij} \right\|^2 + p_\gamma(\eta_{ij}, \lambda) \right\} \end{aligned} \tag{2.12}$$

Different solutions can be obtained, given different setup of the penalty element $p_\gamma(\eta_{ij}, \lambda)$, and here we introduce a second penalty-related parameter γ so that we can extend the function $p(\eta_{ij}, \lambda)$ to MCP and SCAD. This paper is mostly discussed by utilizing penalty of MCP (Zhang 2010).

Component-wise MCP The most natural extension of univariate MCP to multivariate case is to set the penalty term in a component-wise fashion, noting that $\eta_{ij} = \left(\eta_{ij,1}, ..., \eta_{ij,q}\right)^T$. Thus for the l^{th} penalty term, where $l \in \{1, ..., q\}$ the penalty function is

$$p_\gamma(\eta_{ij,l}, \lambda) = p_\gamma(|\eta_{ij,l}|, \lambda) = \lambda \int_0^{|\eta_{ij,l}|} (1 - \frac{x}{\gamma\lambda})_+ dx. \tag{2.13}$$

As a result,

$$\hat{\eta}_{ij,l} = \begin{cases} \frac{\gamma}{\gamma - 1/\upsilon} \text{ST}(\xi_{ij,l}, \frac{\lambda}{\upsilon}), \text{ if } |\xi_{ij,l}| \le \gamma\lambda \\ \xi_{ij,l}, \text{ otherwise.} \end{cases} \tag{2.14}$$

where $\xi_{ij,l}$ is the l^{th} element of $\xi_{ij} = \mu_i - \mu_j + \upsilon^{-1}\nu_{ij}$. And $\text{ST}(z, a) = (1 - \frac{a}{|z|})_+ z$ is a soft-thresholding operator, with $(x)_+ = \max(0, x)$.

Group-wise MCP As $\eta_{ij} = \mu_i - \mu_j$ represents the distance between two subjects i and j after taking out common confounding factors and η_{ij} is in a q dimensional space, it is preferred that each element of η_{ij} is treated as a unit. Following the idea of group LASSO, Huang et. al. (2012) extended MCP into high-dimensional space, and we will term the penalty as the group-wise MCP in this paper.

The group-wise MCP takes form

$$p_\gamma(\|\eta_{ij}\|, \lambda) = \lambda \int_0^{\|\eta_{ij}\|} (1 - \frac{x}{\gamma\lambda})_+ dx, \tag{2.15}$$

and the solution to the equation is

$$\hat{\eta}_{ij} = \begin{cases} \frac{\gamma}{\gamma - 1/\upsilon} \text{ST}_G\left(\xi_{ij}, \frac{\lambda}{\upsilon}\right), \text{ if } \|\xi_{ij}\| \leq \gamma\lambda \\ \xi_{ij}, \text{ otherwise.} \end{cases} \tag{2.16}$$

where $\text{ST}_G(Z, a) = (1 - \frac{a}{\|Z\|})_+ Z$ is a group soft-thresholding operator, and again $\|x\| = (\sum_{i=1}^n x_i^2)^{1/2}$.

2.3 Model 2: Multi-Response Model with Subject-specific Slopes

Model 1 (2.1) can be extended to the case in which one can model the subject-specific slopes, i.e., the heterogeneity presents in response's relationship to covariates. The model can be stated as following

$$y_i = \boldsymbol{B}_i^T z_i + \epsilon_i, \tag{2.17}$$

where $y_i, \epsilon_i \in \mathbb{R}^q$, $z_i \in \mathbb{R}^p$, and $\boldsymbol{B}_i \in \mathbb{R}^{p \times q}$. In this model y_i is a multivariate response variable of subject i, and $\boldsymbol{B}_i$ is the subject-specific slope. We assume that the heterogeneity is driven by these subject-specific slopes. Error term ϵ_i is independent of z_i, and it has mean 0 and variance σ^2. Note that this setup can model subject-specific intercepts with $z_{i1} = 1$.

Following (2.2), the objective function of this model can be displayed as

$$S(\boldsymbol{B}^*, \boldsymbol{H}) = \frac{1}{2}\sum_{i=1}^{n}\left\|y_i - \boldsymbol{B}_i^T z_i\right\|^2 + \sum_{1\le i<j\le n} p(\left\|\boldsymbol{H}_{ij}\right\|, \lambda),$$
$$\text{subject to } \boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} = \boldsymbol{0},$$

where $\boldsymbol{B}^*$ is a set of $\boldsymbol{B}_i$, i.e., $\left\{\boldsymbol{B}_i\right\}_{i=1}^n$, and $\boldsymbol{H} = \left\{\boldsymbol{H}_{ij}\right\}_{1\le i<j\le n}$. Similarly, by augmented Lagrangian, we have loss function

$$\begin{aligned}
L(\boldsymbol{H}, \boldsymbol{B}, \boldsymbol{N}) &= S(\boldsymbol{B}, \boldsymbol{H}) + \sum_{1\le i<j\le n} < \boldsymbol{N}_{ij}, \boldsymbol{B}_i - \boldsymbol{B}_j + \boldsymbol{H}_{ij} >_F + \frac{\upsilon}{2}\left\|\boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij}\right\|^2 \\
&= S(\boldsymbol{B}, \boldsymbol{H}) + \sum_{1\le i<j\le n} < \boldsymbol{N}_{ij}, \boldsymbol{B}_i - \boldsymbol{B}_j + \boldsymbol{H}_{ij} >_F + \frac{\upsilon}{2}\sum_{1\le i<j\le n}\left\|\boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij}\right\|^2 \\
&+ \frac{1}{2}\sum_{1\le i<j\le n}\left\|\boldsymbol{N}_{ij}\right\|^2 - \frac{1}{2}\sum_{i<j}\left\|\boldsymbol{N}_{ij}\right\|^2 \\
&= \frac{1}{2}\sum_{i=1}^{n}\left\|y_i - z_i^T\boldsymbol{B}_i\right\|^2 + \frac{\upsilon}{2}\sum_{1\le i<j\le n}\left\|\boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} + \upsilon^{-1}\boldsymbol{N}_{ij}\right\|^2 + C \\
&= L_1 + L_2 + C,
\end{aligned} \tag{2.18}$$

where $\boldsymbol{N}_{ij}, \boldsymbol{B}_i, \boldsymbol{B}_j \in \mathbb{R}^{p\times q}$, and $\boldsymbol{N}$ is a set of $\boldsymbol{N}_{ij}$, i.e., $\boldsymbol{N} = \left\{\boldsymbol{N}_{ij}\right\}_{1\le i<j\le n}$. And $\langle\cdot,\cdot\rangle_F$ is the Frobenius inner product, $\langle \boldsymbol{A}, \boldsymbol{B}\rangle_F = \sum_{i,j}\boldsymbol{A}_{(i,j)}\boldsymbol{B}_{(i,j)}$, in which A and B are matrices with same dimension, and $\boldsymbol{A}_{(i,j)}$ is the element from the i^{th} row and j^{th} column of matrix $\boldsymbol{A}$. For matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\| = \left(\sum_{i,j}\boldsymbol{A}_{(i,j)}^2\right)^{(1/2)}$.

Let

$$\begin{aligned}
\beta &= \left\{\text{vec}^T(\boldsymbol{B}_1), ..., \text{vec}^T(\boldsymbol{B}_n)\right\}^T \in \mathbb{R}^{npq} \\
\eta &= \left\{\text{vec}^T(\boldsymbol{H}_{ij}), i<j\right\}^T \in \mathbb{R}^{dpq} \\
\nu &= \left\{\text{vec}^T(\boldsymbol{N}_{ij}), i<j\right\}^T \in \mathbb{R}^{dpq},
\end{aligned}$$

where $d = \frac{n(n-1)}{2}$. Next, the goal is to put L_1 and L_2 as a function of β, η, and ν.

Since

$$\begin{aligned}
\boldsymbol{B}_i^T z_i &= \text{vec}\left(\boldsymbol{B}_i^T z_i\right) \\
&= \text{vec}\left(z_i^T\boldsymbol{B}_i\boldsymbol{I}_q\right) \\
&= \left(\boldsymbol{I}_q \otimes z_i^T\right)\text{vec}(\boldsymbol{B}_i),
\end{aligned}$$

again using the fact that $\text{vec}(\boldsymbol{AXB}) = (\boldsymbol{B}^T \otimes \boldsymbol{A})\text{vec}(\boldsymbol{X})$. We let $y = (y_1^T, ..., y_n^T)^T \in \mathbb{R}^{nq}$, and $\boldsymbol{D} = \text{diag}(\boldsymbol{I}_q \otimes z_1^T, ..., \boldsymbol{I}_q \otimes z_n^T)$, which is a block-diagonal and $\boldsymbol{D} \in \mathbb{R}^{nq \times npq}$.

$$\boldsymbol{D}\beta = \begin{bmatrix} \boldsymbol{I}_q \otimes z_1^T \text{vec}(\boldsymbol{B}_1) \\ ... \\ \boldsymbol{I}_q \otimes z_n^T \text{vec}(\boldsymbol{B}_n) \end{bmatrix} = \begin{bmatrix} z_1^T \boldsymbol{B}_1 \\ ... \\ z_n^T \boldsymbol{B}_n \end{bmatrix} \in \mathbb{R}^{nq},$$

and this leads to

$$L_1 = \frac{1}{2} \|y - \boldsymbol{D}\beta\|^2 . \tag{2.19}$$

Next we will show that $L_2 = \frac{\upsilon}{2} \left\|\boldsymbol{A}\beta - \eta + \upsilon^{-1}\nu\right\|^2$. Firstly, let

$$\begin{aligned} \boldsymbol{\Delta} &= \{(e_i - e_j), i < j\}^T \in \mathbb{R}^{d \times n}, \\ \boldsymbol{A} &= \boldsymbol{\Delta} \otimes \boldsymbol{I}_{pq} \in \mathbb{R}^{dpq \times npq.}, \\ \widetilde{\boldsymbol{B}} &= \Big[\text{vec}(\boldsymbol{B}_1), ..., \text{vec}(\boldsymbol{B}_n)\Big] \in \mathbb{R}^{pq \times n}, \end{aligned}$$

we have

$$\begin{aligned} \Big[(e_i - e_j)^T \otimes \boldsymbol{I}_{pq}\Big]\beta &= \Big[(e_i - e_j)^T \otimes \boldsymbol{I}_{pq}\Big]\text{vec}(\widetilde{\boldsymbol{B}}) \\ &= \text{vec}\Big[\boldsymbol{I}_{pq}\widetilde{\boldsymbol{B}}(e_i - e_j)\Big] \\ &= \text{vec}(\boldsymbol{B}_i - \boldsymbol{B}_j). \end{aligned}$$

Thus we can rewrite L_2 such that

$$\begin{aligned} L_2 &= \frac{\upsilon}{2} \sum_{1 \le i < j \le n} \left\|\boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} + \upsilon^{-1}\boldsymbol{N}_{ij}\right\|^2 \\ &= \frac{\upsilon}{2} \sum_{1 \le i < j \le n} \left\|\text{vec}(\boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} + \upsilon^{-1}\boldsymbol{N}_{ij})\right\|^2 \\ &= \frac{\upsilon}{2} \sum_{1 \le i < j \le n} \left\|\text{vec}(\boldsymbol{B}_i - \boldsymbol{B}_j) - \text{vec}(\boldsymbol{H}_{ij}) + \upsilon^{-1}\text{vec}(\boldsymbol{N}_{ij})\right\|^2 \\ &= \frac{\upsilon}{2} \left\|\boldsymbol{A}\beta - \eta + \upsilon^{-1}\nu\right\|^2. \end{aligned} \tag{2.20}$$

With reformulation of L_1 and L_2 in (2.19) and (2.20), the objective function in (2.18) can be expressed as

$$\begin{aligned} L(\boldsymbol{H}, \boldsymbol{B}, \boldsymbol{N}) &= L(\eta, \beta, \nu) \\ &= \frac{1}{2} \|y - \boldsymbol{D}\beta\|^2 + \frac{\upsilon}{2} \left\|\boldsymbol{A}\beta - \eta + \upsilon^{-1}\nu\right\|^2 + C. \end{aligned} \tag{2.21}$$

And using the ADMM algorithm, we can obtain solutions for this model from following 3 steps.

Step 1 Given (η, ν), update β. Now that we rewrite $L(\boldsymbol{B}, \boldsymbol{H}, \boldsymbol{N})$ as a function of β, η, and, ν, i.e., $L(\eta, \beta, \nu)$ in (2.21), we start to solve β by having η and ν fixed.

$$\begin{aligned}
L(\beta; \eta, \nu) &= \frac{1}{2} \|y - \boldsymbol{D}\beta\|^2 + \frac{\upsilon}{2} \left\| \boldsymbol{A}\beta - \eta + \upsilon^{-1}\nu \right\|^2 + C \\
\frac{\partial L}{\partial \beta} &= \boldsymbol{D}^T(\boldsymbol{D}\beta - y) + \upsilon \boldsymbol{A}^T(\boldsymbol{A}\beta - \eta + \upsilon^{-1}\nu) = 0 \\
\boldsymbol{D}^T\boldsymbol{D}\beta + \upsilon \boldsymbol{A}^T\boldsymbol{A}\beta &= \boldsymbol{D}^T y + \upsilon \boldsymbol{A}^T(\eta - \upsilon^{-1}\nu)
\end{aligned}$$

Taking derivative against β, we have

$$\hat{\beta} = \left[\boldsymbol{D}^T\boldsymbol{D} + \upsilon \boldsymbol{A}^T\boldsymbol{A}\right]^{-1} \left[\boldsymbol{D}^T y + \upsilon \boldsymbol{A}^T(\eta - \upsilon^{-1}\nu)\right]. \tag{2.22}$$

Step 2 Given (β, ν), update η. Laying out the objective function $L(\beta, \eta, \nu)$ in terms of η, note again that $\eta = \left\{\text{vec}^T(\boldsymbol{H}_{ij}), i < j\right\}^T$.

$$L(\eta; \beta, \nu) = \frac{\upsilon}{2} \sum_{1 \le i < j \le n} \left\| \boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} + \upsilon^{-1}\boldsymbol{N}_{ij} \right\|^2 + p_\gamma(\left\|\boldsymbol{H}_{ij}\right\|, \lambda) + C^*.$$

Similar to Model 1, minimizing $L(\eta; \beta, \nu)$ can be done piece-wise, i.e.

$$\text{argmin}_\eta L(\eta; \beta, \nu) = \text{argmin}_{\boldsymbol{H}_{ij}} L_{ij}(\boldsymbol{H}_{ij}; \beta, \nu).$$

Let $\boldsymbol{\Xi}_{ij} = \boldsymbol{B}_i - \boldsymbol{B}_j + \upsilon^{-1}\boldsymbol{N}_{ij} \in \mathbb{R}^{p \times q}$, we have

$$\begin{aligned}
L_{ij}(\boldsymbol{H}_{ij}; \beta, \nu) &= \frac{\upsilon}{2} \left\| \boldsymbol{B}_i - \boldsymbol{B}_j - \boldsymbol{H}_{ij} + \upsilon^{-1}\boldsymbol{N}_{ij} \right\|^2 + p_\gamma(\left\|\boldsymbol{H}_{ij}\right\|, \lambda) \\
&= \frac{\upsilon}{2} \left\| \boldsymbol{\Xi}_{ij} - \boldsymbol{H}_{ij} \right\|^2 + p_\gamma(\left\|\boldsymbol{H}_{ij}\right\|, \lambda)
\end{aligned}$$

Similar to solution in Model 1 (2.16), with respect to $\eta_{ij} = \mu_i - \mu_j \in \mathbb{R}^q$, we can obtain solutions of $\boldsymbol{H}_{ij}$ in group-wise and component-wise fashion. We will here demonstrate the solution when group-wise MCP penalty is used.

$$\hat{\boldsymbol{H}}_{ij} = \begin{cases} \text{ST}_G(\boldsymbol{\Xi}_{ij}, \frac{\lambda}{\upsilon}), & \text{if } \left\|\boldsymbol{\Xi}_{ij}\right\| \le \gamma\lambda \\ \boldsymbol{\Xi}_{ij}, & \text{otherwise} \end{cases} \tag{2.23}$$

Step 3 Given (η, β), update ν. Note that $\nu = \left\{\text{vec}^T(N_{ij}), i < j\right\}^T$.

$$\boldsymbol{N}_{ij}^{(m+1)} = \boldsymbol{N}_{ij}^{(m)} + \upsilon\left(\boldsymbol{B}_i^{(m+1)} - \boldsymbol{B}_j^{(m+1)} - \boldsymbol{H}_{ij}^{(m+1)}\right) \in \mathbb{R}^{p\times q} \tag{2.24}$$

To conclude, solutions in these 3 steps from iteration $m+1$ step of depend on solutions from the previous iteration m, we can collectively write the solutions at iteration $m+1$ as:

$$\begin{aligned}
\beta^{(m+1)} &= \left[\boldsymbol{D}^T\boldsymbol{D} + \upsilon\boldsymbol{A}^T\boldsymbol{A}\right]^{-1}\left[\boldsymbol{D}^T y + \upsilon\boldsymbol{A}^T(\eta^{(m)} - \upsilon^{-1}\nu^{(m)})\right], \\
\boldsymbol{H}_{ij}^{(m+1)} &= \begin{cases} \text{ST}_G(\boldsymbol{\Xi}_{ij}^{(m+1)}, \frac{\lambda}{\upsilon}), & \text{if } \left\|\boldsymbol{\Xi}_{ij}^{(m+1)}\right\| \le \gamma\lambda \\ \Xi_{ij}, & \text{otherwise} \end{cases}, \\
\boldsymbol{N}_{ij}^{(m+1)} &= \boldsymbol{N}_{ij}^{(m)} + \upsilon\left(\boldsymbol{B}_i^{(m+1)} - \boldsymbol{B}_j^{(m+1)} - \boldsymbol{H}_{ij}^{(m+1)}\right).
\end{aligned}$$

2.4 Initialization of the Algorithm

2.4.1 Model 1: Subject-specific Intercepts

A simple ordinary least square multi-response regression will be used to generate initial values for Model 1. In this case, we let $\mu_i = \mu$ for all i when initializing $\boldsymbol{B}$. We can write the initial values as

$$\begin{aligned}
\boldsymbol{B}^{(0)} &= \boldsymbol{B}_{\text{OLS}}, \\
\mu_i^{(0)} &= y_i - \boldsymbol{B}^{T(0)}x_i, \\
\eta_{ij}^{(0)} &= \mu_i^{(0)} - \mu_j^{(0)}, \\
\nu_{ij}^{(0)} &= 0.
\end{aligned} \tag{2.25}$$

2.4.2 Model 2: Subject-specific Slopes

To initialize β and $\beta = \left\{\text{vec}^T(\boldsymbol{B}_1), ..., \text{vec}^T(\boldsymbol{B}_n)\right\}^T$, we follow similar approach as specified in Ma et. al (2016). The initialization algorithm takes three steps.

