The thesis of Weixin Yao was read and approved\(^1\) by the following:

Bruce G. Lindsay  
Willaman Professor and Department Head of Statistics  
Thesis Co-Adviser  
Co-Chair of Committee

Runze Li  
Associate Professor of Statistics  
Thesis Co-Adviser  
Co-Chair of Committee

Bing Li  
Professor of Statistics

David Hunter  
Associate Professor of Statistics

Qiang Du  
Willaman Professor of Mathematics

\(^1\text{Signatures on file in the Graduate School.}\)
Abstract

My thesis includes two topics: modal local polynomial regression and label switching for Bayesian mixtures.

Modal Local Polynomial Regression

By combining the ideas of local polynomial regression (LPR) and modal regression, we created a new adaptive robust nonparametric regression method – “modal local polynomial regression (MLPR).” We have successfully proved that asymptotically MLPR produces smaller mean square error (MSE) than LPR when there are outliers or when the error distribution has a heavier tail than the normal distribution. Furthermore, unlike other general robust methods, this new method achieves robustness without sacrificing efficiency. In fact, in cases where there are no outliers or where the error distribution has a light tail (e.g. Gaussian distribution), MLPR produces results that are at least as good as the local polynomial method.

By adding one more tuning parameter, MLPR performs better than the traditional LPR. Specifically, suppose the bivariate data \( \{(x_i, y_i), i = 1, ..., n\} \) are independent and identically sampled from the model: \( Y = m(X) + \epsilon \), where \( E(\epsilon | X) = 0 \). Our focus is to estimate the smooth function \( m(x) \). For any given \( x_0 \), using Taylor’s expansion in the neighborhood of \( x_0 \): \( m(x) \approx \sum_{v=0}^{p} m^{(v)}(x_0)(x - x_0)^v/v! \), LPR fits the above polynomial regression locally at \( x_0 \) by minimizing the weighted least square criterion

\[
\sum_{i=1}^{n} \left[ K_h(x_i - x_0) \left( y_i - \sum_{j=0}^{p} \beta_j(x_i - x_0)^j \right)^2 \right],
\]
where \( K_h(t) = h^{-1}K(t/h), \) \( K(t) \) is a symmetric probability density function. Denote \( \hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_p) \) the solution to minimize the above formula. We estimate \( m^{(v)}(x_0) \) by \( v!\hat{\beta}_v, v = 0, \ldots, p \). Specifically, when \( v = 0 \), we estimate \( m(x_0) \) by \( \hat{\beta}_0 \).

In comparison, MLPR estimates \( \hat{\beta} \) by maximizing the objective function

\[
\sum_{i=1}^{n} \left[ K_{h_1}(x_i - x_0) \phi_{h_2}(y_i - \sum_{j=0}^{p} \beta_j(x_i - x_0)^j) \right],
\]

where \( \phi_{h_2}(t) = h_2^{-1}\phi(t/h_2), \) \( \phi(t) \) is the standard normal density function, and \( h_2 \) is a constant depending on error distribution. If \( p = 0 \), which is the modal local constant model, \( \hat{\beta}_0 \) represents the conditional mode of a kernel density estimator, conditional on \( x = x_0 \). The EM algorithm for finding the modes of mixtures (Li, Ray, and Lindsay, 2007) can be extended to find \( \hat{\beta} \). In the E Step we calculate: \( \pi(j | \beta^{(k)}) \propto K_{h_1}(x_j - x_0)\phi_{h_2}(y_j - \sum_{l=0}^{p} \beta_l(x_j - x_0)^l) \). In the M Step, we find \( \beta^{(k+1)} \) by maximizing the function \( \sum_{j=1}^{n} \left[ \pi(j | \beta^{(k)}) \log \left( \phi_{h_2}(y_j - \sum_{l=0}^{p} \beta_l(x_j - x_0)^l) \right) \right] \) with respect to \( \beta = (\beta_0, \ldots, \beta_p) \). The added robustness can be seen in the E step by the inclusion of \( \phi_{h_2}(y_j - \sum_{l=0}^{p} \beta_l(x_j - x_0)^l) \) as a weight function. As a result, any outliers are weighted less under MLPR. Notice that \( h_2 \to \infty, \pi(j | \beta^{(k)}) \propto K_{h_1}(x_j - x_0), \) then MLPR is exactly the same as LPR. Therefore, robustness is achieved without sacrificing efficiency. We have successfully found the way to select the adaptive optimal bandwidth \( h_2 \) based on the asymptotic MSE.

We can also extend MLPR to simple linear regression to create a robust linear regression – “modal linear regression (MLR)”. Instead of using least square \( \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 \), the loss function \( \sum_{i=1}^{n} \phi_h(y_i - \beta_0 - \beta_1 x_i) \) is used to estimate the regression.
parameters \( \theta = (\beta_0, \beta_1) \). Due to similar reasons stated above, we can achieve robustness without sacrificing efficiency.

**Label Switching for Bayesian Mixtures**

One of the most fundamental problems for Bayesian mixture model estimation is label switching. We mainly propose two methods to solve this problem. One solution is to use the modes of the posterior distribution to do labelling. In order to find the posterior modes, we successfully created an algorithm to find the posterior modes of Bayesian mixtures by using the ideas of ECM (Meng and Rubin, 1993) and Gibbs sampler. This labelling method creates a natural and intuitive partition of the parameter space into labelled regions and has a nice explanation based on the highest posterior region (HPD). The other main solution is to do labelling by minimizing the normal likelihood of the labelled Gibbs samples. Unlike order constraint method, this new method can be easily extended to high dimension case and is scale invariant to the component parameters. In addition, this labelling method can be also used to solve label switching in frequentist case.
# Table of Contents

List of Tables .......................................................... ix

List of Figures .......................................................... x

Acknowledgments .......................................................... xiii

## I Modal Local Polynomial Regression ................................. 1

Chapter 1. Introduction .................................................. 2

1.1 Review of Local Polynomial Regression .......................... 2

1.2 Review of Robust Local Polynomial Regression ................. 12

Chapter 2. Modal Local Polynomial Regression ....................... 17

2.1 Modal local constant regression ................................. 17

2.2 Modal Local Polynomial Regression ............................ 21

2.3 Asymptotic Properties ............................................. 24

2.4 Bandwidth Selection .............................................. 47

2.4.1 Ideal Choice of Bandwidths ................................. 47

2.4.2 Bandwidth Selection in Practice ............................ 52

2.4.3 Relative Efficiency between Modal Local Polynomial and Lo-

cal Polynomial Regression ........................................ 61

2.5 Simulation Study and Real Data Application .................... 65
Chapter 3. Modal Linear Regression

3.1 Introduction of Modal Linear Regression

3.2 Simulation Study and Real Data Application

Chapter 4. Summary and Future Work

II Label Switching for Bayesian Mixtures

Chapter 1. Introduction

1.1 Review of Mixture Models

1.2 Introduction to Label Switching for Finite Mixture Models

1.3 Review of Current Labelling Methods for Bayesian Mixtures

Chapter 2. New Labelling Methods for Bayesian Mixtures

2.1 Ideal Labelling Methods Based on Asymptotics

2.2 Introduction of Labelling Methods

2.2.1 Labelling Based on the ECM algorithm

2.2.2 Computing the HPD modal credibility $1 - \alpha^*$

2.2.3 Labelling Based on Modal Distance

2.2.4 Labelling Based on Loss Functions

2.2.4.1 The K-means method

2.2.4.2 The Determinant Based Loss

2.2.4.3 The Classification MLE Method

2.2.5 Comparing the Proposed Labelling Methods

2.3 Simulation Studies and Real Data Application
List of Tables

2.1 Relative Efficiency between MLL and LL for different error distributions. 65
2.2 Summary of MSE for LL and MLL for Example 2 with $R=500$ . . . . . 67
2.3 Summary of MSE for LL and MLL for Example 2 with $R=500$ . . . . . 69
2.4 Summary of MSE for LL and MLL for Example 3 with $R=500$ . . . . . 70
3.1 Relative Efficiency between MLR and LR for different error distributions. 96
3.2 Summary of MSE of the intercept estimates for Example 1 with $R=10000$ 98
3.3 Summary of MSE of the slope estimates for Example 1 with $R=10000$ 99
3.4 Summary of MSE of the intercept estimates for Example 2 with $R=10000$ 100
3.5 Summary of MSE of the slope estimates for Example 2 with $R=10000$ 101
3.6 Summary of MSE of the intercept estimates for Example 3 with $R=10000$ 101
3.7 Summary of MSE of the slope estimates for Example 3 with $R=10000$ 102
3.8 Summary of MSE of the intercept estimates for Example 4 with $R=10000$ 103
3.9 Summary of MSE of the slope estimates for Example 4 with $R=10000$ 104
3.10 Summary of the linear coefficient estimators. . . . . . . . . . . . . . . . 105
2.1 Sample means of labelled Gibbs samples for different labelling methods. 158
2.2 Sample means of labelled Gibbs samples for different labelling method. . 161
2.3 Sample means of labelled Gibbs samples for different labelling method. . 176
List of Figures

1.1 Plot of fitted regression for different regression methods. .............................. 15
1.2 Plot of local linear fit with outliers. ................................................................. 16

2.1 Plots of local linear fit and modal local linear fit with outliers. ......................... 72
2.2 Plot $R(h)$ vs. $h$ .............................................................................................. 73
2.3 Plot $R(h)$ vs. $h$ for small $h$ ............................................................................ 74
2.4 Plot $R(h)$ vs. $h$ for method 2 ........................................................................... 75
2.5 Plot of $R(h)$ vs. $h$ for method 1 with different bandwidths ......................... 76
2.6 Plot of $R(h)$ vs. $h$ for modified method 1 with different bandwidths .... 77
2.7 Plot of $R(h)$ vs. $h$ when the error is $t_6$. ......................................................... 78
2.8 Plot of fitted curve for Example 1 with $n=250$ ............................................... 79
2.9 MSE of the estimates for LL and MR for Example 1 when $n=250$ .......... 80
2.10 MSE of the estimates by LL and MR for Example 2 with $n=250$ .......... 81
2.11 MSE of the estimates for Example 3 with $n=250$ ........................................ 82
2.12 Plots of fitted regression curves for Education Expenditure Data. ............. 83

3.1 Regression Fit for Expenditure Data assuming linear regression. ............... 106

1.1 Histogram plot of galaxy data. The number of bins used is 30. .................. 129
1.2 Plot of Gibbs samples of component means for Galaxy data. ..................... 130
1.3 Plot of marginal posterior densities of component means based on original

Gibbs samples. ........................................................................................................ 131
2.1 Plot of the two component means of 1000 Gibbs samples for the observations with sample size of 500 generated from $0.5N(0,1) + 0.5N(1.5,1)$. 162
2.2 Plot of the two component means of 1000 Gibbs samples for the observations with sample size of 100 generated from $0.5N(0,1) + 0.5N(1.5,1)$. 163
2.3 Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 1. 164
2.4 Plots of $\mu_1 - \mu_2$ vs. $\pi_1$ for different labelling methods in example 1. The red hexagram points is the posterior modes. 165
2.5 Marginal density plot of parameters of the Gibbs samples labelled by OC and PM(ECM) for example 1. 166
2.6 Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 2. 167
2.7 Plots of $\sigma_1 - \sigma_2$ vs. $\pi_1$ for different labelling methods in example 2. 168
2.8 Plots of $\sigma_1 - \sigma_2$ vs. $\mu_1 - \mu_2$ for different labelling methods in example 2. 169
2.9 Plots of $\mu_1 - \mu_2$ vs. $\pi_1$ for different labelling methods in example 2. 170
2.10 Marginal density plots of component means for different labelling methods in example 2. 171
2.11 Marginal density plots of component standard deviation for different labelling methods in example 2. 172
2.12 Marginal density plots of component proportions for different labelling methods in example 2. 173
2.13 Marginal density plots of the parameters for HPD Gibbs samples labelled by PM(ECM) for example 2. ........................................... 174

2.14 Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 3. ........................................... 178

2.15 Plots of sigma1-sigma2 vs. pi1 for different labelling methods in example 3. .......................................................... 179

2.16 Plots of sigma1-sigma2 vs. mu1-mu2 for different labelling methods in example 3. .......................................................... 180

2.17 Plots of mu1-mu2 vs. pi1 for different labelling methods in example 3. .......................................................... 181

2.18 Marginal density plots of component means for different labelling methods in example 3. ........................................... 182

2.19 Marginal density plots of component standard deviation for different labelling methods in example 3. ........................................... 183

2.20 Marginal density plots of component proportions for different labelling methods in example 3. ........................................... 184

2.21 Marginal density plots of the parameters for HPD Gibbs samples labelled by PM(ECM) for example 3. ........................................... 185

2.22 Plot of Gibbs samples of component means labelled by PM(ECM) for Galaxy data. ........................................... 186

2.23 Plot of marginal posterior densities of component means based on the Gibbs samples labelled by PM(ECM). ........................................... 187
Acknowledgments

I wish to express my great appreciation to my advisors, Bruce G. Lindsay and Runze Li, for the large doses of guidance, patience and encouragement. I also wish to thank my other committee members, Bing Li, David Hunter, Hongyuan Zha, and Qiang Du, for their insightful commentary on my work.

Finally, a special thanks is extended to my wife, Jian Guo, her love, patience, and companionship proved invaluable.

My research was supported in part by the National Science Foundation grant DMS-0348869 and CCF-0430349 and National Institute on Drug Abuse grant NIH-P50-DA10075.
Part I

Modal Local Polynomial Regression
Chapter 1

Introduction

In this part, we will introduce a new adaptive robust nonparametric regression method “modal local polynomial regression”. Modal local polynomial regression is an improved version of local polynomial regression. It is robust to outliers and heavy tailed error distributions. Moreover, it works as well as local polynomial regression when the error is normally distributed. Since we will compare modal local polynomial regression with local polynomial regression through the whole part, in this chapter, we first review the widely used smoothing method called local polynomial regression, and explain why it is not robust to outliers and heavy tail error distribution. After that, we review currently used robust versions of local polynomial regression.

1.1 Review of Local Polynomial Regression

Local polynomial regression is a well developed nonparametric smoothing method. See Fan and Gijbels (1996), Wang and Jones (1995). For simplicity we will introduce the framework of this smoothing method in the case of one-dimensional explanatory variables. The whole theory can be easily extended to multivariate case.

Consider bivariate data \( \{(x_1, y_1), \ldots, (x_n, y_n)\} \), which are independent and identically sampled from the model

\[
Y = m(X) + \varepsilon,
\]
where $\varepsilon$ is random error with $E(\varepsilon \mid X) = 0$, $\text{Var}(\varepsilon \mid X = x) = \sigma^2(x)$ and $X$ and $\varepsilon$ are independent. Let $f(\cdot)$ be the design density of $X$. Our interest is in estimation of the regression function $m(x)$. One of the simplest and most commonly used methods is linear regression (or more generally polynomial regression). This parametric method works well if the linear assumption about $m(x)$ holds. However, if the parametric assumption does not hold, the linear regression fit will give us the misleading results. One good alternative method is nonparametric regression, which does not impose a specified form about $m(x)$.

Nonparametric regression method is widely used to estimate the regression function in an exploratory data analysis stage or when standard parametric regression, such as linear regression, doesn’t work well. There are many nonparametric regression methods such as kernel smoothing, spline smoothing, local regression, and wavelets, see Fan and Gijbels (1996), Härdle (1990), and Wand and Jones (1995). Local polynomial regression (LPR) is one of the most popular smoothing methods, due to both its simplicity of computation and nice asymptotic properties. See Fan (1992,1993) and Ruppert and Wand (1994), among others.

Suppose the $(p + 1)^{th}$ derivative of $m(x)$ at the point $x_0$ exists. Using Taylor’s expansion in the neighborhood of $x_0$:

$$m(x) \approx m(x_0) + m'(x_0)(x - x_0) + \cdots + \frac{m^{(p)}(x_0)}{p!}(x - x_0)^p \triangleq \sum_{j=0}^{p} \beta_j (x - x_0)^j, \quad (1.1)$$

where $m^{(v)}(x_0), v = 0, 1, \ldots, p$ is the $v^{th}$ derivative of $m(x)$ at point $x_0$. 
The approximate polynomial regression in a neighborhood of \( x_0 \) is then fit using a weighted least squares regression problem i.e. minimizing

\[
\sum_{i=1}^{n} \left\{ y_i - \sum_{j=0}^{p} \beta_j (x_i - x_0)^j \right\}^2 K_h(x_i - x_0),
\]

where \( K_h(t) = h^{-1}K(t/h) \), \( h \) is the bandwidth, and \( K(t) \) is a kernel function, which is generally taken as a symmetric probability density function such as uniform kernel, Gaussian Kernel, and Epanichov kernel (\( K(x) = \frac{3}{4}(1 - x^2)I(|x| < 1) \)). Denote by \( \hat{\beta}_j, j = 1, \ldots, p \), the solution to the least squares problem (1.2). The estimate of \( m^{(v)}(x_0) \) is then \( \hat{m}_v(x_0) = v!\hat{\beta}_v, v = 0, \ldots, p \).

Notice that the first part of (1.2) is just the usual least squares criterion for polynomial regression. The only difference between polynomial regression and local polynomial regression is the second part — the kernel weight function \( K_h(x_i - x_0) \). Generally, this weight function, say a Gaussian kernel, downweights the points with \( x_i \) values away from \( x_0 \) and reaches a maximum at \( x_i = x_0 \). Intuitively, data points with \( x \) values away from \( x_0 \) have less information about \( m(x_0) \) and for these points the polynomial approximation (1.1) is also less accurate. Hence, we put smaller weight for the data points with \( x \) values away from \( x_0 \).

When dealing with the bandwidth selection problem, we need to find the asymptotic bias and the variance of the estimators. Ruppert and Wand (1994) give the first order asymptotic expansion for the bias and the variance of the estimator \( \hat{m}_v(x_0) = v!\hat{\beta}_v \) from (1.2). In order to introduce their results, the following notation will be used. The
moments of $K$ and $K^2$ are denoted respectively by

$$
\mu_j = \int t^j K(t) dt \quad \text{and} \quad \nu_j = \int t^j K^2(t) dt .
$$

Some matrices and vectors of moments appear in the asymptotic expansions. Let

$$
S = (\mu_{j+l})_{0 \leq j, l \leq p} \\
\tilde{S} = (\mu_{j+l+1})_{0 \leq j, l \leq p} \\
S^* = (\nu_{j+l})_{0 \leq j, l \leq p} .
$$

Further, we consider the unit vector $e_{v+1} = (0, \ldots, 0, 1, 0, \ldots, 0)^T$, with 1 on the $(v+1)^{th}$ position. Ruppert and Wand (1994) prove the following theorem.

**Theorem 1.1.1.** Assume that $f(x_0) > 0$ and that $f(\cdot), m^{(p+1)}(\cdot)$, and $\sigma^2(\cdot)$ are continuous in a neighborhood of $x_0$. Further, assume that $h \to 0$ and $nh \to \infty$. Then the asymptotic conditional variance of $\hat{m}_v(x_0)$ is given by

$$
\text{Var}\{\hat{m}_v(x_0) \mid \mathbf{X}\} = e^T_{v+1} S^{-1} \tilde{S}^* S^{-1} e_{v+1} \frac{v!^2}{f(x_0)nh^{1+2v}} + o_p\left(\frac{1}{nh^{1+2v}}\right), \quad (1.3)
$$

where $\mid \mathbf{X}$ means conditional on the predictors.

The asymptotic conditional bias of $\hat{m}_v(x_0)$ for $p - v$ odd is given by

$$
\text{Bias}\{\hat{m}_v(x_0) \mid \mathbf{X}\} = e^T_{v+1} S^{-1} c_p \frac{v!}{(p+1)!} m^{(p+1)}(x_0)h^{p+1-v} + o_p(h^{p+1-v}). \quad (1.4)
$$
Further, for \( p - v \) even the asymptotic conditional bias of \( \hat{m}_v(x_0) \) is

\[
\text{Bias}\{\hat{m}_v(x_0) \mid X\} = e_{v+1} S^{-1} \left( \frac{v!}{(p + 2)!} \right) \left\{ m^{(p+2)}(x_0) + (p + 2)m^{(p+1)}(x_0) \frac{f'(x_0)}{f(x_0)} \right\} h^{p+2-v} + o_p(h^{p+2-v}),
\]

provided that \( f'(\cdot) \) and \( m^{(p+2)}(\cdot) \) are continuous in a neighborhood of \( x_0 \) and \( nh^3 \rightarrow \infty \).

By similar arguments, we can easily prove that the unconditional asymptotic bias and variance are the same as above (1.3) – (1.5), see Fan (1992). From the above results, we know there is a theoretical difference between the cases \( p - v \) odd and \( p - v \) even. Rupper and Wand (1994), Fan and Gijbels (1996) prove that when \( p - v \) increases consecutively from an even number to an odd number the asymptotic variance (1.3) does not change, however when it increases consecutively from an odd number to an even number the asymptotic variance increases. We know the higher order polynomial approximations result in a smaller order of the bias. It turns out clearly that polynomial fits with \( p - v \) odd outperform those with \( p - v \) even. Throughout the rest of the chapters, we take \( p - v \) odd.

When using local polynomial regression method, we need to choose the kernel function \( K(\cdot) \) and the smoothing parameter \( h \). It is well known that the choice of the kernel function is of less importance in kernel smoothing. The optimal kernel function is Epanechnikov kernel, which minimizes the asymptotically Mean Squared Error (MSE) and weighted Mean Integrated Squared Error (MISE) of the resulting local polynomial estimators for interior points, which are defined by (1.6) and (1.7). See Gasser, Müller and Mammitzsch (1985) and Granovsky and Müller (1991).
The bandwidth selection is critical. A too large bandwidth possibly under-parameterizes the regression function, causing a large modelling bias and missing to find features of the data. A too small bandwidth over-parameterizes the regression function and results in noisy estimates. When bandwidth goes to infinity, local polynomial regression becomes polynomial regression, while when bandwidth equals zero, the local polynomial regression fit interpolates all the data points. By (1.3) and (1.4), we can also know a large bandwidth $h$ results in a large bias, while a small bandwidth produces an estimator with a large variance. A good choice of bandwidth would balance the bias and variance trade-off.

There are two types of bandwidth — local bandwidth (or variable bandwidth) and global bandwidth (constant bandwidth). A theoretical optimal local (variable) bandwidth for estimating $m^{(v)}(x_0)$ is obtained by minimizing the Mean Squared Error (MSE) given by

$$[	ext{Bias}\{\hat{m}_v(x_0) | X\}]^2 + \text{Var}\{\hat{m}_v(x_0) | X\}. \quad (1.6)$$

The above MSE can be approximated by the asymptotical MSE based on (1.3) and (1.4).

The global bandwidth is obtained by minimizing the global loss. A commonly used and simple measure of global loss is the weighted Mean Integrated Squared Error (MISE)

$$\int \left( [\text{Bias}\{\hat{m}_v(x) | X\}]^2 + \text{Var}\{\hat{m}_v(x) | X\} \right) w(x)dx, \quad (1.7)$$

where $w \geq 0$ is some weight function, such as 1 or design density $f(\cdot)$. Using the asymptotic expression in (1.3) and (1.4), we can find the asymptotic optimal global
bandwidth. For more detail about the theoretical choice of the bandwidth, see Fan, Gijbels, Hu and Huang (1996) and Fan and Gijbels (1996).

For local polynomial regression, if we take $p = 1$, we will get the special case local linear regression. In practice, local linear regression is one of the mostly widely used smoothing methods. The local linear regression is fitted locally by the following weighted least squares regression

$$
\arg \min_{\beta} \sum_{i=1}^{n} \left[ y_i - \left\{ \beta_0 + \beta_1 (x_i - x_0) \right\} \right]^2 K_h(x_i - x_0), \tag{1.8}
$$

where $K_h(t) = h^{-1}K(t/h)$, $h$ is the bandwidth and $K(t)$ is a symmetric probability density function. For example, if the kernel function is uniformly distributed on $(-h, h)$, then (1.8) is simplified to

$$
\sum_{i=1}^{n} \left[ y_i - \left\{ \beta_0 + \beta_1 (x_i - x_0) \right\} \right]^2 I(|x_i - x_0| < h).
$$

Using the above formula, we are in fact fitting the simple linear regression locally for the data points with $x_i$ values within the $h$ distance from $x_0$.

The estimator $\hat{m}(x_0) = \hat{\beta}_0$ by local linear regression is, in fact, a linear function of $y_i$

$$
\hat{m}(x_0) = \sum_{i=1}^{n} w_i(x_0) y_i, \tag{1.9}
$$
where
\[
    w_i(x_0) = \frac{K_h(x_i - x_0)(S_{n,2}(x_0) - (x_i - x_0)S_{n,1}(x_0))}{S_{n,0}(x_0)S_{n,2}(x_0) - (x_i - x_0)^2 S_{n,1}(x_0)}.
\]

\[
    S_{n,j}(x_0) = \sum_{i=1}^{n} K_h(x_i - x_0)(x_i - x_0)^j.
\]

The asymptotic bias and variance of local linear estimator are

\[
    E\{\hat{m}(x_0) | X\} - m(x_0) = \mu_2 \frac{m''(x_0)h^2}{2} + o_p(h^2),
\]

(1.10)

and

\[
    \text{Var}\{\hat{m}(x_0) | X\} = \nu_0 \frac{\sigma^2(x_0)}{f(x_0)nh} + o_p(\frac{1}{nh}),
\]

(1.11)

provided that the bandwidth $h$ tends to 0 such that $nh \to \infty$, where $\mu_2 = \int_{-\infty}^{\infty} t^2 K(t)dt$, $\nu_0 = \int_{-\infty}^{\infty} K^2(t)dt$ and $f(x)$ is the marginal density of $X$. Based on the above asymptotic expansion of bias and variance, we can find the asymptotically optimal bandwidth using the similar method to the local polynomial regression. See Fan (1992) and Fan and Gijbels (1996).

**Theorem 1.1.2.** By (1.10) – (1.11), the asymptotically optimal local bandwidth, which minimizes the asymptotic MSE of $\hat{m}(x_0)$, is given by

\[
    h_{\text{opt}}(x_0) = \left( \frac{\nu_0 \sigma^2(x_0)}{\mu_2 \left( m''(x_0) \right)^2 f(x_0)} \right)^{1/5} n^{-1/5},
\]

(1.12)
the asymptotically optimal global bandwidth, which minimizes the asymptotic weighted MISE of \( \hat{m}(x) \), is given by

\[
h_{opt} = \left( \frac{\nu_0 \int \sigma^2(x)f^{-1}(x)w(x)dx}{\mu_2^2 \int \{m''(x)\}^2 w(x)dx} \right)^{1/5} n^{-1/5},
\]

(1.13)

where \( \mu_2 = \int_{-\infty}^{\infty} t^2 K(t)dt \), \( \nu_0 = \int_{-\infty}^{\infty} K^2(t)dt \), and \( w(x) \) is a weight function.

The above optimal bandwidth involves the unknown quantities such as the second derivative of the regression function, \( m''(x) \), the design density \( f(\cdot) \), and the variance function \( \sigma^2(x) \). Hence the results (1.12) and (1.13) can not be used directly. One possible solution is to replace the unknown quantities with pilot estimates. This is the plug in method. See Fan and Gijbels (1996) and Ruppert, Sheather and Wand (1995). Another method is to try several bandwidths and choose the one which makes the best plot. This is generally called “trial and error” method. We can also use cross-validation criterion to measure the overall “prediction errors”, which is defined by

\[
CV(h) = \frac{1}{n} \sum_{i=1}^{n} \{y_i - \hat{m}_{h,(-i)}(x_i)\}^2,
\]

where \( \hat{m}_{h,(-i)}(x_i) \) is the local linear regression estimator at \( x_i \) with bandwidth \( h \) without using the \( i^{th} \) observation \( (x_i, y_i) \). We choose the \( \hat{h}_{CV} \) which minimizes \( CV(h) \). The other popular methods include nearest neighbor bandwidth, Sheather and Jones’ bandwidth selector, see Fan and Gijbels (1994) and Sheather and Jones (1991), among others.
Now let us use a simulation example to see how local linear regression works.