Step 1

$$\begin{aligned}
L_R(\beta) &= \frac{1}{2}\sum_{i=1}^{n}\left\|y_i - \boldsymbol{B}_i^T z_i\right\|^2 + \frac{\lambda^*}{2}\sum_{i<j}\left\|\boldsymbol{B}_i - \boldsymbol{B}_j\right\|^2 \\
&= \frac{1}{2}\left\|y - \boldsymbol{D}\beta\right\|^2 + \frac{\lambda^*}{2}\left\|\boldsymbol{A}\beta\right\|^2
\end{aligned}$$

If taking derivatives against β from L_R, we will get

$$\begin{aligned} \boldsymbol{D}^T(\boldsymbol{D}\beta - y) + \lambda^* \boldsymbol{A}^T \boldsymbol{A}\beta &= 0 \\ \boldsymbol{D}^T\boldsymbol{D}\beta + \lambda^* \boldsymbol{A}^T \boldsymbol{A}\beta &= \boldsymbol{D}^T y, \end{aligned}$$

then we have

$$\beta_R(\lambda^*) = (\boldsymbol{D}^T\boldsymbol{D} + \lambda^* \boldsymbol{A}^T \boldsymbol{A})^{-1}\boldsymbol{D}^T y, \tag{2.26}$$

and λ^* is chosen to have a small value. Ma (2016) uses $\lambda^* = 0.001$.

Step 2 Assign n subjects to K^* groups by ranking median of $\text{vec}(\boldsymbol{B}_{R,i}(\lambda^*))$, and we let $K^* = \lfloor n \rfloor^{1/2}$. Note that $\beta_R(\lambda^*) = \left\{\text{vec}^T\left(\boldsymbol{B}_{R,1}(\lambda^*)\right), ..., \text{vec}^T\left(\boldsymbol{B}_{R,n}(\lambda^*)\right)\right\}^T$.

Step 3 Find $\beta^{(0)}$ from multi-response regression within each of these K^* groups, and let $\boldsymbol{H}_{ij}^{(0)} = \boldsymbol{B}_i^{(0)} - \boldsymbol{B}_j^{(0)}$ and $\boldsymbol{N}_{ij}^{(0)} = \boldsymbol{0}$.

CHAPTER 3

SIMULATION STUDIES

In this chapter, experimental simulations are conducted to evaluate numerical performance of our method.

To select the tuning parameter λ, we use the modified Bayesian Information Criteria (BIC) for high-dimensional data. The modified BIC works by minimizing:

$$\text{BIC}(\lambda) = \log\Big[\sum_{i=1}^{n}\Big\|y_i - \hat{\mu}_i(\lambda) - \hat{\boldsymbol{B}}^T(\lambda)x_i\Big\|^2/n\Big] + C_n\frac{\log n}{n}(\hat{k}(\lambda)q + pq) \tag{3.1}$$

with respect to λ, where C_n is a positive number that diverges to infinity as n increases. When $C_n = 1$, the modified BIC reduces to the traditional BIC (Schwarz 1978). Following the similar strategy used in Ma and Huang (2016), we use $C_n = \log(nq + pq)$. λ is selected by minimizing the modified BIC, while ADMM dual variables ν and υ, and one other MCP tuning parameter γ were fixed.

3.1 Example 1

We simulate $n = 100$ data points from the model

$$y_i = \mu_i + \boldsymbol{B}^T x_i + \epsilon_i, \text{ i=1,...,n,}$$

where $x_i \in \mathbb{R}^p, p = 3$ is generated from a multivariate normal distribution with mean 0 and covariance matrix $\boldsymbol{I}_3$. The error term $\epsilon_i \in \mathbb{R}^q$, here we take $q = 2$. The error term is generated from multivariate normal $N(0, \boldsymbol{\Sigma})$ with diagonal elements as 1 and off-diagonal elements as ρ. We simulated μ_i from two groups with equal probabilities. Each of these groups have different mean vectors, and we notate the

true value of group 1 as α_1 and group 2 as α_2. We conduct our simulation with $\alpha_1 = a_1 \mathbb{1}_2$ and $\alpha_2 = a_2 \mathbb{1}_2$, where $\mathbb{1}_2^T = (1, 1)$. We then take two different set of values for a_1 and a_2 to create 2 levels of separations. These set of values are (1) $a_1 = 3$ and $a_2 = -1$; (2) $a_1 = 2$ and $a_2 = -1$. In our simulation study, we fix ADMM dual variables $\nu = 0$ and $\nu = 1$, and one of MCP tuning parameters $\gamma = 2$.

3.1.1 Component-wise vs. group-wise penalty

With simulated data, we first compare performance of the component-wise and group-wise MCP.

Firstly, we observe how both methods converge with a series of λ values, in the case where $a_1 = 3$ and $a_2 = -1$. Figure 3.1 presents solution paths for l_2 norm of estimated $(\mu_1, ..., \mu_n)$ versus λ by using component-wise and group-wise MCP methods. These two methods have similar solution paths, in which group-wise MCP converges to the true number of groups at norm values of $\sqrt{2}$ and 4, while component-wise MCP converges to 1 group with norm value 1.5.

Next, we continue the simulated experiment by selecting λ that minimizes the modified BIC. Note that we select $C_n = \log(nq + pq)$, where $(nq + pq)$ is the number of elements to be estimated in μ and $\boldsymbol{B}$. Table 3.1 reports the mean, median, and standard error (SE) of the estimated number of groups $\hat{K}$ by component-wise and group-wise MCP based on 100 simulations. To study the estimation accuracy, we report in Table 3.2 the mean and SE of square root of the Mean Squared Error (RMSE) for μ and $\boldsymbol{B}$. The RMSE for μ and $\boldsymbol{B}$ are defined as $\|\hat{\mu} - \mu\| / \sqrt{nq}$ and $\left\|\hat{\boldsymbol{B}} - \boldsymbol{B}\right\| / \sqrt{pq}$ of each iteration. For both sets of (a_1, a_2), we observe that component-wise MCP tend to select more groups compared to group-wise MCP, while group-wise MCP is much more successful in recovering the true number of groups, with lower variability.

This is within expectation. Group-wise MCP penalizes the collective representation of every component from η_{ij}, while component-wise MCP penalizes these components separately. Viewing all components in η_{ij} as a unit makes better sense. The reason is that the norm of η_{ij} collectively becomes 0 sounds more reasonable only when every component within this unit gets penalized to 0, and it indicates that every element of μ_i is the same as every element in μ_j when they are considered from the same group. In other words, different components in η_{ij} may end up with

conflicting results when component-wise penalties are used. As a result, $\hat{\mu}$ is better estimated by group-wise MCP. Surprisingly, the estimation of $\hat{\boldsymbol{B}}$ did not differ much between these two types of penalty setup. Possible reasons may be that despite the performance of $\boldsymbol{B}$ should be affected by estimation performance of μ, the influence is negligible. Visualization of estimation differences are presented in Figure 3.2.

To evaluate clustering results, we use the Rand Index measure (Rand 1971). The Rand Index is viewed as a measure for the amount correct decisions made by an algorithm as a proportion over the overall number of decisions it needs to make. It is defined as

$$\text{RI} = \frac{\text{TP} + \text{TN}}{\text{TP+FP+FN+TN}}, \tag{3.2}$$

where a true positive (TP) decision is one that assigns two observations from the same ground-truth group to the same cluster, and a true negative (TN) decision is one that assigns two observations from different groups to different clusters. A false positive (FP) decision assigns two observations from different groups to the same cluster, and a false negative (FN) decision assigns two observations from the same group to different clusters. Value of Rand Index ranges from 0 to 1, while higher values indicate better performance. We report rand index results of above simulations in Table 3.3.

In the case where less separation between two groups is injected, the group-wise MCP yields a much more significant advantage in clustering accuracy over the component-wise MCP based model. Despite both models have relatively small variability, group-wise MCP still outperforms component-wise MCP in terms of stability.

3.1.2 Weighted l_1 vs. MCP

We continue to experiment our methods with two types of penalties, MCP and weighted l_1. The weighted l_1 takes the form

$$p(\mu_i - \mu_j, \lambda) = \lambda w_{ij} \left\| \mu_i - \mu_j \right\|,$$

which requires specification of weights w_{ij}. Following Ma and Huang (2017), we select the Gaussian kernel (1.5) defined based on the distance between two points

$$w_{ij} = \exp(\phi \left\|\mu_i - \mu_j\right\|^2),$$

with a non-negative constant ϕ. When $\phi = 0$, it corresponds to the l_1 penalty (the LASSO).

Figure 3.1 shows the solution paths by using both MCP and weighted l_1 penalties with $\phi = 0$, 0.2, and 0.5. The various l_1 penalties showcase solution paths that substantially differ from those of MCP. Moreover, these solution paths seem quite different when ϕ differ. This corresponds the fact pointed out by Ma and Huang (2017) that choice of weights can dramatically affect estimation results. Observing different ϕ values at 0, 0.2, and 0.5, one can see that as ϕ increases, the estimated $\|\mu\|$ converge to one point more slowly.

We continue our experiments using weighted l_1 penalty with $\phi = 0.2$, and $\phi = 0.2$ was selected because it had a reasonable convergence rate. We first select λ that minimizes the modified BIC, and proceed model constructions with the selected λ. Table 3.1 and Table 3.2 report mean and SE of RMSE for $\hat{K}$, $\hat{\mu}$, and $\hat{\boldsymbol{B}}$, and Table 3.3 reports clustering accuracy measured by Rand Index. In general, the model with MCP penalty generates better clustering results, and gives more accurate and less variable estimations of μ and $\boldsymbol{B}$. This observation falls in line with observations from univariate case, as described in Ma and Huang (2017).

In the simulation we also presented results for both group-wise and component-wise type of penalties for models with MCP and weighted l_1. Similar to what were observed for MCP, in most cases, group-wise weighted l_1 outperforms the component-wise setup. However, the differences between group-wise and component-wise setup of weighted l_1 are not as drastic as of those in models with MCP penalty. In the case of less separation between groups, i.e., $a_1 = 2$ and $a_2 = -1$, component-wise weighted l_1 outperformed group-wised one in clustering results.

TABLE 3.1: Mean, median, and standard error (SE) of $\hat{K}$ by group-wise and component-wise MCP ($\gamma = 2$) and weighted l_1 ($\phi = 0.2$)

	$a_1 = 3, a_2 = -1$			$a_1 = 2, a_2 = -1$		
Method	Mean	Median	SE	Mean	Median	SE
grp. MCP	2.01	2	0.0100	2.30	2	0.0670
comp. MCP	4.11	4	0.0827	4.10	4	0.1856
grp. l_1	2.50	2	0.0798	1.81	1.5	0.0940
comp. l_1	2.77	2	0.1004	1.49	1	0.0810

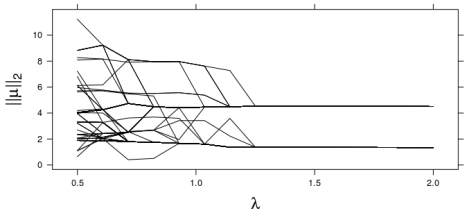
TABLE 3.2: Mean (and standard error, SE) of RMSE for $\hat{\mu}$ and $\hat{\boldsymbol{B}}$ by group-wise and component-wise MCP ($\gamma = 2$) and weighted l_1 ($\phi = 0.2$)

	μ		$\boldsymbol{B}$	
Method	$a_1 = 3, a_2 = -1$	$a_1 = 2, a_2 = -1$	$a_1 = 3, a_2 = -1$	$a_1 = 2, a_2 = -1$
grp. MCP	0.392 (0.0216)	0.570 (0.0179)	0.159 (0.0073)	0.169 (0.0074)
comp. MCP	0.581 (0.0267)	0.973 (0.0332)	0.175 (0.0079)	0.207 (0.0079)
grp. l_1	0.672 (0.3980)	1.335 (0.0284)	0.188 (0.0103)	0.220 (0.0088)
comp. l_1	0.748 (0.0423)	1.457 (0.0169)	0.193 (0.0105)	0.227 (0.0088)

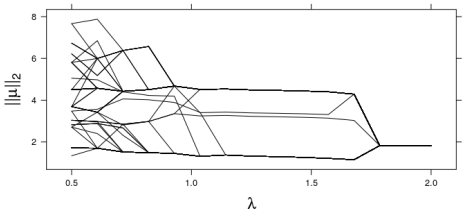
TABLE 3.3: Mean and standard error (SE) of Rand Index by group-wise and component-wise MCP ($\gamma = 2$) and weighted l_1 ($\phi = 0.2$)

	$a_1 = 3, a_2 = -1$		$a_1 = 2, a_2 = -1$	
Method	Mean	SE	Mean	SE
grp. MCP	0.983	0.0024	0.937	0.0044
comp. MCP	0.958	0.0053	0.808	0.0162
grp. l_1	0.969	0.0079	0.652	0.0208
comp. l_1	0.964	0.0085	0.562	0.0143

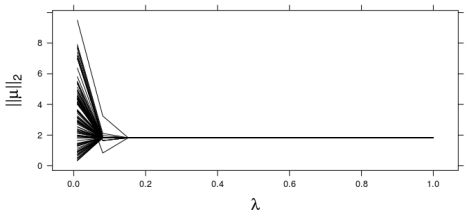
(A) Group-wise MCP



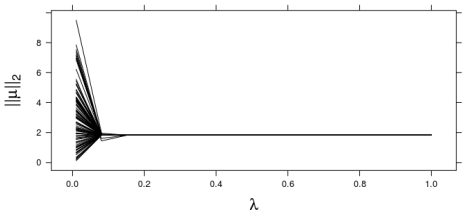
(B) Component-wise MCP



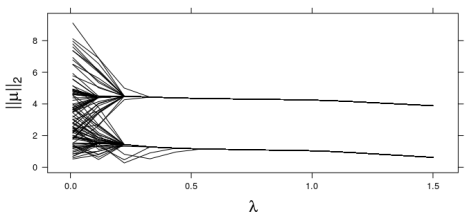
(C) Group-wise weighted l_1, $\phi = 0$



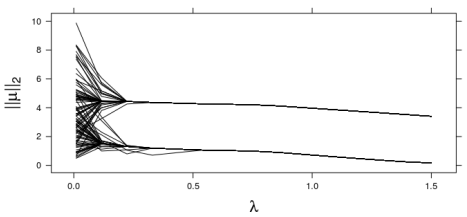
(D) Component-wise weighted l_1, $\phi = 0$



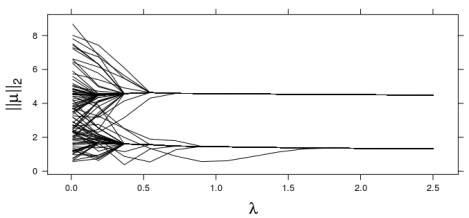
(E) Group-wise weighted l_1, $\phi = 0.2$



(F) Component-wise weighted l_1, $\phi = 0.2$



(G) Group-wise weighted l_1, $\phi = 0.5$



(H) Component-wise weighted l_1, $\phi = 0.5$

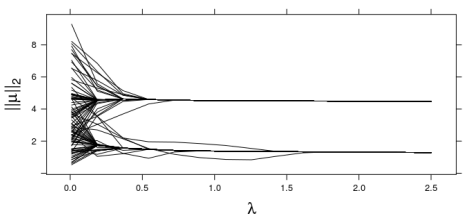


FIGURE 3.1: Solution Path of group-wise and component-wise MCP and weighted l_1 ($\phi = 0$, 0.2, and 0.5)

3.2 Example 2

With the same model as in **Example 1**, in the case $q = 2$, and $a_1 = 2$ and $a_2 = -1$, we compare the results from multi-response model (2.1) with separate single-response models as set in Ma and Huang (2017). The univariate model takes the form

$$y_i = \mu_i + x_i^T \beta + \epsilon_i, i = 1, ..., n,$$

where y_i, μ_i, and $\epsilon \in \mathbb{R}^1$, and x_i and $\beta \in \mathbb{R}^p$. The comparison aims to identify situations in which it is appropriate to choose between simultaneous clustering/estimation and separate clustering/estimation. In this example, we will only explore performance of multi-response method compared with the single-response using group-wise MCP.

Table 3.4 reports K estimated by univariate model separately in each of $\boldsymbol{Y}$'s directions, $\boldsymbol{Y}_{(1)}$ and $\boldsymbol{Y}_{(2)}$, and the multi-response model. Note that $\boldsymbol{Y} = (\boldsymbol{Y}_{(1)}, \boldsymbol{Y}_{(2)})$. Table 3.5 reports estimation performance of μ and $\boldsymbol{B}$, and here $\boldsymbol{B} = (\beta_1, \beta_2)$. Table 3.6 presents clustering accuracy measured by the Rand Index. A visualization of clustering and estimation performances of multivariate (multi-response) and univariate methods at different correlation ρ between $\boldsymbol{Y}_{(1)}$ and $\boldsymbol{Y}_{(2)}$ are presented in Figure 3.2.

In most cases, multi-response model gives estimate of K, μ, and $\boldsymbol{B}$ that are closer to the ground truth, with better stability. Except for in the case of $\rho = 0.5$ and 0.8, multi-response method estimates larger number of groups, while single-response method in both directions gives K that is closer to 2.

Looking at multi-response method alone, one can observe that as the correlation between responses, $\boldsymbol{Y}_{(1)}$ and $\boldsymbol{Y}_{(2)}$, grows from -1 to 1, the performance of estimation decreases (larger RMSE), and the algorithm yields most accurate estimation and clustering results when correlation between responses are negative. As correlation grows to 0 and continues into positive, performance in estimation and clustering results both get compromised, and the advantages of conducting multi-response modeling as compared to separate single-response models fade away.

This fact is intuitive. As shown in Figure 3.3, when the correlation ρ is negative, much overlap can be seen in each individual axes. However, no overlap is observed in the 2-dimensional space and the distance between these two clusters is

TABLE 3.4: Mean, median, and standard error (SE) of $\hat{K}$ by group-wise MCP ($\gamma = 2$) with varying correlation ρ between $\boldsymbol{Y}_{(1)}$ and $\boldsymbol{Y}_{(2)}$: $\alpha_1 = (2,2)^T$ and $\alpha_2 = (-1,-1)^T$

	$\boldsymbol{Y}_{(1)}$			$\boldsymbol{Y}_{(2)}$			$\boldsymbol{Y}$		
ρ	Mean	Median	SE	Mean	Median	SE	Mean	Median	SE
-0.8	2.46	2	0.0937	2.31	2	0.0929	2.35	2	0.0592
-0.5	2.41	2	0.1006	2.40	2	0.0910	2.19	2	0.0419
-0.2	2.34	2	0.0956	2.45	2	0.0999	2.11	2	0.0314
0	2.45	2	0.1049	2.43	2	0.0956	2.11	2	0.0345
0.2	2.45	2	0.0999	2.41	2.5	0.0922	2.30	2	0.0522
0.5	2.40	2	0.0919	2.41	2	0.0965	2.51	2	0.0772
0.8	2.33	2	0.0933	2.44	2	0.0914	3.14	3	0.1146

visually more significant. By modeling the clusters from a higher dimensional space, multi-response model extracts this extra information caused by separation between clusters, which becomes visible only in the 2-dimensional space. As a result, the multi-response model outperforms results from separate single-response models, in clustering accuracy and estimation of μ. Note that the estimation of $\boldsymbol{B}$, which is itself not a source of heterogeneity, is also improved in estimation accuracy and stability. This differs from the estimation done by OLS in classical multi-response regression, which can be equally done by estimating from separate single response regression (Section 1.5). As ρ grows to 0, the separation in individual dimension does not change much, however the separation in 2-dimensional space becomes smaller. As ρ grows past 0 and becomes positive, overlaps in each axes stay about the same, while points from two-dimensional space start to mingle and it causes the performance in estimation and clustering to drop for multi-response model.

3.3 Example 3

With different setup of intercept vectors α_1 and α_2, the behaviors of uni-variate and multi-variate methods can be reversed. Continuing the model used in Example 2, we set these vectors $\alpha_1 = a\mathbb{1}_2$ and $\alpha_2 = (a,-1)^T$. With different choices of a, different separations are created to amplify the variations in performance.

Estimation for number of groups K, μ and $\boldsymbol{B}$, as well as Rand Index are shown in Figure 3.4 ($a = 3$) and Figure 3.5 ($a = 2$). The setup of intercept vectors in this

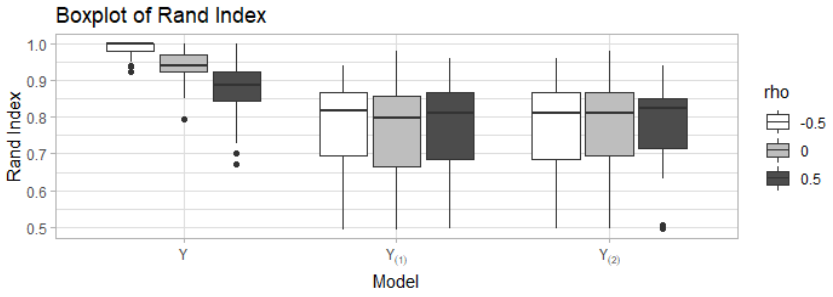
TABLE 3.5: Mean (and standard error, SE) of RMSE for $\hat{\mu}$ and $\hat{\boldsymbol{B}}$ or $\hat{\beta}_1$ and $\hat{\beta}_2$ by group-wise MCP ($\gamma = 2$): $\alpha_1 = (2,2)^T$ and $\alpha_2 = (-1,-1)^T$

Estimation Results of μ			
ρ	$\boldsymbol{Y}_{(1)}$	$\boldsymbol{Y}_{(2)}$	$\boldsymbol{Y}$
-0.8	1.025 (0.0261)	1.040 (0.0279)	0.317 (0.0172)
-0.5	1.014 (0.0272)	1.006 (0.0270)	0.319 (0.0134)
-0.2	1.006 (0.0276)	1.010 (0.0266)	0.447 (0.0159)
0	1.023 (0.0276)	1.021 (0.0275)	0.530 (0.0158)
0.2	1.010 (0.0265)	1.038 (0.0274)	0.641 (0.0173)
0.5	1.006 (0.0270)	1.031 (0.0269)	0.775 (0.0184)
0.8	1.006 (0.02914)	1.007 (0.0262)	0.927 (0.0219)
Estimation Results of $\boldsymbol{B} = (\beta_1, \beta_2)$			
ρ	$\boldsymbol{Y}_{(1)}$	$\boldsymbol{Y}_{(2)}$	$\boldsymbol{Y}$
-0.8	0.213 (0.0107)	0.203 (0.0110)	0.140 (0.0058)
-0.5	0.214 (0.0096)	0.212 (0.0113)	0.144 (0.0051)
-0.2	0.212 (0.0102)	0.208 (0.0105)	0.153 (0.0055)
0	0.216 (0.0097)	0.214 (0.0102)	0.163 (0.0060)
0.2	0.208 (0.0106)	0.211 (0.0114)	0.175 (0.0072)
0.5	0.211 (0.0114)	0.212 (0.0107)	0.191 (0.0089)
0.8	0.213 (0.0115)	0.210 (0.0107)	0.200 (0.0089)

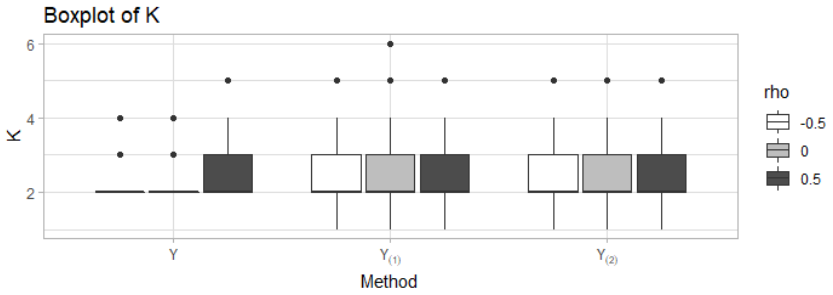
TABLE 3.6: Mean and standard error (SE) of Rand Index by group-wise MCP ($\gamma = 2$): $\alpha_1 = (2,2)^T$ and $\alpha_2 = (-1,-1)^T$

	$\boldsymbol{Y}_{(1)}$		$\boldsymbol{Y}_{(2)}$		$\boldsymbol{Y}$	
ρ	Mean	SE	Mean	SE	Mean	SE
-0.8	0.762	0.0135	0.752	0.0144	0.989	0.0026
-0.5	0.765	0.0142	0.763	0.0138	0.987	0.0019
-0.2	0.768	0.0144	0.761	0.0137	0.962	0.0031
0	0.757	0.0143	0.76	0.0142	0.944	0.0037
0.2	0.761	0.0137	0.756	0.0142	0.916	0.005
0.5	0.763	0.0138	0.761	0.0142	0.877	0.006
0.8	0.762	0.0148	0.773	0.0134	0.823	0.0084

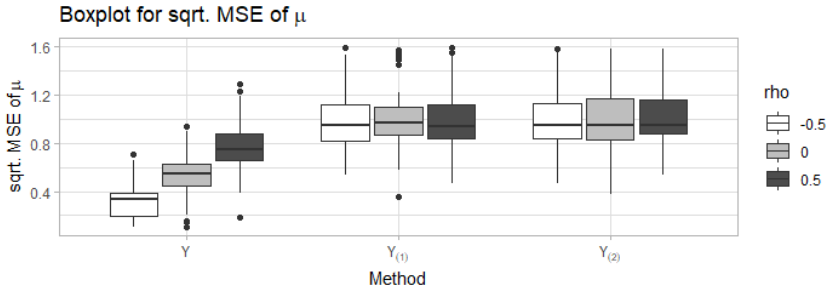
(A)



(B)



(C)



(D)

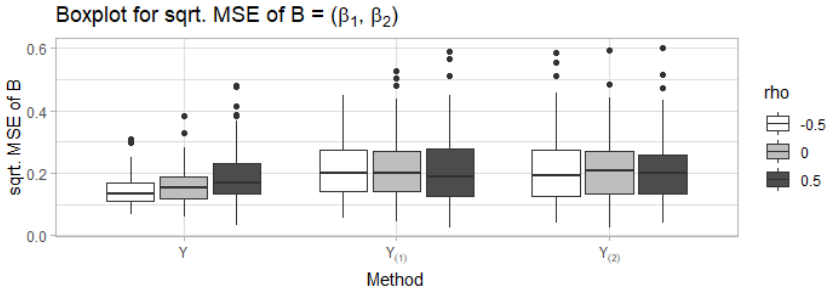


FIGURE 3.2: Clustering and Estimation Results of $\alpha_1 = a_1\mathbb{1}_2$ and $\alpha_2 = a_1\mathbb{1}_2$, where $a_1 = 2$ and $a_2 = -1$

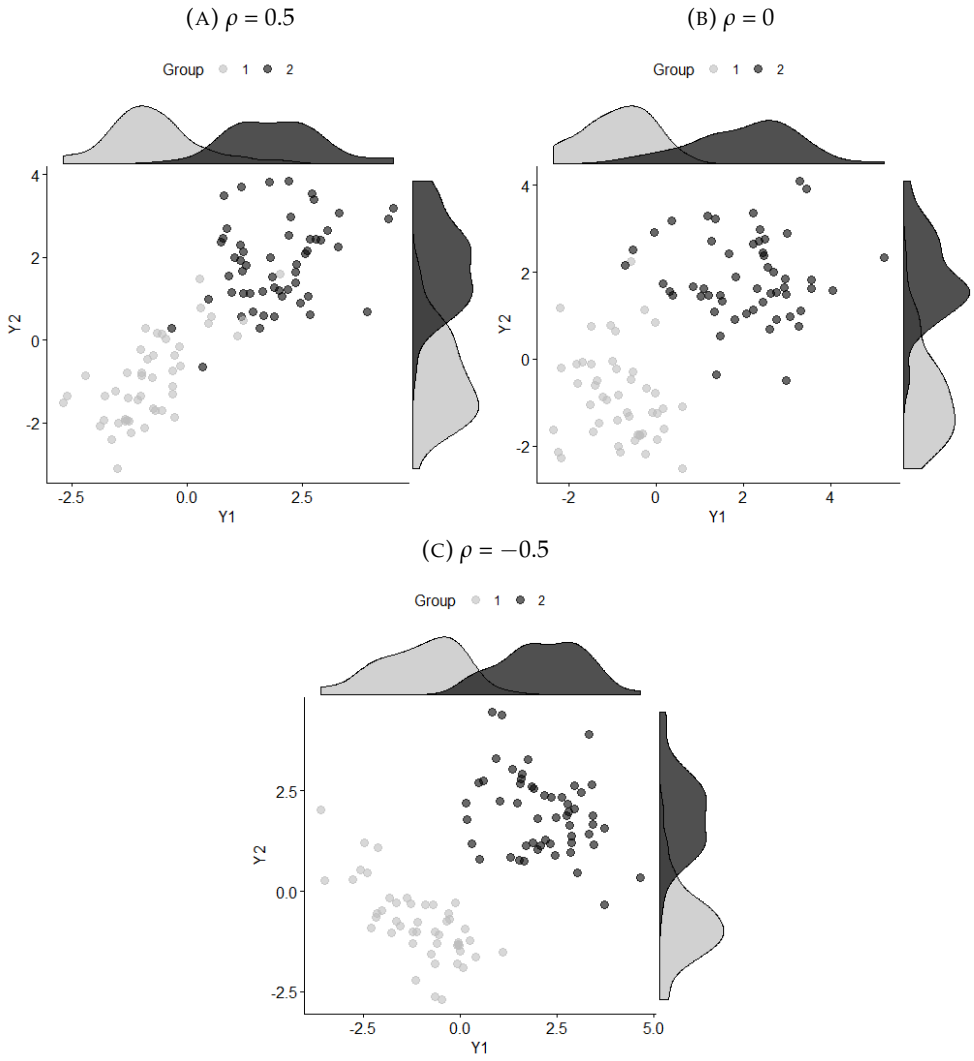
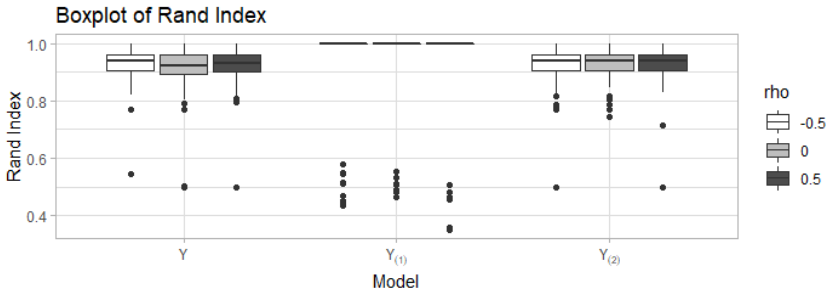


FIGURE 3.3: Separation in Groups with Varying Correlation

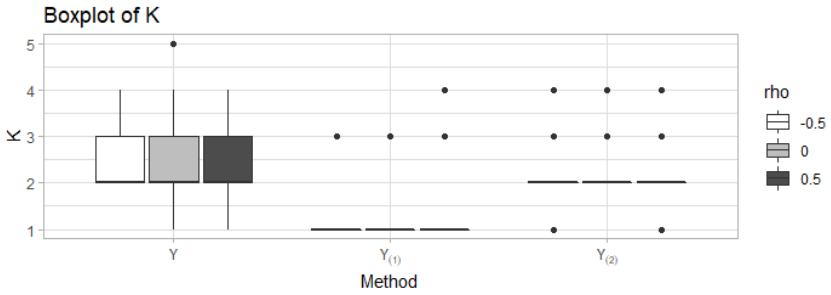
example only differ in one dimension. In such scenario, compared to the estimation from separate single-response model, results given by multi-response method are not as ideal. Interesting fact to observe is that, despite less promising performance in clustering results, the estimations of μ and $\boldsymbol{B}$ are as competent as the single-response model in the direction where heterogeneity exists.