**Example 1:** Generate data \( \{(x_i, y_i), i = 1, \ldots, 100\} \) independent and identically distributed from the model

\[
Y = 2\sin(2\pi X) + \epsilon,
\]

where \( X \sim U(0, 1) \) and \( \epsilon \sim N(0, 1) \).

Figure 1.1 is the scatter plot of the simulated data with the local linear fit (dashed curve). The bandwidth for local linear fit is chosen by the plug-in method (Ruppert, Sheather and Wand, 1995). For comparison, we also show the true regression curve (solid curve) and the linear regression fit (dash dotted line). Notice that in this example the true regression function, \( 2\sin(2\pi x) \), is not linear. From the Figure 1.1, we can see the linear regression fit does not estimate the true regression function well. Although the true regression curve is not linear globally, each small piece of the regression curve is approximately linear. Thus we can fit the linear regression locally in each small area. This is the motivation of local linear regression. From the Figure 1.1 we can see the local linear fit is very close to true regression curve. Hence the local linear regression works quite well.

If we take \( p = 0 \), we will get the local constant estimator — the Nadaraya-Watson (NW) estimator (Nadaraya, 1964; Watson, 1964). It finds \( \beta_0 \) by

\[
\hat{\beta}_0 = \arg \min_{\beta_0} \sum_{i=1}^{n} (y_i - \beta_0)^2 K_h(x_i - x_0) .
\]
The regression estimator is

$$\hat{m}(x_0) = \hat{\beta}_0 = \frac{\sum_{i=1}^{n} K_h(x_i - x_0) y_i}{\sum_{i=1}^{n} K_h(x_i - x_0)}.$$ 

The Nadaraya-Watson (NW) estimator is one of the first and simplest smoothing methods. There is some literature comparing the Nadaraya-Watson (NW) estimator with local linear regression. In summary, local linear regression is superior to the NW estimator in terms of mean squared error (MSE), minimax efficiency, design adaptiveness, and boundary effects. See Chu and Marron (1991), Fan (1992, 1993), and Hastie and Loader (1993).

1.2 Review of Robust Local Polynomial Regression

From the discussion of the last section, we know that the local polynomial regression is defined as the solution to the following weighted least squares problem: Find \( \beta = (\beta_0, \ldots, \beta_p) \) to minimize

$$\sum_{i=1}^{n} \left[ y_i - \sum_{j=0}^{p} \beta_j (x_i - x_0)^j \right]^2 K_h(x_i - x_0),$$

where \( K_h(t) = h^{-1} K(t/h), \) \( h \) is the bandwidth and \( K(t) \) is a symmetric probability density function.

The above criterion is based on the least squares principle. However, the least squares criterion is sensitive to outliers and heavy tailed error distributions. Hence, one possible drawback of local polynomial regression is lack of robustness.
Figure 1.2 is the plot of the local linear fit (dashed curve) when we added two outliers (star dots) to the data set in Figure 1.1. For comparison, we also show the original local linear fit (dash-dot) without the added outliers and the true regression curve (solid curve). The big difference of the two local linear fits with $x$ values from 0.8 to 1 in Figure 1.2 shows that the local linear regression is very sensitive to the outliers. The outliers raise the local mean curve of local linear fit and create the big difference between local linear fit and the true regression curve.

In order to solve the non-robustness problem of local polynomial regression, Fan and Gijbels (1996) and Fan and Jiang (1999) propose to “robustify” local polynomial regression by replacing the $L_2$ loss with an outlier resistant loss. They propose to find $\beta = (\beta_0, \ldots, \beta_p)$ by minimizing

$$
\sum_{i=1}^{n} K_h(x_i - x_0) \rho(y_i - \beta_0 - \beta_1(x_i - x_0)),
$$

where $\rho(\cdot)$ is an outlier resistant loss function. Generally if the first derivative of $\rho(\cdot)$, $\rho'(\cdot)$, exists and is bounded, then $\rho(\cdot)$ is an outlier resistant loss function. One of the commonly used $\rho(\cdot)$ is Huber’s $\psi$-function (Huber 1981) $\psi_c(t) = \rho'(t) = \max\{-c, \min(c, t)\}$. If $c = \infty$, the function $\rho(\cdot)$ becomes the $L_2$-loss, and as $c \to 0$, it becomes the $L_1$ loss when suitably normalized. Huber (1981) recommends using $c = 1.345$. Another possibility for $\rho(\cdot)$ is Turkey’s bisquare function $\psi_c(t) = t\{1 - (t/c)^2\}^2_+$, which weighs down the tail contribution of $t$ by a biweight function. In the parametric robustness literature the use of $c = 4.685$ is recommended. If we use $L_1$ loss function $\rho(t) = |t|$, we will get another commonly used robust regression method — median regression.
These estimators are in fact M-type estimators, a popular method to produce robust estimators. See Härdle and Gasser (1984), Tsybakov (1986), Besl et al. (1988), Fan et al. (1994), and Fan and Jiang (1999). The above robust methods will be more efficient than local polynomial regression (LPR) when there are outliers or the error distribution has a heavy tail. However, they will lose some efficiency when there are no outliers or the error distribution is normal. Thus the currently used robust local polynomial regression methods achieve robustness by sacrificing efficiency. Our proposed new robust method — “modal local polynomial”, which will be introduced in the next chapter — can achieve both robustness and efficiency.
Fig. 1.1. Plot of fitted regression for different regression methods.
Fig. 1.2. Plot of local linear fit with outliers.
Chapter 2

Modal Local Polynomial Regression

From the previous chapter, we can see that the widely used nonparametric regression method called local polynomial regression is sensitive to outliers. The existing robust versions of local polynomial regression will lose efficiency when the error distribution is normal and thus sacrifice efficiency to gain robustness. We seek a method which can achieve both robustness and efficiency. Motivated by this, we create a new adaptive robust nonparametric regression method we call “modal local polynomial regression”. In section 2.1, we introduce the motivation of our new method. In section 2.2 and 2.3, we introduce modal local polynomial regression and provide its asymptotic properties. In section 2.4, we discuss how to choose the bandwidth. In the last section 2.5, we use the simulation studies and a real data application to compare modal local polynomial regression with local polynomial regression and to assess the finite sample performance of modal local polynomial regression.

2.1 Modal local constant regression

For simplicity, we mainly use a one dimensional predictor to introduce the proposed new method — “modal local polynomial regression”. The whole idea can be easily extended to higher dimensions. To motivate this method, we first introduce the
modal local constant regression, also called modal regression (Scott, 1992). Modal regression is to use the mode of conditional density, instead of conditional mean used by general regression methods, to report the relationship between response and explanatory variables.

Given bivariate data \( \{(x_i, y_i), i = 1, ..., n\} \), which are independent and identically distributed with joint density function \( f(x, y) \). Our method is to estimate the regression function \( m(x) = E(Y \mid X = x) \) by the mode (with the highest density) of the conditional density function \( f(y \mid x) \).

Since \( f(y \mid x) = f(x, y) / f(x) \), finding the mode of \( f(y \mid x) \) is equivalent to finding the mode of \( f(x, y) \) with \( x \) fixed. We can estimate \( f(x, y) \) by the kernel density estimator, i.e.

\[
\hat{f}(x, y) = \frac{1}{n} \sum_{i=1}^{n} K_{h_1}(x_i - x)K_{h_2}(y_i - y),
\]

where \( K_h(x) = h^{-1}K(x/h) \), \((h_1, h_2)\) are the bandwidths and \( K(x) \) is the kernel function. So the problem is simplified to find the mode of (2.1) with \( x \) fixed. This method was initially proposed by Scott (1992) and called modal regression, because it uses the modes of the conditional density function to get the alternative regression estimator. Chu et al. (1998) also applied a similar method to image processing and proved its edge-preserving property. Since the mode is a robust statistic compared to the mean, any regression method based on the mode, such as modal regression, should be robust.

A general method used to find the conditional mode of (2.1) is the Newton-Raphson algorithm. A drawback of this algorithm is that it is not guaranteed to converge. Moreover the found solution can even be a local minimum. Notice that (2.1) is in fact
a mixture of distributions which can have multiple modes and saddle points. Here we propose to use the Modal EM algorithm (Li, Ray and Lindsay, 2007) to find the mode of (2.1).

Suppose we wish to find the mode of the finite mixture model

\[ f(x) = \sum_{i=1}^{m} \pi_i f_i(x), \quad (2.2) \]

where \( \sum_{i=1}^{m} \pi_i = 1 \), and \( f_i(x) \) is the density function of \( i^{th} \) component. In order to find the mode of \( f(x) \), Li, Ray, and Lindsay (2007) propose the following Modal EM algorithm: given the initial value \( x_0 \), in the \( (k+1)^{th} \) step of EM algorithm,

**E Step:** Set

\[ \pi(j \mid x^{(k)}) = \frac{\pi_j f_j(x^{(k)})}{\sum_{i=1}^{m} \pi_i f_i(x^{(k)})}. \]

**M Step:** Update \( x^{(k+1)} \) by maximizing the following objective function

\[ \sum_{j=1}^{m} \left[ \pi(j \mid x^{(k)}) \log f_j(x) \right]. \]

The E step is the “Expectation” step where the posterior probability of each mixture component \( j, 1 \leq j \leq m \), at the current point \( x^{(k)} \) is computed. The M step is the “Maximization” step where the mode of the mixture density (2.2) is updated. If \( f_j(x) \) is the Gaussian density with mean \( \mu_j \) and common covariance matrix \( \Sigma \), \( \log f_j(x) \)
is a quadratic function of $x$ and thus the update of $x^{(k+1)}$ in M step is simply

$$x^{(k+1)} = \sum_{j=1}^{m} \pi(j \mid x^{(k)}) \mu_j .$$

Notice that the objective function (2.1) is a mixture distribution. Hence the Modal EM algorithm can be applied to find its mode. Specifically, given the initial value, the $(k + 1)^{th}$ step of EM algorithm to find $m(x_0)$ is:

**E Step:**

$$\pi(j \mid m^{(k)}) = \frac{K_{h_1}(x_j - x_0)K_{h_2}(y_j - m^{(k)})}{\sum_{i=1}^{n} [K_{h_1}(x_i - x_0)K_{h_2}(y_i - m^{(k)})]} .$$

(2.3)

**M Step:** Find $m^{(k+1)}$ by maximizing the following objective function

$$\sum_{j=1}^{n} \left[ \pi(j \mid m^{(k)}) \log K_{h_2}(y_j - m) \right] ,$$

(2.4)

where $K_h(t) = h^{-1}K(t/h)$, $h$ is the bandwidth and $K(t)$ is a symmetric probability density function such as Gaussian Kernel.

**Remark 2.1.1.** When $K(x)$ is standard normal density function, the objective function (2.4) in M step can be simplified to a weighted least squares and thus the M step is equivalent to

$$m^{(k+1)} = \sum_{j=1}^{n} \pi(j \mid m^{(k)}) y_j .$$
When $h_2 \to \infty$,

$$
\pi(j \mid m^{(k)}) = \frac{K_{h_1}(x_j - x_0)K_{h_2}(y_j - m^{(k)})}{\sum_{i=1}^n [K_{h_1}(x_i - x_0)K_{h_2}(y_i - m^{(k)})]} \to \frac{K_{h_1}(x_j - x_0)}{\sum_{i=1}^n K_{h_1}(x_i - x_0)} \propto K_{h_1}(x_j - x_0)
$$

and this estimator converges to the local constant estimator (Nadaraya-Watson estimator). Hence, local constant estimator is a limiting case of modal local constant. Notice that modal regression only uses one term (the intercept) for the conditional mode, like the local constant estimator. Hence, here we also call this modal regression “modal local constant regression”.

In E step, in fact, we need not normalize $\pi(j \mid m^{(k)})$. We can simply replace (2.3) by

$$
\pi(j \mid m^{(k)}) = K_{h_1}(x_j - x_0)K_{h_2}(y_j - m^{(k)}).
$$

### 2.2 Modal Local Polynomial Regression

It is known that the local linear estimator is superior to the local constant one, as discussed in the first chapter. So we may want to extend the idea of modal local constant to a local linear case, or more generally to local polynomial case. That is the motivation of our new proposed method we call “modal local polynomial regression”.

Suppose that the $(p + 1)^{th}$ derivative of $m(x)$ at the point $x_0$ exists. We can approximate the regression function $m(x)$ locally by a polynomial of order $p$. By Taylor expansion, for $x$ in a neighborhood of $x_0$,

$$
m(x) \approx m(x_0) + m'(x_0)(x - x_0) + \cdots + \frac{m^{(p)}(x_0)}{p!}(x - x_0)^p \Delta \sum_{j=0}^p \beta_j(x - x_0)^j, \quad (2.5)
$$
where \( \beta_v = m^{(v)}(x_0)/v!, v = 0, \ldots, p. \)

The modal local polynomial regression is to find \( \beta = (\beta_1, \ldots, \beta_p) \) by maximizing the following objective function

\[
\sum_{i=1}^{n} \left[ K_h(x_i - x_0) \phi_h \left( y_i - \sum_{j=0}^{p} \beta_j (x_i - x_0)^j \right) \right],
\]

where \( \phi_h(t) = h^{-1} \phi(t/h) \) and \( \phi(t) \) is the standard normal density function. Denote \( \hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_p) \) the solution to the above formula. We estimate \( m^{(v)}(x_0) \) by \( \hat{m}_v(x_0) = v! \hat{\beta}_v, v = 0, \ldots, p. \) Specifically, we estimate \( m(x_0) \) by \( \hat{\beta}_0. \)

Here \( -\phi(\cdot) \) plays the role of the loss function \( \rho(\cdot) \) of (1.14). So the minimization is changed to maximization in the objective function (2.6). Notice that our method is in fact also a M-type estimator and the first derivative of \( \phi(\cdot) \) is bounded. This explains why modal local polynomial regression is robust. One reason why we use the Gaussian density is for the simplicity of the calculation and the easy comparison between modal local polynomial regression and local polynomial regression. We will explain those in the following several chapters. It also offers a modal interpretation to the estimator, like modal regression (Scott, 1992), when the errors are not symmetric or there are multiple modes.

Notice that there is no explicit solution \( \beta = (\beta_1, \ldots, \beta_p) \) to the objective function (2.6). We propose to extend the idea of the Modal EM algorithm (Li, Ray and Lindsay, 2007) to find the mode of (2.6). From the EM algorithm theory, we know the Modal EM algorithm (Li, Ray and Lindsay, 2007) can be also applied to the situation when \( f_i(x) \) of (2.2) is any nonnegative integrable function, not necessarily the density function.
Noticing that \( \phi_2(y_i - \sum_{j=0}^{p} \beta_j(x_i - x_0)^j) \) of (2.6) is an integrable function of \( \beta = (\beta_1, \ldots, \beta_p) \), so we can extend the idea of the Modal EM algorithm (Li, Ray and Lindsay, 2007) to find the maximum of (2.6).

The \((k+1)\)th step of the EM algorithm to find \( \beta = (\beta_1, \ldots, \beta_p) \), given the initial value, is:

**E Step:** Set

\[
\pi(j \mid \beta^{(k)}) = \frac{K_{h_1}(x_j - x_0) \phi_{h_2} \left( y_j - \sum_{l=0}^{p} \beta_l^{(k)}(x_j - x_0)^l \right)}{\sum_{i=1}^{n} \left[ K_{h_1}(x_i - x_0) \phi_{h_2} \left( y_i - \sum_{l=0}^{p} \beta_l^{(k)}(x_i - x_0)^l \right) \right]}
\]

**M Step:** Find \( \beta^{(k+1)} \) by maximizing the following objective function

\[
\sum_{j=1}^{n} \left[ \pi(j \mid \beta^{(k)}) \log \phi_{h_2} \left( y_j - \sum_{l=0}^{p} \beta_l(x_j - x_0)^l \right) \right]
\]

where \( K_h(t) = h^{-1}K(t/h) \), \( h \) is the bandwidth and \( K(t) \) is a symmetric probability density function such as Gaussian Kernel.

** Remark 2.2.1.** Since \( \phi(x) \) is standard normal density function, M step is simplified to

\[
\beta^{(k+1)} = \arg \min_{\beta} \sum_{j=1}^{n} \left[ \pi(j \mid \beta^{(k)}) \left( y_j - \sum_{l=0}^{p} \beta_l(x_j - x_0)^l \right)^2 \right]
\]

Hence

\[
\beta^{(k+1)} = (X^T W_k X)^{-1} X^T W_k Y
\]
where \( X = (x_1^*, \ldots, x_n^*)^T \) with \( x_i^* = (1, x_i - x_0)^T \), \( W_k = \text{diag}(\pi(j \mid \beta^{(k)})) \) and \( Y = (y_1, \ldots, y_n)^T \).

When \( h_2 \to \infty \), from the E step we can know

\[
\pi(j \mid \beta^{(k)}) \to \frac{K_{h_1}(x_j - x_0)}{\sum_{i=1}^n K_{h_1}(x_i - x_0)} \propto K_{h_1}(x_j - x_0)
\]

and thus this new estimator converges to the local polynomial estimator. So local polynomial regression is a limiting case of modal local polynomial regression.

In E step, in fact, we need not normalize \( \pi(j \mid \beta^{(k)}) \). We can replace the E step by

\[
\pi(j \mid \beta^{(k)}) = K_{h_1}(x_j - x_0) \phi_{h_2} \left( y_j - \sum_{l=0}^p \beta_l^{(k)}(x_j - x_0)^l \right).
\]

From this algorithm, we can see the major difference between local polynomial and modal local polynomial lies in E step. For local polynomial, the weight of contribution of each observation, \( K_h(x_i - x_0) \), only depends on how close \( x_i \) is to \( x_0 \). For modal local polynomial, the weight also depends on how close \( y_i \) is to the regression curve. Generally, modal local polynomial downweights the observations away from the regression curve. That is how modal local polynomial regression achieves robustness.

### 2.3 Asymptotic Properties

In this section, we will derive some asymptotic properties of the modal local polynomial estimator. Denote the marginal density of \( X \), also called design density, by \( f(\cdot) \) and the density of \( \epsilon \) by \( g(t) \). For any given point \( x_0 \), the following notation and
assumptions are needed. The moments of $K$ and $K^2$ are denoted respectively by

$$
\mu_j = \int t^j K(t) dt \quad \text{and} \quad \nu_j = \int t^j K^2(t) dt.
$$

Let

$$
S = (\mu_{j+l})_{0 \leq j, l \leq p} \quad \text{and} \quad \epsilon_p = (\mu_{p+1}, \ldots, \mu_{2p+1})^T
$$

$$
\tilde{S} = (\mu_{j+l+1})_{0 \leq j, l \leq p} \quad \text{and} \quad \tilde{\epsilon}_p = (\mu_{p+2}, \ldots, \mu_{2p+2})^T
$$

$$
S^* = (\nu_{j+l})_{0 \leq j, l \leq p}.
$$

Further, we consider the unit vector $e_{v+1} = (0, \ldots, 0, 1, 0, \ldots, 0)^T$, with 1 on the $(v+1)^{th}$ position.

**Assumption 2.3.1.**

1. $m(x)$ has bounded $(p+1)^{th}$ derivative in a neighborhood of $x_0$.
2. $f(x)$ has continuous and bounded second derivative in a neighborhood of $x_0$.
3. The probability density function $g(\cdot)$ of $\epsilon$ is supported on $\mathbb{R}$, symmetric about 0.
4. $g(\cdot)$ has bounded second derivative.
5. $K(\cdot)$ is a symmetric probability density with compact support.
6. $E[\phi''(\epsilon/h)] < 0$ for any $h > 0$, where the error term $\epsilon$ has the density $g(t)$.

For simplicity of the proof, here we assume $K(\cdot)$ has compact support. In fact we only need to put some conditions on the tail of $K(\cdot)$. The other conditions may also be relaxed.
However the conditions here are still not strict and can hold for general situations. If $g'(t) > 0$ when $t < 0$ and $g'(t) < 0$ when $t > 0$, then the last assumption $E[\phi''(\epsilon/h)] < 0$ holds. To see this, notice that

$$E[\phi''(\epsilon/h)] = \int_{-\infty}^{\infty} \phi''(t/h)g(t)dt$$

$$= h \int_{-\infty}^{\infty} \phi''(t)g(th)dt$$

$$= h\phi'(t)g(th) \bigg|_{-\infty}^{\infty} - h \int_{-\infty}^{\infty} \phi'(t)g'(th)dt$$

$$= h^2 \int_{-\infty}^{\infty} t\phi(t)g'(th)dt < 0. \quad (2.7)$$

Now let us establish some asymptotic properties for the modal local polynomial estimator.

**Theorem 2.3.1.** Under the Assumption 2.3.1, if $h_1 \to 0$, $nh_1 \to \infty$, and $h_2$ does not depend on $n$, with probability approaching to 1 (wpa1), there exists a consistent local maximizer $\hat{\beta} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)$ of (2.6), such that $h_1^v \left( \hat{m}_v(x_0) - m^{(v)}(x_0) \right) = O_p \left( (nh_1)^{-1/2} + h_1^{p+1} \right)$, where $\hat{m}_v(x_0) = v!\hat{\beta}_v$ is the estimator of $m^{(v)}(x_0)$ and $m^{(v)}(x)$ is $v^{th}$ derivative of $m(x)$ at $x_0$.

**Proof:** Denote $H = \text{diag}\{1, h_1, \ldots, h_1^p\}$, $\theta = H\beta$, and $Z_i = H^{-1}X_i^{*}$, where $X_i^{*} = [1, X_i - x_0, \ldots, (X_i - x_0)^p]^T$ and $\beta = (\beta_0, \ldots, \beta_p)$. Let $\theta_0 = \left( m(x_0), \ldots, h_1^p m^{(p)}(x_0)/p! \right)^T$ be the true value for $\theta$. If $\hat{\beta}$ is a local maximizer of (2.6), $\hat{\theta} = H\hat{\beta}$ will be a local maximizer of

$$l_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} \left[ K_{h_1}(X_i - x_0)\phi_{h_2}(Y_i - \theta^T Z_i) \right].$$
Denote

\[ W_n \triangleq \frac{\partial l_n(\theta_0)}{\partial \theta} = -\frac{1}{n h_2^2} \sum_{i=1}^{n} K_{h_1}(X_i - x_0) \phi'(\frac{Y_i - \theta_0^T Z_i}{h_2}) Z_i \]

\[ \Delta_n \triangleq \frac{\partial^2 l_n(\theta_0)}{\partial \theta \partial \theta^T} = \frac{1}{n h_2^3} \sum_{i=1}^{n} K_{h_1}(X_i - x_0) \phi^{(2)}(\frac{Y_i - \theta_0^T Z_i}{h_2}) Z_i Z_i^T \]

where \( \phi^j(t) \) is the \( j^{th} \) derivative of \( \phi(t) \) with \( \phi'(t) = (-t) \phi(t), \phi''(t) = (t^2 - 1) \phi(t) \) and \( \phi^{(3)}(t) = (3t - t^3) \phi(t) \).

Let

\[ S_{j,k} = K_{h_1}(X - x_0)(\frac{X - x_0}{h_1})^j (\frac{Y - \theta_0^T Z}{h_2})^k \phi_{h_2}(Y - \theta_0^T Z). \]

If \( k \) is even, noticing that \( g'(\cdot) \) and \( f''(\cdot) \) are bounded and the conditional density of \( Y \) given \( X = x \) is \( g(y - m(x)) \), we have

\[
E(S_{j,k}) = \int K(h_1) \frac{(X - x_0)}{h_1} \left( \int \frac{1}{h_2} (\frac{Y - \theta_0^T Z}{h_2})^k \phi(\frac{Y - \theta_0^T Z}{h_2}) g(y - m(x)) dy \right) f(x) dx
\]

\[
= \int K(s) s^j \left[ \int t^k \phi(t) g(th_2) + \sum_{j=0}^{p} \frac{m^{(j)}(x_0)}{j!} (sh_1)^j (m(x_0 + sh_1)) dt \right] f(x_0 + sh_1) ds
\]

\[
= \int K(s) s^j \left[ \int t^k \phi(t) \left( g(th_2) + O(h_1^{p+1}) s^{p+1} \right) dt \right] f(x_0 + sh_1) ds
\]

\[
= \int K(s) s^j \left[ \int t^k \phi(t) g(th_2) dt + O(h_1^{p+1}) s^{p+1} \right] \left( f(x_0) + f'(x_0) sh_1 + O(h_1^2) s^2 \right) ds
\]

\[
= \int t^k \phi(t) g(th_2) dt f(x_0) \mu_j + f'(x_0) \mu_{j+1} h_1 + O(h_1^2) \quad (2.8)
\]
where $\mu_j = \int t^j K(t) dt$.

If $k$ is odd, $\int t^k \phi(t) g(th_2) dt = 0$. By noticing that $g''(\cdot)$ is bounded, we have

$$E(S_{j,k}) = \int K(s) s^j \left\{ \int t^k \phi(t) \left[ g(th_2) + g'(th_2) \left( \beta_{p+1}(sh_1)^{p+1} + \beta_{p+2}(sh_1)^{p+2} + O(h_1^{p+3}) s^{p+3} \right) \right.ight.
$$

$$+ O(h_1^{2p+2}) s^{2p+2} \left\} dt \right\} f(x_0 + sh_1) ds
$$

$$= \int K(s) s^j \left\{ \int t^k \phi(t) g'(th_2) \left( \beta_{p+1}(sh_1)^{p+1} + \beta_{p+2}(sh_1)^{p+2} \right) dt \right\}
$$

$$\times \left( f(x_0) + f'(x_0) sh_1 + O(h_1^2) s^2 \right) ds + O(h_1^{p+3})
$$

$$= H_k(h_2) h_1^{p+1} \left[ \beta_{p+1} f(x_0) \mu_{p+1+j} + h_1 \left( \beta_{p+1} f'(x_0) + \beta_{p+2} f(x_0) \right) \mu_{p+2+j} \right] + O(h_1^{p+3}),
$$

(2.9)

where $\beta_j = m^{(j)}(x_0)/j!$, $H_k(h) = \int t^k \phi(t) g'(th) dt$ and $g'(t)$ is the first derivative of $g(t)$.

Let

$$T_{j,k} = K_{h_1} (X - x_0) \left( \frac{X - x_0}{h_1} \right)^j \left( \frac{Y - \theta_0^T Z}{h_2} \right)^{2k} \phi_{h_2} (Y - a_n \theta_0^T Z).$$
\[ E(T_{j,k}) = E \left[ K_h^2 (X - x_0) \left( \frac{X - x_0}{h_1} \right)^j \left( Y - \theta_0^T Z \right)^{2k} \phi_{h_2}^2 (Y - a_n \theta_0^T Z) \right] \]

\[ = \frac{1}{h_1 h_2^2} \int K^2 \left( \frac{x - x_0}{h_1} \right) \left( \frac{x - x_0}{h_1} \right)^j \]

\[ \times \left[ \int \left( \frac{y - \theta_0^T Z}{h_2} \right)^{2k} \phi^2 \left( \frac{y - \theta_0^T Z}{h_2} \right) g(y - m(x))dy \right] f(x)dx \]

\[ = \frac{1}{h_1 h_2} \int K^2 (s) s^j \left[ \int t^{2k} \phi^2 (t) g(th_2 + \sum_{j=0}^{p} m(j)(x_0) (sh_1)^j - m(x_0 + sh_1))dt \right] f(x_0 + sh_1)ds \]

\[ = \frac{1}{h_1 h_2} \left( f(x_0) \nu_j \int t^{2k} \phi^2 (t) g(th_2)dt \right) \left( 1 + O(h_1) \right) , \quad (2.10) \]

where \( \nu_j = \int t^j K^2 (t)dt. \)

Based on (2.8), we have

\[ E(\Delta_n) = E_{h_2} \left( S_{i+j,2} - S_{i+j,0} \right)_{0 \leq i,j \leq p} = h_2^{-2} F(h_2)(f(x_0)S + f'(x_0)\tilde{S}h_1) + O(h_1^2) . \]

where \( S = (\mu_{i+j})_{0 \leq i,j \leq p}, \quad F(h_2) = \int (t^2 - 1) \phi(t)g(th)dt = \int \phi''(t)g(th)dt, \) and \( \tilde{S} = (\mu_{i+j+1})_{0 \leq i,j \leq p} . \)

From (2.10), we know \( E(T_{j,k}) = O(h_1^{-1}). \) By noticing that

\[ E(S_{j,k}^2) = E(T_{2j,k}) = O \left( h_1^{-1} \right), \]

we have

\[ \text{Var}\{\{\Delta_n\}_{i,j}\} = n^{-1} h_2^{-4} \text{Var}\{S_{i+j,2} - S_{i+j,0}\} \leq O \left( 2n^{-1} E[S_{i+j,2}^2 + S_{i+j,0}^2] \right) = O((nh_1)^{-1}) . \]
Based on the result $X_n = E(X_n) + O_p \left( \sqrt{\text{Var}(X_n)} \right)$, we have

$$
\Delta_n = h_2^{-2} F(h_2) (f(x_0)S + f'(x_0)\hat{S}h_1) + O_p(b_n),
$$

(2.11)

where $b_n = h_1^2 + 1/\sqrt{n}h_1$. Based on the Assumption 2.11, we know $E[\phi''(\epsilon/h_2)] < 0$.

Thus,

$$
F(h_2) = \int \phi''(t)g(th_2)dt = h_2^{-1} \phi''(t/h_2)g(t)dt = h_2^{-1}E[\phi''(\epsilon/h_2)] < 0.
$$

(2.12)

From (2.9), we have

$$
E(W_n) = \frac{1}{h_2} E(S_{i,1})_{0 \leq i \leq p}
$$

$$
= \frac{1}{h_2} H_1(h_2) h_1^{p+1} \left[ (\beta_{p+1}f(x_0)c_p + h_1(\beta_{p+1}f'(x_0) + \beta_{p+2}f(x_0))\tilde{c}_p) \right] \left[ 1 + O(h_1) \right],
$$

(2.13)

where $c_p = (\mu_{p+1+j})_{0 \leq j \leq p}$ and $\tilde{c}_p = (\mu_{p+2+j})_{0 \leq j \leq p}$.

Also

$$
\text{Cov}(W_n) = \frac{1}{nh_2^2} \text{Cov} \left[ K_{h_2} (X_i - x_0) \phi' \left( \frac{Y_i - a_n^T Z_i}{h_2} \right) Z_i \right]
$$

$$
= \frac{1}{nh_2^2} \left[ E(T_{i+j,1})_{0 \leq i,j \leq p} + (E(S_{i,1})_{0 \leq i \leq p}) (E(S_{i,1})_{0 \leq i \leq p})^T \right]
$$

$$
= \frac{1}{nh_1h_2} f(x_0)G(h_2)S^* (1 + o(1)),
$$

(2.14)

where $G(h) = \int t^2 \phi^2(t)g(th)dt$ and $S^* = (\nu_{i+j})_{0 \leq i,j \leq p}$. 
Denote \( \alpha_n = (nh_1)^{-1/2} + h_1^{p+1} \), then

\[
l(\theta_0 + \alpha_n \mu) - l(\theta_0) = \alpha_n W_n^T \mu + \frac{1}{2} \alpha_n^2 \mu^T \Delta_n \mu - \frac{\alpha_n^3}{6nh_2^2} \sum_{i=1}^{n} K_{h_1}(x_i - x_0) \phi(3) \left( \frac{y_i - \theta^* \mu}{h_2} \right) (z_i^T \mu)^3
\]

\[
= A + B + C ,
\]

where \( ||\mu|| = c \), which will be defined later, and \( ||\theta^* - \theta_0|| \leq c \alpha_n \).

From (2.13) and (2.14), we can get \( W_n = O_p \left( h_1^{p+1} + (nh_1)^{-1/2} \right) = O_p(\alpha_n) \) and hence \( A = O_p(\alpha_n^2) \). Based on (2.11),

\[
B = \frac{1}{2h_2^2} F(h_2) f(x_0) \alpha_n^2 \mu^T S \mu (1 + o_p(1)).
\]

Since \( \phi(3)(\cdot) \) is bounded and \( K(\cdot) \) has compact support. Thus

\[
C = \alpha_n^3 O_p \left( E[K_{h_1}(X - x_0)] \right) = O_p(\alpha_n^3).
\]

We can choose \( c \) big enough, such that the second term \( B \) dominates the other two terms in (2.15) wpa1 So \( l(\theta_0 + \alpha_n \mu) - l(\theta_0) < 0 \) wpa1. with \( ||u|| = c \). Hence wpa1 there exists a local maximizer \( \hat{\theta} \) such that \( ||\hat{\theta} - \theta_0|| \leq \alpha_n c = O_p(\alpha_n) \), where \( \alpha_n = (nh_1)^{-1/2} + h_1^{p+1} \).

We can also get, wpa1 \( \left| h_1^v \left( \hat{m}_v(x_0) - m^{(v)}(x_0) \right) \right| = O_p \left( (nh_1)^{-1/2} + h_1^{p+1} \right) \). □

Next we provide the asymptotic expansion of the bias and variance of the consistent estimator \( \hat{m}_v(x_0) \) given in Theorem 2.3.1.

**Theorem 2.3.2.** Under the same assumption as in Theorem 2.3.1, if \( h_1 \to 0, nh_1 \to \infty \), and \( h_2 \) does not depend on \( n \), the asymptotic variance of \( \hat{m}_v(x_0) \), found in Theorem 2.3.1,
is given by

\[
\text{Var}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} v S^{-1} e_{v+1} + \frac{h_2 v^2}{f(x_0) n h_1^{v+2}} G(h_2) F(h_2)^{-2} + o\left(\frac{1}{nh_1^{v+2}}\right). \tag{2.16}
\]

where \(F(h) = \int (t^2 - 1) \phi(t) g(th) dt\) and \(G(h) = \int t^2 \phi^2(t) g(th) dt\).

The asymptotic bias for \(p - v\) odd is given by

\[
\text{Bias}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} T^{p+1} m_{p+1}^{(p+1)}(x_0) - m_{p+1}^{(p+1)}(x_0) + o(h_1^{p+1-v}) \tag{2.17}
\]

Further, for \(p - v\) even the asymptotic bias is

\[
\text{Bias}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} T^{p+1} m_{p+1}^{(p+1)}(x_0) \left( T^{p+2} + o(h_1^{p+2-v}) \right) \tag{2.18}
\]

provided that \(h_1^{v} \left( \hat{m}_v(x_0) - m^{(v)}(x_0) \right) = o_p(h), m^{(p+2)}(\cdot)\) are continuous in a neighborhood of \(x_0\), and \(nh_1^{3} \to \infty\).

Proof: Denote \(H = \text{diag}\{1, h_1, \ldots, h_p\}, \theta = H \beta\), and \(Z_i = H^{-1} X^*, \text{where } X^* = [1, X_i - x_0, \ldots, (X_i - x_0)^p]^T\) and \(\beta = (\beta_0, \ldots, \beta_p)\). Let \(\theta_0 = \left( m(x_0), \ldots, h_1^p m(x_0)/p! \right)^T\) be the true value for \(\theta\). Since \(\hat{\beta}\) is a local maximizer of (2.6), \(\hat{\theta} = H \hat{\beta}\) is a local maximizer of

\[
l_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} \left[ K_{h_1}(X_i - x_0) \phi_{h_2}(Y_i - \theta^T Z_i) \right].
\]
Denote

\[ W_n \triangleq \frac{\partial l_n(\theta_0)}{\partial \theta} = -\frac{1}{nh^2} \sum_{i=1}^{n} K_{h_1}(X_i - x_0)\phi'(\frac{Y_i - \theta_0^T Z_i}{h_2})Z_i \]

\[ \Delta_n \triangleq \frac{\partial^2 l_n(\theta_0)}{\partial \theta^2} = \frac{1}{nh^3} \sum_{i=1}^{n} K_{h_1}(X_i - x_0)\phi''(\frac{Y_i - a_n\theta_0^T Z_i}{h_2})Z_iZ_i^T \]

where \( \phi^j(t) \) is the \( j \)th derivative of \( \phi(t) \) with \( \phi'(t) = (-t)\phi(t), \phi''(t) = (t^2 - 1)\phi(t) \) and \( \phi^{(3)}(t) = (3t - t^3)\phi(t) \).

From (2.11), (2.13) and (2.14), we know

\[ E(W_n) = \frac{1}{h_2} H_1(h_2)h_1^{p+1} \left[ (\beta_{p+1} f(x_0) c_p + h_1(\beta_{p+1} f'(x_0) + \beta_{p+2} f(x_0))\tilde{c}_p) \right] \left[ 1 + O(h_1) \right], \]

\[ \text{Cov}(W_n) = \frac{1}{nh_1 h_2^3} f(x_0) G(h_2)S^*(1 + o(1)), \]

\[ \Delta_n = h_2^{-2} F(h_2)(f(x_0)S + f'(x_0)\tilde{S} h_1) + o(h_1) \quad (2.19) \]

Since \( \hat{\theta} \) is the consistent local maximizer of \( l_n(\theta), \partial l_n(\hat{\theta})/\partial \theta = 0 \). By the Taylor expansion, we have

\[ 0 = \frac{\partial l_n(\hat{\theta})}{\partial \theta} = W_n + (\Delta_n + J_n)(\hat{\theta} - \theta_0), \quad (2.20) \]

where

\[ J_n = -\frac{1}{2nh^2} \sum_{i=1}^{n} \left[ K_{h_1}(X_i - x_0)\phi^{(3)}(\frac{Y_i - a_n\theta_0^T Z_i}{h_2})((\hat{\theta} - \theta_0)^T Z_i)Z_iZ_i^T \right] \]
where \( \| \theta^* - \theta_0 \| \leq \| \hat{\theta} - \theta_0 \| \). Since \( \phi^{(3)}(\cdot) \) is bounded, \( K(\cdot) \) has compact support, and
\[
\| \hat{\theta} - \theta \| = O_p(c_n) \text{ with } c_n = (nh_1)^{-1/2} + h_1^{p+1},
\]
we have
\[
J_n \leq O(\| \hat{\theta} - \theta \| n^{-1} \sum_{i=1}^{n} K(X_i - x_0) = O(c_n K_n),
\]
where \( K_n = n^{-1} \sum_{i=1}^{n} K_h(X_i - x_0) \). Notice that
\[
E(|K_n|) = h_1^{-1} \int K\{ (x - x_0)/h_1 \} f(x) dx
\]
\[
= \int K(s) f(x_0 + sh_1) ds
\]
\[
= \int K(s) f(x_0) ds + O(h_1) < \infty.
\]
Thus \( K_n = E(K_n) + o_p(1) = O(1) \). Hence
\[
J_n = O(c_n K_n) = O_p(c_n).
\]
Note that if \( A_n^{-1} = O(1) \) and \( B_n = o_p(1) \), we have
\[
(A_n + B_n)^{-1} = A_n^{-1} - A_n^{-1} B_n A_n^{-1} (1 + o_p(1)).
\]
(2.21)
From (2.11), (2.20), and (2.21),

\[
\hat{\theta} - \theta_0 = -(\Delta_n + J_n)^{-1} W_n \\
= - \left[ \frac{1}{h_2^2} F(h_2) f(x_0) S + o_p(1) \right]^{-1} W_n \\
= - \left[ \frac{1}{h_2^2} F(h_2) f(x_0) S \right]^{-1} W_n \left[ 1 + o_p(1) \right]. \tag{2.22}
\]

Based on the results (2.22) and (2.19), we can get that the asymptotic expansion for the variance term of \( \hat{\theta} \) is

\[
\text{Cov}(\hat{\theta}) = \left[ \frac{1}{h_2^2} F(h_2) f(x_0) S \right]^{-1} \left[ \frac{1}{nh_1 h_2^3} f(x_0) G(h_2) S^{-1} S^{-1} + o(1) \right] \left[ \frac{1}{h_2^2} F(h_2) f(x_0) S \right]^{-1} + o(1)
\]

\[
= \frac{h_2}{f(x_0)} G(h_2) F(h_2)^{-2} S^{-1} S^{-1} + o(1).
\]

From equation (2.22) and results (2.19), we have

\[
\text{Bias}(\hat{\theta}) = - \left[ \frac{1}{h_2^2} F(h_2) f(x_0) S \right]^{-1} \times \left[ \frac{1}{h_2} H_1(h_2) h_1^{p+1} \beta p+1 f(x_0) c_p \right] \left( 1 + o(1) \right)
\]

\[
= - h_2 F(h_2)^{-1} H_1(h_2) \beta p+1 h_1^{p+1} S^{-1} c_p (1 + o(1)).
\]

From (2.7), we know

\[
H_1(h_2) = \int t \phi(t) g'(th_2) dt = h_2^{-1} \int \phi''(t) g(th_2) dt = h_2^{-1} F(h_2). \tag{2.23}
\]

So

\[
\text{Bias}(\hat{\theta}) = - h_2 \beta p+1 h_1^{p+1} S^{-1} c_p (1 + o(1)).
\]
Notice that the \((v + 1)^{th}\) element of \(S^{-1}c_p\) is zero for \(p - v\) even. So we need higher order expansion for \(p - v\) even. From (2.11), (2.20), and (2.21), if \(nh_1^3 \to \infty\), we have \(c_n = o(h_1)\) and thus

\[
\hat{\theta} - \theta_0 = -(\Delta_n + J_n)^{-1} W_n
\]

\[
= -\left\{ \frac{1}{h_2^2} F(h_2) \left( f(x_0)S + f'(x_0)\tilde{S}h_1 \right) + o_p(h_1) \right\}^{-1} W_n
\]

\[
= -\left\{ \frac{1}{h_2^2} F(h_2)(f(x_0)S + f'(x_0)\tilde{S}h_1) \right\}^{-1} W_n (1 + o_p(h_1)) . \tag{2.24}
\]

From equations (2.13), (2.24) and (2.23), we have

\[
\text{Bias}(\hat{\theta}) = -\left\{ \frac{1}{h_2^2} F(h_2)(f(x_0)S + f'(x_0)\tilde{S}h_1) \right\}^{-1}
\]

\[
\times \left\{ \frac{1}{h_2} H_1(h_2) h_1^{p+1} \left[ \beta_{p+1}f(x_0)c_p + h_1 \left( \beta_{p+1}f'(x_0) + \beta_{p+2}f(x_0) \right) \tilde{c}_p \right] \right\} (1 + o(1))
\]

\[
= -h_2 h_1^{p+1} H_1(h_2) F(h_2)^{-1} \{ \beta_{p+1}S^{-1}c_p + h_1 b^*(x_0) \} + o(h_1^{p+2}) \tag{2.25}
\]

\[
= -h_1^{p+1} \{ \beta_{p+1}S^{-1}c_p + h_1 b^*(x_0) \} + o(h_1^{p+2}), \tag{2.26}
\]

where

\[
b^*(x_0) = \frac{\beta_{p+1}f'(x_0) + \beta_{p+2}f(x_0)}{f(x_0)} S^{-1} \tilde{c}_p - \frac{f'(x_0)}{f(x_0)^3} \beta_{p+1}S^{-1} \tilde{S}S^{-1}c_p .
\]

Noticing that \(\hat{\theta}_v = h_1^v \hat{m}(v)(x_0)/v!\), we obtain the results of (2.16), (2.17), and (2.18). □

Our method has a very strong tie to the methods proposed by Rue et al. (2002) and Hwang (2004). Rue et al. (2002) proposes the local linear M-smoother by extending
the local constant M-smoother (Chu et al. 1998). Hwang (2004) extends the local constant and local linear M-smoother methods to local polynomial M-smoothers. Rue et al. (2002) and Hwang (2004) assume that the bandwidth \( h_2 \) depends on the sample size \( n \) and tends to 0 as \( n \) goes to infinity. The choice of the bandwidth \( h_2 \) in their papers does not depend on the error distribution. Thus their methods are not adaptive to different error distributions and will lose efficiency when the error distribution is normal. Here, we assume the bandwidth \( h_2 \) is a constant, which does not depend on sample size \( n \). Later we will see the asymptotic optimal bandwidth \( h_2 \) only depends on the error distribution. This is one of the major advantage of our methods. The tuning parameter \( h_2 \) makes our method adaptive to different error distributions.

Under the assumption in Theorem 2.3.2, from Theorem 1.1.1 we know that for local polynomial regression the asymptotic variance of \( \hat{m}_v(x_0) \) is given by

\[
\text{Var}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} S^* e_{v+1} \frac{\sigma^2 v!^2}{f(x_0) n h^{1+2v}} + o\left(\frac{1}{n h^{1+2v}}\right),
\]

the asymptotic bias for \( p - v \) odd is given by

\[
\text{Bias}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} \tilde{c}_p \frac{v!}{(p+1)!} m^{(p+1)}(x_0) h^{p+1-v} + o(h^{p+1-v}),
\]

and for \( p - v \) even the asymptotic bias is

\[
\text{Bias}\{\hat{m}_v(x_0)\} = e_{v+1}^T S^{-1} \tilde{c}_p \frac{v!}{(p+2)!} \left\{ m^{(p+2)}(x_0) + (p+2) m^{(p+1)}(x_0) \frac{f'(x_0)}{f(x_0)} \right\} h^{p+2-v} + o(h^{p+2-v}).
\]
From (2.16) – (2.18), we can easily see that the modal local polynomial estimator has the same asymptotic bias as the local polynomial estimator. The ratio of the asymptotic variance of the modal local polynomial estimator to that of the local polynomial estimator is $R(h_2) \triangleq h_2^2 G(h_2)F(h_2)^{-2} \sigma^{-2}$. If there exists some $h_2$ such that $h_2^2 G(h_2)F(h_2)^{-2} \sigma^{-2} < 1$, the modal local polynomial will have smaller asymptotic MSE than local polynomial regression. Hence $R(h_2)$ determines the relative efficiency between modal local polynomial estimator and local polynomial estimator. The next theorem gives us the properties of $R(h_2)$.

**Theorem 2.3.3.** Suppose $F(h_2) = \int (t^2 - 1)\phi(t)g(th_2)dt$, $G(h_2) = \int t^2\phi^2(t)g(th_2)dt$, $\phi(t)$ is standard normal density function, $g(t)$ is a density function of the error term $\epsilon$ with mean 0 and variance $\sigma^2$. Then we have

(a) $\lim_{h_2 \to \infty} R(h_2) = 1$ and hence $\inf_{h_2} R(h_2) \leq 1$;

(b) $R(h_2) \to \infty$ as $h_2 \to 0$, if $g(t)$ has bounded third derivative;

(c) When $g(t)$ is normal distribution, $R(h_2) > 1$ for any finite $h_2$ and $\inf_{h_2} R(h_2) = 1$.

**Proof:** (a) Notice that

$$h_2^3 G(h_2) = h_2^3 \int t^2 \phi^2(t)g(th_2)dt = \int t^2 \phi^2(t/h_2)g(t)dt;$$

$$h_2^2 F(h_2) = h_2 \int (t^2 - 1)\phi(t)g(th_2)dt = \int (t^2/h_2^2 - 1)\phi(t/h_2)g(t)dt.$$
Then $h_2 \to \infty, h_2^2 G(h_2) \to \phi^2(0)\sigma^2$ and $h_2 F(h_2) \to -\phi(0)$. Thus $h_2^2 G(h_2) F(h_2)^{-2} = h_2^3 G(h_2)/(h_2^2 F(h_2))^2 \to \sigma^2$, when $h_2 \to \infty$. So

$$\lim_{h_2 \to \infty} R(h_2) = \lim_{h_2 \to \infty} h_2^2 G(h_2) F(h_2)^{-2} \sigma^{-2} = 1. \quad (2.27)$$

From (2.27) we can get $\inf_{h_2} R(h_2) \leq \lim_{h_2 \to \infty} R(h_2) = 1$.

(b) Suppose that $h_2 \to 0$, then

$$F(h_2) = \int (t^2 - 1)\phi(t)g(th_2)dt$$

$$= \int (t^2 - 1)\phi(t)[g(0) + g'(0)th_2 + \frac{1}{2}g''(0)t^2 h_2^2 + o(h_2^2) t^3]dt$$

$$= \frac{1}{2}g''(0)h_2^2 \int (t^4 - t^2)\phi(t)dt + o(h_2^2)$$

$$= g''(0)h_2^2 + o(h_2^2), \quad (2.28)$$

and

$$G(h_2) = \int t^2\phi^2(t)g(th_2)dt$$

$$= \int t^2\phi^2(t)[g(0) + o(1)t]dt$$

$$= g(0)\nu_2 + o(1). \quad (2.29)$$
Thus

\[
R(h_2) = h_2 G(h_2) F(h_2)^{-2} / \sigma^2
\]

\[
= g(0) h_2 \nu_2 + o(h_2) \cdot \frac{1}{g''(0) h_2^4 + o(h_2^4)} \cdot \frac{1}{\sigma^2}
\]

\[
= \frac{g(0) \nu_2}{g''(0) h_2^3 \sigma^2} (1 + o(1)). \quad (2.30)
\]

So \(R(h_2) \to \infty\) as \(h_2 \to 0\).

(c) Suppose \(g(t)\) is the density function of \(N(0, \sigma^2)\). Then

\[
G(h_2) = \int t^2 \phi^2(t) g(th_2) dt = \int t^2 \frac{1}{2\pi} \exp\left\{ -\frac{t^2}{2} \right\} \frac{1}{\sqrt{2\pi} \sigma} \exp\left\{ -\frac{t^2 h_2^2}{2\sigma^2} \right\} dt
\]

\[
= \frac{1}{2\pi \sigma} \int \frac{1}{\sqrt{2\pi}} t^2 \exp\left\{ -\frac{t^2}{2} \left(2 + \frac{h_2^2}{\sigma^2}\right) \right\} dt
\]

\[
= \frac{1}{2\pi \sigma} \left( \frac{\sigma^2}{2\sigma^2 + h_2^2} \right)^{3/2}
\]

\[
= \frac{\sigma^2}{2\pi} \left(2\sigma^2 + h_2^2\right)^{-3/2},
\]
and

$$F(h_2) = \int (t^2 - 1) \phi(t) g(h_2) dt = \int (t^2 - 1) \frac{1}{\sqrt{2\pi}} \exp\{-\frac{t^2}{2}\} \frac{1}{\sqrt{2\pi\sigma}} \exp\{-\frac{t^2 h_2^2}{2\sigma^2}\} dt$$

$$= \frac{1}{2\pi\sigma} \int (t^2 - 1) \exp\{-\frac{t^2}{2}(1 + \frac{h_2^2}{\sigma^2})\} dt$$

$$= \frac{1}{2\pi\sigma} \sqrt{2\pi} \left[ \left( \frac{\sigma^2}{\sigma^2 + h_2^2} \right)^{3/2} - \left( \frac{\sigma^2}{\sigma^2 + h_2^2} \right)^{1/2} \right]$$

$$= -\frac{1}{\sqrt{2\pi\sigma}} \left( \frac{\sigma^2}{\sigma^2 + h_2^2} \right)^{1/2} \frac{h_2^4}{\sigma^2 + h_2^2}$$

$$= -\frac{h_2^2}{\sqrt{2\pi}} (\sigma^2 + h_2^2)^{-3/2}.$$

Then we get

$$h_2 G(h_2) F(h_2)^{-2} = \frac{h_2 \sigma^2 (2 \sigma^2 + h_2^2)^{-3/2}}{h_2^3 (\sigma^2 + h_2^2)^{-3}} = \frac{\sigma^2 (\sigma^2 + h_2^2)^3}{h_2^3 (2 \sigma^2 + h_2^2)^{3/2}}$$

$$= \frac{\sigma^2 (\sigma^4 + h_2^4 + 2 \sigma^2 h_2^4)^{3/2}}{(2 \sigma^2 h_2^2 + h_2^4)^{3/2}}$$

$$= \left( \frac{\sigma^4 + h_2^4 + 2 \sigma^2 h_2^4}{2 \sigma^2 h_2^2 + h_2^4} \right)^{3/2} \sigma^2.$$ 

Thus $h_2 G(h_2) F(h_2)^{-2} > \sigma^2$ for finite $h_2$ and $\inf_{h_2} h_2 G(h_2) F(h_2)^{-2} = \sigma^2$. From (b) we can get $\inf_{h_2} h_2 G(h_2) F(h_2)^{-2} = \sigma^2$. □

**Remark 2.3.1.** From (a), we know the performance of the modal local polynomial regression can be arbitrary close to that of local polynomial regression in terms of the asymptotic MSE. If there exists some $h_2$ such that $h_2 G(h_2) F(h_2)^{-2} < \sigma^2$, then the modal local polynomial estimator has smaller asymptotic MSE than local polynomial estimator.
As stated in the last section, when \( h_2 \to \infty \), modal local polynomial regression becomes the same as local polynomial regression. Hence both methods have the same asymptotic mean squared error (MSE). The result \( \lim_{h_2 \to \infty} R(h_2) = 1 \) of (a) reassures this result.

From (c), we know when \( h_2 \to 0 \), \( R(h_2) \to \infty \) and the relative efficiency between modal local polynomial and local polynomial will go to zero. So the optimal choice of \( h_2 \) for modal local polynomial should not go to 0. That is the reason why we do not assume that \( h_2 \) goes to zero as \( n \to \infty \). Instead, we assume \( h_2 \) is a constant. Later, we will see the optimal choice of \( h_2 \) only depends on the error distribution.

From (d), we know that when the error term has normal distribution, the optimal modal local polynomial will be the same as local polynomial regression and in this situation \( h_2 \to \infty \).

The following also provides the asymptotic normality result for modal local polynomial estimator.

**Theorem 2.3.4.** Under the same assumption as Theorem 2.3.1, we have the following asymptotic normality

\[
\sqrt{n}h_1 \left[ \hat{m}_v(x_0) - m^{(v)}(x_0) - b_v(x_0) + o_p(h_1^{p+1-v}) \right] \xrightarrow{L} N(0, \sigma_v(x_0)),
\]

where

\[
b_v(x_0) = h_1^{p+1-v} \frac{m^{(p+1)}(x_0)}{c_{v+1}^T S^{-1} c_p \frac{v!}{(p+1)!}}
\]

and \( \sigma_v(x_0) = h_2 G(h_2) F^{-2}(h_2) v!^2 f(x_0)^{-1} h_1^{-2v} e_{v+1}^T S^{-1} S^* S^{-1} e_{v+1} \) is the \( (v+1) \)th diagonal element of \( h_2 G(h_2) F^{-2}(h_2) v!^2 f(x_0)^{-1} h_1^{-2v} S^* S^{-1} \).
Proof: Denote $H = \text{diag}\{1, h_1, \ldots, h_p\}$, $\theta = H \beta$, and $Z_i = H^{-1}X^*$, where $X^* = [1, X_i - x_0, \ldots, (X_i - x_0)^p]^T$ and $\beta = (\beta_0, \ldots, \beta_p)$. Let $\theta_0 = \left(m(x_0), \ldots, h_p^m(x_0)/p!\right)^T$ be the true value for $\theta$. Since $\hat{\beta}$ is a local maximizer of (2.6), $\hat{\theta} = H\hat{\beta}$ is a local maximizer of

$$l_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} \left[K_{h_1}(X_i - x_0)\phi_{h_2}(Y_i - \theta^T Z_i)\right].$$

Denote

$$W_n \triangleq \frac{\partial l_n(\theta_0)}{\partial \theta} = -\frac{1}{nh_2^2} \sum_{i=1}^{n} K_{h_1}(X_i - x_0)\phi'(\frac{Y_i - \theta_0^T Z_i}{h_2})Z_i,$$

$$\Delta_n \triangleq \frac{\partial^2 l_n(\theta_0)}{\partial \theta^2} = \frac{1}{nh_2^3} \sum_{i=1}^{n} K_{h_1}(X_i - x_0)\phi^{(2)}(\frac{Y_i - a_{\phi} \theta_0^T Z_i}{h_2})Z_iZ_i^T,$$

where $\phi^j(t)$ is the $j$th derivative of $\phi(t)$ with $\phi'(t) = (-t)\phi(t)$, $\phi''(t) = (t^2 - 1)\phi(t)$ and $\phi^{(3)}(t) = (3t - t^3)\phi(t)$.