Comparing Figure 3.4 and Figure 3.5, one can observe that the smaller the separation of the two true groups, the more significant is the advantage of conducting clustering and estimation using separate single-response models.

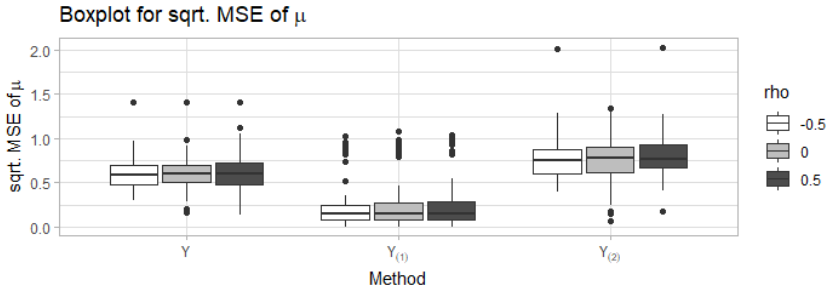
(A)



(B)



(C)



(D)

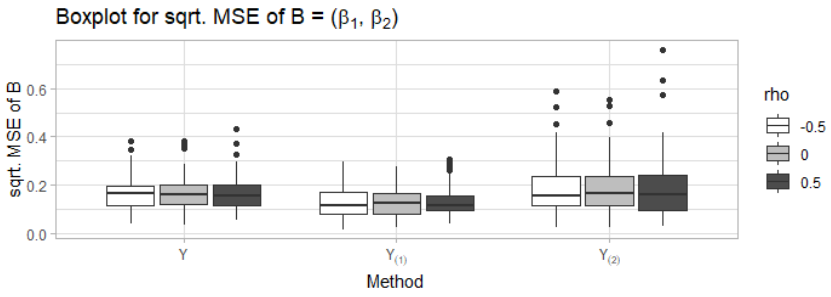
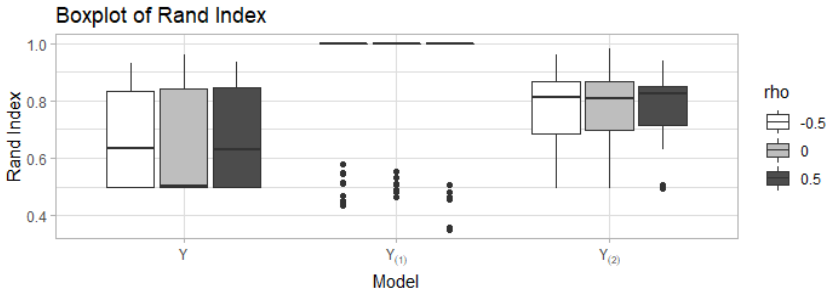
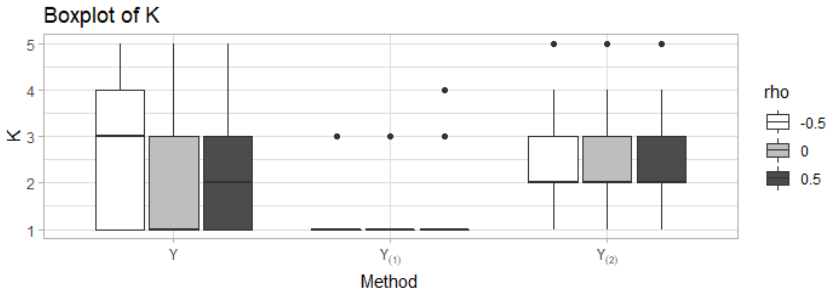


FIGURE 3.4: Clustering and Estimation Results of $\alpha_1 = a\mathbb{1}_2$ and $\alpha_2 = (a, -1)^T$, where $a = 3$

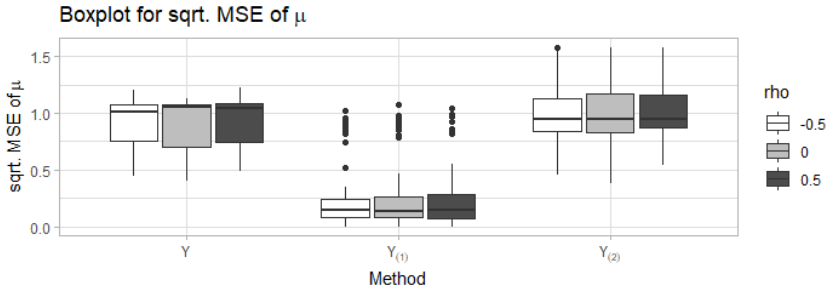
(A)



(B)



(C)



(D)

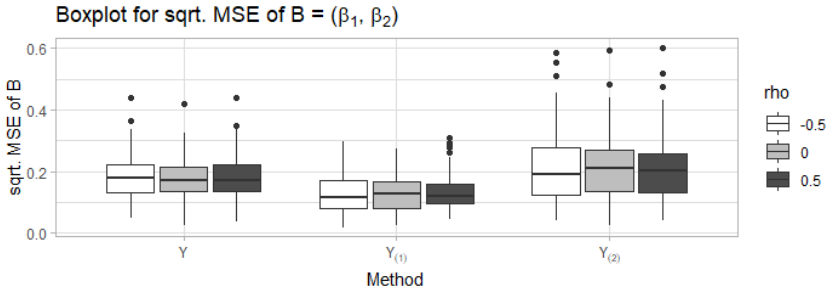


FIGURE 3.5: Clustering and Estimation Results of $\alpha_1 = a\mathbb{1}_2$ and $\alpha_2 = (a, -1)^T$, where $a = 2$

CHAPTER 4

EMPIRICAL STUDY

In this section, the dataset from Hlavnicka (2016) was used to illustrate the developed methods. The dataset was collected from a study run in Prague, Czech. This study enrolled 130 participants' data, 50 healthy subjects, 50 subjects with idiopathic rapid eye movement sleep behavior disorder (RBD), and 30 newly diagnosed, untreated Parkinson's disease (PD) patients. Hlavnicka et. al. focused on using speech data collected from these participants for development of acoustic biomarkers that may be used for early diagnosis of PD among patients with RBD. RBD is a population with high risk of developing PD (>80%). Twelve acoustic features from 2 context (reading passage and monologue) were generated, and the authors assessed capability of these features to differentiate the 2 cohorts de novo PD patients and healthy subjects. Hlavnicka et. al. designed a set of 12 acoustic features and used them to train a classifier that differentiates between Healthy participants from PD patients.

Duration of pause intervals (DPI) and Rate of speech timing (RST) from monologue were used in our analysis. According to analysis done by Hlavnicka et. al., these were two of the highest ranking features that can separate healthy subjects from PD patients.

DPI: evaluates a speaker's ability to initiate speech. Complex speech impairment can cause difficulties in initiating speech, which cause prolongation of pauses. The higher value in DPI seems to indicate more difficulties to initiate speech.

RST: provides a robust estimate of speech rate impairment. It considers not only pause but both voiced and unvoiced intervals. It is computed as the regression slope, regressing Time over the cumulative sum of number of intervals over

Time. The lower RST seems to indicate that the more impaired is one's speech rate.

Our interest is to conduct subgroup analysis for monologue DPI and RST, after adjusting for available demographic information (age and gender) and whether one is on psychotic medication. In this exercise, we use two approaches to conduct our analysis: (1) single response subgroup analysis separately for DPI and RST, using Ma and Huang (2017); and (2) multi-response subgroup analysis for DPI and RST simultaneously.

As the first step, we plot the OLS regression residuals after adjusting for covariates in Figure 4.1a. Heterogeneity obviously presents. It may be caused by unobserved latent factors. Heterogeneity led to the fact that with OLS regression and single estimate of intercept, the model is insufficient to describe the data. When viewing from the DPI direction, one can see a small bi-modality around the value 0 and a shoulder around 0.5 (see Figure 4.1c). And from the direction of RST, there seems to be less of such heterogeneity, and rather scattered data points are seen at the stretched left tail (see Figure 4.1e). Modified BIC was used to tune the model and MCP penalty was used. We compare the clustering and estimation results from the aforementioned approaches in Table 4.1 and 4.2.

Following Ma and Huang (2017), we compared results using estimated number of groups K, coefficient of determination R^2, and Davies-Boulding Index (DB-index). The DB-index is used to assess the quality of clustering algorithms.

$$\text{DB-index} = \hat{K}^{-1} \sum_{j=1}^{\hat{K}} \max_{j \neq j'} \frac{\sigma_j + \sigma_{j'}}{d(c_j, c_{j'})}, \tag{4.1}$$

where c_j is the centroid of cluster j, σ_j is the average distance of all observations in cluster j to centroid c_j, and $d(\cdot)$ is the distance between two centroids c_j and $c_{j'}$. The smaller this metric the better performance of the clustering algorithm.

Separate single-response results seem to split data points in much more subgroups as compared with multi-response results. The split of membership interacted by RST and DPI models is similar to the one modeled by the multi-response model. In Figure 4.2, two obvious groups were clustered. The larger cluster is similarly produced by these two approaches. The dissimilarity emerges from the fact that

TABLE 4.1: Subgroup analysis results of the Parkinson's disease dataset from Hlavnicka (2016).

Parameters	DPI	RST	(DPI, RST)
K	2	3	2
R^2	0.42	0.46	0.61
DB-index	0.55	0.37	0.60

TABLE 4.2: Regression Parameter Estimates of the Parkinson's disease dataset from Hlavnicka (2016)

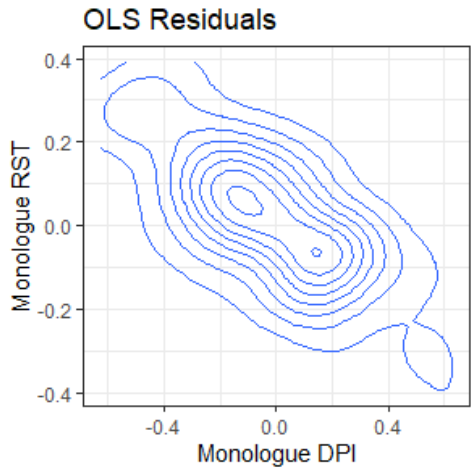
Covariate	DPI	RST	(DPI, RST)
Age	-0.0072	-0.0049	(-0.0120, -0.0036)
Gender (F)	-0.1559	0.0769	(-0.1108, 0.0511)
Antidepressant (Yes)	0.1269	-0.04089	(0.1244, -0.0600)

the separate single-response models further split the smaller cluster (Figure 4.2a). Such split seems unnecessary, especially in the direction of RST. This is evident as R^2 value from the multi-response results are higher than either of the single-response ones. However, the DB-index seems to favor single-response results.

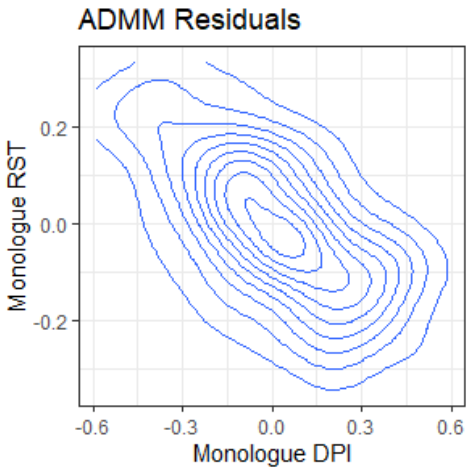
Estimations of covariates' coefficients are similar between both approaches (Table 4.2). To conduct modeling with both approaches are capable to remove heterogeneity and produce smooth residual density plots (see Figure 4.1).

It is worth noting that neither approaches results reflect the original membership given in Hlavnicka (2016). This is not unexpected. As seen in Figure 4.2c, despite tested significant in two sample statistical tests, the original membership does not seem to separate monologue RST and DPI well enough to be identified in a distance based clustering method. And whether there are really three natural groups, based on information given from monologue RST and DPI, is debatable. There may be unobserved latent variables that can explain such separation, and this is the fundamental purpose of unsupervised learning, i.e., to discover data structure that is not known conventionally and yet to be explained.

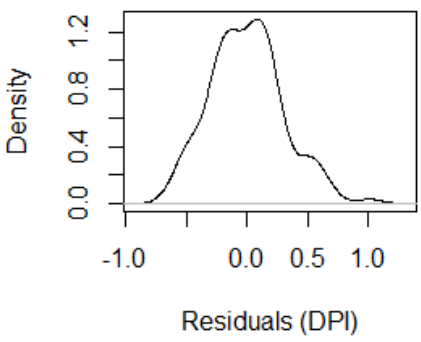
FIGURE 4.1: Residual Plots of OLS and ADMM



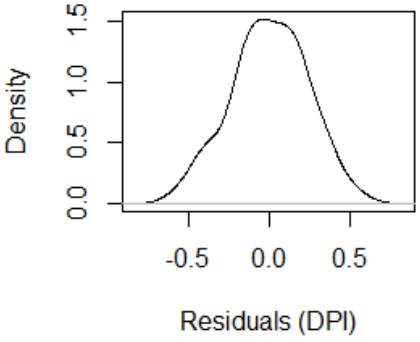
(A) OLS Residual Density Contour



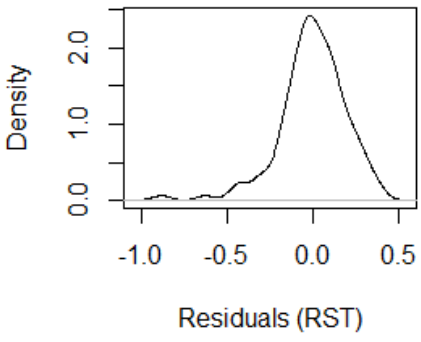
(B) Multi-response ADMM Residual Contour