Based on the result (2.22), we have

$$\hat{\theta} - \theta_0 = -\left\{h_2^{-2}F(h_2)f(x_0)S\right\}^{-1} W_n(1 + o_p(1))$$

$$= -\Delta^{-1} W_n(1 + o_p(1)), \tag{2.31}$$

where $\Delta = h_2^{-2}F(h_2)f(x_0)S$.

Let $W_n^* = \sqrt{nh_1} W_n$. Now we prove the asymptotic normality for $W_n^*$. For any unit vector $d \in \mathbb{R}^{p+1}$, we prove

$$\{d^T \text{Cov}(W_n^*) d\}^{-\frac{1}{2}} \{d^T W_n^* - d^T E(W_n^*)\} \overset{L}{\rightarrow} N(0, 1)$$
Let

\[ \xi_i = -\frac{1}{\sqrt{nh_1h_2^2}} K(\frac{X_i - x_0}{h_1}) \phi' \left( \frac{Y_i - a_\theta^T Z_i}{h_2} \right) d^T Z_i. \]

Then \( d^T W_n^* = \sum_{i=1}^n \xi_i \). We check the Lyapunov’s condition. Based on the results (2.19), we know \( \text{Var}(d^T W_n^*) = nh_1d^T \text{Cov}(W_n)d = O(1) \). So we only need to prove \( nE|\xi_1|^3 \to 0 \).

Notice that \( (d^T Z)^2 \leq \|Z\|^2 \|d\|^2 = \|Z\|^2 \), \( \phi'(\cdot) \) is bounded, and \( K(\cdot) \) has compact support.

\[
nE|\xi_1|^3 \leq O(nm^{-3/2}h_1^{-3/2}) \sum_{j=0}^p E \left| K^3 \left( \frac{X - x_0}{h_1} \right) \left( \phi' \left( \frac{Y - \theta^T Z}{h_2} \right) \right)^3 \left( \frac{X - x_0}{h_1} \right)^{3j} \right|
= O(n^{-1/2}h_1^{-3/2}) \sum_{j=0}^p O(h_1)
= O((nh_1)^{-1/2}) \to 0.
\]

So the asymptotic normality for \( W_n^* \) holds. From (2.31) and Slutsky’s theorem, we have

\[
\sqrt{nh_1} \left( \hat{\theta} - \theta_0 - h_1^{p+1}\beta_{p+1} S^{-1} c_p + o_p(h_1^{p+1}) \right) \overset{L}{\to} N \left( 0, h_2 G(h_2) F^{-2}(h_2) f^{-1}(x_0) S^{-1} S^* S^{-1} \right).
\]

Denote \( e_{v+1} = (0, \ldots, 0, 1, 0, \ldots, 0)^T, v = 0, \ldots, p, \) with 1 on the \((v+1)^{th}\) position. Notice that \( \hat{\theta}_v = h_1^\ast \hat{m}_v(x_0)/v! \) and \( \beta_{p+1} = m^{(p+1)}(x_0)/(p+1)! \). Then we have

\[
\sqrt{nh_1} \left( \hat{m}_v(x_0) - m^{(v)}(x_0) - h_1^{p+1-v} m^{(p+1)}(x_0) e_{v+1}^T S^{-1} c_p \frac{v!}{(p+1)!} + o_p(h_1^{p+1-v}) \right) \overset{L}{\to} N \left( 0, \sigma_v(x_0) \right),
\]
where $\sigma_v(x_0) = h_2G(h_2)F(h_2)^{-2}v!^2 f(x_0)^{-1}h_1^{-2v}e_{v+1}^T S^{-1} S e_{v+1}$ is the $(v+1)^{th}$ diagonal element of $h_2G(h_2)F^{-2}(h_2)v!^2 f(x_0)^{-1}h_1^{-2v}S^{-1} S$. □

Similar to local polynomial regression, when $p - v$ increases consecutively from even number to odd number the asymptotic variance (2.16) of the modal local polynomial estimator does not change, however when it increases consecutively from odd number to even number the asymptotic variance increases. We know the higher order polynomial approximations result in a smaller order of the asymptotic bias. It turns out clearly that polynomial fits with $p - v$ odd outperform those with $p - v$ even. See Ruppert and Wand (1994) and Fan and Gijbels (1996). Throughout the rest of the chapters, we take $p - v$ odd. For example if we want to estimate the regression function ($v = 0$), the fit of odd order of $2q + 1$ has the same asymptotic variance in comparison with fitting of even order $2q$. However, with an extra parameter, there may be a bias reduction especially in the boundary regions or in highly clustered design regions. In the following chapters, we will always assume $p - v$ is odd ( $p - v = 1$ for example). Generally $p = 1$ is enough for the estimation of regression function $m(x_0)$. Next let us look at this special case — modal local linear regression.

**Proposition 2.3.1.** Suppose $K(t)$ is symmetric about 0. For modal local linear regression the asymptotic bias of $\hat{m}(x_0)$ is

$$
\text{Bias}\{\hat{m}(x_0)\} = \frac{1}{2} m''(x_0)\mu_2 h_1^2 + o(h_1^2); \quad (2.32)
$$
the asymptotic variance of $\hat{m}(x_0)$ is

$$
\text{Var}\{\hat{m}(x_0)\} = \frac{v_0 h_2}{n h_1 f(x_0)} G(h_2) F(h_2)^{-2} + o(\frac{1}{nh_1});
$$

and

$$
\sqrt{nh_1} \left[ \hat{m}(x_0) - m(x_0) - \frac{1}{2} m''(x_0) \mu_2 h_1^2 + o_p(h_1^2) \right] \overset{L}{\to} N \left( 0, \frac{v_0 h_2}{f(x_0)} G(h_2) F(h_2)^{-2} \right),
$$

where $F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt$ and $G(h_2) = \int t^2 \phi^2(t) g(th_2) dt$.

Prove: When $K(t)$ is symmetric about 0, the $\mu_j$ will be 0 for odd $j$. In Theorem 2.3.2, take $p = 1$ and $v = 0$. Notice that

$$
S = \begin{pmatrix} 1 & 0 \\ 0 & \mu_2 \end{pmatrix}, \quad S^* = \begin{pmatrix} \nu_0 & \nu_1 \\ \nu_1 & \nu_2 \end{pmatrix}, \quad c_1 = (\mu_2, \mu_3).
$$

Input the above formula to (2.16), (2.17), and Theorem 2.3.4, we will obtain the results (2.32), (2.33), and (2.34).

For local linear regression with bandwidth $h_1$, we have the following asymptotic expansion for bias and variance of $\hat{m}(x_0)$ based on (1.10) and (1.11)

$$
\text{Bias}\{\hat{m}(x_0)\} = \frac{1}{2} m''(x_0) \mu_2 h_1^2 + o(h_1^2),
$$

$$
\text{Var}\{\hat{m}(x_0)\} = \frac{\sigma^2 \nu_0}{nh f(x_0)} + o(\frac{1}{nh}),
$$
where $\text{Var}(Y \mid X) = \text{Var}(\epsilon) = \sigma^2$. So the modal local linear estimator has the same asymptotic bias as the local linear estimator. The ratio of the asymptotic variance of modal local linear estimator to that of the local linear estimator is $R(h_2) = h_2 G(h_2) F(h_2) \sigma^{-2}$. Since $\inf_{h_2} R(h_2) \leq 1$, based on the results in Theorem 2.3.3, model local linear regression works at least as well as local linear regression in terms of the MSE.

Figure 2.1 is the plot by adding the modal local linear fit (red dotted curve) to the Figure 1.2. From the Figure 2.1, we can see the modal local linear fit is almost the same as the local linear fit (dash-dot curve) without outliers. The modal local linear fit automatically downweights outliers. Hence, unlike local linear regression, modal local linear regression is robust to outliers.

### 2.4 Bandwidth Selection

#### 2.4.1 Ideal Choice of Bandwidths

The choice of bandwidth is very crucial. In this section we will discuss how to determine the optimal bandwidth. There are two types of bandwidth — local bandwidth (variable bandwidth) and global bandwidth (constant bandwidth). Borrowing the idea from local polynomial regression, a theoretical optimal local bandwidth for estimating $m^{(e)}(x_0)$ is obtained by minimizing the Mean Square Error (MSE)

$$[\text{Bias}\{\hat{m}^{(e)}(x_0)\}]^2 + \text{Var}\{\hat{m}_{v}(x_0)\}.$$
The above MSE can be approximated by the asymptotic MSE based on (2.16) and (2.17). For simplicity let us mainly focus on modal local linear regression. Without any difficulty, the results in this section can be extended to modal local polynomial regression.

**Theorem 2.4.1.** For modal local linear regression, the asymptotically optimal bandwidths \((h_1, h_2)\) that minimize the asymptotic MSE based on (2.32) and (2.33) are

\[
 h_{2\text{opt}} = \arg \min_h h G(h) F(h)^{-2}, \\
 h_{1\text{opt}}(x_0) = \left( \frac{\nu_0 h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2}}{n f(x_0) m''(x_0)^2 \mu_2^2} \right)^{1/5},
\]

(2.35)

where \(G(h_2) = \int t^2 \phi^2(t) g(th_2) dt, F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt, \phi(t)\) is the standard normal density function and \(g(t)\) is the density function of the error term.

**Proof:** Based on (2.32) and (2.33), the asymptotic MSE of \(\hat{m}(x_0)\) is

\[
 \left( \frac{1}{2} m''(x_0) \mu_2 h_1^2 \right)^2 + \frac{\nu_0 h_2}{nh_1 f(x_0)} G(h_2)^{-2} = \frac{1}{4} m''(x_0)^2 \mu_2^2 h_1^4 + \frac{\nu_0 h_2}{nh_1 f(x_0)} G(h_2) F(h_2)^{-2}.
\]

(2.36)

The minimization of above formula leads to

\[
 h_{2\text{opt}} = \arg \min_h h G(h) F(h)^{-2}, \\
 h_{1\text{opt}}(x_0) = \left( \frac{\nu_0 h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2}}{n f(x_0) m''(x_0)^2 \mu_2^2} \right)^{1/5},
\]

where \(G(h_2) = \int t^2 \phi^2(t) g(th_2) dt, F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt.\) □
From Theorem 2.3.3, we know that when the error is normal distribution, $R(h_2)$ reach the minimum value 1 only when $h_2$ goes to infinity. However, we can make $R(h_2)$ arbitrary close to 1, by taking a large enough $h_2$ value. Hence without making any big difference, for simplicity of explanation, here $h_{2_{opt}}$ can also be taken as $\infty$. In practice, when this happens, we can just simply take a very large $h_2$ value for $h_{2_{opt}}$.

From Theorem 1.1.2, we know that the asymptotically optimal local bandwidth for local linear regression is

$$h_{opt}(x_0) = \left( \frac{\nu_0 \sigma^2}{n \mu_{x_0}^2 m''(x)^2 f(x)} \right)^{1/5}.$$ 

So the asymptotically optimal local bandwidth $h_1$ for modal local linear regression is

$$h_{1_{opt}}(x_0) = \left( \frac{\nu_0 h_{2_{opt}} G(h_{2_{opt}}) F(h_{2_{opt}})^{-2}}{n f(x_0) m''(x_0)^2 \mu_{x}^2} \right)^{1/5} \left( h_{2_{opt}} G(h_{2_{opt}}) F(h_{2_{opt}})^{-2} \sigma^{-2} \right)^{1/5}$$

$$= h_{opt}(x_0) (R(h_{2_{opt}}))^{1/5},$$

where $R(h_{2_{opt}}) = h_{2_{opt}} G(h_{2_{opt}}) F(h_{2_{opt}})^{-2} \sigma^{-2}$. Hence for bandwidth $h_1$, the asymptotically optimal local bandwidth of modal local linear regression is $R(h_{2_{opt}})^{1/5}$ times that of local linear regression.

The optimal global bandwidth is obtained by minimizing the global loss. A commonly used and simple measure of global loss is the weighted Mean Integrated Square
Error (MISE)
\[
\int \left( \left| \text{Bias}\{\hat{m}_v(x)\} \right|^2 + \text{Var}\{\hat{m}_v(x)\} \right) w(x) dx,
\]
where \( w \geq 0 \) is some weight function, such as 1 and design density \( f(x) \). Using the asymptotic expression in (2.16) and (2.17), we can find the asymptotically optimal global bandwidth. In the following, we give the result for modal local linear regression. Similar results can be found for the higher order situation.

**Theorem 2.4.2.** For modal local linear regression, the asymptotically optimal global bandwidths \((h_1, h_2)\) that minimize the asymptotic MISE based on (2.32) and (2.33) are

\[
\begin{align*}
 h_{2\text{opt}} & = \arg \min_{h_2} h_2 G(h_2) F(h_2)^{-2}, \\
 h_{1\text{opt}} & = \left( \frac{\nu_0 h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \int w(x) / f(x) dx}{n \mu_2^2 \int m''(x)^2 w(x) dx} \right)^{1/5},
\end{align*}
\]

(2.37)

where \( G(h_2) = \int t^2 \phi^2(t) g(th_2) dt \), \( F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt \), \( \phi(t) \) is the standard normal density function and \( g(t) \) is the density function of the error term.

**Proof:** Based on (2.32) and (2.33), the asymptotic MISE is

\[
\frac{1}{4} \mu_2^2 h_1^4 \int m''(x)^2 w(x) dx + \frac{\nu_0 h_2^2 G(h_2) F(h_2)^{-2}}{n h_1} \int \frac{w(x)}{f(x)} dx,
\]

(2.38)

where \( G(h_2) = \int t^2 \phi^2(t) g(th_2) dt \), and \( F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt \). The minimization of above formula leads to (2.37). \( \square \)
From Theorem 1.1.2, we know the asymptotically optimal global bandwidth for local linear regression is

\[ h_{\text{opt}} = \left( \frac{\nu \sigma^2 \int w(x)/f(x)dx}{n \mu_2^2 \int m''(x)^2 w(x)dx} \right)^{1/5}. \]

So the asymptotically optimal global bandwidth of \( h_1 \) for modal local linear regression is

\[ h_{1\text{opt}} = \left( \frac{\nu \sigma^2 \int w(x)/f(x)dx}{n \mu_2^2 \int m''(x)^2 w(x)dx} \right)^{1/5} \left( h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \sigma^{-2} \right)^{1/5} \]

\[ = h_{\text{opt}} (R(h_{2\text{opt}}))^{1/5}, \]

where \( R(h_{2\text{opt}}) = h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \sigma^{-2} \). So for \( h_1 \), the asymptotically optimal global bandwidth of modal local linear regression is \( R(h_{2\text{opt}})^{1/5} \) times that of local linear regression.

From the above two theorems, we can see that for \( h_2 \), both the asymptotically optimal local and global bandwidth are the same. Denote the asymptotically optimal bandwidth for \( h_2 \) by \( h_{2\text{opt}} \). For \( h_1 \), both the asymptotically optimal local and global bandwidth of modal local linear regression are \( R(h_{2\text{opt}})^{1/5} \) times that of local linear regression.

From Theorem 2.3.3, we know \( R(h_{2\text{opt}}) \leq 1 \). So for \( h_1 \), the asymptotically optimal local and global bandwidth of modal local linear regression are smaller than that of local linear regression. In general modal local polynomial regression uses bandwidth \( h_2 \) to
control the effect of outliers and thus to reduce the variance of its estimator. In order to
balance the bias and variance tradeoff, modal local polynomial regression uses a smaller
bandwidth \( h_1 \) than local polynomial regression to reduce the bias of its estimator.

### 2.4.2 Bandwidth Selection in Practice

From Theorems 2.4.1 and 2.4.2, we know the asymptotically optimal bandwidths
depend on the unknown quantities such as the error density \( g(\cdot) \), the design density \( f(\cdot) \),
and the derivative function \( m''(\cdot) \). Thus, we can not use the results directly. One of the
commonly used approaches, in practice, is to substitute the unknown quantities by pilot
estimates, leading to so-called ‘plug-in’ type bandwidth selectors.

Based on the discussion of Section 2.4.1, we know the asymptotically optimal
bandwidth of \( h_1 \) depends on the asymptotically optimal bandwidth of \( h_2 \). Hence, let us
first focus on how to find the asymptotically optimal bandwidth for \( h_2 \) in practice. Based
on (2.35) and (2.37), we know the asymptotically optimal global (local) bandwidth for
\( h_2 \) is

\[
h_2^{opt} = \arg \min_{h_2} h_2 G(h_2) F(h_2)^{-2}.
\]

Notice that since there is no explicit solution for \( h_2 \), we propose to use grid search method.
In order to use grid search method we need to find a way to estimate the function \( F(h_2) \)
and \( G(h_2) \). The only unknown quantity in \( F(h_2) \) and \( G(h_2) \) is the density function \( g(t) \)
of the error term.

One way to estimate \( F(h_2) \) and \( G(h_2) \) is to estimate \( g(t) \) first and then input it
to the function \( F(h_2) \) and \( G(h_2) \). To estimate \( g(t) \), we first get the estimates of the error
term: \( \hat{\epsilon}_i = y_i - \hat{m}(x_i) \) by fitting the data using local polynomial regression or any simple robust smoothing method such as LOWESS, then we estimate \( g(t) \) by the kernel density estimator
\[
\hat{g}(t) = \frac{1}{n} \sum_{i=1}^{n} K_d(\hat{\epsilon}_i - t),
\]
where \( d \) is the bandwidth for kernel density estimator. If \( K(\cdot) \) is a Gaussian kernel, we can get the following simplified formulas of \( \hat{F}(h_2) \) and \( \hat{G}(h_2) \), which do not involve any integration,
\[
\hat{F}(h_2) = \frac{1}{\sqrt{2\pi nd}} \sum_{i=1}^{n} \exp\left\{-\frac{\hat{\epsilon}_i^2}{2(h_2^2 + d^2)} \right\} (h_2^2 + 2d^3) \frac{h_2^2d\hat{\epsilon}_i^2 - 2d - h_2^2d^3}{(h_2^2 + d^2)^{5/2}},
\]
\[
\hat{G}(h_2) = \frac{1}{2\pi nd} \sum_{i=1}^{n} \exp\left\{-\frac{\hat{\epsilon}_i^2}{h_2^2 + 2d^2} \right\} \frac{h_2^2 + 2d^3}{(h_2^2 + 2d^2)^{5/2}} + h_2^2d\hat{\epsilon}_i^2 .
\]

For a second method to estimate \( F(h_2) \) and \( G(h_2) \), notice that
\[
F(h_2) = \int (t^2 - 1) \phi(t)g(\theta_2)dt = h_2^{-1} \int (t^2/h_2^2 - 1) \phi(t/h_2)g(t)dt = h_2^{-1} E[(\epsilon_2/h_2 - 1)\phi(\epsilon/h_2)];
\]
\[
G(h_2) = \int t^2 \phi^2(t)g(\theta_2)dt = h_2^{-3} \int t^2 \phi^2(t/h_2)g(t)dt = h_2^{-3} E[\epsilon^2 \phi^2(\epsilon/h_2)],
\]
where the error term \( \epsilon \) has the density \( g(t) \). By approximating the above expectations by their sample means, we can estimate \( F(h_2) \) and \( G(h_2) \) based on \( \hat{\epsilon}_1, \ldots, \hat{\epsilon}_n \)
\[
\hat{F}(h_2) = \frac{1}{nh_2} \sum_{i=1}^{n} \left[ (\hat{\epsilon}_i^2/h_2^2 - 1)\phi(\hat{\epsilon}_i/h_2) \right],
\]
\[
\hat{G}(h_2) = \frac{1}{nh_2^3} \sum_{i=1}^{n} \left[ \hat{\epsilon}_i^2 \phi^2(\hat{\epsilon}_i/h_2) \right].
\]
For the second method this gives

$$h_2 \hat{G}(h_2) \hat{F}(h_2)^{-2} = \frac{n^{-1} \sum_{i=1}^{n} [\hat{\epsilon}_i^2 \phi^2(\hat{\epsilon}_i/h_2)]}{\{n^{-1} \sum_{i=1}^{n} [(\hat{\epsilon}_i^2/h_2^2 - 1)\phi(\hat{\epsilon}_i/h_2)]\}^2}.$$  

This method works well for large $h_2$. However, for small $h_2$, this method does not approximate $F(h_2)$ and $G(h_2)$, especially $F(h_2)$, well. This is due to the large estimation error of the sample means for the expectations in (2.42). To see the reason, notice that when $h_2 \to 0$,

$$E[(\epsilon^2/h_2^2 - 1)^2 \phi^2(\epsilon/h_2)] = \int (t^2/h_2^2 - 1)^2 \phi^2(t/h_2)g(t)$$

$$= h_2 \int (t^2 - 1)^2 \phi^2(t)g(th_2)dt$$

$$= g(0)h_2 \int (t^2 - 1)^2 \phi^2(t) + o(h_2)$$

and

$$E[\epsilon^4 \phi^4(\epsilon/h_2)] = \int t^4 \phi^4(t/h_2)g(t)dt$$

$$= h_2^5 \int t^4 \phi^4(t)g(th_2)dt$$

$$= h_2^5 g(0) \int t^4 \phi^4(t)dt + o(h_2^5).$$

From (2.28) and (2.29), we have

$$E[(\epsilon^2/h_2^2 - 1)\phi(\epsilon/h_2)] = F(h_2)h_2 = g''(0)h_2^3 + o(h_2^3),$$
and

\[ E[\epsilon^2 \phi^2(\epsilon/h_2)] = G(h_2)h_2^3 = g(0)\nu_2 h_2^3 + o(h_2^3). \]

By the above results and \( X = E(X) + O_p(\sqrt{\text{Var}(X)}) \), if using the second method, we have

\[
h_2 \hat{G}(h_2) \hat{F}(h_2)^{-2} = \frac{\frac{1}{n} \sum_{i=1}^{n} \left[ \epsilon_i^2 \phi^2(\epsilon_i/h_2) \right]}{\left\{ \frac{1}{n} \sum_{i=1}^{n} \left[ (\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2) \right] \right\}^2}
\]

\[
= \frac{E[\epsilon^2 \phi^2(\epsilon/h_2)] + O_p(\sqrt{\text{Var}[\epsilon^2 \phi^2(\epsilon/h_2)]/n})}{\left\{ E[(\epsilon^2/h_2^2 - 1)\phi(\epsilon/h_2)] + O_p(\sqrt{\text{Var}[(\epsilon^2/h_2^2 - 1)\phi(\epsilon/h_2)]/n}) \right\}^2}
\]

\[
= \frac{g(0)\nu_2 h_2^3 + o(h_2^3)}{\left\{ g''(0) h_2^3 + o(h_2^3) + O_p(\sqrt{h_2/\nu}) \right\}^2}.
\]

(2.45)

When using the second method, for any finite sample size \( n \), if \( h_2 \to 0 \), the orders of the standard error of the sample means for \( \epsilon^2 \phi^2(\epsilon/h_2) \) and \( (\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2) \) are even smaller than the orders of their expectations. Thus the estimations of the expectation terms in (2.42) are dominated by the variability of their sample means. The denominator in (2.45) has much smaller order of the standard error than the nominator. Thus in practice, when \( h_2 \) goes to zero, \( h_2 \hat{G}(h_2) \hat{F}(h_2)^{-2} \) estimated by the second method most likely goes to zero. The selected \( h_2 \) will also most likely be very small. From Theorem 2.3.3, we know \( h_2 G(h_2)F(h_2)^{-2} \to \infty \) as \( h_2 \to 0 \). So the asymptotically optimal \( h_2 \) will not be very small. From the above discussion, we can see that if \( h_2 \) is too small, the second method by (2.43) and (2.44) gives the wrong estimate of \( h_2 G(h_2)F(h_2)^{-2} \) and thus the wrong selection of \( h_2 \) (generally too small). In order to get the more accurate order of \( h_2 G(h_2)F(h_2)^{-2} \), we suggest the order of \( h_2 \) be smaller than \( (1/n)^{1/5} \).
Once we know how to estimate $G(h_2)$ and $F(h_2)$, by the above two methods, we can use the grid search method to find the asymptotic bandwidth $h_2$. We first estimate $H(h_2) = h_2G(h_2)F(h_2)^{-2}$ for some grid points of $h_2$ such as $h_2 = 0.5\hat{\sigma} \times 1.1^j$, $j = 0, \ldots, g$, by either of the two proposed methods, where $\hat{\sigma}^2$ is the sample variance of $(\hat{\epsilon}_1, \ldots, \hat{\epsilon}_n)$, then we choose $\hat{h}_{2opt}$ which minimizes $\hat{H}(h_2)$.

Now let us estimate the asymptotically optimal bandwidth $h_1$. Based on Section 2.4.1, we know the asymptotically optimal bandwidth for $h_1$ of modal local polynomial is $[R(h_{2opt})]^{1/5}$ times that of local polynomial regression. Suppose the selected $h_2$ is $\hat{h}_{2opt}$, we can estimate $[R(h_{2opt})]^{1/5}$ by $[H(\hat{h}_{2opt})/\hat{\sigma}^2]^{1/5}$. Then we can make use of the bandwidth selection method for local polynomial regression, such as the Plug In method (Ruppert, Sheather and Wand, 1995). Suppose the optimal bandwidth selected for local polynomial regression is $\hat{h}_{1opt}$, then $h_1$ is estimated by $\hat{h}_{1opt} = [H(\hat{h}_{2opt})/\hat{\sigma}^2]^{1/5}\hat{h}_{opt}$. So for modal local polynomial regression, the only new burden for us is to select the bandwidth $h_2$.

Next let us use some examples to see how the two proposed methods of the bandwidth selection for $h_2$ works.

**Example 1**: Generate the independent and identically distributed (i.i.d.) data $\{(x_i, y_i), i = 1, \ldots, 300\}$ from the model

$$Y_i = 2 \sin(2\pi X_i) + \epsilon_i,$$

where $X_i \sim U(0, 1)$ and $\epsilon_i \sim 0.95N(0, 1) + 0.05N(0, 3^2)$. 

The variance of the error distribution \( g(t) \) is 1.4. Estimated variance based on the generated error \((\epsilon_1, \ldots, \epsilon_{300})\) is 1.3550. Estimated variance based on the estimated error \((\hat{\epsilon}_1, \ldots, \hat{\epsilon}_{300})\) is 1.3416. We use three methods to find \( R(h_2) = h_2G(h_2)F(h_2)^{-2}/1.4 : \)

- Method 1: Estimate \( R(h_2) = h_2G(h_2)F(h_2)^{-2}/1.4 \) by (2.40) and (2.41).
- Method 2: Estimate \( R(h_2) = h_2G(h_2)F(h_2)^{-2}/1.4 \) by (2.43) and (2.44).
- True: Find \( R(h_2) = h_2G(h_2)F(h_2)^{-2}/1.4 \) based on the true density \( g(t) \) of the error distribution.

Figure 2.2 is the plot \( R(h_2) \) versus \( h_2 \), with \( h_2 = 0.8 \times 1.04^{i-1}, i = 1, \ldots, 100 \) for the three methods. The dotted curve is for “Method 1”, the star curve is for “Method 2”, and the solid curve is for “True”. From the plot we can see \( R(h_2) \) has a unique minimum and the three curves reach the minimum at almost the same \( h_2 \) value 2.6 (though the \( R(h_2) \) values are different for the three methods). So both of the Method 1 and Method 2 find the minima \( h_2 \) very well in this example. Notice that, for all three curves, the minimum of \( R(h_2) \) is less than 1, so we can get smaller MSE if we use the new method modal local polynomial regression instead of local polynomial regression.

Figure 2.3 is the plot of \( R(h_2) \) versus \( h_2 \) for small bandwidths \( h_2 \) with \( h_2 = 0.01 \times 1.06^{i-1}, i = 1, \ldots, 50 \). From this plot, we can see there is a big difference of the estimates of \( R(h_2) \) between Method 2 and True, especially for small \( h_2 \) values. The true \( R(h_2) \) values go to \( +\infty \) when \( h_2 \to 0 \). However the estimated \( R(h_2) \) values for Method 2 are very small when \( h_2 \) is small and go to zero when \( h_2 \to 0 \). Thus, Method 2 does not work well for small \( h_2 \) values, as suggested earlier in this section. However, Method 1 approximates \( R(h_2) \) well even for small \( h_2 \) values.
Figure 2.4 is the plot of $R(h_2)$ versus $h_2$ for small bandwidths $h_2$ with $h_2 = 0.01 \times 1.06^{(i-1)}, i = 1, \ldots, 50$ for Method 2 alone. From this plot, we can see if we do the grid search started from the very small value, method 2 will most likely choose the very small $h_2$, which is not correct based on the true curve. As explained before, for very small $h_2$ values, the variance of $(\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2)$ is very large compared to its expected value and thus the expectation $E[(\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2)]$ can not be estimated well by the sample mean in (2.43), if the sample size is not big enough. Hence, $F(h_2)$ can not be estimated well by Method 2. For example, if $h_2 = 0.01$, $G(h_2) = 0.0544$, the expectation of $(\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2)$ is $-3.8 \times 10^{-7}$, and the standard deviation is 0.0286. Thus, for sample size 100, the standard error of the sample mean of $(\epsilon_i^2/h_2^2 - 1)\phi(\epsilon_i/h_2)$ is 0.00286, which is much bigger than the absolute value of its expectation. Even for sample size $10^4$, the standard error of the sample mean is $2.86 \times 10^{-4}$, which is still much bigger than the absolute value of its expectation $-3.8 \times 10^{-7}$.

From the above three plots, we can know that generally both two proposed bandwidth selection methods work well for not too small $h_2$ values and can find the minima of $R(h_2)$ very well. The disadvantage of Method 2 is that it does not work well for small $h_2$ values. Based on the experience of the simulation study, we may start the grid search from 0.5 times the standard error of the error term. The disadvantage of Method 1 is that we need to select an addition bandwidth for the kernel density estimation of the error term. We will next show that the bandwidth $d$ for the kernel density estimation (2.39) is in fact not crucial to the selection of bandwidth $h_2$.

Figure 2.5 are the plot of $R(h_2)$ versus $h_2$, with $h_2 = 0.8 \times 1.04^{i-1}, i = 1, \ldots, 100$, for method 1 with different bandwidths — the bandwidth, $d$, selected by any bandwidth
selection method for kernel density estimation, $d/2$, and $2d$. From the plot, we can see method 1 with different bandwidths reach the minimum of $R(h_2)$ at almost the same $h_2$ value, although the estimates of $R(h_2)$ are different. So, in fact, the choice of the bandwidth $d$ for the density estimation of the error term does not affect the bandwidth selection for $h_2$ too much.

We may also notice, the plot of the four curves in Figure 2.5 are almost parallel to each other for large $h_2$ values. We know $H(h_2) = h_2 G(h_2) F(h_2)^{-2} \rightarrow \sigma^2$ when $h_2 \rightarrow \infty$, where $\sigma^2$ is the variance of $g(t)$. For $\hat{H}(h_2)$, $g(t)$ is replaced by $\hat{g}(t)$ of (2.39) and its variance in fact is $d^2 + \hat{\sigma}^2$ where $\hat{\sigma}^2$ is the sample variance of $(\epsilon_1, \ldots, \epsilon_{300})$, since $\hat{g}(t)$ is a smoothed version of $g(t)$ or say a mixture of normal distribution. Thus $\hat{R}(h_2)$ estimated by Method 1 is close to $1 + d^2 / \hat{\sigma}^2$ when $h_2$ is very large. That is the possible reason why minimum of $R(h_2)$ by method 1 with twice the bandwidth is bigger than 1. 

Thus, in practice, we recommend evaluating $(\hat{R}(h_2) - d^2) / \hat{\sigma}^2$ for Method 1. We call this method modified method 1. Notice that this modification will not affect the bandwidth selection for $h_2$, since $(\hat{R}(h_2) - d^2) / \hat{\sigma}^2$ is a monotone function of $\hat{R}(h_2)$. However, after modification, we can get a more accurate estimation of $R(h_2)$ and can better understand how much is the improvement of the modal local polynomial regression over the local polynomial regression.

Figure 2.6 are the plot of $R(h_2)$ versus $h_2$, with $h_2 = 0.8 \times 1.04^{i-1} , i = 1, \ldots, 100$, for modified method 1 with different bandwidths. From the plot, we can see that $R(h_2)$ found by the modified method 1 are almost the same for different bandwidths of kernel density estimator when $h_2$ values are large. So the effect of the bandwidth of the kernel density estimator seems to be negligible in this example.
Example 2: Generate the independent and identically distributed (i.i.d.) data \( \{(x_i, y_i), i = 1, \ldots, 300\} \) from the model

\[
Y_i = 2 \sin(2\pi X_i) + \epsilon_i,
\]

where \( X_i \sim U(0, 1) \) and \( \epsilon_i \sim t_6 \).

Now the error distribution is the \( t \) distribution. Figure 2.7 is the plot of \( R(h_2) \) versus \( h_2 \) for the modified method 1 and the true one. From the plot, we can know \( R(h_2) \) estimated by the modified method 1 only has one minima and reaches the minimum at almost the same \( h \) value 2.4 as the true one.

In summary, for modal local polynomial regression, we recommend using the modified method 1 for the bandwidth selection in practice. This method can find the bandwidth \( h_2 \) very effectively and accurately and it is not affected too much by the bandwidth selection for the kernel density estimator of the error term. The selection of the bandwidth \( h_1 \) is very simple by using the existing bandwidth selection method for local polynomial regression. Suppose the selected asymptotically optimal bandwidth for local polynomial regression is \( \hat{h} \), the asymptotically optimal bandwidth for \( h_1 \) of modal local polynomial can be estimated by \( [\hat{R}(\hat{h}_{2opt})]^{1/5} \hat{h} \), where \( \hat{R}(\hat{h}_{2opt}) \) is calculated when finding the bandwidth \( \hat{h}_{2opt} \).

The two proposed bandwidth selection methods in this section tried to minimize \( H(h) = hG(h)F(h)^{-2} \) directly. Instead, we can also try to find the solution to
\[ \frac{d}{dh} \log(H(h)) = 0. \] By some simple calculations, we can get

\[ \frac{d}{dh} \log(H(h)) = \frac{2}{hG(h)F(h)} \left[ F(h) \int_{-\infty}^{\infty} \phi^2(t)t^4 g(th) dt + G(h) \int_{-\infty}^{\infty} (3t^2 - t^4) \phi(t) g(th) dt \right]. \]

So we find the root \( h \) such that

\[ F(h) \int_{-\infty}^{\infty} \phi^2(t)t^4 g(th) dt + G(h) \int_{-\infty}^{\infty} (3t^2 - t^4) \phi(t) g(th) dt = 0 \quad (2.46) \]

We can use similar ways as method 1 and 2 to estimate \( G(h), F(h) \), and the two integrations in (2.46). This method may be more efficient and numerically more stable. However, if there is more than one root, we many need to find all of them and select the one with the lowest \( H(h) \) value. We will spend more time on this bandwidth selection method in our future research.

2.4.3 Relative Efficiency between Modal Local Polynomial and Local Polynomial Regression

From the last few sections, we know modal local polynomial works arbitrary close to local polynomial when the error distribution is normal and works better than local polynomial regression when the error distribution has a heavy tail or there are outliers in the data set. In this section, we will find the relative efficiency between the modal local polynomial estimator and the local polynomial estimator for any error distribution. Let us first see the results for local linear case. Suppose both estimators use the asymptotically optimal (local or global) bandwidths.
From the formula (1.10) and (1.11), we know the asymptotic bias and variance of the local linear estimator are

\[
\text{Bias}\{\hat{m}(x_0)\} = \mu_2 \frac{m''(x_0) h^2}{2} + o(h^2),
\]
\[
\text{Var}\{\hat{m}(x_0)\} = \nu_0 \frac{\sigma^2(x_0)}{f(x_0) nh} + o\left(\frac{1}{nh}\right).
\]

Hence, the asymptotic mean squared error of the local linear estimator (MSE(LL)) with the asymptotically optimal bandwidth \(h_{opt}\) at \(x_0\) is

\[
\text{MSE}(\text{LL}) = \mu_2 \frac{m''(x_0) h_{opt}^4}{4} + \nu_0 \frac{\sigma^2}{f(x_0) nh_{opt}}.
\]

From the formula (2.32) and (2.33), we know the asymptotic bias and variance of the modal local linear estimator are

\[
\text{Bias}\{\hat{m}(x_0)\} = \frac{1}{2} m''(x_0) \mu_2 h_{1}^2 + o(h_{1}^2),
\]
\[
\text{Var}\{\hat{m}(x_0)\} = \nu_0 h_2 \frac{G(h_2) F(h_2) - 2}{nh_1 f(x_0)} + o\left(\frac{1}{nh_1}\right),
\]

where \(G(h_2) = \int t^2 \phi_2^2(t) g(th_2) dt\), and \(F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt\). Based on the results of subsection 2.4.1, we know

\[
h_{1\text{opt}} = h_{\text{opt}} \cdot R(h_{2\text{opt}})^{1/5},
\]

where \(h_{1\text{opt}}\) and \(h_{2\text{opt}}\) are the asymptotically optimal bandwidths for \(h_1\) are \(h_2\) and \(R(h_{2\text{opt}}) = h_{2\text{opt}} G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \sigma^{-2}\). As before, for simplicity of explanation, here
we also suppose that $h_{2opt}$ can be taken as $\infty$. Hence, the asymptotic mean squared error of the modal local linear estimator (MSE(MLL)) with the asymptotically optimal bandwidths $h_{1opt}$ and $h_{2opt}$ is

$$
\text{MSE}(\text{MLL}) = \frac{\mu^2}{4} \frac{m''(x_0)^2 h_{1opt}^4}{4} + \nu_0 \frac{\sigma^2}{f(x_0) n h_{1opt}} \cdot h_{2opt} G(h_{2opt}) F(h_{2opt})^{-2} / \sigma^2
$$

$$
= \mu^2 \frac{m''(x_0)^2 h_{opt}^4}{4} + \nu_0 \frac{\sigma^2}{f(x_0) n h_{opt} R(h_{2opt})} R(h_{2opt})^{4/5}
$$

$$
= \left\{ \frac{\mu^2}{4} \frac{m''(x_0)^2 h_{opt}^4}{4} + \nu_0 \frac{\sigma^2}{f(x_0) n h_{opt}} \right\} R(h_{2opt})^{4/5}
$$

$$
= \text{MSE}(\text{LL}) \cdot R(h_{2opt})^{4/5}.
$$

We can see that the ratios of both the asymptotic variance and the square of asymptotic bias of the modal local linear estimator to those of the local linear estimator are $R(h_{2opt})^{4/5}$. Thus, the asymptotic relative efficiency between the modal local linear estimator and the local linear estimator is

$$
\frac{\text{MSE}(\text{LL})}{\text{MSE}(\text{MLL})} = R(h_{2opt})^{-4/5}.
$$

Using similar arguments as above, we can extend the above results to the local polynomial situation.

Theorem 2.4.3. If we use the asymptotically optimal (local or global) bandwidths for both modal local polynomial regression and local polynomial regression, the asymptotic relative efficiency between the modal local polynomial estimator and the local polynomial
estimator of \( m^{(v)}(x_0) \) with order \( p \) is

\[
\frac{\text{MSE}(LP)}{\text{MSE}(MLP)} = R(h_{2\text{opt}})^{-2p^2/(2p+3)}
\]

where \( \text{MSE}(MLP) \) and \( \text{MSE}(LP) \) are the asymptotic mean squared error of the modal local polynomial and the local polynomial estimator of \( m^{(v)}(x_0) \), respectively. \( R(h_{2\text{opt}}) = h_{2\text{opt}}^2 G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \) and \( h_{2\text{opt}} = \arg \min_{h_2} h_2 G(h_2) F(h_2)^{-2} \) with

\[
G(h_2) = \int t^2 \phi^2(t) g(th_2) dt \quad \text{and} \quad F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt.
\]

Take \( p = 1 \) and \( v = 0 \), we have the following special case.

**Theorem 2.4.4.** If we use the asymptotically optimal (local or global) bandwidths for both local linear regression and modal local linear regression, the asymptotic relative efficiency between the modal local linear estimator and the local linear estimator of \( m(x_0) \) is

\[
\frac{\text{MSE}(LL)}{\text{MSE}(MLLL)} = R(h_{2\text{opt}})^{-4/5},
\]

where \( R(h_{2\text{opt}}) = h_{2\text{opt}}^2 G(h_{2\text{opt}}) F(h_{2\text{opt}})^{-2} \) and \( h_{2\text{opt}} = \arg \min_{h_2} h_2 G(h_2) F(h_2)^{-2} \) with

\[
G(h_2) = \int t^2 \phi^2(t) g(th_2) dt \quad \text{and} \quad F(h_2) = \int (t^2 - 1) \phi(t) g(th_2) dt.
\]

From the Theorem 2.3.3, we know

\[
R(h_{2\text{opt}}) \leq 1 \Rightarrow R(h_{2\text{opt}})^{-2p^2/(2p+3)} \geq 1.
\]

Thus, the asymptotic relative efficiency between the modal local polynomial estimator and the local polynomial estimator is always bigger than or equal to 1. Strictly speaking,
if the error distribution is normal, the relative efficiency can only be arbitrary close to one for any finite $h_2$ based on Theorem 2.3.3.

Table 2.1 are the asymptotic relative efficiency between the modal local linear estimator and the local linear estimator for some special error distributions.

Table 2.1.
Relative Efficiency between MLL and LL for different error distributions.

<table>
<thead>
<tr>
<th>Error Distribution</th>
<th>0.95N(0, 1) + 0.05N(0, 3^2)</th>
<th>0.95N(0, 1) + 0.05N(0, 4^2)</th>
<th>$t_6$</th>
<th>$t_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative Efficiency</td>
<td>1.1745</td>
<td>1.3971</td>
<td>1.1281</td>
<td>1.3167</td>
</tr>
</tbody>
</table>

One nice explanation of the relative efficiency is the relative sample sizes for both estimators to achieve the same accuracy. Based on Theorem 2.4.3, we know that in order to get the same accuracy of the estimators in terms of asymptotic MSE, the sample size needed for local polynomial regression is $R(h_{2opt})^{-(2p-2v+2)/(2p+3)} (\geq 1)$ times the sample size needed for modal local polynomial regression. From the Table 2.1, we know that in order to get the same accuracy of the estimators in terms of asymptotic MSE, the ratios of the sample size needed for local linear regression to the sample size needed for modal local linear regression is 1.1745, 1.3971, 1.1281, and 1.3167 for the error distributions $0.95N(0, 1) + 0.05N(0, 3^2)$, $0.95N(0, 1) + 0.05N(0, 4^2)$, $t_6$, and $t_4$, respectively.

### 2.5 Simulation Study and Real Data Application

In this section, we will use three simulation examples and a real data application to compare the modal local linear with the local linear regression and to assess the finite sample performance of the modal local linear regression.
In the following three simulation examples, we used three different error distributions to investigate how well modal local linear regression works. For the local linear regression, we used Ruppert-Sheather-Wand Plug In bandwidth.

**Example 1:** Generate the independent and identically distributed (i.i.d.) data \( \{(x_i, y_i), i = 1, \ldots, n\} \) from the model

\[
Y_i = 2\sin(2\pi X_i) + \epsilon_i,
\]

where \( X_i \sim U(0, 1) \) and \( \epsilon_i \sim N(0, 1) \).

The two smoothing methods we used here are: local linear regression (LL) and modal local linear regression (MLL).

Figure 2.8 is the plot of fitted regression curves for different smoothing methods with sample size \( n = 250 \) for one specific simulation. From the plot we can see that both methods worked quite well. Figure 2.9 is the plot of the mean squared error (MSE) on 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL when sample size is 250. The dashed curve is for “Local linear” and the circles are for “Modal local linear”. From the graph, we can see the two smoothing methods had almost the same MSE for all the grid points.

Table 2.2 is the summary of the average of MSE over 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL for different sample size \( n \), where \( RE = LL/MLL \) is the relative efficiency between the modal local linear estimator and the local linear estimator and \( R \) is the number of replications. From the table 2.2, we can see MLL had almost the same MSE as LL from small sample size 50 to large
Table 2.2.
Summary of MSE for LL and MLL for Example 2 with R=500

<table>
<thead>
<tr>
<th>n</th>
<th>LL</th>
<th>MLL</th>
<th>RE</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.1408</td>
<td>0.1427</td>
<td>0.9868</td>
</tr>
<tr>
<td>100</td>
<td>0.0737</td>
<td>0.0744</td>
<td>0.9909</td>
</tr>
<tr>
<td>250</td>
<td>0.0314</td>
<td>0.0315</td>
<td>0.9963</td>
</tr>
<tr>
<td>500</td>
<td>0.0175</td>
<td>0.0175</td>
<td>0.9975</td>
</tr>
</tbody>
</table>

sample size 500. Notice that in this example the error distribution is exactly normal and there are no outliers. Hence we need not use a robust method and LL should work the best in this case, which is the reason why the RE is smaller than one. However the new adaptive robust method MLL worked almost as well as LL in this example when the error distribution is exactly the normal distribution.

The computation of MLL is very fast. For example, to estimate the regression function for the 46 grid points when sample size is 500, it only took about 0.3 second, using the personal desktop with Pentium 4 CPU 2.40GHz.

**Example 2:** Generate the independent and identically distributed (i.i.d.) data \( \{(x_i, y_i), i = 1, \ldots, n\} \) from the model

\[
Y_i = 2\sin(2\pi X_i) + \epsilon_i ,
\]

where \( X_i \sim U(0, 1) \) and \( \epsilon_i \sim 0.95N(0, 1) + 0.05N(0, 3^2) \).

Now the error distribution is a mixture of normal distribution with 95% data from \( N(0, 1) \) and 5% data from \( N(0, 3^2) \). The 5% data from \( N(0, 3^2) \) are most likely
to be outliers. This kind of mixture distribution is also called contaminated normal distribution, which is generally used to evaluate the robustness of a regression method.

Figure 2.10 is the plot of mean squared error (MSE) of the 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL when sample size is 250. From the graph, we can see MLL had smaller MSE than LL for all the grid points.

Table 2.3 is the summary of the average of MSE on 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL for different sample size $n$. From the Table 2.3, we can see MLL had smaller MSE than LL across all the sample sizes. The relative efficiency between MLL and LL is about 1.0798 for small sample size and 1.1762 for large sample size, which is quite close to the theoretical value, 1.1745, reported on the Table 2.1. Hence when there are outliers in the data set, MLL works better than LL.

The possible reason why RE is bigger for the larger sample size is that we can get more accurate estimation of the asymptotically optimal bandwidths for the larger sample size. However, even for small sample size like 50, our new method MLL still works better than LL, which also shows that the proposed bandwidth selection method works well for small sample sizes, too. If the sample size is too small, such as 20, both of the two methods perform almost the same. In this situation, modal local linear regression had trouble to detect outliers and thus selected the very large bandwidth $h_2$ in most cases.

**Example 3:** Generate the independent and identically distributed (i.i.d.) data $\{(x_i, y_i), i = 1, \ldots, n\}$ from the model

$$Y_i = 2\sin(2\pi X_i) + \epsilon_i,$$
where $X_i \sim U(0, 1)$ and $\epsilon_i$ are $t_6$ distributed.

Now the error distribution is the $t_6$ distribution. A $t$ distribution with degree of freedom from 4 to 6 is generally used to represent the heavy tailed distribution.

Figure 2.11 is the plot of the mean squared error (MSE) of 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL when sample size is 250. From the graph, we can see that MLL had a smaller MSE than LL for all the grid points.

Table 2.4 is the summary of the average of MSE on 46 grid points (equal space from 0.05 to 0.95 based on 500 replications) for LL and MLL for different sample size $n$. From the Table 2.4, we can also see that MLL had smaller MSE than LL across all the sample sizes. The relative efficiency between MLL and LL is about 1.0421 for small sample size and 1.1273 for large sample size, which is quite close to the theoretical value, 1.1281, reported on the Table 2.1. If we use $t_4$ as the error distribution, the relative efficiency will be 1.3167 for large sample size. Hence when the error distribution is heavy tailed distribution, MLL works better than LL.

**Real Data Application:** Education Expenditure Data (Chatterjee and Price, 1977): There are 50 observations from 50 states, one for each state. The two variables we are
Table 2.4.
Summary of MSE for LL and MLL for Example 3 with $R = 500$

<table>
<thead>
<tr>
<th>n</th>
<th>LL</th>
<th>MLL</th>
<th>RE</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.1893</td>
<td>0.1817</td>
<td>1.0421</td>
</tr>
<tr>
<td>100</td>
<td>0.1028</td>
<td>0.0949</td>
<td>1.0834</td>
</tr>
<tr>
<td>250</td>
<td>0.0442</td>
<td>0.0394</td>
<td>1.1224</td>
</tr>
<tr>
<td>500</td>
<td>0.0242</td>
<td>0.0215</td>
<td>1.1273</td>
</tr>
</tbody>
</table>

interested are

X: Number of residents per thousand residing in urban areas in 1970

Y: Per capita expenditure on public education in a state, projected for 1975.

Our interest is to find the relationship between $Y$ and $X$ without any parametric assumption. Here we use two methods to fit the data. One is local linear regression, the other is modal local linear regression.

Figure (2.12) is the scatter plot of original observations and the fitted regression curves by the local linear regression and the modal local linear regression. From the Figure (2.12), we can see there is an extreme observation, also called outlier in statistics. This extreme observation is from Hawaii, which has very high per capita expenditure on public education. This observation created the big difference between the two fitted curves around $x = 500$. The observations with $x$ around 500 appear go down in that area. Thus the regression function should also go down. The modal local linear did reflect this fact. However the local linear fit went up in that area. It is due to the big impact of the extreme observation from Hawaii. In fact this extreme observation received around 10% weight for the local linear fit at $x = 500$, compared to nearly 0% weight for the modal local linear fit. The modal local linear fit automatically assigns very small weight to the
observations which are away from the fitted regression curve. Hence, modal local linear regression is more robust to outliers than local linear regression.
Fig. 2.1. Plots of local linear fit and modal local linear fit with outliers.
Fig. 2.2. Plot $R(h)$ vs. $h$
Fig. 2.3. Plot $R(h)$ vs. $h$ for small $h$
Fig. 2.4. Plot $R(h)$ vs. $h$ for method 2
Fig. 2.5. Plot of $R(h)$ vs. $h$ for method 1 with different bandwidths
Fig. 2.6. Plot of $R(h)$ vs. $h$ for modified method 1 with different bandwidths
Fig. 2.7. Plot of $R(h)$ vs. $h$ when the error is $t_6$. 
Fig. 2.8. Plot of fitted curve for Example 1 with n=250
Fig. 2.9. MSE of the estimates for LL and MR for Example 1 when n=250
Fig. 2.10. MSE of the estimates by LL and MR for Example 2 with \( n=250 \)
Fig. 2.11. MSE of the estimates for Example 3 with n=250
Fig. 2.12. Plots of fitted regression curves for Education Expenditure Data.
Chapter 3

Modal Linear Regression

From the last chapter, we can see that modal local polynomial regression works better than local polynomial regression for heavy tailed error and almost as well as local polynomial regression for normal distributed error distribution. By replacing the $L_2$ loss with the Gaussian kernel for local polynomial regression, modal local polynomial regression can be adaptive to different error distributions and robust to the outliers. This idea can be extended to other models which use the least squares criterion. In this chapter we introduce how to extend the idea of modal local polynomial regression to linear regression to create an adaptive robust linear regression we call “modal linear regression”. Similar to the properties of modal local polynomial regression, modal linear regression is as efficient as linear regression when the error term is normally distributed. When the error distribution has a heavy tail or there are outliers in the data set, modal linear regression is more efficient than linear regression.

3.1 Introduction of Modal Linear Regression

Linear regression is one of the simplest regression method. However the least squares criterion makes this method sensitive to outliers and the heavy tailed error distribution. The general method to get a robust linear regression is regression M-estimators, which replace the least squares criterion with some outlier resistant function,
such as $L_1$ loss, Huber’s $\psi$-function (Huber 1981) $\psi_c(t) = \rho'(t) = \max\{-c, \min(c, t)\}$, and Turkey’s bisquare function $\psi_c(t) = t\{1 - (t/c)^2\}^2$. See Huber (1973, 1981), Andrews (1974), Beaton and Tukey (1974), Holland and Welsch (1977), and Hampel (1986). Generally, these robust methods will be more efficient than linear regression when there are outliers or the error distribution has a heavy tail. However, they will lose some efficiency when there are no outliers or the error distribution is normal. In this chapter, we extend the idea of modal local polynomial to linear regression to create a new robust linear regression — “modal linear regression”, which achieves the robustness without sacrificing efficiency.

Suppose the data $(x_i, y_i)$ are independent and identically sampled from the model

$$Y = \alpha + X^T\beta + \epsilon$$

where $E(\epsilon) = 0, \Var(\epsilon \mid X) = \sigma^2$, $X$ and $\epsilon$ are independent and $X$ is $p$-dimension vector. We denote the density of $\epsilon$ by $g(t)$. Denote $z_i = (1, x_i^T)^T$ and $\theta = (\alpha, \beta^T)^T$.

The most commonly used and simplest method to estimate the parameter $\theta$ is the least squares estimation

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} (y_i - z_i^T \theta)^2.$$
Similar to modal local polynomial regression, we propose to estimate the unknown regression parameter $\beta$ by maximizing the following objective function

$$Q_h(\theta) = \frac{1}{n} \sum_{i=1}^{n} \phi_h(y_i - z_i^T \theta), \quad (3.1)$$

where $\phi_h(t) = h^{-1}\phi(t/h)$ and $\phi(t)$ is standard normal density function. This method is called “modal linear regression”. Note that for $h$ large, the criterion $Q_h(\theta)$ becomes nearly equivalent to least squares, as we will demonstrate soon. However, for small $h$, we have something like a kernel density estimator, only for the regression parameters. In fact, as $h$ becomes small, there are many modes to the objective (3.1), corresponding to all the regression parameters $\theta$ that can be generated using least squares on a subset of $p + 1$ data points $(x_i, y_i)$.

To find the modes of $Q(\theta)$, notice that the above objective function has a mixture form. Thus we can extend the Modal EM algorithm (Li, Ray and Lindsay, 2007) to find $\theta$. Specifically, in the $(k+1)^{\text{th}}$ step of EM algorithm:

**E Step:** Set

$$\pi(j \mid \theta^{(k)}) = \frac{\phi_h(y_j - z_j^T \theta^{(k)})}{\sum_{i=1}^{n} \phi_h(y_i - z_i^T \theta^{(k)})} \propto \phi_h(y_j - z_j^T \theta^{(k)})$$

**M Step:** Find $\theta^{(k+1)}$ by maximizing the following objective function

$$\sum_{j=1}^{n} \left[ \pi(j \mid \theta^{(k)}) \log \phi_h(y_j - z_j^T \theta) \right]$$
Remark 3.1.1. Denote $X = (z_1, \ldots, z_n)^T$ and $Y = (y_1, \ldots, y_n)$. Notice that $\phi(\cdot)$ is Gaussian density. The M step is simplified to

$$\theta^{(k+1)} = \arg \min_{\theta} \sum_{j=1}^{n} \left[ \pi(j \mid \theta^{(k)})(y_j - z_j^T \theta)^2 \right],$$

hence

$$\theta^{(k+1)} = (X^T W X)^{-1} X^T W Y,$$

where $W = \text{diag}\{\pi(j \mid \theta^{(k)})\}$. If $h \to \infty$, then $\pi(j \mid \theta^{(k)}) \to 1/n$ and so modal linear regression converges to linear regression in this situation. So linear regression is the limiting case of modal linear regression.

In order to give the asymptotic properties for the modal linear estimator, we need the following assumptions.

Assumption 3.1.1.

(a). $X^T X/n = A + o(1)$, where $A = E(ZZ^T)$ is a positive symmetric matrix.

(b). $E(||Z||^3) < \infty$.

(c). The error density $g(t)$ is symmetric about the mean 0 with the unique mode at 0.