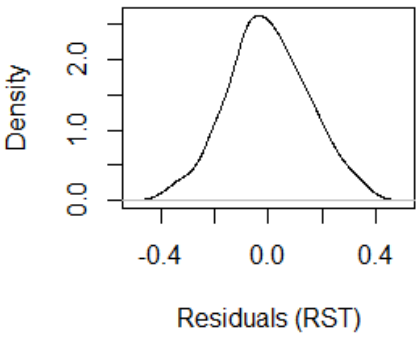
(C) OLS Residual Density Plot: DPI



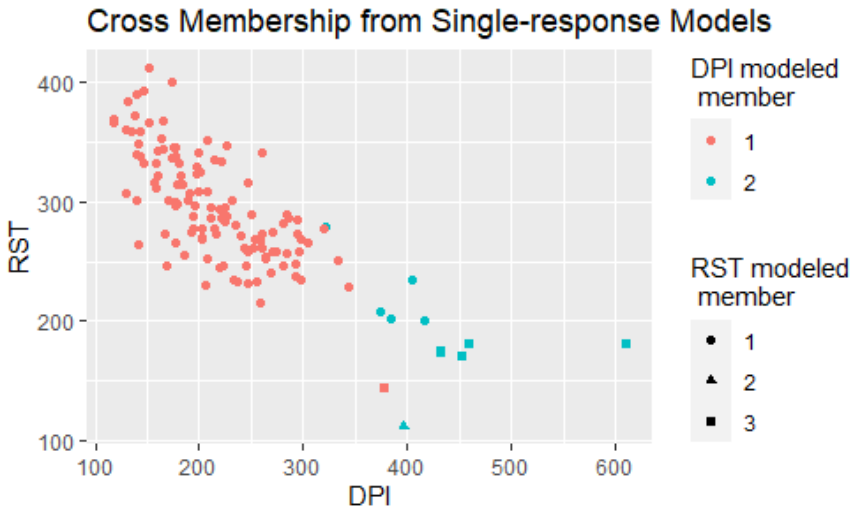
(D) ADMM Residual Plot: DPI



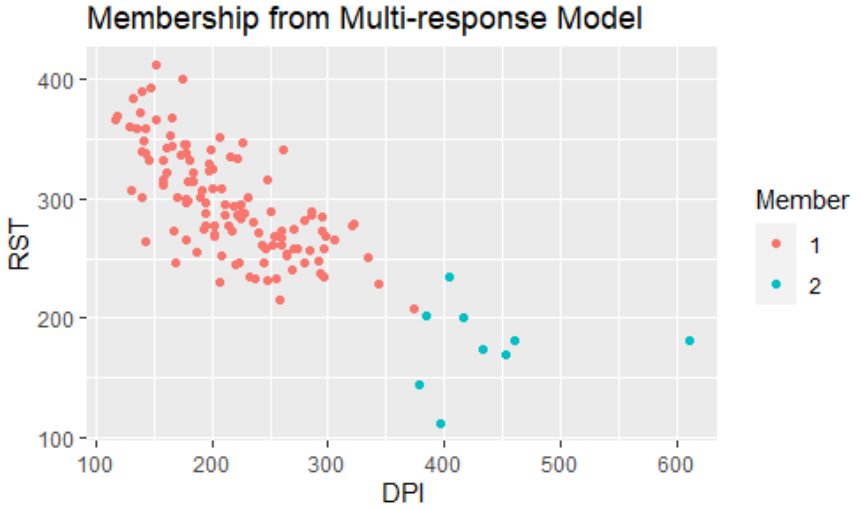
(E) OLS Residual Density Plot: RST



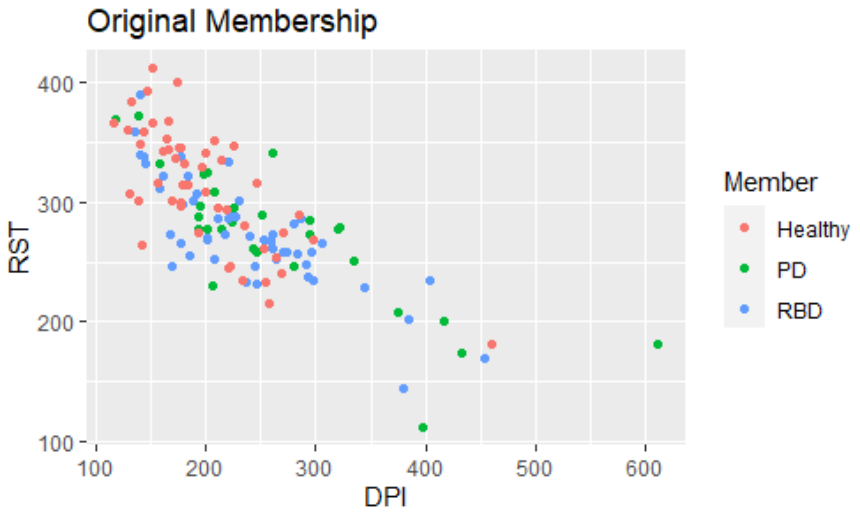
(F) ADMM Residual Plot: RST



(A)



(B)



(C)

FIGURE 4.2: Clustering Results Mapped to Original Data

Part II

ROBUST SOLUTIONS

CHAPTER 5

ITERATIVE REWEIGHTED LEAST SQUARES

5.1 Model Formulation with Robust Loss

Given formulation of Model 1, see Section 2.1 (equation 2.1), if we expect presence of outliers in our modeling and are borrowing the strategy of using robust loss to down-weight outliers' influence on the goodness-of-fit term of the objective function. Specifically, the estimation problem in (2.3) can be updated as

$$\begin{aligned} S_h(\mu, \boldsymbol{B}, \eta) &= \sum_{i=1}^{n} h(y_i - \mu_i - \boldsymbol{B}^T x_i) + \sum_{1 \le i < j \le n} p(\eta_{ij}, \lambda) \\ &\text{subject to } \mu_i - \mu_j = \eta_{ij}, \end{aligned} \tag{5.1}$$

where $h(\cdot)$ is a robust loss function. Following the derivation in Section 1.6.1, (5.1) can be updated to the weight least square formulation.

$$\begin{aligned} S(\mu, \boldsymbol{B}, W, \eta) &= \sum_{i=1}^{n} w_i \left\| y_i - \mu_i - \boldsymbol{B}^T x_i \right\|^2 + \sum_{1 \le i < j \le n} p(\eta_{ij}, \lambda) \\ &\text{subject to } \mu_i - \mu_j = \eta_{ij}, \end{aligned} \tag{5.2}$$

where $W = \text{diag}\{w_i, i = 1, \ldots, n\} \in \mathbb{R}^n$.

The Lagrangian form of above objective function is

$$\begin{aligned} L(\mu, \boldsymbol{B}, W, \eta, v) =& S(\mu, \boldsymbol{B}, W, \eta) + \sum_{1 \le i < j \le n} v_{ij}^T (\mu_i - \mu_j - \eta_{ij}) \\ &+ \frac{\upsilon}{2} \sum_{1 \le i < j \le n} \left\| \mu_i - \mu_j - \eta_{ij} \right\|^2. \end{aligned} \tag{5.3}$$

To solve the added weights, an additional step is inserted at to the original ADMM steps as laid out in Section 2.1. The updated algorithm is termed as the IRLS-ADMM Algorithm.

5.2 IRLS-ADMM Algorithm

The matrix form of (5.3) is

$$\begin{aligned} & L(\mu, \boldsymbol{B}, W, \eta, \nu) \\ & = (\boldsymbol{Y} - \boldsymbol{M} - \boldsymbol{XB})^T W (\boldsymbol{Y} - \boldsymbol{M} - \boldsymbol{XB}) + \nu^T (\boldsymbol{A}\mu - \eta) + \frac{\upsilon}{2} \|\boldsymbol{A}\mu - \eta\|^2 \\ & = (\boldsymbol{Y}^* - \boldsymbol{XB})^T W (\boldsymbol{Y}^* - \boldsymbol{XB}) + \nu^T (\boldsymbol{A}\mu - \eta) + \frac{\upsilon}{2} \|\boldsymbol{A}\mu - \eta\|^2 . \end{aligned} \tag{5.4}$$

Below is a reminder of notations copied from Section 2.1.

$$\begin{aligned} \mu &= \{\mu_1^T, \mu_2^T, \ldots, \mu_n^T\}^T \in \mathbb{R}^{nq} \\ \eta &= \{\eta_{ij}^T, i < j\}^T \in \mathbb{R}^{dq}, d = \frac{n(n-1)}{2} \\ \nu &= \{\nu_{ij}^T, i < j\}^T \in \mathbb{R}^{dq} \\ \boldsymbol{A} &= \boldsymbol{\Delta} \otimes I_q \in \mathbb{R}^{dq \times nq} \\ \boldsymbol{\Delta} &= \{(e_i - e_j), i < j\}^T \in \mathbb{R}^{d \times n} \\ \boldsymbol{M} &= (\mu_1, \mu_2, \ldots, \mu_n)^T \in \mathbb{R}^{n \times q} \\ \boldsymbol{Y} &= (y_1, y_2, \ldots, y_n)^T \in \mathbb{R}^{n \times q} \\ \boldsymbol{Y}^* &= \boldsymbol{Y} - \boldsymbol{M} \in \mathbb{R}^{n \times q} \\ \boldsymbol{X} &= (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^{n \times p} \\ \epsilon_i &= y_i - \mu_i - \boldsymbol{B}^T x_i \in \mathbb{R}^q \\ \boldsymbol{E} &= (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T \in \mathbb{R}^{n \times q} \end{aligned}$$

With results from the t^{th} iteration, below are the steps of the IRLS-ADMM algorithm for the $(t+1)^{th}$ update.

Step 1: Given (μ, β, η, ν)**, update** W

This is the added step to the original ADMM algorithm in Section 2.1 to handle the weights brought in to solve robust loss. These weights take the form as specified in

(1.17). Depending on the robust loss function $h(\cdot)$ of selection, the weights differ.

$$
\begin{aligned}
W^{(t+1)} &= \text{diag}\{w_i(\mu^{(t)}, \boldsymbol{B}^{(t)})\} \\
w_i(\mu^{(t)}, \boldsymbol{B}^{(t)}) &= \frac{h'(\left\| y_i - \mu_i^{(t)} - \boldsymbol{B}^{(t)^T} x_i \right\|)}{2\left\| y_i - \mu_i^{(t)} - \boldsymbol{B}^{(t)^T} x_i \right\|} \\
&= \frac{h'(||\epsilon_i||)}{2||\epsilon_i||}.
\end{aligned}
$$

The most natural choice is to use absolute loss, $h(x) = |x|$ for $q = 1$ cases and l_2 norm $h(x) = \|x\| = \sqrt{x^T x}$ for $q = 2$ cases. Using the weight function derived directly from the absolute loss, we would encounter singularity at $x = 0$. Following Schuberg (2019) we adopted a numeric solution to get around the issue, which takes the form as below.

$$
w_i = \frac{1}{\max\{r, \|\epsilon_i\|\}}
$$

It turned out that above numeric solution of weights is equivalent to if the weights have been derived from Huber's loss as laid out in Fountoulakis and Gondzio (2016). In their work, they laid out Huber's loss as

$$
h(\epsilon_i, r) = \begin{cases} \frac{1}{2}\frac{\|\epsilon_i\|^2}{r} & \text{if } \|\epsilon_i\| \leq r \\ \|\epsilon_i\| - \frac{1}{2}r & \text{if } \|\epsilon_i\| > r \end{cases} \tag{5.5}
$$

The smaller the value of r is the better Huber function approximates the absolute loss. As a result, Schuberg (2019) has termed this solution as Huber's approxmation to least absoluate deviance (H-LAD) and suggested to use the value $r = 10^{-4}$.

Alternatively, widely used statistical softwares, such as SAS (*PROC ROBUSTREG*) and R (the *rlm()* function), have adopted Huber's loss in the below format.

$$
h(\epsilon_i, r) = \begin{cases} \frac{1}{2}\|\epsilon_i\|^2 & \text{if } \|\epsilon_i\| \leq r \\ r\|\epsilon_i\| - \frac{1}{2}r^2 & \text{if } \|\epsilon_i\| > r \end{cases} \tag{5.6}
$$

$$
w_i = \begin{cases} 1 & \text{if } \|\epsilon_i\| \leq r \\ \frac{r}{\|\epsilon_i\|} & \text{if } \|\epsilon_i\| > r \end{cases}
$$

Commonly in this format, r is set to be 1.345. The loss function and the weights are applied to scaled residuals ϵ_i/s, where s is an estimated scale parameter. The median absolute deviation (MAD) of the residuals scaled by a constant 0.6745 is used to estimate s, i.e.

$$\begin{aligned} \hat{s} &= \frac{\text{MAD}}{0.6745} \\ &= \frac{\text{median}(||\epsilon_i||)}{0.6745} \end{aligned} \tag{5.7}$$

Step 2: Given (W, η, ν)**, update** $(\mu, \boldsymbol{B})$

Updates to $\boldsymbol{B}$ follows weighted least square, i.e.

$$\boldsymbol{B}^{(t+1)} = (\boldsymbol{X}^T W^{(t+1)} \boldsymbol{X})^{-1} \boldsymbol{X}^T W^{(t+1)} (\boldsymbol{Y} - \boldsymbol{M}^{(t)})$$

To solve for μ we need to reorganize elements in (5.4). Let

$$\boldsymbol{Q}^w = \boldsymbol{X}(\boldsymbol{X}^T W \boldsymbol{X})^{-1} \boldsymbol{X}^T W,$$

then we have

$$\begin{aligned} \boldsymbol{Y} - \boldsymbol{M} - \boldsymbol{XB} &= (\boldsymbol{Y} - \boldsymbol{M}) - \boldsymbol{Q}^w (\boldsymbol{Y} - \boldsymbol{M}) \\ &= (\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}). \end{aligned}$$

Next we want to turn the goodness-of-fit part in (5.4) from matrix form to a long vector, so that it becomes a quadratic problem w.r.t the long vector μ.

$$\begin{aligned} &(\boldsymbol{Y} - \boldsymbol{M} - \boldsymbol{XB})^T W (\boldsymbol{Y} - \boldsymbol{M} - \boldsymbol{XB}) \\ &= \Big[(\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big]^T W \Big[(\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big] \\ &= \text{trace} \Big\{ W \Big[(\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big] \Big[(\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big]^T \Big\} \\ &= \text{trace} \Big\{ \Big[W^{1/2} (\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big] \Big[W^{1/2} (\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big]^T \Big\} \\ &= \text{vec}^T \Big[W^{1/2} (\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big] \text{vec} \Big[W^{1/2} (\boldsymbol{I}_n - \boldsymbol{Q}^w)(\boldsymbol{Y} - \boldsymbol{M}) \Big]. \end{aligned} \tag{5.8}$$

The last line of above derivation is because of (2.7), and based on

$$\text{vec}(AB) = (I_m \otimes A)\text{vec}(B),$$

we can show that

$$\begin{aligned}
&\mathrm{vec}^T\Big[W^{1/2}\big(\boldsymbol{I}_n-\boldsymbol{Q}^w\big)\big(\boldsymbol{Y}-\boldsymbol{M}\big)\Big]\\
&=\boldsymbol{I}_{\boldsymbol{q}}\otimes W^{1/2}\big(\boldsymbol{I}_n-\boldsymbol{Q}^w\big)\mathrm{vec}\big(\boldsymbol{Y}-\boldsymbol{M}\big)\\
&=\boldsymbol{I}_{\boldsymbol{q}}\otimes W^{1/2}\big(\boldsymbol{I}_n-\boldsymbol{Q}^w\big)\big(y-\mu\big)\\
&=\boldsymbol{D}^w\big(y-\mu\big),
\end{aligned}\tag{5.9}$$

where

$$\boldsymbol{D}^w=\boldsymbol{I}_{\boldsymbol{q}}\otimes W^{1/2}\big(\boldsymbol{I}_n-\boldsymbol{Q}^w\big).$$

Plugging (5.9) into (5.8), we can update (5.4) to

$$\begin{aligned}
&L(\mu;\boldsymbol{B},W,\eta,\nu)\\
&=(\boldsymbol{Y}-\boldsymbol{M}-\boldsymbol{X}\boldsymbol{B})^TW(\boldsymbol{Y}-\boldsymbol{M}-\boldsymbol{X}\boldsymbol{B})+\nu^T(\boldsymbol{A}\mu-\eta)+\frac{\upsilon}{2}\left\|\boldsymbol{A}\mu-\eta\right\|^2\\
&=\left\|\boldsymbol{D}^w\big(y-\mu\big)\right\|^2+\frac{\upsilon}{2}\left\|\boldsymbol{A}\mu-\eta+\upsilon^{-1}\nu\right\|^2+C,
\end{aligned}\tag{5.10}$$

where C is a constant w.r.t μ. With the loss function rearranged into (5.10) we can solve for μ, i.e.

$$\mu^{(t+1)}=\big(\boldsymbol{D}^{w^T}\boldsymbol{D}^w+\upsilon\boldsymbol{A}^T\boldsymbol{A}\big)^{-1}\Big[\boldsymbol{D}^{w^T}\boldsymbol{D}^wy+\upsilon\boldsymbol{A}^T(\eta-\upsilon^{-1}\nu)\Big].\tag{5.11}$$

Remaining steps and convergence

The rest of steps follow Section 2.1.

- **Step 3** to update η given $(\mu,\boldsymbol{B},W,\nu)$ is the same as **Step 2** in Section 2.1.
- **Step 4** to update ν given $(\mu,\boldsymbol{B},W,\nu)$ follows **Step 3** in Section 2.1.
- Iterations will stop when the pre-defined convergence criteria are met.

CHAPTER 6

MEAN SHIFT CLUSTERING

6.1 Model Formulation with Mean Shift Vector

Given formulation of Model 1 in Section 2.1, if outliers are expected and we use mean-shift approach to handle them, the minimization problem in (1.18) can be combined with the formulation in (2.1), it leads to

$$\begin{aligned} &\min_{S,\boldsymbol{E},\boldsymbol{B}} \sum_{j=1}^{K} \sum_{i \in S_j} \left\| y_i - \boldsymbol{B}^T x_i - e_i - \mu_j \right\|^2 + \sum_{i=1}^{n} p(||e_i||, \lambda) \\ &\text{s.t. } \mu_j = \frac{1}{\text{card } S_j} \sum_{i \in S_j} \left\| y_i - \boldsymbol{B}^T x_i \right\|^2, \\ &\bigcup_{j=1}^{K} S_j = \{1, ..., n\}, \end{aligned} \tag{6.1}$$

where

6.2 The Mean-shift Algorithm

To solve for this formulation (6.1), we adopted the algorithm proposed by Witten (2013), with an added step to solve for $\boldsymbol{B}$.