(d). $E[\phi''(\epsilon/h) < 0]$ for any $h > 0$, where the error $\epsilon$ has the density $g(t)$.

From Assumption 3.1.1 (a), we can know that $A$ is the limit of $X^T X/n$ and $X^T X/n = O(1/n)$. Assumption 3.1.1 (b) requires the predictor has finite third moment. Assumption 3.1.1 (c) and (d) of $g(t)$ are also used by modal local polynomial regression.
Theorem 3.1.1. Under the Assumption 3.1.1, we have the following three results.

(1). With probability approaching to 1 (wpa), there exists a local maximizer $\hat{\theta}$ of (3.1) such that $\|\hat{\theta} - \theta_0\| = O_p(\sqrt{1/n})$.

Let $\hat{\theta}$ be the found consistent local maximizer of (3.1) in (1).

(2). The asymptotic conditional bias of $\hat{\theta}$ is $O_p(1/n)$. The asymptotic conditional variance is

$$\text{Cov}(\hat{\theta} \mid X) = hG(h)F(h)^{-2}(X^TX)^{-1} + o_p(1/n).$$  \hspace{1cm} (3.2)

(3). $\sqrt{n}(\hat{\theta} - \theta_0) \xrightarrow{L} N(0, hG(h)F(h)^{-2}A^{-1})$, where $A$ is the limit of $X^TX/n$.

Proof: 1.) Let $X = (z_1, \ldots, z_n)^T$ and $Y = (y_1, \ldots, y_n)$. Denote

$$W_n \triangleq \frac{\partial Q_h(\theta_0)}{\partial \theta} = -\frac{1}{nh^2} \sum_{i=1}^n \phi' \left( \frac{y_i - z_i^T \theta_0}{h} \right) z_i$$  \hspace{1cm} (3.3)

$$\Delta_n \triangleq \frac{\partial^2 Q_h(\theta_0)}{\partial \theta \partial \theta^T} = \frac{1}{nh^3} \sum_{i=1}^n \phi'' \left( \frac{y_i - z_i^T \theta_0}{h} \right) z_i z_i^T,$$  \hspace{1cm} (3.4)

where $Q_h(\theta)$ is defined in (3.1), $\theta_0$ is the true value, and $\phi^{(j)}(t)$ is the $j^{th}$ derivative of $\phi(t)$ with $\phi'(t) = (-t)\phi(t), \phi''(t) = (t^2 - 1)\phi(t)$, and $\phi^{(3)}(t) = (3t - t^3)\phi(t)$.

Noticing that

$$\int \left( \frac{y - z^T \theta}{h} \right)^j \phi^k \left( \frac{y - z^T \theta}{h} \right) g(y - z^T \theta) dy = h \int t^j \phi^k(t) g(th) dt.$$  \hspace{1cm} (3.5)
Using Assumption 3.1.1 (a, c), by some simple calculations, we have

\[ E(\Delta_n | X) = \frac{1}{nh^2} F(h) \sum_{i=1}^{n} z_i z_i^T = \frac{1}{nh^2} F(h) X^T X, \quad (3.6) \]

\[ \text{Var}(\Delta_n | X) = O_p\left(\frac{1}{n}\right) \quad (3.7) \]

\[ E(W_n | X) = 0 \quad (3.8) \]

\[ \text{Cov}(W_n | X) = \frac{1}{n^2 h^3} G(h) \sum_{i=1}^{n} z_i z_i^T = \frac{1}{n^2 h^3} G(h) X^T X \quad (3.9) \]

where \( F(h) = \int (t^2 - 1) \phi(t) g(th) dt \) and \( G(h) = \int t^2 \phi^2(t) g(th) dt \). By default, when we calculate the variance of a matrix, we find the variance of each element of the matrix. Using the result \( X = E(X) + O_p(\sqrt{\text{Var}(X)}) \), we have

\[ \Delta_n = \frac{1}{nh^2} F(h) X^T X + O_p(1/\sqrt{n}). \quad (3.10) \]

Notice that

\[ Q_h(\theta_0 + n^{-1/2} \mu) - Q_h(\theta_0) = n^{-1/2} W_n^T \mu + \frac{1}{2n} \mu^T \Delta_n \mu - \frac{1}{6nh^{5/2}} \sum_{i=1}^{n} \phi^{(3)}(y_i - z_i^T \theta^*) (z_i^T \mu)^3 \]

\[ = A + B + C, \quad (3.11) \]

where \( ||u|| = c \) and \( ||\theta^* - \theta_0|| \leq cn^{-1/2} \). The constant \( c \) will be defined later. From (3.8), (3.9), and the Assumption (3.1.1) (a), we get \( W_n = O_p(n^{-1/2}) \) and hence \( A = O_p(n^{-1}) \).

Based on (3.10),

\[ B = \frac{1}{2nh^2} F(h) \mu^T A \mu (1 + o_p(1)). \]
Since $\phi^{(3)}(\cdot)$ is bounded, from the Assumption (3.1.1) (b), we have

$$C = n^{-3/2}O_p(E||Z||^3) = O_p(n^{-3/2}) = o_p(n^{-1}).$$

We can choose $c$ big enough, such that the second term $B$ dominates the other two terms in (3.11) wpa1. From (2.12), we know $F(h) < 0$. So $Q_h(\theta_0 + n^{-1/2}\mu) - Q_h(\theta_0) < 0$ wpa1 with $||u|| = c$. Hence, wpa1, there exists a local maximizer $\hat{\theta}$ such that $||\hat{\theta} - \theta_0|| \leq cn^{-1/2}$.

2.) Since $\hat{\theta}$ is the consistent solution to $\partial Q_h(\theta)/\partial \theta$, by the Taylor expansion, we have

$$0 = \frac{\partial Q_h(\hat{\theta})}{\partial \theta} = W_n + (\Delta_n + J_n)(\hat{\theta} - \theta_0), \tag{3.12}$$

where

$$J_n = -\frac{1}{2nh^4} \sum_{i=1}^{n} \left[ \phi^{(3)}(Y_i - \theta^{\ast}Tz_i)\frac{(\hat{\theta} - \theta_0)^Tz_i}{h}z_i z_i^T \right],$$

where $||\theta^\ast - \theta_0|| \leq ||\hat{\theta} - \theta_0||$. Since $\phi^{(3)}(\cdot)$ and $E(||Z||^3)$ is bounded, we can easily check that $J_n = O_p(||\hat{\theta} - \theta_0||) = O_p(1/\sqrt{n})$. From (3.10) and (3.12), noticing $W_n = O_p(1/\sqrt{n})$, we have

$$\hat{\theta} - \theta_0 = (\Delta_n + J_n)^{-1}W_n = \left[ \frac{1}{nh^2}F(h)X^TX \right]^{-1}W_n + O_p(1/n). \tag{3.13}$$
So, we can easily get

\[
\text{Bias}\{\hat{\theta} \mid X\} = O_p(1/n),
\]

\[
\text{Cov}\{\hat{\theta} \mid X\} = \left(\frac{1}{nh^2} F(h)X^T X\right)^{-1} \left(\frac{1}{n^2h^3} G(h)X^T X\right) \left(\frac{1}{nh^2} F(h)X^T X\right)^{-1} + o_p(1/n)
\]

\[
= hG(h)F(h)^{-2}(X^T X)^{-1} + o_p(1/n).
\]

3.) From (3.13), we know

\[
\hat{\theta} - \theta_0 = \left[\frac{1}{nh^2} F(h)X^T X\right]^{-1} W_n + O_p(1/n) = h^2F(h)^{-1}A^{-1}W_n + o_p(1/\sqrt{n}) \quad (3.14)
\]

Let \(W_n^* = \sqrt{n}W_n\). Now we prove asymptotic normality for \(W_n^*\). For any unit vector \(d \in \mathbb{R}^{p+1}\), we prove

\[
\{d^T \text{Cov}(W_n^*)d\}^{-\frac{1}{2}} \{d^T W_n^* - d^T E(W_n^*)\} \overset{L}{\to} N(0, 1)
\]

Let

\[
\xi_i = -\frac{1}{\sqrt{nh^2}} \varphi\left(\frac{Y_i - z_i^T \hat{\theta}_0}{h}\right)d^T Z_i.
\]

Then \(d^T W_n^* = \sum_{i=1}^n \xi_i\). We check the Lyapunov's condition. Based on the results (3.9), we know

\[
\text{Cov}(W_n) = \frac{1}{nh^3} G(h)X^T X = \frac{1}{nh^3} G(h)A + o(1/n).
\]
Hence \( \text{Var}(d^T W_n^*) = nd^T \text{Cov}(W_n)d = O(1) \). So we only need to prove \( nE|\xi_1|^3 \to 0 \).

Notice that \( (d^T Z)^2 \leq \|Z\|^2\|d\|^2 = \|Z\|^2 \) and \( \phi'(\cdot) \) is bounded.

\[
nE|\xi_1|^3 \leq O(\sqrt{n}^{-3/2})E(\|Z\|^3) = O(n^{-1/2}) \to 0.
\]

So asymptotic normality for \( W_n^* \) holds. From (3.14) and Slutsky’s theorem, we have

\[
\sqrt{n}(\hat{\theta} - \theta_0) \overset{L}{\to} N(0, hG(h)F(h)^{-2}A^{-1}) . \quad \square
\]

Notice that the order of the square of asymptotic conditional bias is \( 1/n^2 \) and the order of the asymptotic conditional variance is \( 1/n \). Compared to the order of conditional variance, the asymptotic conditional bias is negligible. Thus we can consider the asymptotic bias is 0 and focus on the asymptotic variance.

For linear regression, the variance of \( \hat{\theta} \) is \( \sigma^2(X^TX)^{-1} \), where \( \sigma^2 \) is the variance of the error density \( g(t) \). Similar to modal local polynomial regression, define \( R(h) = hG(h)F(h)^{-2}\sigma^{-2} \). Then the ratio of the asymptotic mean squared error (MSE) of modal linear regression to that of linear regression is \( R(h) \). If \( R(h) < 1 \), modal linear regression will have smaller MSE of the regression parameter estimates than linear regression. Based on Theorem 2.3.3, we know \( \inf_h R(h) \leq 1 \). Hence modal linear regression works at least as well as linear regression. When the error is normally distributed, the optimal \( h \) should go to infinity, hence strictly speaking, modal local linear regression can only work arbitrary close to linear regression in this situation.
In order to apply modal linear regression, we need to select the bandwidth $h$ first. We propose to select the bandwidth $h$ by minimizing the asymptotic mean squared error of $\hat{\theta}$. Then we get the following theorem.

**Theorem 3.1.2.** Under the Assumption 3.1.1, the asymptotically optimal bandwidth $h$ for the modal linear regression which minimizes the asymptotic mean squared error of $\hat{\theta}$ is

$$h_{\text{opt}} = \arg\min_h hG(h)F(h)^{-2}.$$ 

Proof: Based on the Theorem 3.1.1, we know the asymptotic bias of $\hat{\theta}$ is negligible compared to its asymptotic variance. Thus we select the bandwidth $h$ by minimizing the asymptotic variance of $\hat{\theta}$. From Theorem 3.1.1, we know

$$\text{Cov}\{\hat{\beta} \mid X\} = hG(h)F(h)^{-2}(X^T X)^{-1} + o_p(1/n).$$

So minimizing the asymptotic variance of $\hat{\theta}$ is equivalent to minimizing $hG(h)F(h)^{-2}$. Hence the asymptotically optimal bandwidth $h$ which minimizes the asymptotic mean squared error of $\hat{\theta}$ is

$$h_{\text{opt}} = \arg\min_h hG(h)F(h)^{-2}. \quad \square$$

In practice we can use the similar method for modal local polynomial regression to estimate $G(h)$ and $H(h)$ and use the grid search method to find the asymptotically optimal $h$. Specifically we estimate $g(t)$ first and then input it to the function $F(h)$ and $G(h)$. To estimate $g(t)$, we first get the estimates of the error term $\hat{\epsilon}_i = y_i - \hat{\beta}_i^T \hat{\theta}$ by fitting the data using the linear regression or any simple robust linear regression method.
Then we estimate \( g(t) \) by the kernel density estimator

\[
\hat{g}(t) = \frac{1}{n} \sum_{i=1}^{n} K_d(\hat{\epsilon}_i - t).
\]

If \( K(\cdot) \) is the Gaussian kernel, we can get the following simplified formulas of \( \hat{F}(h) \) and \( \hat{G}(h) \),

\[
\hat{F}(h) = \frac{1}{\sqrt{2\pi nd}} \sum_{i=1}^{n} \exp \left\{ -\frac{\epsilon_i^2}{2(h^2 + d^2)} \right\} \frac{h^2 d^2 - h^4 d - h^2 d^3}{(h^2 + d^2)^{5/2}},
\]

\[
\hat{G}(h) = \frac{1}{2\pi nd} \sum_{i=1}^{n} \exp \left\{ -\frac{\epsilon_i^2}{h^2 + 2d^2} \right\} \frac{(h^2 + 2d^2)d^3 + h^2 d^2}{(h^2 + 2d^2)^{5/2}}.
\]

We first estimate \( H(h) = hG(h)F(h)^{-2} \) for some grid points of \( h \) such as \( h = 0.5\hat{\sigma} \times 1.1^j, j = 0, \ldots, g \), where \( \hat{\sigma}^2 \) is the sample variance of \( \hat{\epsilon}_i, i = 1, \ldots, n \). Then we choose \( \hat{h}_{\text{opt}} \) which minimizes \( \hat{H}(h) = hG(h)\hat{F}(h)^{-2} \).

Now let us see the relative efficiency between modal linear regression and linear regression.

**Theorem 3.1.3.** Suppose the variance of the error density \( g(t) \) is \( \sigma^2 \). Under the same condition of Theorem 3.1.1, the asymptotic relative efficiency between the modal linear regression with the bandwidth \( h \) and linear regression is \( R(h)^{-1} \), where \( R(h) = hG(h)F(h)^{-2}\sigma^{-2} \), with \( G(h) = \int t^2 \phi^2(t) g(t) dt \) and \( F(h) = \int (t^2 - 1) \phi(t) g(th) dt \).

Proof: The mean squared error of the regression parameter estimate by the linear regression is \( \sigma^2 (X^TX)^{-1} \). From the Theorem 3.1.1, we know the asymptotic mean square
error of the regression parameter estimate by modal linear regression is

\[ hG(h)F(h)^{-2}(X^TX)^{-1}. \]

Thus the asymptotic relative efficiency between modal linear regression and linear regression is

\[ \frac{\sigma^2(X^TX)^{-1}}{hG(h)F(h)^{-2}(X^TX)^{-1}} = \frac{\sigma^2}{hG(h)F(h)^{-2}} = R(h)^{-1}, \]

where \( R(h) = hG(h)F(h)^{-2}\sigma^2 \), with \( G(h) = \int t^2\phi^2(t)g(t)dt \) and \( F(h) = \int (t^2 - 1)\phi(t)g(th)dt \). □

We can easily know the above theorem also holds for the unconditional situation since \( R(h) \) does not involve \( X \). By default, we refer the modal linear regression to the one with the asymptotically optimal bandwidth chosen by Theorem 3.1.2. From the Theorem 2.3.3, we know

\[ R(h_{2opt}) \leq 1 \Rightarrow R(h_{2opt})^{-1} \geq 1. \]

Thus, the asymptotic relative efficiency between modal linear regression and linear regression is always bigger than or equal to 1. Specifically, \( R(h_{2opt})^{-1} > 1 \) if the error distribution has a heavy tail and \( R(h_{2opt})^{-1} = 1 \) (arbitrary close to 1, strictly speaking) if the error distribution is normal. Notice that the asymptotic relative efficiency between modal local polynomial regression and local polynomial regression is \( R(h_{2opt})^{-4/5} \). So for the same error distribution, the asymptotic relative efficiency between modal linear
regression and linear regression is $R(h_{2opt})^{-1/5}(\geq 1)$ times the asymptotic relative efficiency between modal local linear regression and local linear regression. Hence, modal linear regression improves more over linear regression than modal local linear regression over local linear regression.

Table 3.1 are the asymptotic relative efficiency between modal linear regression (MLR) and linear regression (LR) for some special error distributions.

<table>
<thead>
<tr>
<th>Error Distribution</th>
<th>Relative Efficiency</th>
<th>$t_6$</th>
<th>$t_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0.95N(0,1) + 0.05N(0,3^2)$</td>
<td>1.2227</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$0.95N(0,1) + 0.05N(0,4^2)$</td>
<td>1.5189</td>
<td>1.1626</td>
<td>1.4105</td>
</tr>
</tbody>
</table>

Based on Theorem 3.1.3, we know that in order to get the same accuracy of the estimators in terms of the asymptotic MSE, the sample size needed for the linear regression is $R(h_{2opt})^{-1} (\geq 1)$ times the sample size needed for the modal linear regression. From the Table 3.1, we know that in order to get the same accuracy of the estimators in terms of the asymptotic MSE, the ratios of the sample size needed for linear regression to the sample size needed for modal linear regression is 1.2227, 1.5189, 1.1626, and 1.4105 for the error distributions $0.95N(0,1) + 0.05N(0,3^2)$, $0.95N(0,1) + 0.05N(0,4^2)$, $t_6$, and $t_4$, respectively.
3.2 Simulation Study and Real Data Application

In this section, we will use three simulation examples and a real data application to compare model linear regression (MLR) with linear regression (LR) and to assess the finite sample performance of modal linear regression.

In the simulation examples, we will use three different error distributions to show how modal linear regression works. For comparison, we also report the maximum likelihood estimator (MLE) assuming the true error distribution is known. Thus MLE will be the most efficient estimator. To compare different regression methods, we will report the mean square error (MSE) of the regression intercept and slope estimates of the method LR, MLR, and MLE. To better see how much the improvement of the new method MLR, we also report the relative efficiency between MLR and LR, denoted by REMLR, and the relative efficiency between MLE and LR, denoted by REMLE. All above values are based on 10000 replications for different sample size $n$ of the data.

Example 1: Generate the independent and identically distributed (i.i.d.) data\[ \{(x_i, y_i), i = 1, \ldots, n\} \] from the model

\[ Y_i = 1 + 3X_i + \epsilon_i, \]

where $X_i \sim U(0, 1)$ and $\epsilon_i \sim N(0, 1)$.

The two methods we used to estimate the regression parameters are: the linear regression using the least squares (LR) and modal linear regression (MLR).
In this example, the error distribution is normal distribution. In this case, the linear regression estimator by least squares criterion is the maximum likelihood estimator and thus the most efficient estimator. Table 3.2 is the summary of the MSE of the intercept estimates for LR and MLR. REMLR is the relative efficiency between MLR and LR, defined by $\text{MSE}(\text{LR})/\text{MSE}(\text{MLR})$, for different sample size $n$ based on 10000 replications. Table 3.3 is the summary of the MSE of the slope estimates for LR and MLR and the relative efficiency (REMLR) between MLR and LR for different sample size $n$ based on 10000 replications ($R = 10000$).

**Table 3.2.**
Summary of MSE of the intercept estimates for Example 1 with $R = 10000$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>REMLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.2213</td>
<td>0.2242</td>
<td>0.9871</td>
</tr>
<tr>
<td>50</td>
<td>0.0843</td>
<td>0.0853</td>
<td>0.9888</td>
</tr>
<tr>
<td>100</td>
<td>0.0412</td>
<td>0.0415</td>
<td>0.9943</td>
</tr>
<tr>
<td>250</td>
<td>0.0159</td>
<td>0.0160</td>
<td>0.9964</td>
</tr>
<tr>
<td>500</td>
<td>0.0080</td>
<td>0.0080</td>
<td>0.9977</td>
</tr>
</tbody>
</table>

From the Tables 3.2 and 3.3, we can see the MSE of the regression parameter estimates for LR and MLR are almost the same and the relative efficiency is quite close to 1. When the error distribution is normal distribution, we know the linear regression is the most efficient method. From this example, we can see our proposed robust method — modal linear regression — works as efficiently as linear regression when the error is normally distributed.
Table 3.3.
Summary of MSE of the slope estimates for Example 1 with $R = 10000$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>RE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.6695</td>
<td>0.6794</td>
<td>0.9854</td>
</tr>
<tr>
<td>50</td>
<td>0.2518</td>
<td>0.2546</td>
<td>0.9888</td>
</tr>
<tr>
<td>100</td>
<td>0.1244</td>
<td>0.1249</td>
<td>0.9965</td>
</tr>
<tr>
<td>250</td>
<td>0.0476</td>
<td>0.0478</td>
<td>0.9956</td>
</tr>
<tr>
<td>500</td>
<td>0.0237</td>
<td>0.0237</td>
<td>0.9989</td>
</tr>
</tbody>
</table>

The computation of MLR is very fast. For example, to estimate the regression parameters when sample size is 500, it only took about 0.04 second, using the personal desktop with Pentium 4 CPU 2.40GHz.

**Example 2:** Generate the independent and identically distributed (i.i.d.) data \{$(x_i, y_i), i = 1, \ldots, n$\} from the model

$$Y_i = 1 + 3X_i + \epsilon_i,$$

where $X_i \sim U(0, 1)$ and $\epsilon_i \sim 0.95N(0, 1) + 0.05N(0, 3^2)$.

Now the error distribution is a mixture of normal distribution with 95% data from $N(0, 1)$ and 5% data from $N(0, 3^2)$. The 5% data from $N(0, 3^2)$ are most likely to be outliers.

Table 3.4 is the summary of the MSE of the intercept estimates by LR, MLR, and MLE for different sample size $n$. REMLR is the relative efficiency between MLR and LR and REMLE is the relative efficiency between MLE and LR. Table 3.5 is the
summary of the MSE of the regression slope estimates and the relative efficiency for different estimators. All the reported values are based on 10000 replications.

Table 3.4.
Summary of MSE of the intercept estimates for Example 2 with $R = 10000$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.3113</td>
<td>0.2637</td>
<td>0.2533</td>
<td>1.1804</td>
<td>1.2289</td>
</tr>
<tr>
<td>50</td>
<td>0.1183</td>
<td>0.0980</td>
<td>0.0953</td>
<td>1.2066</td>
<td>1.2413</td>
</tr>
<tr>
<td>100</td>
<td>0.0574</td>
<td>0.0471</td>
<td>0.0463</td>
<td>1.2199</td>
<td>1.2400</td>
</tr>
<tr>
<td>250</td>
<td>0.0222</td>
<td>0.0183</td>
<td>0.0182</td>
<td>1.2150</td>
<td>1.2256</td>
</tr>
<tr>
<td>500</td>
<td>0.0110</td>
<td>0.0089</td>
<td>0.0089</td>
<td>1.2304</td>
<td>1.2420</td>
</tr>
</tbody>
</table>

From the Table 3.4 and Table 3.5, we can see MLR estimator had a smaller MSE for the regression parameter estimates than LL across all the sample sizes from 20 to 500. The relative efficiency between MLR and LL is 1.18 for small sample size 20 and around 1.23 for large sample size, which is consistent with the theoretical value reported in Table 3.1. In addition, we can see MLR had almost the same efficiency as MLE for large sample size. For small sample size, MLE worked a little better than MLR, since for finite sample size MLE gained some efficiency by directly using the true parameter values, missing proportions and component variance in this example, of the error distribution. So MLR is more efficient for estimating the regression parameters than LL and is nearly the optimal estimator in this example when the error has a contaminated normal distribution.
Table 3.5.
Summary of MSE of the slope estimates for Example 2 with $R = 10000$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.9414</td>
<td>0.7963</td>
<td>0.7633</td>
<td>1.1822</td>
<td>1.2334</td>
</tr>
<tr>
<td>50</td>
<td>0.3563</td>
<td>0.2946</td>
<td>0.2874</td>
<td>1.2092</td>
<td>1.2398</td>
</tr>
<tr>
<td>100</td>
<td>0.1710</td>
<td>0.1408</td>
<td>0.1386</td>
<td>1.2142</td>
<td>1.2335</td>
</tr>
<tr>
<td>250</td>
<td>0.0662</td>
<td>0.0544</td>
<td>0.0538</td>
<td>1.2172</td>
<td>1.2294</td>
</tr>
<tr>
<td>500</td>
<td>0.0330</td>
<td>0.0268</td>
<td>0.0265</td>
<td>1.2323</td>
<td>1.2439</td>
</tr>
</tbody>
</table>

Example 3: Generate the independent and identically distributed (i.i.d.) data

$\{(x_i, y_i), i = 1, \ldots, n\}$ from the model

$$Y_i = 1 + 3X_i + \epsilon_i,$$

where $X_i \sim U(0, 1)$ and $\epsilon_i \sim t_6$.

Now the error distribution is the $t_6$ distribution. We use the $t$ distribution to represent the heavy tailed distribution. Table 3.6 is the summary of MSE and the relative efficiency for different estimators of regression intercept. Table 3.7 is the summary of MSE of regression slope estimates and the relative efficiency for different estimators.

Table 3.6.
Summary of MSE of the intercept estimates for Example 3 with $R = 10000$.

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.3240</td>
<td>0.3000</td>
<td>0.2869</td>
<td>1.0798</td>
<td>1.1291</td>
</tr>
<tr>
<td>50</td>
<td>0.1212</td>
<td>0.1096</td>
<td>0.1061</td>
<td>1.1062</td>
<td>1.1423</td>
</tr>
<tr>
<td>100</td>
<td>0.0612</td>
<td>0.0546</td>
<td>0.0530</td>
<td>1.1215</td>
<td>1.1542</td>
</tr>
<tr>
<td>250</td>
<td>0.0237</td>
<td>0.0206</td>
<td>0.0203</td>
<td>1.1530</td>
<td>1.1680</td>
</tr>
<tr>
<td>500</td>
<td>0.0119</td>
<td>0.0102</td>
<td>0.0102</td>
<td>1.1589</td>
<td>1.1678</td>
</tr>
</tbody>
</table>
Table 3.7.
Summary of MSE of the slope estimates for Example 3 with $R = 10000$.

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.9964</td>
<td>0.9281</td>
<td>0.8867</td>
<td>1.0736</td>
<td>1.1236</td>
</tr>
<tr>
<td>50</td>
<td>0.3653</td>
<td>0.3304</td>
<td>0.3178</td>
<td>1.1055</td>
<td>1.1493</td>
</tr>
<tr>
<td>100</td>
<td>0.1816</td>
<td>0.1616</td>
<td>0.1573</td>
<td>1.1238</td>
<td>1.1545</td>
</tr>
<tr>
<td>250</td>
<td>0.0708</td>
<td>0.0619</td>
<td>0.0609</td>
<td>1.1444</td>
<td>1.1634</td>
</tr>
<tr>
<td>500</td>
<td>0.0357</td>
<td>0.0309</td>
<td>0.0306</td>
<td>1.1557</td>
<td>1.1666</td>
</tr>
</tbody>
</table>

From the Table 3.6 and Table 3.7, we can see MLR estimator had smaller MSE of the intercept and slope estimates than LL across all the sample sizes from 20 to 500. The relative efficiency between MLR and LL is around 1.08 for small sample size and 1.16 for large sample size, which is quite close to the theoretical value reported in Table 3.1. In addition, MLR had almost the same efficiency as MLE for large sample size. So MLR is more efficient for estimating the regression parameters than LL and is nearly the optimal one in this example when the error distribution is $t_6$.

**Example 4:** Generate the independent and identically distributed (i.i.d.) data \{$(x_i, y_i), i = 1, \ldots, n$\} from the model

$$Y_i = 1 + 3X_i + \epsilon_i,$$

where $X_i \sim U(0, 1)$ and $\epsilon_i$ has a Laplace(0, 1) distribution (also called double exponential distribution).

In this case, median regression estimator is the MLE. Table 3.8 is the summary of MSE and the relative efficiency for different estimators of regression intercept.
3.9 is the summary of MSE of regression slope estimates and the relative efficiency for different estimators.