Step 1 Initialize $E^{(0)}$ and $\boldsymbol{B}^{(0)}$.

1a Run an OLS regression that estimates $\boldsymbol{B}^{(0)} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$.

1b For 90% of observations that are closest to the overall mean of the residuals $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{B}^{(0)}$, set their errors to $E^{(0)} = 0$, and others $\boldsymbol{E}^{(0)} = \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{B}^{(0)}$.

Step 2 Iterate until converge.

2a Perform K means clustering on $\boldsymbol{Y} - \boldsymbol{XB} - \boldsymbol{E}$, which gives solution to $\mu_j^{(t+1)}$,

2b Update $\boldsymbol{B}$ by solving

$$\min_{\boldsymbol{B}} \sum_{i=1}^{n} \left\| y_i - e_i - \mu_j - \boldsymbol{B}^T x_i \right\|^2,$$

which can be obtained from a least square solution, i.e.

$$\boldsymbol{B}^{(t+1)} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\big(\boldsymbol{Y} - \boldsymbol{E}^{(t)} - \boldsymbol{M}^{(t+1)}\big),$$

where $\boldsymbol{M} = (\mu_{S(1)}, ...\mu_{S(n)})^T \in \mathbb{R}^{nq}$ and $\mu_{S(i)}$ indicates the current cluster assignment of the i^{th}observation.

2c Update e_i by solving the minimization problem below for each element of $\boldsymbol{E}$, i.e., e_i.

$$\min_{e_i} \left\| y_i - \boldsymbol{B}^T x_i - \mu_{S(i)} - e_i \right\|^2 + p(||e_i||, \lambda).$$

We follow Witten (2013) to use group lasso as the penalty function, and we will get

$$e_i^{(t+1)} = \left(y_i - \boldsymbol{B}^T x_i - \mu_{S(i)}^{(t+1)}\right) \max\left[0, 1 - \frac{\lambda}{\left\| y_i - \boldsymbol{B}^T x_i - \mu_{S(i)}^{(t+1)} \right\|}\right].$$

Step 3 For observations with $||e_i|| = 0$, conduct a new round of K means and update estimation of parameters $\mu_j^{(\text{final})}$, $S_i^{(\text{final})}$, and $\boldsymbol{B}^{(\text{final})}$

To check for convergence, we check the difference between iteration t and $t-1$ in the estimation of error $\boldsymbol{E}$. If $\left\| \boldsymbol{E}^{(t)} - \boldsymbol{E}^{(t-1)} \right\|_\infty < \epsilon$, the model is determined as converged and $\|\mathbf{x}\|_\infty := \max_i |x_i|$.

CHAPTER 7

SIMULATION STUDIES

7.1 Simulation Study 1

To investigate the performance of the proposed robust solutions compared to the original ADMM algorithm as evaluated in Chapter 4, we conducted a set of simulation studies. In this set of examples, we fix ADMM dual variables $\upsilon = 0$ and $\nu = 0$. For both ADMM and IRLS-ADMM, group-wise MCP was used with $\gamma = 2$. H-LAD was used in IRLS-ADMM.

Similar to Chapter 3, we use the modified BIC to select the tuning parameter λ for ADMM and IRLS-ADMM, with a small tweak in the goodness-of-fit term for IRLS-ADMM. The tweak adopts Manhattan distance or $l2$ norm for the goodness-of-fit term for $q > 1$ cases, instead of Euclidean distance (least-square).

To measure clustering accuracy, we continue to use Rand Index (RI). And to measure estimation accuracy, in addition to RMSE, we also report AAD (average absolute deviation), which may better represent errors when outliers are present. AAD is defined as

$$\text{AAD} = \frac{1}{nq} \sum_{i=1}^{n} \sum_{j=1}^{q} |\hat{a}_{ij} - a|.$$

It is worth noting that in our simulation studies when computing performance measures, all data were included in computation. Performance should be improved when outliers are excluded from computation. Schuberg (2019) excluded all data that have a squared residual greater than $\chi^2_{0.95}$ when reporting performance measures.

We simulate $n = 100$ data points from below model

$$y_i = \mu_i + x_i^T \beta + \epsilon_i, i = 1, \ldots, n,$$

where $x_i \in \mathbb{R}^p$, $p = 3$ is generated from a multivariate normal distribution with mean 0 and covariance matrix I_3. μ_i is a scalar and we simulated it from two values (3 or -1) with equal probabilities and β is simulated from Uniform$(0.5, 1)$. The error term ϵ_i is a scalar. We simulated these errors from a few different settings.

1. All from $N(0, 1)$,
2. 95% from $N(0, 1)$, 5% from $N(0, 100)$,
3. 90% from $N(0, 1)$, 10% from $N(0, 100)$,
4. t distribution with degree of freedom (d.f.) as 2,
5. Laplace distribution.

In each of above scenarios, we compared IRLS-ADMM and Mean-shift clustering with ADMM. Results are shown in Table 7.1 and visualized in Figure 7.1. A hundred repetitions were run for each setting. Mean and standard error (SE) of the performance measures are reported. Performance measures include Rand Index (RI), root of mean squared error (RMSE) of μ and β, and average absolute deviation (AAD) of μ and β.

As seen from the results, least square based ADMM method has reduced performance when outliers are present or the distribution is long-tail (comparing Setting 1 and the others). The reduction is especially significant in estimation accuracy of μ. The reduction seems to be heavily led to by outliers because of the magnitude of differences between RMSE and AAD of μ, comparing Setting 1 to the others. Little impact seems to happen to estimation of β. It is interesting to see that with 5% outliers added, the clustering accuracy (RI) decreased by 5% (Setting 1 vs. Setting 2), which is expected because all data were used in computing performance measures for ADMM and IRLS-ADMM.

When the errors are generated from Normal distribution, both IRLS-ADMM and Mean-shift clustering appear to have better clustering results and estimation accuracy when the proportion of outliers is 5% (Setting 2). The advantage diminishes when the proportion increase to 10% (Setting 3), while mean-shift clustering still seem to perform well in estimating μ. For t distribution, there don't seem to be much difference among the 3 methods, despite mean-shift clustering still seem to have better accuracy when it comes to estimating μ. For Laplace distribution, IRLS-ADMM performs better in both clustering and estimating of model parameters. This is expected as $l1$ loss is known to behave well when the underlying distribution is Laplace. All three methods do not seem to differ much in estimating the regression slopes β.

It is interesting to see that in Setting 1, when the simulation setting does not generate outliers nor is the distribution long-tailed, IRLS-ADMM outperforms ADMM.

7.2 Simulation Study 2

In this section we look at the the anatomy of both robust solution models through a simple example continued from Figure 1.2 in Section 1.6.1. We randomly split the non-outliers to two groups with probability $\pi = 0.5$, one group has intercept 5 while the other has intercept -1. Regression slope is set to 1.5. Both x and the errors are drawn from standard normal. Scatter plot is updated in Figure 7.2a. Note that the outlier in this scenario may influence both clustering and regression.

Solution path of the IRLS-ADMM method is presented in Figure 7.2b. In this case, the IRLS-ADMM finds its optimum at $\lambda = 1.61$ and successfully identifies the clusters and the outlier. Estimated slope from IRLS-ADMM is 1.34, intercepts for group 1 is -1.17 and 5.15 for group 2. When we look into how the weights in the optimal model changes over iterations, it follows quite strange a pattern, and it is certainly not monotonic (see Figure 7.2d). When viewed together with changes in estimation of μ of the outlier and β over the iterations, it seems that the iteration may have entered into a steady period in which weights of the outlier no longer matters. Specifically, we noted that the estimated μ for the outlier was constantly pushed afar from where the majority of data points locate (see Figure 7.2e), while the estimate

of β approaches true value and becomes steady. Interestingly, the fluctuation of the outlier's weight starts around the time (iteration 150 onwards) when the estimate of its μ was pushed furthest and became steady, and it is also the time when the estimate of β for the model became steady. This may imply that the estimates of the outlier gets into a point that its weight no longer matters. Specifically, when minimizing over the coordinate of μ for $||y - \mu - x\beta||$, it is so far from the rest of data points it gets its own center and makes up its own cluster. And when minimizing over the coordinate of β, its location in relation with the other points was shifted by the operation of $y - \mu$ and settled in a place that it trends with the rest of data points.

The phenomenon that μ of the outlier is constantly pushed afar from where the majority of points locate persisted when we changed the seed and repeated the simulation. This may imply that during the model iterations when it comes to clustering operation, the outlier was left out from the clustering of the main cohort. And when it comes to the operation of estimating regression slope, the outlier's influence was either gradually downweighted as compared to the rest of data points or it got pulled by the pulled-afar μ value so much so that it started to trend in the direction together with the majority of data points.

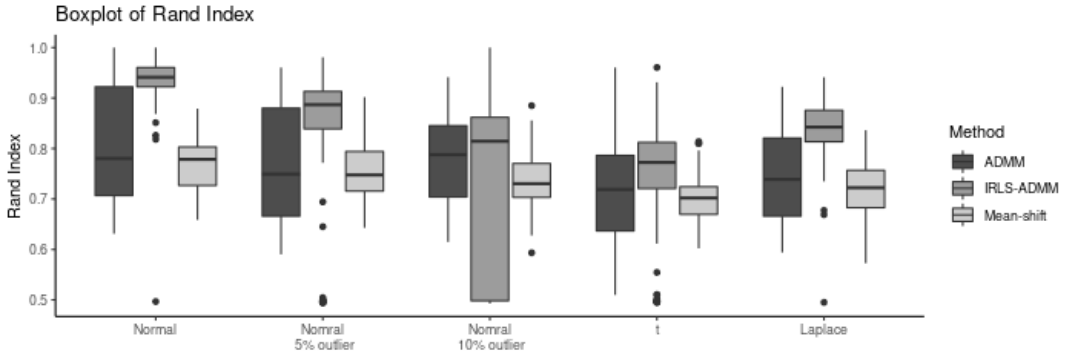
The philosophy of alienating the outlier(s) mirrors with that of the mean-shift based method. Such connection may be explained by both She and Owen (2011) and Witten (2013), which demonstrate that mean-shift based optimization finds the same minimum as the one based on Huber's loss. However, in our example Huber's loss based IRLS-ADMM method did not end up having the same estimate of β as compared to the mean-shift method. This may be due to the facts that (1) numeric solutions may not have settled on the optimum, (2) mean-shift method renders the traditional K-means algorithm while IRLS-ADMM uses convex-ified K means search through ADMM algorithm. To understand the theoretical connections between these two methods further research if required.

Moreover, it is important to point out that we also see similar phenomenon in the ADMM-based method. Viewing from its solution path in Figure 7.2c, this method also alienates the outlier and pushes its μ value afar from the rest of data points. If this is indeed true, the real difference between ADMM and IRLS-ADMM may only lie in how they estimate β, i.e., whether to use ordinary least square or weighted

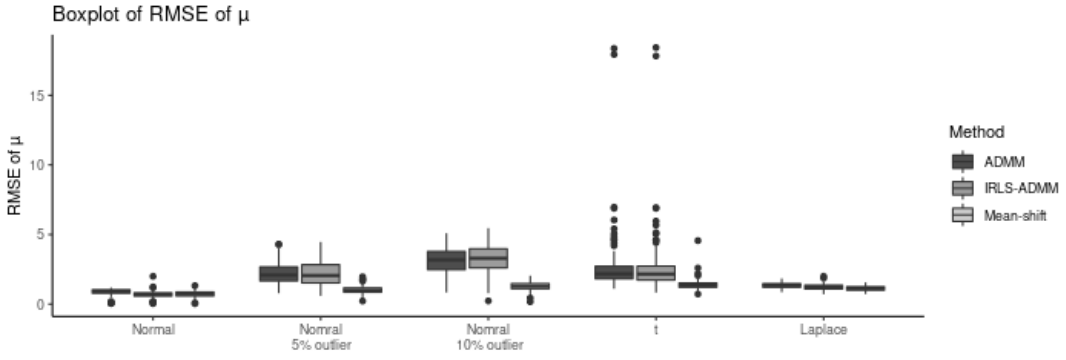
least square. In fact, by playing with different factors in the simulation setup, these two methods' results mostly differ in their estimations of β, and IRLS-ADMM appears to have some advantages over ADMM, when both models are given careful attentions for tuning. This may have explained why we have not seen significant differences between these two methods in previous simulation studies (see Section 7.1). In other words, least-square loss based models (ADMM) in this case may be a self-sufficient robust model because of the phenomenon of alienating outlier(s) and the benefits added by replacing least-square loss with robust loss may be minuscule. Such unclear advantages transiting from least-square solution to robust solutions in the setting of utilizing fusion penalty for clustering is also seen in Zhang (2019).

TABLE 7.1: Mean and standard error (SE) of Performance Measurements for ADMM, IRLS-ADMM, and Mean-shift Clustering (RI = Rand Index, RMSE = root of mean squared error, AAD = average absolute devaition)

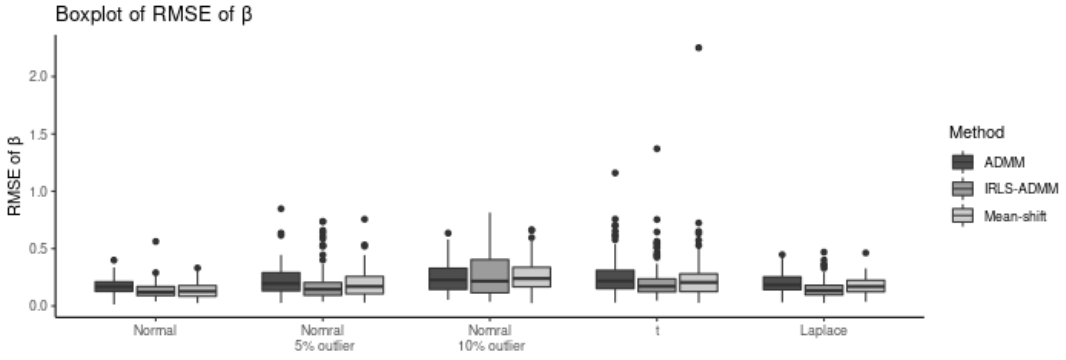
	Statistic	ADMM	IRLS-ADMM	Mean-shift
Setting 1	RI	0.807 (0.011)	0.935 (0.004)	0.769 (0.005)
	RMSE μ	0.861 (0.025)	0.697 (0.022)	0.730 (0.022)
	RMSE β	0.170 (0.008)	0.135 (0.006)	0.138 (0.006)
	AAD μ	0.581 (0.027)	0.288 (0.014)	0.277 (0.013)
	AAD β	0.150 (0.007)	0.117 (0.005)	0.120 (0.006)
Setting 2	RI	0.768 (0.011)	0.823 (0.015)	0.755 (0.005)
	RMSE μ	2.228 (0.080)	2.224 (0.090)	1.024 (0.028)
	RMSE β	0.225 (0.014)	0.201 (0.016)	0.192 (0.012)
	AAD μ	0.980 (0.039)	0.931 (0.072)	0.438 (0.021)
	AAD β	0.199 (0.013)	0.176 (0.015)	0.168 (0.011)
Setting 3	RI	0.776 (0.009)	0.719 (0.018)	0.736 (0.005)
	RMSE μ	3.154 (0.091)	3.262 (0.099)	1.277 (0.033)
	RMSE β	0.254 (0.013)	0.288 (0.020)	0.269 (0.014)
	AAD μ	1.269 (0.042)	1.607 (0.088)	0.656 (0.028)
	AAD β	0.222 (0.012)	0.252 (0.018)	0.235 (0.013)
Setting 4	RI	0.714 (0.009)	0.744 (0.011)	0.702 (0.005)
	RMSE μ	2.808 (0.247)	2.793 (0.249)	1.424 (0.042)
	RMSE β	0.265 (0.018)	0.227 (0.018)	0.247 (0.024)
	AAD μ	1.242 (0.056)	1.178 (0.068)	0.705 (0.041)
	AAD β	0.229 (0.016)	0.215 (0.023)	0.215 (0.023)
Setting 5	RI	0.745 (0.009)	0.829 (0.007)	0.720 (0.005)
	RMSE μ	1.345 (0.017)	1.264 (0.025)	1.126 (0.018)
	RMSE β	0.198 (0.009)	0.155 (0.009)	0.178 (0.007)
	AAD μ	0.803 (0.024)	0.585 (0.029)	0.473 (0.013)
	AAD β	0.171 (0.008)	0.135 (0.008)	0.152 (0.007)



(A) Rand Index



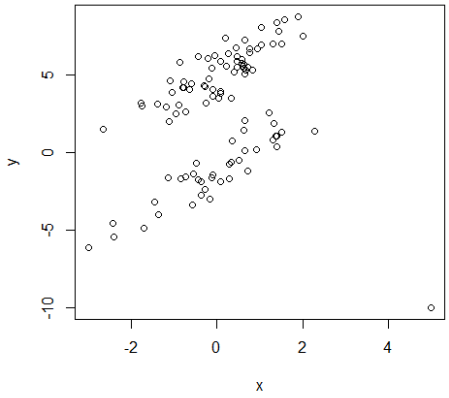
(B) RMSE of μ



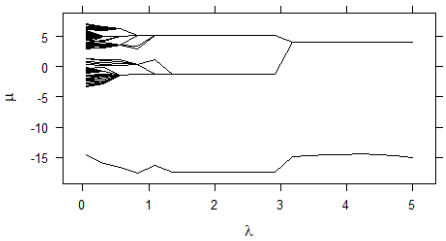
(C) RMSE of β

FIGURE 7.1: Performance Measurements for ADMM, IRLS-ADMM, and Mean-shift Clustering (IPOD)

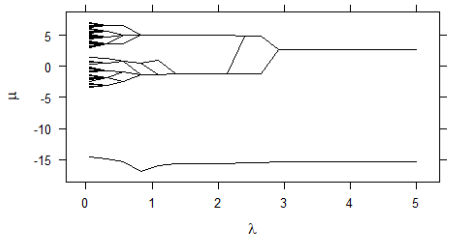
(A) Scatter plot



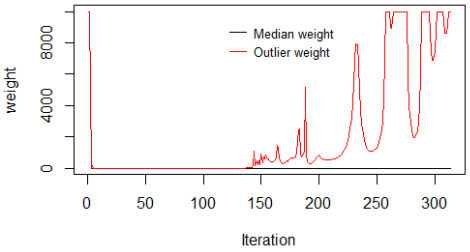
(B) Solution path of IRLS-ADMM Model



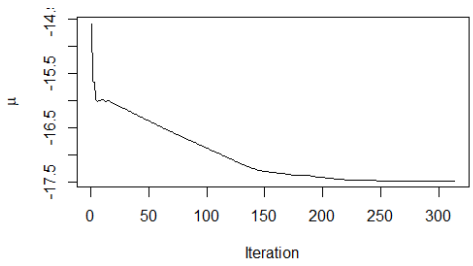
(C) Solution path of ADMM Model



(D) Weights of IRLS-ADMM Model



(E) μ of Outlier



(F) β of Outlier

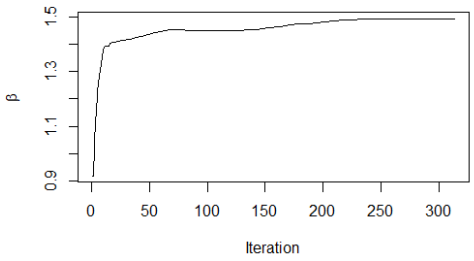


FIGURE 7.2: Continued example from Figure 1.2 splitted to 2 groups

CHAPTER 8

EMPIRICAL STUDIES

8.1 AIDS CD4 Data

In this section, we use developed robust methods to analyze dataset from Hammer SM et. al. (1996). This dataset was collected from a clinical trial designed to compare efficacy from multiple therapies in treating HIV-infected adults with CD4 cell counts from 200 to 500 per cubic millimeter. Change from baseline at Week 96 in count of CD4 is the endpoint being analyzed. We included in analysis only data points from two treatment groups, Zidovudine and Zidovudine + Didanosine. Five hundred and twenty two patients were randomized to these two treatment groups. Their CD4 cell counts at baseline (week 0), week 20, and week 96 were recorded. After adjusting to common demographic information and baseline characteristics (x1=age, x2=weight in kg, x3=Karnofsky score, x4=CD4 count at baseline, x5=gender, x6=Yes or No to drug use, x7=Yes or No to history of antiretroviral treatment, x8=symptomatic or not), we clearly see bi-modality presents in residuals (Figure 8.1a), as well as an enlonged right tail. We utilize the robust solutions developed to analyze the data, and compare our results with the ADMM-based regression from Ma and Huang (2017). Clustering and Estimation Results are listed in Table 8.1.

IRLS-ADMM clustering ended up with 1 cluster. Looking at the final residual plot from IRLS-ADMM, Figure 8.1c, IRLS-ADMM seems to have failed to pick up the hetergeneity. IRLS-ADMM has similar solution path (see Figure 8.2) as compared with ADMM. ADMM found its optimal $\lambda = 0.3$ while IRLS-ADMM had it at $\lambda = 2.1$. This fact of similar solution path but different optimum may be led to by the different loss function used in the modified BIC formula. Absolute loss was used for

IRLS-ADMM and squared loss was used for ADMM. In addition, looking at both solution paths, it looks rather unlikely the stable division of groups occurs around $\lambda = 0.3$ or $\lambda > 2$. It is hard to conclude whether optimal solution was found in either method.

Mean-shift clustering results with a preset $K = 2$ were also included. Its final residual plot (Figure 8.1d) had heterogeneity removed. Looking at residual plot for each cluster in Figure 8.1f, we see that the distribution is more homogeneous within each identified cluster than the distribution of all response values as shown in Figure 8.1a, which is not the case for ADMM's within-cluster residuals (Figure 8.1e). However, ADMM has better results when evaluated by R^2 and DB-index.

8.2 Parkinson's and Rapid Eye Movement Data Continued

Continued from Chapter 4, we applied robust solutions to the dataset. IRLS-ADMM cluster all data points into one cluster. It is the case from separate single-response models and from multi-response model, and will be omitted from presentation here. In this case, IRLS-ADMM models boild down to a simple weighted least square regression. Estimations of coefficients of the covariates are similar to ADMM results.

Mean-Shift model was also run and its results compared with ADMM are presented in Table 8.2. R^2 is higher in the Mean-shift model, while the DB-index performs worse. This implies that result from Mean-shift model seems to outperforms ADMM in fitting the data, however, clusters defined by Mean-shift model seem to lack internal coherent when compared with ADMM. DB-index E has its distance measured by Euclidean distance, while DB-index M uses Manhattan distance. The number of groups for the Mean-shift model is set as $K = 2$. One reminder is that throughout our work, we assume the number of groups for the Mean-shift model is known. As compared with the fused-penalty based models (ADMM and IRLS-ADMM), this is a downside of the Mean-shift model. K will need to be jointly tuned with λ and an updated algorithm to conduct such tuning is yet to be developed. The regression estimates from both models are similar, except for the slope estimate for DPI. ADMM gives negative estimation while mean-shift method gives positive.

TABLE 8.1: Subgroup analysis results of the AIDS CD4 Count dataset from Hammer SM, et. al. (1996)

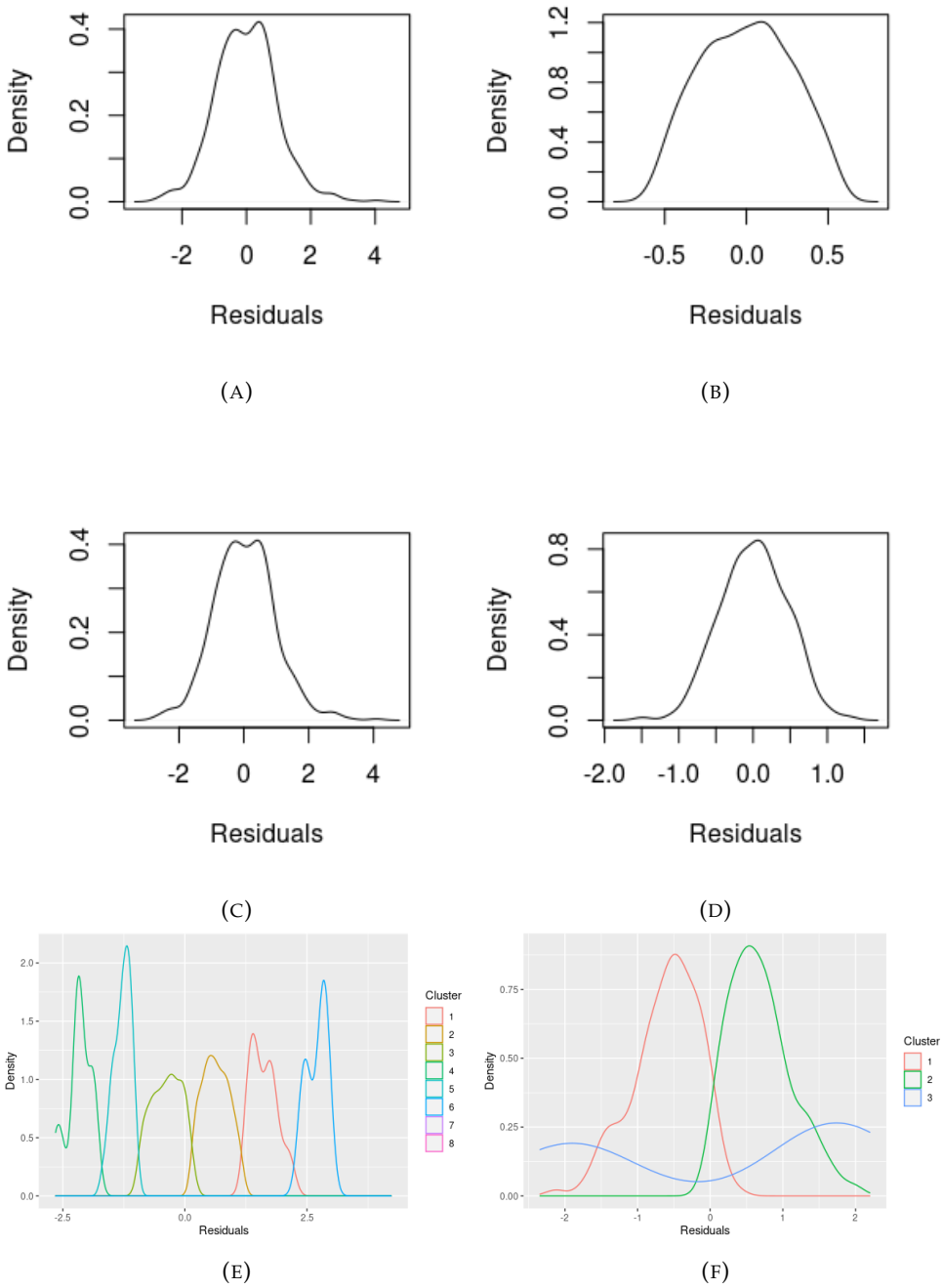
Parameter	OLS	ADMM	IRLS-ADMM	Mean-shift
K	NA	8	1	2
R^2	0.112	0.9232	0.1075	0.7037
DB-Index	NA	0.4352	NA	0.6943
$\hat{\beta}_{\text{age}}$	0.0568	0.0518	0.0665	0.072
$\hat{\beta}_{\text{weight}}$	0.0631	0.0539	0.0397	0.0169
$\hat{\beta}_{\text{karnofsky}}$	0.0333	0.0362	0.0721	0.0465
$\hat{\beta}_{\text{CD4 at w0}}$	-0.276	-0.2654	-0.2828	-0.1129
$\hat{\beta}_{\text{gender:M}}$	0.0451	0.1148	0.0365	0.1195
$\hat{\beta}_{\text{drug use}}$	0.0005	-0.0295	-0.0237	0.1108
$\hat{\beta}_{\text{antiretroviral}}$	-0.3711	-0.352	-0.02855	-0.1654
$\hat{\beta}_{\text{symptom}}$	-0.1873	-0.1918	-0.1598	-0.127

TABLE 8.2: Subgroup analysis results of the Parkinson's disease dataset from Hlavnicka (2016).

Parameter	ADMM	Mean-Shift
K	2	2
R^2	0.606	0.785
DB-index E	0.602	0.870
DB-index M	0.432	0.634
	(DPI, RST)	(DPI, RST)
$\hat{\beta}_{\text{age}}$	(-0.0120, -0.0036)	(0.0291, -0.0282)
$\hat{\beta}_{\text{gender:F}}$	(-0.1108, 0.0511)	(-0.0280, 0.0351)
$\hat{\beta}_{\text{antidepress}}$	(0.1244, -0.0600)	(0.1257, -0.0259)

Interpreting DPI, duration of pause interval, it should increase as one ages. Thus the positive slope makes more sense.

Mean-shift modeled membership mapped on original data (monologue DPI and RST) is plotted in Figure 8.3. Outliers are grouped in group Member $= 3$. The clustering results look very different as compared with that of ADMM and IRLS-ADMM models. IRLS-ADMM model considers all data points to be in one group, and ADMM model considers majority of the data points in one group, while the smaller cluster seems to be a set of outliers rather than a new group.



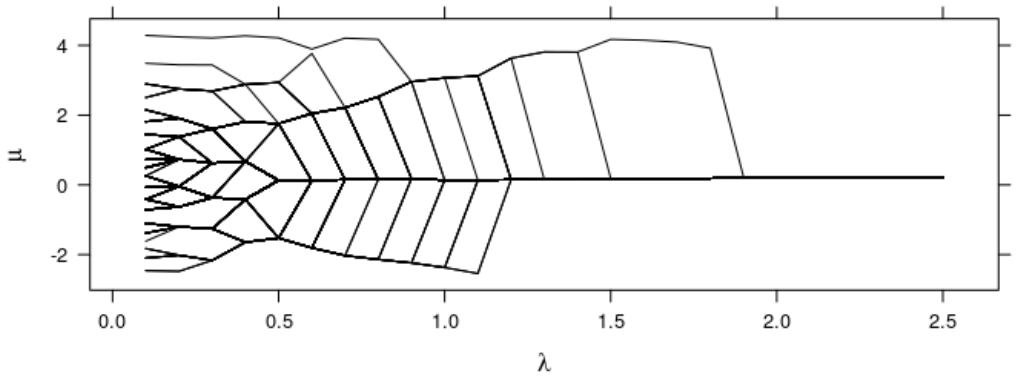
(A) (B)

(C) (D)

(E) (F)

FIGURE 8.1: Density Plot of Modeling Residuals for the AIDS CD4 Count dataset (A: OLS residuals; B: ADMM residuals; C: IRLS residuals; D: Mean-shift residuals; E: ADMM residuals by clusters, cluster 7 and 8 have only 1 data points thus omitted; F: Mean-shift residuals by clusters, cluster 3 is the group of identified outliers.)

(A) Solution path from ADMM



(B) Solution path from IRLS-ADMM

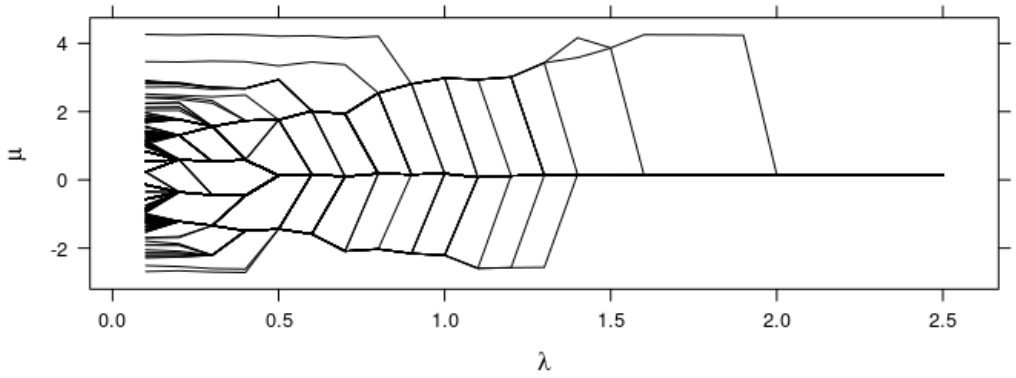


FIGURE 8.2: Solution Path of the AIDS CD4 Count dataset using ADMM and IRLS-ADMM clustering

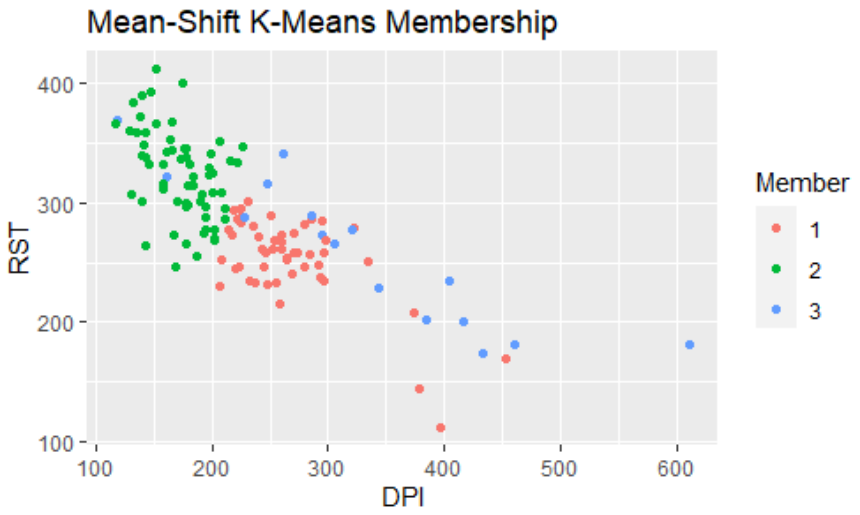


FIGURE 8.3: Clustering Results Mapped to Original Data for the RBD AND PD dataset, Members = 3 are identified outliers.