Table 3.8.
Summary of MSE of the intercept estimates for Example 4 with $R = 10000.$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.4367</td>
<td>0.3766</td>
<td>0.3756</td>
<td>1.1597</td>
<td>1.1626</td>
</tr>
<tr>
<td>50</td>
<td>0.1657</td>
<td>0.1304</td>
<td>0.1164</td>
<td>1.2706</td>
<td>1.4233</td>
</tr>
<tr>
<td>100</td>
<td>0.0812</td>
<td>0.0613</td>
<td>0.0519</td>
<td>1.3240</td>
<td>1.1564</td>
</tr>
<tr>
<td>250</td>
<td>0.0314</td>
<td>0.0225</td>
<td>0.0186</td>
<td>1.3975</td>
<td>1.6878</td>
</tr>
<tr>
<td>500</td>
<td>0.0157</td>
<td>0.0109</td>
<td>0.0088</td>
<td>1.4405</td>
<td>1.7895</td>
</tr>
</tbody>
</table>

From the Table 3.8 and Table 3.9, we can see MLR had smaller MSE of the regression parameter estimates than LL across all the sample sizes from 20 to 500. The relative efficiency between MLR and LL is around 1.16 for small sample size and 1.44 for large sample size. So MLR is more efficient than LL for estimating the regression parameters. For double exponential error distribution, MLE (median regression) worked better than MLR except when sample size is too small (20, for example).

In fact the double exponential density function does not have the first derivative around 0, thus violates the assumption for the asymptotic results of MLR. The selected bandwidth based on the original asymptotic results will not be the optimal one. That is the reason why MLE worked better than MLR with the original bandwidth selection method. We have found empirically that there do exist values that make MLR nearly as efficient as median regression for this example.
Table 3.9.
Summary of MSE of the slope estimates for Example 4 with $R = 10000.$

<table>
<thead>
<tr>
<th>n</th>
<th>LR</th>
<th>MLR</th>
<th>MLE</th>
<th>REMLR</th>
<th>REMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1.3185</td>
<td>1.1397</td>
<td>1.1562</td>
<td>1.1568</td>
<td>1.1404</td>
</tr>
<tr>
<td>50</td>
<td>0.4892</td>
<td>0.3876</td>
<td>0.3591</td>
<td>1.2621</td>
<td>1.3622</td>
</tr>
<tr>
<td>100</td>
<td>0.2462</td>
<td>0.1855</td>
<td>0.1581</td>
<td>1.3274</td>
<td>1.1569</td>
</tr>
<tr>
<td>250</td>
<td>0.0946</td>
<td>0.0676</td>
<td>0.0562</td>
<td>1.3989</td>
<td>1.6833</td>
</tr>
<tr>
<td>500</td>
<td>0.0470</td>
<td>0.0327</td>
<td>0.0265</td>
<td>1.4376</td>
<td>1.7736</td>
</tr>
</tbody>
</table>

Now, let us also apply the modal local linear to the Education Expenditure Data (Chatterjee and Price, 1977).

**Real Data Application:** Education Expenditure Data (Chatterjee and Price, 1977):

There are 50 observations from 50 states, one for each state. The two variables we are interested are

X: Number of residents per thousand residing in urban areas in 1970

Y: Per capita expenditure on public education in a state, projected for 1975.

Our aim is to find the relationship between $Y$ and $X$ assuming a linear form for the regression function. Here we use two methods to fit the data. One is the least squares, the other is the modal linear regression. For comparison, we also add the linear regression fit for the data without the outlier collected from Hawaii.

Figure 3.1 is the plot of regression fit for Expenditure Data using the linear regression, modal linear regression, and the linear regression without the outlier. The solid line is the linear regression fit, the dashed line is the modal linear fit, and the dash dotted line is the linear regression fit without the outlier. The observation from Hawaii state is an extreme observation. This outlier made the linear regression fit and
the modal linear regression fit different. The linear regression fit was pulled towards the outlier. However, the modal linear regression fit was not affected too much by the outlier. From the figure, we can see the modal linear regression fit is almost the same as the linear regression fit without the outlier. Hence the modal linear regression automatically considered the observation from Hawaii as an outlier and put a very small weight for this observation. Specifically, this extreme observation only got 0.16% weight for the modal linear regression fit compared to the average of 2.04% weight for all other observations.

Table 3.10 is the summary of the regression parameter estimates by different methods. From the table, we can see that there is a big difference of the regression parameter estimates between LR with and without the outlier. Hence the outlier from the Hawaii state had a big impact on the linear regression fit and thus the linear regression fit is sensitive to the outlier. From the table, we can also see that the modal linear regression estimates are almost the same as the linear regression estimates without the outlier. Hence, the modal linear regression is robust to the outlier.

Table 3.10.
Summary of the linear coefficient estimators.

<table>
<thead>
<tr>
<th></th>
<th>LR</th>
<th>MLR</th>
<th>LR without the outlier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>194.9740</td>
<td>157.9424</td>
<td>155.7074</td>
</tr>
<tr>
<td>Slope</td>
<td>0.1363</td>
<td>0.1831</td>
<td>0.1868</td>
</tr>
</tbody>
</table>
Fig. 3.1. Regression Fit for Expenditure Data assuming linear regression.
Chapter 4

Summary and Future Work

Local polynomial regression is one of the most popular smoothing methods. However the least squares criterion make this method sensitive to outliers and heavy tailed error distributions. Existing robust methods, which replace the $L_2$ loss with the outlier resistent loss such as $L_1$ loss, Huber’s $\psi$-function (Huber 1981), and Turkey’s bisquare function, will lose some efficiency when there are no outliers or the error distribution is normal distribution. In order to achieve both robustness and efficiency, we propose the new adaptive robust smoothing method we call “modal local polynomial regression”. Modal local polynomial uses a double kernel in the objective function, which is motivated by (Scott, 1992). The locations of the maxima of the objective function can be interpreted as the conditional modes of a kernel density estimator based on the data. With one more tuning parameter, asymptotically, the modal local polynomial fit produces smaller mean squared error (MSE) than the local polynomial fit when there are outliers or the error distribution has a heavy tail. When the error distribution is the Gaussian distribution, modal local polynomial still works at least as well as local polynomial regression. Thus this new method achieves the robustness without sacrificing efficiency and can be adaptive to different error distributions. Modal local polynomial regression can be considered as an improved version of local polynomial regression. Based on the asymptotic bias and variance analysis, we successfully find a way to select the bandwidth
in practice. From the simulation studies, we can see the bandwidth selection method works quite well, even for small sample size. The whole theory of modal local polynomial can be easily extended to the high dimensional case for the predictors. It is hard to implement though, due to the “curse of dimensionality”.

The idea of the modal local polynomial can be also easily extended to linear regression to create an adaptive robust linear regression we call “modal linear regression”. Similar to the properties of modal local polynomial regression, modal linear regression works more efficiently than linear regression when there are outliers or the error distribution has a heavy tail. It works as well as linear regression when the error distribution is normally distributed. Moreover, modal linear regression estimators are nearly optimal for certain class of error distributions based on the simulation studies. Hence modal linear regression achieves robustness without sacrificing the efficiency, unlike other existing robust linear regression.

In the future work, we will try to extend the idea of the modal local polynomial regression to other models, such as spline smoothing, the semi-parametric model, the partial linear model, and the varying coefficient model. In addition, we also want to apply the modal local polynomial regression to the dependent data and the discontinuous regression function. Finally, we will try to use the modal local polynomial regression to estimate mixture of regressions.
Part II

Label Switching for Bayesian Mixtures
Chapter 1

Introduction

Label switching is a well known problem for finite mixtures. It arises from the invariance of the likelihood under relabelling of the mixture components. For Bayesian mixtures, if the prior is symmetric for the component parameters, the posterior distribution is also invariant under relabelling of the mixture components. Label switching is one of the fundamental problems for Bayesian mixture analysis. In this chapter, we will first review mixture models, then describe how the label switching occurs in finite mixture models, and finally, review currently used methods to solve the label switching for Bayesian mixtures.

1.1 Review of Mixture Models

Mixture models have experienced increased interest over last decades. Mixture models can be used for cluster analysis, latent class analysis, discriminant analysis, image analysis, survival analysis, disease mapping, meta analysis, and more. They provide extremely flexible descriptive models for distributions in data analysis and inference. For a general introduction to mixture models, see Lindsay (1995), Böhning (1999) and McLachlan and Peel (2000).

Finite mixture models are most commonly used for mixture models. They are natural models for unobserved population heterogeneity and are generally applicable
when one samples from a population which consists of several homogeneous subpopulation. The homogeneous subpopulations will be called components of the population. Let \((X_1, \ldots, X_n)\) be the sample from this population and \(J_i\) be the index number of component to which \(X_i\) belongs, also called the latent component label. The number of components will generally be denoted by \(m\), which can be known or unknown. Denote the density function of \(j\)th component by \(f(x; \lambda_j), j = 1, \ldots, m\), where \(\lambda_j\) is a component specific parameter, which can be scalar or vector. For simple notation, \(f(\cdot)\) can be the density of discrete or continuous random variables and generally is called component density. Hence we have

\[
p(x_i|J_i = j) = f(x_i; \lambda_j) \quad (i = 1, \ldots, n; j = 1, \ldots, m).
\]

where \(p(\cdot)\) is the general notation for density function of both discrete and continuous function. The proportion of the total population that is in the \(j\)th component will be denoted by \(\pi_j\) and called the component weight. Thus we have

\[
P(J_i = j) = \pi_j \quad (i = 1, \ldots, n; j = 1, \ldots, m),
\]

where \(0 \leq \pi_j \leq 1, \ j = 1, \ldots, m\) and \(\sum_{j=1}^{m} \pi_j = 1\).

If we could observe the latent labels \((J_i, i = 1, \ldots, n)\), then the variables \((X_i, J_i), i = 1, \ldots, n\) are a random sample from the joint density of the form

\[
p(x, j) = p(x \mid J = j)P(J = j) = f(x; \lambda_j)\pi_j.
\]
If the component labels \((J_1, \ldots, J_n)\) are missing, we only observe \((X_1, \ldots, X_n)\) from the marginal density of \(X\), which is the mixture density

\[
p(x; \theta) = \sum_{j=1}^{m} p(x \mid J = j) P(J = j) = \sum_{j=1}^{m} \pi_j f(x; \lambda_j),
\]

(1.1)

where \(\theta = (\lambda_1, \ldots, \lambda_m, \pi_1, \ldots, \pi_m)\). A family of distributions having density (1.1) are generally called a \(m\)-component finite mixture model. When \(m = 1\), there is just one component and \(p(\cdot)\) will be called unicomponent mixture density.

To estimate the unknown parameters \(\theta = (\lambda_1, \ldots, \lambda_m, \pi_1, \ldots, \pi_m)\) in (1.1), we generally use the maximum likelihood estimator (MLE): Find \(\hat{\theta} = (\hat{\lambda}_1, \ldots, \hat{\lambda}_m, \hat{\pi}_1, \ldots, \hat{\pi}_m)\) to maximize

\[
L(\theta; x) = \prod_{i=1}^{n} \sum_{j=1}^{m} \pi_j f(x_i; \lambda_j).
\]

(1.2)

Feng and McCulloch (1996) and Cheng and Liu (2001) prove the consistency of the MLE of the finite mixture models. There is no explicit formula for the MLE, but generally we can use EM algorithm (Dempster et al., 1977) to find the MLE for finite mixture models. The EM algorithm is a general-purpose algorithm to iterative computation of maximum-likelihood estimates when the observations can be viewed as the incomplete data.

In general, suppose the observed data \(x = (x_1, \ldots, x_n)\) are from \(g(x \mid \theta)\). The MLE of \(\theta\) is \(\hat{\theta} = \arg \max_{\theta} l(\theta)\), where \(l(\theta) = \sum_{i=1}^{n} \log g(x_i \mid \theta)\). Sometime, it is hard to find \(\hat{\theta}\) when there is no closed form. Suppose for the complete data \(y = (x, z)\) with the log likelihood \(l_c(\theta)\), there is a simple or closed form of the ML estimator, where
$z = (z_1, \ldots, z_n)$ are the unobserved (or missing) data. We can use the EM algorithm to find the $\hat{\theta}$. The EM algorithm is performed as follows:

1. Start with initial parameter values $\theta^{(0)}$.

2. Expectation Step (E step): at the $(k + 1)^{th}$ step, compute the conditional expectation of the complete-data log-likelihood $l_c(\theta)$ given the observed data $x$, using current estimate $\theta^{(k)}$

$$Q(\theta, \theta^{(k)}) = E_{\theta^{(k)}}(l_c(\theta) | x).$$

3. Maximization Step (M step): determine the new estimate $\theta^{(k+1)}$ as the maximizer of $Q(\theta, \theta^{(k)})$ over $\theta$

$$\theta^{(k+1)} = \arg\max_{\theta} Q(\theta, \theta^{(k)}).$$

4. Iterate steps 2 and 3 until some convergence criterion is attained.

There are a variety of methods available for choosing the initial values. One natural way is to choose the initial values by other estimation methods, such as the $K$-means (MacQueen, 1967) and the moment method (Furman and Lindsay, 1994 and Lindsay and Basak, 1993). Böhning (1994) proposed to use well-separated initial values. In their experience, the algorithm could then converge faster. Böhning (1999) also proposed an initial partition of the data by maximizing the within sum of squares criterion. The initial parameter values can be easily calculated based on the initial partition. We can also just start from a random point or the best of several random points (for each initial values, the likelihood was calculated and the set with the largest likelihood was considered as the best and was used as the starting point). For other
methods, see McLachlan (1988), Finch et al. (1989), and Seidel (2000). Karlis and Xekalaki (2002) provided a good review and comparison of different methods to choose initial values.

Notice that if $l_c(\theta)$ is a linear function of the missing data $z$, then in the E step we can simply calculate the conditional expectation of the missing data $z$ and input the expected value to $l_c(\theta)$.

Several convergence criteria have been proposed. Lindstrom and Bates (1988) proposed to stop the algorithm based on the relative change of the parameters and/or of the loglikelihood. These criteria only indicate lack of progress rather than actual convergence. Böhning et al. (1994) and Pilla et al. (2001) proposed to use Aitken’s acceleration scheme as the stopping rule. The gradient function (Lindsay, 1983, 1995; Pilla and Lindsay, 2001) can be also used to check if we have found the MLE.

Dempster et al. (1977) showed that the (incomplete data) log likelihood function $l(\theta)$ is not decreased after an EM iteration; that is, $l(\hat{\theta}^{(k+1)}) \geq l(\hat{\theta}^{(k)})$ for $k = 0, 1, 2, \ldots$. More generally, this nice property holds as long as we increase the objective function in M step, that is $Q(\hat{\theta}^{(k+1)}, \hat{\theta}^{(k)}) > Q(\hat{\theta}^{(k)}, \hat{\theta}^{(k)})$. This is the idea of Generalized-EM (GEM) algorithm (Dempster et al., 1977). Convergence to a critical point of the likelihood occurs under mild regularity conditions (Wu, 1983). In practice, the EM algorithm has been observed to be very slow in some applications. As noted in Dempster et al. (1977), the convergence rate of EM is linear and governed by the fraction of missing information.

For finite mixture models, we can define the missing data $(Z_{ij}, i = 1, \ldots, n, j = 1, \ldots, m)$, where $Z_{ij} = 1$ if the $i^{th}$ observation $x_i$ is from the $j^{th}$ component and 0
otherwise. Then the complete-data log-likelihood is

$$l_c(\theta) = \sum_{i=1}^{n} \sum_{j=1}^{m} z_{ij} \log(\pi_j f(x_i; \lambda_j)),$$

where $\theta = (\lambda_1, \ldots, \lambda_m, \pi_1, \ldots, \pi_m)$. For example, if $x_i$ is from $m$-component normal mixture model with $\lambda_j = (\mu_j, \sigma_j^2)$, then the EM algorithm is as follows:

1. Take initial guesses of the parameters, say $\hat{\mu}_j^{(0)}, \hat{\sigma}_j^{2(0)}, \hat{\pi}_j^{(0)}, j = 1, \ldots, m$.

2. E Step: at the $(k+1)^{th}$ step, compute the classification probability that the observation $x_i$ comes from component $j$ based on the current estimates,

$$\hat{p}_{ij}^{(k+1)} = E(Z_{ij} | x, \theta^{(k)}) = \frac{\hat{\pi}_j^{(k)} f(x_i; \hat{\lambda}_j^{(k)})}{\sum_{l=1}^{m} \hat{\pi}_l^{(k)} f(x_i; \hat{\lambda}_l^{(k)})}, i = 1, \ldots, n, j = 1, \ldots, m$$

3. M step: compute the component means, variance and the mixing probability:

$$\hat{\mu}_j^{(k+1)} = \frac{\sum_{i=1}^{n} \hat{p}_{ij}^{(k+1)} x_i}{\sum_{i=1}^{n} \hat{p}_{ij}^{(k+1)}}$$

$$\hat{\sigma}_j^{2(k+1)} = \frac{\sum_{i=1}^{n} \hat{p}_{ij}^{(k+1)} (x_i - \hat{\mu}_j^{(k+1)})^2}{\sum_{i=1}^{n} \hat{p}_{ij}^{(k+1)}}$$

$$\hat{\pi}_j^{(k+1)} = \frac{\sum_{i=1}^{n} \hat{p}_{ij}^{(k+1)}}{n}$$

(1.5)

for $j = 1, \ldots, m$.

4. Iterating steps 2 and 3 until convergence.
Notice that the complete likelihood (1.4) is the linear function of the missing data $z$. Hence in E step, we only need to simply calculate the conditional expectation of $Z_{ij}$.

The normal mixture model with unequal variance for different component has an unbounded likelihood function. When the variance of one component goes to zero and the corresponding mean is equal to any observation, the likelihood value will go to infinity. In this case, our interest will be the consistent local maximizer instead of the global maximizer. In order to avoid the unboundness, one can maximize the likelihood function $L(\theta)$ in (1.2) over the constrained parameter space

$$\Omega_C = \{\theta \in \Omega : \sigma_h^2/\sigma_j^2 \geq c > 0, 1 \leq h \neq j \leq m\},$$

where $\Omega$ denotes the unconstrained parameter space. Hathaway (1985) showed that the global maximizer $\hat{\theta}$ of $L(\theta)$ over $\Omega_C$ exists and that $\hat{\theta}$ is strongly consistent for the true value $\theta$ provided that the true value of $\theta$ lies in $\Omega_C$.

For the univariate normal distribution, in order to incorporate the constraint (1.6), Hathaway (1983,1986) and Bezdak, Hathaway, and Huggins (1985) propose a constrained version of the EM algorithm. For multivariate normal mixture with unequal covariance matrix, $\Sigma_i (i = 1 \ldots, m)$, the likelihood function is also unbounded. Similarly to the univariate case, we can put some constraints on the covariance matrix to get the constrained global maximizer. For example, we constrain all the eigenvalues of $\Sigma_h \Sigma_j^{-1} (1 \leq h \neq j \leq m)$ to be greater than or equal to some minimum value $C > 0$ or $|\Sigma_h|/|\Sigma_j| \geq C > 0 (1 \leq h \neq j \leq m)$. 
If we assume that the component variance $\sigma_j^2$ are equal, the likelihood function is bounded and the formula (1.5) in M step is replaced by

$$\hat{\sigma}^{2(k+1)} = \frac{\sum_i^n \sum_{j=1}^m \hat{p}_{ij}^{(k+1)} (x_i - \hat{\mu}_j^{(k+1)})^2}{n}. $$

In the finite mixture models, the unknown parameters are

$$\begin{pmatrix}
\pi_1 & \cdots & \pi_m \\
\lambda_1 & \cdots & \lambda_m
\end{pmatrix}, \quad (1.7)$$

each column corresponding to a component. The weights satisfy

$$0 \leq \pi_j \leq 1, \ j = 1, \ldots, m \quad \text{and} \quad \sum_{j=1}^m \pi_j = 1.$$ 

Define the latent variables $\Phi_i, \ i = 1 \cdots, n$ to be the values of the parameter $\lambda$ for sample $X_i, \ i = 1 \cdots, n$. We can consider $\Phi_i$ to be a random sample from the discrete probability measure $Q$ that put mass $\pi_j$ at the support point $\lambda_j$; that is

$$P(\Phi = \lambda_j) = Q(\{\lambda_j\}) = \pi_j.$$ 

Thus we can equate the set of unknown parameters (1.7) uniquely with a discrete probability measure $Q$ on the parameter space for $\lambda$, with $m$ points of support $\{\lambda_1, \cdots, \lambda_m\}$ and corresponding masses $\{\pi_1, \cdots, \pi_m\}$. So estimating the unknown parameters in (1.7) is the same as finding an unknown distribution $Q$. Therefore, we will replace (1.7) by
$Q$ in the notation. The mixture density is denoted by $p(x; Q)$. The distribution $Q$ is usually called the mixing distribution or latent distribution.

There is a natural extension of the finite mixture models by allowing the mixing distribution $Q$ to be continuous. In such a situation, the mixture density becomes

$$p(x; Q) = \int f(x; \lambda) dQ(\lambda).$$

Many other models can be considered as the above mixture models, such as random effects models, repeated measures models, latent class and latent trait models, missing covariates and data etc. See Lindsay (1995) for details.

Estimation of $Q$ is done conventionally by maximum likelihood, that is we find $\hat{Q}$ which maximizes log-likelihood

$$l(Q) = \sum_{i=1}^{n} \log p(x_i; Q). \quad (1.8)$$

If the distribution $Q$ is treated as completely unspecified as to whether it is discrete, continuous or in any particular family of distributions, then $\hat{Q}$ which maximizes (1.8), if it exists, is called nonparametric maximum likelihood estimator (NPMLE). Conditions for the existence of NPMLE were established by Kiefer and Wolfowitz (1956). Its computation and properties were defined by Laird (1978) and Lindsay (1983a, 1983b). Full discussions are given in Lindsay (1995) and Böhning (1999). The important function
for determining the properties of NPMLE is the \textit{gradient function}, defined as

\[
D_Q(\lambda) = \sum_{i=1}^{n} \left( \frac{p(x_i; \lambda)}{p(x_i; Q)} - 1 \right). \tag{1.9}
\]

We have the “fundamental theorem of nonparametric mixture maximum likelihood estimation” (Lindsay, 1983):

1. \textit{Existence and discreteness}. Under the condition\(^1\) that the components density \(f(x_i, \lambda)\) are both nonnegative and bounded as function of \(\lambda\), there \textit{exists} a maximum likelihood estimator \(\hat{Q}\) that is a \textit{discrete} distribution with the number of support points less than the number of distinct points in sample \(x_i, i = 1, \ldots, n\).

2. \textit{Gradient characterization}. \(\hat{Q}\) is NPMLE if and only if

\[
D_{\hat{Q}}(\lambda) \leq 0 \text{ for all } \lambda. \tag{1.10}
\]

3. \textit{Support point properties}. If \(\lambda\) is a support point for the \(\hat{Q}\) that maximize the likelihood, then \(D_{\hat{Q}}(\lambda) = 0\).

4. \textit{Uniqueness}. The fitted values of the likelihood, i.e. \((p(x_1, \hat{Q}), \ldots, p(x_n, \hat{Q}))\) are uniquely determined. That is, even if there were two distributions maximizing the likelihood, they would generate the same vector of likelihood fitted values.

A Bayesian approach to mixture analysis is now feasible by simulating from the posterior distribution with recently developed Markov Chain Monte Carlo (MCMC)

\(^1\)technically, there are another two assumptions, which generally don’t present a genuine difficulty. Please see Lindsay, 1995 for detail
methods (Tanner and Wong, 1987 and Gelfand and Smith, 1990). Important initial papers on Bayesian analysis of mixtures using MCMC methods include Diebolt and Robert (1994) and Escobar and West (1995). Following the work of Diebolt and Robert (1994), Bayesian mixture models could be applied routinely when the number of components $k$ is assumed known. When the number of components is unknown, Bayesian analysis via mixture models can be done using the methods of Escobar and West (1995, Dirichlet process mixtures), Mengersen and Robert (1996, distributional distances) and more recently developed methods by Richardson and Green (1997, reversible jump MCMC) and Stephens (2000a, birth-and-death MCMC).

1.2 Introduction to Label Switching for Finite Mixture Models

Supposing that \( \{x_1, \ldots, x_n\} \) is a random sample from the \( m \)-component mixture density, the likelihood for \( x = (x_1, \ldots, x_n) \) is

\[
L(\theta; x) = \prod_{i=1}^{n} \left\{ \pi_1 f(x_i; \lambda_1) + \pi_2 f(x_i; \lambda_2) + \cdots + \pi_m f(x_i; \lambda_m) \right\}.
\]

Denote the parameter vector by

\[
\theta = \left[ \begin{array}{c} \pi_1 \\ \lambda_1 \\ \vdots \\ \pi_m \\ \lambda_m \end{array} \right].
\] (1.11)
For any permutation $\sigma = \{\sigma(1), \ldots, \sigma(m)\}$ of the integers $\{1, \ldots, m\}$, define the corresponding permutation of the parameter vector $\theta$ by

$$
\theta^\sigma = \left[\left(\frac{\pi_1^\sigma}{\lambda_1^\sigma}\right), \ldots, \left(\frac{\pi_m^\sigma}{\lambda_m^\sigma}\right)\right] = \left[\left(\frac{\pi_{\sigma(1)}}{\lambda_{\sigma(1)}}\right), \ldots, \left(\frac{\pi_{\sigma(m)}}{\lambda_{\sigma(m)}}\right)\right].
$$

For any permutation $\sigma$, $L(\theta^\sigma; \mathbf{x})$ will be exactly the same as $L(\theta; \mathbf{x})$. Hence if $\hat{\theta}$ is the MLE, then $\hat{\theta}^\sigma$ is also the MLE for any permutation $\sigma$. In a technical sense, this means that the subscripts we assign to the $\pi$’s and $\lambda$’s are not identifiable unless we put additional restrictions on the model. That is, $\hat{\lambda}_1$ is not a meaningful symbol. However, there is a form of asymptotic identifiability that suggests that in large enough samples, we can meaningfully assign labels.

Suppose that the parameter space is the full product space $\Omega$. ($\pi$’s in the simplex, $\lambda$’s in cross product space). For any point $\theta$ in the full parameter space, define an equivalence class of points, call it $\text{EC}(\theta)$, that contains all parameter values $\theta'$ that generate the same distribution as $\theta$. In this space, there are two kinds of nonidentifiability. The first type, degenerate nonidentifiability, arises when $\theta$ corresponds to a mixing distribution with fewer than $m$ components (at least two of the $\lambda$’s are equal or one of the $\pi$’s is zero). These nonregular points create pathologies in $\Omega$. For example, suppose $k = 2$, and that the two component parameters $\lambda_1$ and $\lambda_2$ are set equal to $\lambda$. In this case, the weight parameters $\pi_1$ and $\pi_2$ are not identified, so that even though one permutes them, one gets exactly the same density. One representative for the equivalence class is the permutation symmetric $\theta = \left(\begin{smallmatrix} 5 \\ \lambda \end{smallmatrix}\right), \left(\begin{smallmatrix} 5 \\ \lambda \end{smallmatrix}\right)$, but the class also contains all the elements of the form $\left(\begin{smallmatrix} \pi_1 \\ \lambda \end{smallmatrix}\right), \left(\begin{smallmatrix} \pi_2 \\ \lambda \end{smallmatrix}\right)$, as well as all elements of the form $\left(\begin{smallmatrix} 1 \\ \lambda \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 \\ \lambda \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} 0 \\ \lambda \end{smallmatrix}\right), \left(\begin{smallmatrix} 1 \\ \lambda \end{smallmatrix}\right)$ for arbitrary
The second type, *labelling nonidentifiability*, arises because every relabelling of \( \theta \)'s columns yields exactly the same distribution. In a standard problem, for most values of \( \theta \), this will be the only nonidentifiability problem, and for them \( \text{EC}(\theta) \) will consist of the \( m! \) points \( \theta^\sigma \) that correspond to possible relabellings. We will say that \( \theta \) is *regular* if these are the only elements of \( \text{EC}(\theta) \).

Let us say that a subset \( S \) of \( \Omega \) is an *identifiable subset* if all its points are regular, and if for every \( \theta \in S \), we have \( \text{EC}(\theta) \cap S = \{ \theta \} \). If we restrict the parameters to lie in an identifiable subset, then the parameters are identifiable, as there is a one to one map from parameters to distributions. If we have an identifiable subset \( S \), then we can create image sets by permutation: \( S^\sigma = \{ \theta^\sigma : \theta \in S \} \). Each of the image sets is also identifiable and they are disjoint from each other. One can find a small neighborhood around every regular point that is identifiable, and so in this sense, the regular points are *locally identifiable*.