CHAPTER 9

CONCLUDING REMARKS

9.1 Conclusions and Contributions

The first part of our work introduced solution path (convex-ified K-means) clustering in the setting of multi-response regression. It is an extension from Ma and Huang (2017). Ma and Huang's work implemented solution path clustering in the single-response regression setting. We tested our model using both convex (LASSO) and concave (MCP) penalties, and we demonstrated that concave penalties in modeling of multi-response regression outperformed convex penalties. We introduced two types of penalty constructs, component-wise and group-wise. When comparing their performance, we noted that the group-wise construct of penalties generated more intuitive results and the accuracy was improved for both clustering and estimation of regression coefficients. We also compared performance of our multi-response extension with results from single-response models that were run separately for each response dimension. The multivariate extension has the advantage to uncover information that exists in multi-dimensional space while such information is not approachable through modeling from uni-dimension. This fact resulted in improved accuracy in clustering and estimation of regression coefficients. Depending on the locations of clusters, such advantages may amplify as the correlation between response variables varies.

We later implemented robust solutions to the solution path clustering of regressions. Two approaches handling outliers and/or heavy-tail distributions were introduced. The first approach is to replace least-square loss with robust-loss. This line of work is based on the setting of convex-ified clustering and it was first seen

in the doctoral work of Schuberg (2019). Zhang (2019) also pursued such extension, i.e., solution path clustering in regression setting, with an absolute loss, and they approached the solution through the local linear approximation (LLA, Zou 2008) algorithm. The second approach is to add a mean-shift component in the framework of K means clustering. This addition was inspired by She and Owen (2011). It was first combined with K means clustering by Witten (2013). We further added the regression component to the formulation in Witten (2013). Through simulation studies we demonstrated that when the underlying distribution was long-tailed (e.g. student's t and Laplace) and when the amount of outliers was small ($\sim$5%), the robust-loss based methods outperformed the least-square loss based method. Mean-shift clustering performed sub-optimally in terms of clustering accuracy when compared with robust-loss based convex clustering. It had similar performance when compared with least-square based convex clustering. However, the mean-shift clustering method had improved accuracy for estimating the heterogeneous intercepts. All three methods did not seem to differ much in estimating the regression slopes. We also explored the connections between these two robust solutions. From simulated examples we discovered that these two methods used similar principles to handle outlier(s), i.e. alienating outlier(s) while modeling the majority of data points. We also discussed the possibility that the method that bases on least-square loss combined with concave fusion penalty (ADMM) may have sufficient robustness than what is otherwise perceived.

9.2 Future Research Directions

The most critical challenges for the proposed methods resides in computation. These methods (ADMM and IRLS-ADMM) utilize pairwise fusion penalty to conduct clustering. It requires the model to conduct $\binom{n}{2}$ comparisons at each iteration. Such conduct results in a n^2 computational cost, i.e. as n grows the number of comparisons needed grows at a rate of n^2. Potential directions for improvement include utilizing a k-nearest-neighbor approach and/or naive random selection to down-size the number of pairs to be compared. Zhang (2019) has attempted such solution using the

idea of divide-and-conquer from massive data analysis and it has achieved promising improvements in computation time. In addition, their work used an algorithm in replacement of the ADMM, which seemed to converge faster than ADMM.

Methods used to select the number of clusters need to be improved. When implementing the mBIC for convex clustering methods (ADMM and IRLS-ADMM), we noticed that the construct of goodness-of-fit term could heavily influence the selected number of groups. As a result, it influenced clustering and estimation accuracy. When using the IRLS-ADMM method, we used Manhattan distance to construct the goodness-of-fit term and it may have biased IRLS-ADMM results towards selecting smaller number of groups. Moreover, picking the constant for mBIC is itself a work of art. This constant weighs how mBIC favors between goodness-of-fit and the complexity component. It is not as simple as the literature has claimed. Specifically, small change in the constant may change the optimal selection of K dramatically. And for the mean-shift clustering based method, the method to select the number of clusters is underdeveloped. Methods implemented by traditional K-means may be considered and will need to be implemented together with the tuning of sparsity for the mean-shift vector.

The multi-response extension of our work may be improved by taking into consideration of multivariate covariance structure. We explored how the different correlations may impact the clustering results, however, we did not take into consideration of the covariance at the time of model construction. Such understanding may improve model performance and provide a foundation for development of inferential framework. This line of work will also help to improve the heterogeneous regression slope models (Section 2.4.2). Further simulation and empirical studies are needed to understand the basic behavior of the heterogeneous regression slope models. A nice review of current efforts is presented and discussed in Price et. al. (2021).

The mean-shift approach proposed in Chapter 6 can be easily extended to the convex-ified clustering framework. Specifically, the shifted-mean vector $\boldsymbol{E}$ may be

added to objective function 2.2 resulting in

$$S(\mu, \boldsymbol{B}, \boldsymbol{E}) = \frac{1}{2}\sum_{i=1}^{n}\left\|y_i - \mu_i - \boldsymbol{B}^T x_i - e_i\right\|^2 + \sum_{1\le i<j\le n} p(\mu_i - \mu_j, \lambda_1) + \sum_{i=1}^{n} g(||e_i||, \lambda_2),$$

where $g(\cdot)$ is any regularization function with tuning parameter λ_2. This model may be solved by adding one layer of iteration in the ADMM algorithm which solves for $\boldsymbol{E}$. With such replacement, the investigation for connections between robust loss and mean-shift formulation can be properly conducted. Study of this connection may lead to a unified algorithm and improved modeling procedure may be discovered. In the current work, we only explored behaviors and performances of these two formulations via simulations and empirical examples, further research are required.

BIBLIOGRAPHY

[1] Aftab, Khurrum, Hartley, Richard, and Trumpf, Jochen. "Generalized weiszfeld algorithms for Lq optimization". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (4 Apr. 2015), pp. 728–745. ISSN: 01628828. DOI: 10.1109/TPAMI.2014.2353625.

[2] Beaton, Albert E. and Tukey, John W. "FITTING OF POWER SERIES, MEANING POLYNOMIALS, ILLUSTRATED ON BAND-SPECTROSCOPIC DATA." In: *Technometrics* 16 (2 1974). ISSN: 00401706. DOI: 10.2307/1267936.

[3] Boyd, Stephen P. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Vol. 3. Now Publishers Inc., 2010, pp. 1–122. URL: https://web.stanford.edu/~boyd/papers/pdf/admm_distr_stats.pdf.

[4] Chi, Eric C and Lange, Kenneth. "Splitting Methods for Convex Clustering". In: *Journal of Computational and Graphical Statistics* 24 (2015), pp. 994–1013. DOI: 10.1080/10618600.2014.948181.

[5] Cott, S et al. "A TRIAL COMPARING NUCLEOSIDE MONOTHERAPY WITH COMBINATION THERAPY IN HIV-INFECTED ADULTS WITH CD4 CELL COUNTS FROM 200 TO 500 PER CUBIC MILLIMETER". In: *The New England Journal of Medicine* 335 (16 1996), pp. 1081–1090.

[6] Fan, Jianqing and Li, Runze. "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties". In: *Journal of the American Statistical Association* 96 (2001), pp. 1348–1360. DOI: 10.1198/016214501753382273.

[7] Fop, Michael and Murphy, Thomas Brendan. *Variable selection methods for model-based clustering*. 2018. DOI: 10.1214/18-SS119.

[8] Fountoulakis, Kimon and Gondzio, Jacek. "A second-order method for strongly convex (Formula presented.) -regularization problems". In: *Mathematical Programming* 156 (1-2 Mar. 2016), pp. 189–219. ISSN: 14364646. DOI: 10.1007/s10107-015-0875-4.

[9] Fox, John and Weisberg, Sanford. *An R Companion to Applied Regression, Third Edition*. 2019.

[10] Gupta, Shalmoli et al. "Local search methods for k-means with outliers". In: vol. 10. 2017. DOI: 10.14778/3067421.3067425.

[11] Hocking, Toby Dylan et al. "Clusterpath An Algorithm for Clustering using Convex Fusion Penalties". In: *28th international conference on machine learning* (2011).

[12] Huang, Jian, Breheny, Patrick, and Ma, Shuangge. "A Selective Review of Group Selection in High-Dimensional Models". In: *Statistical Science* 27 (4 2012), pp. 481–499. DOI: 10.1214/12-STS392.

[13] Huber, Peter J. "Robust Estimation of a Location Parameter". In: *The Annals of Mathematical Statistics* 35 (1 1964). ISSN: 0003-4851. DOI: 10.1214/aoms/1177703732.

[14] Izenman, Alan. *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. 2013. ISBN: 9780387781884; 0387781889; 9780387781891; 0387781897. DOI: 10.1007/978-0-387-78189-1.

[15] Li, Gen et al. "Integrative multi-view regression: Bridging group-sparse and low-rank models.(Report)". In: *Biometrics* 74 (2 2019), pp. 593–603. DOI: 10.1111/biom.13006.

[16] Liang, Baosheng et al. "Regression and subgroup detection for heterogeneous samples". In: *Computational Statistics* 35 (4 2020). ISSN: 16139658. DOI: 10.1007/s00180-020-00965-5.

[17] Lindsten, Fredrik, Ohlsson, Henrik, and Ljung, Lennart. "Just Relax and Come Clustering ! A Convexification of k-Means Clustering". In: *Control* (2011).

[18] Liu, Tzu Ying and Jiang, Hui. "Minimizing Sum of Truncated Convex Functions and Its Applications". In: *Journal of Computational and Graphical Statistics* 28 (1 Jan. 2019), pp. 1–10. ISSN: 15372715. DOI: 10.1080/10618600.2017.1390471.

[19] Lloyd, Stuart P. "Least Squares Quantization in PCM". In: *IEEE Transactions on Information Theory* 28 (2 1982). ISSN: 15579654. DOI: 10.1109/TIT.1982.1056489.

[20] Lu, Wenqi et al. "Multiply robust subgroup identification for longitudinal data with dropouts via median regression". In: *Journal of Multivariate Analysis* 181 (2021). ISSN: 10957243. DOI: 10.1016/j.jmva.2020.104691.

[21] Ma, Shujie and Huang, Jian. "A Concave Pairwise Fusion Approach to Subgroup Analysis". In: *Journal of the American Statistical Association* 112 (517 2017). ISSN: 1537274X. DOI: 10.1080/01621459.2016.1148039.

[22] Ma, Shujie et al. "Exploration of heterogeneous treatment effects via concave fusion". In: *The International Journal of Biostatistics* 16 (1 2016).

[23] Marchetti, Yuliya and Zhou, Qing. "Solution path clustering with adaptive concave penalty". In: *Electronic Journal of Statistics* 8 (2014), pp. 1569–1603. ISSN: 19357524. DOI: 10.1214/14-EJS934.

[24] Menezes, D. Q.F. de et al. "A review on robust M-estimators for regression analysis". In: *Computers and Chemical Engineering* 147 (2021). ISSN: 00981354. DOI: 10.1016/j.compchemeng.2021.107254.

[25] Peker, Eli and Wiesel, Ami. "Fitting Generalized Multivariate Huber Loss Functions". In: *IEEE Signal Processing Letters* 23 (11 2016). ISSN: 10709908. DOI: 10.1109/LSP.2016.2612170.

[26] Phillips, Robert F. *Least absolute deviations estimation via the EM algorithm*. 2002, pp. 281–285.

[27] Price, Bradley S., Allenbrand, Corban, and Sherwood, Ben. "Detecting clusters in multivariate response regression". In: *Computational Statistics* (2021). ISSN: 19390068. DOI: 10.1002/wics.1551.

[28] Quandt, Richard E. and Ramsey, James B. "Estimating mixtures of normal distributions and switching regressions". In: *Journal of the American Statistical Association* 73 (364 1978). ISSN: 1537274X. DOI: 10.1080/01621459.1978.10480085.

[29] Rand, William M. "Objective Criteria for the Evaluation of Clustering Methods". In: *Journal of the American Statistical Association* 66 (1971), pp. 846–850. DOI: 10.1080/01621459.1971.10482356.

[30] Raskutti, Garvesh, Yuan, Ming, and Chen, Han. "Convex regularization for high-dimensional multiresponse tensor regression". In: *Ann. Statist.* 47 (3 2019), pp. 1554–1584. DOI: 10.1214/18-AOS1725.

[31] Raykov, Yordan P et al. "What to Do When K-Means Clustering Fails: A Simple yet Principled Alternative Algorithm". In: *PLOS ONE* (2016). DOI: 10.1371/journal.pone.0162259.

[32] Schuberg, Edward. "Solution Path Clustering with Robust Loss and Concave Penalty". 2019.

[33] Schwarz, Gideon. "Estimating the Dimension of a Model". In: *The Annals of Statistics* 6 (2 1978), pp. 461–464. DOI: 10.1214/aos/1176344136.

[34] She, Yiyuan. "SPARSE REGRESSION WITH EXACT CLUSTERING A DISSERTATION SUBMITTED TO THE DEPARTMENT OF DEPARTMENT OF STATISTICS AND THE COMMITTEE ON GRADUATE STUDIES OF STANFORD UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY". 2008.

[35] Sun, Wei, Wang, Junhui, and Fang, Yixin. "Regularized k-means clustering of high-dimensional data and its asymptotic consistency". In: *Electronic Journal of Statistics* 6 (2012). ISSN: 19357524. DOI: 10.1214/12-EJS668.

[36] Tan, Kean Ming and Witten, Daniela. "Statistical properties of convex clustering". In: *Electronic Journal of Statistics* 9 (2 2015). ISSN: 19357524. DOI: 10.1214/15-EJS1074.

[37] Tibshirani, Robert. "Regression shrinkage and selection via the lasso: A retrospective". In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 73 (3 2011). ISSN: 13697412. DOI: 10.1111/j.1467-9868.2011.00771.x.

[38] Tibshirani, Robert et al. "Sparsity and smoothness via the fused lasso". In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 67 (1 2005). ISSN: 13697412. DOI: 10.1111/j.1467-9868.2005.00490.x.

[39] Wang, Binhuan et al. "Sparse Convex Clustering". In: *Journal of Computational and Graphical Statistics* 27 (2 2018). ISSN: 15372715. DOI: 10.1080/10618600.2017.1377081.

[40] Wang, Hansheng, Li, Guodong, and Jiang, Guohua. "Robust regression shrinkage and consistent variable selection through the LAD-lasso". In: *Journal of Business and Economic Statistics* 25 (3 2007), pp. 347–355. ISSN: 07350015. DOI: 10.1198/073500106000000251.

[41] Wang, Li, Gordon, Michael D., and Zhu, Ji. "Regularized least absolute deviations regression and an efficient algorithm for parameter tuning". In: 2006, pp. 690–700. ISBN: 0769527019. DOI: 10.1109/ICDM.2006.134.

[42] Welsch, Roy E. "Robust regression using iteratively reweighted least-squares". In: *Communications in Statistics - Theory and Methods* 6 (9 Jan. 1977), pp. 813–827. ISSN: 1532415X. DOI: 10.1080/03610927708827533.

[43] Whelan, Christopher, Harrell, Greg, and Wang, Jin. "Understanding the K-Medians Problem". In: 2015.

[44] Witten, Daniela M. "Penalized unsupervised learning with outliers". In: *Statistics and its Interface* 6 (2 2013). ISSN: 19387997. DOI: 10.4310/SII.2013.v6.n2.a5.

[45] Zhang, Cun-Hui. "NEARLY UNBIASED VARIABLE SELECTION UNDER MINIMAX CONCAVE PENALTY". In: *The Annals of Statistics* 38 (2 2010), pp. 894–942.

[46] Zhang, Yingying, Wang, Huixia Judy, and Zhu, Zhongyi. "Robust subgroup identification". In: *Statistica Sinica* 29 (4 2020), pp. 1873–1889. ISSN: 10170405. DOI: 10.5705/ss.202017.0179.

[47] Zhu, Changbo et al. “Convex optimization procedure for clustering: Theoretical revisit”. In: vol. 2. 2014.

[48] Zou, Hui and Li, Runze. “One-step sparse estimates in nonconcave penalized likelihood models”. In: *Annals of Statistics* 36 (4 2008). ISSN: 00905364. DOI: 10.1214/009053607000000802.