In a standard mixture problem, one says that estimator \( \hat{\theta}_n \) is *consistent* for true value \( \theta_\tau \) if there exists a sequence of labels \( \sigma_n \) such that \( \hat{\theta}^{\sigma_n}_n \to \theta_\tau \). That is, given any ball about \( \theta_\tau \), the probability that one element of \( \text{EC}(\hat{\theta}_n) \) is in that ball goes to one. It will be *consistent and asymptotically normal* if \( \sqrt{n}(\hat{\theta}^{\sigma_n}_n - \theta_\tau) \to_d N(0, \Sigma) \). In a standard mixture problem, these results hold for the regular MLE sequence, and are clearly related to the fact that the parameters are locally identifiable. Hence, when sample size is large enough, we can meaningfully assign labels, such that all the labelled \( \theta \) values should be close to the same labelled true value.

The label switching problem concerns about how we might choose a permutation \( \sigma \) so that we can view \( \hat{\theta}^\sigma \) as the estimate of the true parameter \( \theta_\tau \). In order to
compare different estimation methods, we might want to compare their bias, variance, mean square error, and more, using a simulation study. Given a sequence of estimates \( \hat{\theta}_1, \ldots, \hat{\theta}_n \), we must first determine which is the right permutation of \( \hat{\theta}_i, i = 1, \ldots, n \) to estimate \( \theta_\tau \), then we can use the labelled estimates to measure the uncertainty (such as standard error) of the corresponding estimation method. When we want to use bootstrap method to approximate, say, standard error of the estimates, we face similar problems.

Label switching also exists for Bayesian mixtures. Bayesian mixtures require the specification of a proper prior for the parameters of the mixture model. The inference is based on the simulated samples from the posterior distribution using Markov chain Monte Carlo (MCMC) methods, Gibbs sampler for example. If we do not have prior information that distinguishes between the components of the mixture (i.e. the prior distribution is symmetric for the component parameters), the posterior distribution will be symmetric and invariant to the permutation of the component parameters. The marginal posterior densities for the parameters will be identical for every component. Hence the posterior means of all the component specific parameters are the same for all components and are thus poor estimates of these parameters. A similar problem will occur when we try to estimate quantities relating to individual components of the mixture such as the predictive component densities and the marginal classification probabilities. However, even though these quantities are not well defined, there is an asymptotic sense in which labels can be assigned in large samples.

In Bayesian mixture analysis, after we get a sequence of simulated samples \((\theta_1, \ldots, \theta_N)\) from the posterior distribution of \(\theta\) using any MCMC sampling method, the label switching problem concerns about how we might find the “right” labels for \(\theta_t, t = 1, \cdots, N\).
Only after we solve the label switching problem, can we use the labelled Gibbs samples to do inference.

We can illustrate the label switching problem using the galaxy data (Roeder, 1990). The data set consists of the velocities (in thousands of kilometers per second) of 82 distant galaxies diverging from our own galaxy. They are sampled from six well-separated conic sections of the corona borealis. This data set has been analyzed by many researchers including Crawford (1994), Chib (1995), Carlin and Chib (1995), Escobar and West (1995), Phillips and Smith (1996), and Richardson and Green (1997). Stephens (2000b) also used this data set to explain the label switching problem. A histogram of the 82 data points is shown in Figure 1.1. We fit this data by 6-component normal mixture:

$$p(x; \pi, \mu, \sigma^2) = \pi_1 N(x; \mu_1, \sigma^2_1) + \ldots + \pi_6 N(x; \mu_6, \sigma^2_6)$$  \hspace{1cm} (1.12)

We use Gibbs sampling and the conjugate priors used by Richardson and Green (1997) to estimate the unknown parameters in (1.12). That is to assume

$$\pi \sim D(\delta, \delta, \ldots, \delta), \mu_i \sim N(\xi, \kappa^{-1}), \sigma^{-2}_j \sim \Gamma(\alpha, \beta).$$

where $D(\cdot)$ is Dirichlet distribution. Following the suggestion of Richardson and Green (1997), we let $\delta = 1, \xi$ equal the sample mean of the observations, $\kappa$ equal $1/R^2$, $\alpha = 2$, and $\beta = R^2/200$, where $R$ is the range of the observations.

Figure 1.2 is the plot of 10000 original Gibbs samples for 6 component means. From the plot, we can see there are distinct jumps in the traces of the means as the Gibbs samples moves between the permuted modal regions. Figure 1.3 is the plot of
estimated marginal posterior densities based on the original Gibbs samples. There are multiple modes for each marginal density. Theoretically, if we run the MCMC scheme for sufficiently long, the Gibbs samples should visit all the 6! modal regions and the marginal posterior densities of component parameters would be exactly the same as each other. That is how label switching happens for Bayesian analysis.

1.3 Review of Current Labelling Methods for Bayesian Mixtures

From the last section, we can see label switching is one of the fundamental problems of Bayesian mixtures. Many methods have been proposed to deal with label switching in Bayesian analysis. The easiest way to solve the label switching is to use an explicit parameter constraint (such as $\mu_1 < \mu_2 < \ldots < \mu_6$ or $\pi_1 < \pi_2 < \ldots < \pi_6$ for galaxy data example) so that only one permutation can satisfy it. This method is initially used by Diebolt and Robert (1994) and Dellaportas, et al (1996). Stephens (1997a) proved that inference conditional on an identifiability constraint can be performed when the constraint is imposed after the MCMC run. Such procedures are equivalent to changing the prior distribution. Frühwirth-Schnatter (2001) proposed to use an identifiability constraint on the MCMC output from the random permutation sampler. For random permutation sampler, at every iteration of an MCMC sampler, a Metropolis-Hastings move is used to propose a new permutation of the labels, which ensures the sampler visits all $m!$ symmetric modes. Celeux (1997), Celeux, Hurn and Robert (2000) and Stephens (1997a, 1997b, 2000) all expressed their concerns about imposing an identifiability constraint. One problem with identifiability constraint labelling is the choice of constraint, especially for multivariate problems. Different order constraint may generate markedly
different results as demonstrated by Celeux, Hurn and Robert (2000). Moreover it is
difficult to anticipate the overall effect. Many choices of identifiability constraint do not
completely remove the symmetry of the posterior distribution. As a result, label switching
problem may remain after imposing an identifiability constraint. See the example by

Two kinds of labelling methods have been proposed. The first kind we consider
use batch processing. That is after the samples have all been collected, one goes back
and assign optimal permutations \( (\theta_{1}^{\sigma}, \ldots, \theta_{N}^{\sigma}) \) so that the \( \theta \) values are clustered tightly
together. Stephens (2000b) and Celeux (1998) developed such a relabelling algorithm.
For a given loss function \( L: A \times \theta \rightarrow [0, \infty) \) such that

\[
L(a, \theta) = \min_{\sigma} L_{0}(a, \theta^{\sigma})
\]

for some \( L_0 : A \times \theta \rightarrow [0, \infty) \), the risk function is: \( R(a) = \int_{\theta} L(a, \theta)p(\theta | x) \, d\theta \), where
\( p(\theta | x) \) is the posterior distribution of \( \theta \). Our aim is to find \( a \) which minimizes \( R(a) \). It
is hard to compute \( R(a) \) directly, we can approximate it by Monte Carlo (MC) estimate.
Then the algorithm is as follows:
Starting with some initial values for \( (\sigma_{1}, \ldots, \sigma_{N}) \) (setting them all to the identity per-
mutation for example), iterate the following steps until a fixed point is reached.
Step 1: choose \( \hat{a} \) to minimize \( \sum_{t=1}^{N} L_{0}(\hat{a}, \theta_{t}^{\sigma_{t}}) \).
Step 2: for \( t = 1, \ldots, N \) choose \( \sigma_{t} \) to minimize \( L_{0}(\hat{a}, \theta_{t}^{\sigma_{t}}) \). □
Stephens (2000b) suggests a particular choice $L_0$

$$L_0(Q; \theta) = \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}(\theta^\sigma) \log \left[ \frac{p_{ij}(\theta^\sigma)}{q_{ij}} \right].$$

Here Stephens uses the Kullback-Liebler (KL) divergence to measure the loss of reporting $Q = \{q_{ij}\}_{1 \leq i \leq n, 1 \leq j \leq m}$ when the true classification probability is $\{p_{ij}(\theta)\}_{1 \leq i \leq n, 1 \leq j \leq m}$.

The above relabelling algorithm becomes the following algorithm, referred as KL algorithm.

Starting with some initial values for $(\sigma_1, \ldots, \sigma_N)$ (setting them all to the identity permutation for example),

Step 1: Choose $\hat{Q} = (\hat{q}_{ij})$ to minimize

$$\sum_{t=1}^{N} \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}(\theta^\sigma) \log \left[ \frac{p_{ij}(\theta^\sigma)}{\hat{q}_{ij}} \right].$$

Step 2: for $t = 1, \ldots, N$ choose $\sigma_t$ to minimize

$$\sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}(\theta^\sigma) \log \left[ \frac{p_{ij}(\theta^\sigma)}{\hat{q}_{ij}} \right],$$

where $p_{ij}(\theta^\sigma) = \frac{\pi_j^\sigma f(x_i; \lambda_i^\sigma)}{\sum_{l=1}^{m} \pi_l^\sigma f(x_i; \lambda_l^\sigma)}$ is the classification probability that the observation $x_i$ comes from the component $j$ based on the permutated parameter $\theta^\sigma$. □

Notice that the above algorithm requires $m!$ comparisons for each $t$ in step 2.

Hence the computation speed will be slow when $m$ is large. Marin, Mengersen and Robert (2005) proposed a method related to relabelling algorithm. They found the MAP
(maximum a posterior) estimate of the parameters based on all the MCMC samples and then label the samples by minimizing their distance to the MAP estimate. Celeux, Hurn and Robert (2000) and Hurn, Justel and Robert (2003) proposed to use label invariant loss functions—a decision theoretic procedure. They proposed to define a label invariant loss function for each quantity we want to estimate. Both of the relabelling algorithm and the labelling method using label invariant loss functions require batch processing of the Bayesian samples. For this kind of labelling method, we must do labelling for all the Bayesian samples simultaneously, which demands high storage requirements.

Chung, Loken, and Schafer (2004) proposed to use Data-dependent priors to solve the labelling problems. They proposed to break the symmetry of the prior and thus the posterior distribution by assigning one observation to each component. Jasra, Holmes and Stephens (2005) provide a good review about the existing methods to solve the label switching problem in Bayesian mixture modelling.
Fig. 1.1. Histogram plot of galaxy data. The number of bins used is 30.
Fig. 1.2. Plot of Gibbs samples of component means for Galaxy data.
Fig. 1.3. Plot of marginal posterior densities of component means based on original Gibbs samples.
Chapter 2

New Labelling Methods for Bayesian Mixtures

After the MCMC run, suppose we get the Bayesian samples \((\theta_1, \ldots, \theta_N)\) based on the observation \((x_1, \ldots, x_n)\). As explained before, there are \(m!\) symmetric major modes and locally identifiable modal regions. Now our aim is to find the “right” labels for the Bayesian samples \((\theta_1, \ldots, \theta_N)\). We will propose and assess five methods.

The first labelling method, denoted by PM(ECM), is based on the posterior modes and ECM algorithm (Meng and Rubin, 1993). The PM(ECM) labelling method automatically matches the labels in the highest posterior density (HPD) region, \(\{\theta : p(\theta) > c\}\), where \(p(\theta)\) is the posterior density. Because of this important nice property, this method will be our standard labelling method. A second method is based on the distance to posterior modes, an approximation to PM(ECM) that avoids iteration. The other three methods are based on loss functions. We will use simulation studies to evaluate the other labelling methods by comparing to our proposed standard labelling method – PMECM.

2.1 Ideal Labelling Methods Based on Asymptotics

We consider the implications of the asymptotic theory (Le Cam 1953, 1986) on the posterior. Suppose that the posterior density \(p(\theta)\) attains its maximum at a regular point \(\hat{\theta}\), with value \(p(\hat{\theta}) = d\). It will also attain this maximum at all points in \(\text{EC}(\hat{\theta})\), where \(\text{EC}(\hat{\theta})\) is the equivalent class of \(\hat{\theta}\) which contains all parameter values \(\theta'\) that generate
the same posterior distribution as \( \hat{\theta} \). More generally, if there are other local maxima at the regular points they must occur in entire sets of equivalence classes. Let us suppose that our goal is to build credible regions for the parameters, for any fixed credibility level \( 1 - \alpha \), using regions of highest posterior density (HPD). Such credible regions have the theoretical justification of being the smallest volume credible regions at a fixed level.

To be specific, let the regions have the form \( \psi_c = \{ \theta : p(\theta) \geq c \} \), where \( c = c_\alpha \) is chosen to give the target credibility level. From the proceeding asymptotic theory, one can say that for any fixed \( \alpha \), for \( n \) sufficiently large the HPD region will consist of \( m! \) disjoint sets \( S_j \). Each \( S_j \) will be an identifiable set, and will tend towards an elliptical shape around \( \theta^\sigma \), for some \( \sigma \). And each set \( S_j \) must be a permutation image \( S^\sigma_1 \) of \( S_1 \) for some \( \sigma \). In such a setting, one can use the identifiable set \( S_1 \) to describe the HPD region, as all others are just permuted images of \( S_1 \). The parameters have unique labels in \( S_1 \), so the HPD region can be described using labelled parameters. It is also reasonable to consider using asymptotic normality for the labelled parameters of \( S_1 \). In this sense we can consider the problem to be well-labelled at infinity.

Figure 2.1 is the plot of the two component means \((\mu_1, \mu_2)\) of 1000 Gibbs samples, assuming equal variance, and their permutation \((\mu_2, \mu_1)\), based on the 500 observations generated from \(0.5N(0,1)+0.5N(1.5,1)\). The pentagram is the true values for component means. Now the sample size of 500 is large. From Figure 2.1, we can see that either one of the clusters can be used to assign labels to each estimate as there appears to be no uncertainty about which cluster a sample is in.

Figure 2.2 is the plot of the two component means of 1000 Gibbs samples, assuming equal variance, and their permutations based on the 100 observations generated
from $0.5N(0,1) + 0.5N(1.5,1)$, which is the same as above. Now the sample size of 100 is not very large. From the Figure 2.2, we can see that the two permuted regions are connected together. Now there are not two well-separated clusters. One can define a boundary line separating two identifiable regions, but it is, in a sense, arbitrary.

We need to transfer the asymptotic labelling theory to finite $n$. For a given maximal mode $\hat{\theta}$, we define $S_c(\hat{\theta})$ to be the maximal connected subset of the HPD region $\psi_c$ that contains $\hat{\theta}$, where $\psi_c = \{\theta : p(\theta) \geq c\}$, with $c = c_\alpha$ chosen to give the target credibility level. We will call $S_c(\hat{\theta})$ the modal region defined by $c$ and $\hat{\theta}$. When $c = p(\hat{\theta})$, $\psi_c$ equals $EC(\hat{\theta})$, and $S_c(\hat{\theta})$ is the single point $\{\hat{\theta}\}$. As $c$ decreases, the size of the set $S_c(\hat{\theta})$ increases. Note also that $S^\sigma_c(\hat{\theta})$ is automatically the maximal connected subset that contains $\hat{\theta}^\sigma$, i.e. $S^\sigma_c(\hat{\theta}) = S^\sigma_c(\hat{\theta}^\sigma)$. In a smooth problem, as long as $c$ is sufficiently large that no other modes attain height $c$, the set $\psi_c$ will be the union of disjoint $S^\sigma_c(\hat{\theta})$ over all permutations $\sigma$. We will discuss how to find such $c$ later. If there are other relevant modes, each new equivalence class of modes will potentially generate other connected subsets, one around each new mode. For now, we assume that the posterior has just one equivalence class of regular modes.

If the modal region $S_c(\hat{\theta})$ is an identifiable subset, we can mimic the asymptotic theory and use $S_c(\hat{\theta})$, with its labels, to describe the HPD region of the posterior distribution, as the rest of the region is just permuted copies of $S_c(\hat{\theta})$.

In this case the identifiable HPD region gives, to all $\theta$ values it contains, a natural labelling from the local identifiability property. We will call these labels the HPD labels, and consider them to be the ideal labels, when they exist. We will judge our various methods based on their ability to replicate these ideal labels.
Unfortunately, there will always exist a value of $c$, say $c^*$, and corresponding credibility $(1 - a^*)$, such that the HPD region around one mode intersects the HPD region around another mode, thereby creating an HPD region that is not identifiable. For any point $\theta$ with $P(\theta) \leq c^*$, we will say that the labelling is HPD ambiguous.

Consider the following modal labelling method. Take any smooth algorithm that, for any initial value $\theta$, takes short steps in $\theta$ until it reaches a mode, while never decreasing the posterior. Say that any two points $\theta_1$ and $\theta_2$ have the same labels if they reach the same mode using this algorithm. This gives labels to all values of $\theta$ save those that iterate to a saddle point; the $\theta$ values converging to a saddle point become the boundary points of the labelled regions.

If started at a point with HPD labels, such an algorithm necessarily stays in the same identifiable region until it reaches the mode within that region. Thus if you label points by the mode they reach, these labels will exactly reconstruct the HPD labels for all $\theta$ with $p(\theta) > c^*$. Thus any such an algorithm matches the ideal method. However, different such algorithms will potentially differ on points with $p(\theta) \leq c^*$, the HPD ambiguous points.

In section 2.2.2, we will consider the calculation of $c^*$ and $(1 - a^*)$ as a way of determining which points have HPD labels, and what fraction of points are ideally labelled.

### 2.2 Introduction of Labelling Methods

In this section we start by introducing a modal algorithmic method that will reproduce the HPD labels when they exist. We will consider this our best method for
describing the HPD region, and so use it as a benchmark for comparison. We will also consider a computationally easier modal method for the purposes of seeing how well it captures the ideal HPD labels. We will also develop several new loss based methods with hopes that one or more can recapture the HPD labels well. In the process we will find one that is very effective based on the classification MLE. It is superior to the Kullback-Leibler loss method of Stevens (2000b).

2.2.1 Labelling Based on the ECM algorithm

We know there are $m!$ major modes for the posterior distribution of the $m$-component finite mixture models, due to the symmetry of their posterior distributions. In this subsection, we propose two methods to do labelling based on the posterior modes.

The posterior mode is generally very useful for Bayesian methods. It can be seen as an alternative point estimator to the general Bayesian estimator – the posterior mean. The posterior mode can also be a good starting value for the MCMC sampler. If we use the posterior mode as the initial value, we do not need to use a burn-in period. In the frequentist case, we often use the EM algorithm (Dempster, et al. 1977) to find the MLE. For Bayesian mixtures, it is difficult to directly use the EM algorithm to find the modes of the posterior distribution, since the objective function in the M step is very complicated. By combining the ideas of ECM (Meng and Rubin, 1993), a class of GEM algorithm (Dempster et al., 1977), and properties of Gibbs sampler, we proposed the following algorithm to find the posterior modes of Bayesian mixtures.
Algorithm 2.2.1. ECM Algorithm for Bayesian mixtures (ECM(BM))

Iterate the following two steps, given the initial parameter values, until a fixed point is reached.

**E-step:** Given the current estimates of the parameters, we find the conditional expectation of the log complete posterior distribution (missing variable is the latent $Z$ that indicate which component each observation is from). Notice that if the log complete posterior distribution is linear in the latent variable $Z$, which is the case for Gibbs sampler, then we only need to calculate the conditional expectation of $Z$, i.e the classification probability for each observation.

**M-step** For Gibbs sampler, we have the conditional posterior distribution for each parameter given all other parameters and missing data $Z$. In this M step, we estimate the parameters by maximizing the conditional posterior distribution (conditional on all other parameters, like Gibbs sampler, or on part of them, like data augmentation) with the latent variable $Z$ replaced by the classification probability calculated from E-step. □

From the theory of ECM (Meng and Rubin, 1993) and GEM (Dempster, et al. 1997), we know that by maximizing the conditional posterior distribution, we increase the conditional expectation of log complete posterior distribution and thus we increase the posterior distribution after each iteration based on the EM algorithm theory. Later we will see that the main advantage of our proposed standard labelling method PM(ECM) depends on this ascending property.

Let us use two examples to show how ECM(BM) algorithm works.
Example 1: Suppose \( \{x_1, \ldots, x_n\} \) come from a \( m \)-component normal mixture

\[
p(x; \pi, \mu, \sigma^2) = \pi_1 N(x; \mu_1, \sigma_1^2) + \pi_2 N(x; \mu_2, \sigma_2^2) + \cdots + \pi_m N(x; \mu_m, \sigma_m^2).
\]

Conjugate priors for normal densities have been considered by Diebolt and Robert (1994) and Bensmail et al. (1997). They are given by

\[
\pi \sim D(\delta_1, \delta_2, \ldots, \delta_m), \quad \mu_j \sim N(\xi_j, \sigma_j^2 / \gamma_j), \quad \sigma_j^{-2} \sim \Gamma(\alpha_j, \beta_j).
\]

where \( D(\cdot) \) is Dirichlet distribution. Let

\[
Z_{ij} = \begin{cases} 
1, & \text{if } x_i \text{ comes from the } j^{th} \text{ component;} \\
0, & \text{otherwise.}
\end{cases}
\]

Then we have the following conditional posterior distribution

\[
\pi \mid \cdots \sim D(\delta_1 + n_1, \ldots, \delta_m + n_m)
\]

\[
\mu_j \mid \cdots \sim N\left(\frac{n_j \overline{x}_j + \gamma_j \xi_j}{n_j + \gamma_j}, \frac{\sigma_j^2}{n_j + \gamma_j}\right)
\]

\[
\sigma_j^{-2} \mid \cdots \sim \Gamma\left(0.5 + 0.5n_j + \alpha_j, \beta_j + 0.5\gamma_j(\mu_j - \xi_j)^2 + 0.5\sum_{i=1}^n Z_{ij}(x_i - \mu_j)^2\right)
\]

\[
P(Z_{ij} = 1 \mid \cdots) \propto \pi_j N(x_i; \mu_j, \sigma_j^2),
\]

(2.1)

where \( n_j = \sum_{i=1}^n Z_{ij}, \overline{x}_j = \sum_{i=1}^n Z_{ij} x_i \), and ‘ \mid \cdots ’ denotes conditioning on all other parameters. Then the \((k + 1)^{th}\) step of ECM(BM) algorithm is as follows:
**E step:** Since the log complete posterior distribution is a linear function of the latent variable $Z$, we only need to find the conditional expectation of $Z$,

$$p_{ij}^{(k+1)} = E(Z_{ij} | \cdots) = \frac{\pi_j^{(k)} N(x_i; \mu_j^{(k)}, \sigma_j^{2(k)})}{\sum_{l=1}^{m} \pi_l^{(k)} N(x_i; \mu_l^{(k)}, \sigma_l^{2(k)})}.$$  

Denote $n_j^{(k+1)} = \sum_{i=1}^{n} p_{ij}^{(k+1)}$, $\bar{x}_j^{(k+1)} = \sum_{i=1}^{n} p_{ij}^{(k+1)} x_i$.

**M step:** Update the parameters by maximizing the conditional posterior distribution (2.1) with $Z_{ij}$ replaced by $p_{ij}^{(k+1)}$,

$$\pi_j^{(k+1)} = \frac{n_j^{(k+1)} - 1 + \delta_j}{n - m + \sum_{l=1}^{m} \delta_l}$$

$$\mu_j^{(k+1)} = \frac{n_j \bar{x}_j^{(k+1)} + \gamma_j \xi_j}{n_j + \gamma_j}$$

$$\sigma_j^{2(k+1)} = \frac{\beta_j + 0.5 \gamma_j (\mu_j^{(k+1)} - \xi_j)^2 + 0.5 \sum_{i=1}^{n} p_{ij}^{(k+1)} (x_i - \mu_j^{(k+1)})^2}{0.5 n_j^{(k+1)} + \alpha_j - 0.5}.$$

**Example 2:** (Example 1 continued)

Another popular conjugate priors for normal densities have been considered by Phillips and Smith (1996) and Richardson and Green (1997). They are given by

$$\pi \sim D(\delta, \delta, \cdots, \delta), \ \mu_i \sim N(\xi, \kappa^{-1}), \ \sigma_j^{-2} \sim \Gamma(\alpha, \beta).$$
Then we have the following conditional posterior distribution

\[
\begin{align*}
\pi \mid \cdots & \sim D(\delta + n_1, \ldots, \delta + n_m) \\
\mu_j \mid \cdots & \sim N \left( \frac{\sigma_j^{-2} n_j \bar{x}_j + \kappa \xi}{\sigma_j^{-2} n_j + \kappa}, \frac{1}{n_j \sigma_j^{-2} + \kappa} \right) \\
\sigma_j^{-2} \mid \cdots & \sim \Gamma \left( 0.5 n_j + \alpha, \beta + 0.5 \sum_{i=1}^{n} Z_{ij} (x_i - \mu_j)^2 \right), \\
P(Z_{ij} = 1 \mid \cdots) & \propto \pi_j \, N(x_i; \mu_j, \sigma_j^2) 
\end{align*}
\]

Then the \((k + 1)\)th step of ECM(BM) algorithm is as follows:

**E step:** Find the conditional expectation of \(Z\),

\[
p_{ij}^{(k+1)} = E(Z_{ij} \mid \cdots) = \frac{\pi_j^{(k)} N(x_i; \mu_j^{(k)}, \sigma_j^{2(k)})}{\sum_{l=1}^{m} \pi_l^{(k)} N(x_i; \mu_l^{(k)}, \sigma_l^{2(k)})}.
\]

Denote \(n_j^{(k+1)} = \sum_{i=1}^{n} p_{ij}^{(k+1)}\), \(\bar{x}_j^{(k+1)} = \sum_{i=1}^{n} p_{ij}^{(k+1)} x_i\).

**M step:** Update the parameters estimate,

\[
\begin{align*}
\pi_j^{(k+1)} &= \frac{n_j^{(k+1)} - 1 + \delta}{n - m + m \delta} \\
\mu_j^{(k+1)} &= \frac{\sigma_j^{-2(k+1)} n_j \bar{x}_j^{(k+1)} + \kappa \xi}{\sigma_j^{-2(k+1)} n_j + \kappa} \\
\sigma_j^{2(k+1)} &= \frac{\beta + 0.5 \sum_{i=1}^{n} p_{ij}^{(k+1)} (x_i - \mu_j^{(k+1)})^2}{0.5 n_j^{(k+1)} + \alpha - 1}.
\end{align*}
\]

\[(2.3)\]
Suppose one of the major mode is $\hat{\theta}$, which can be found by the Algorithm 2.2.1. The algorithm of our benchmark method to find the labels of $(\theta_1, \ldots, \theta_N)$ is as follows.

Algorithm 2.2.2. Labelling Based on Posterior Modes and the ECM(BM) algorithm (PM(ECM))

**Step 1:** Taking the Bayesian sample $\{\theta_t, t = 1, \ldots, N\}$ as the initial value, find the corresponding mode $\{m_t, t = 1, \ldots, N\}$ using ECM(BM) algorithm.

**Step 2:** Label $\{\theta_t, t = 1, \ldots, N\}$ by the associated converged mode $\{m_t, t = 1, \ldots, N\}$, i.e.

$$\sigma_t = \arg\min_{\sigma} (m_\sigma^\sigma - \hat{\theta})^T (m_\sigma^\sigma - \hat{\theta}).$$

The above method is to label the Bayesian samples by the associated converged modes based on Algorithm 2.2.1 (ECM(BM)). If there are only $m!$ major modes, $m_t$ will be one of the $m!$ modes. So the step 2 in Algorithm 2.2.2 is to find the label $\sigma_t$ that makes $m_\sigma^\sigma$ the same as the referenced major mode $\hat{\theta}$. Using the objective function in step 2 to decide the labels of $m_t$, we can relax the stopping rule for ECM(BM) when finding $\{m_t, t = 1, \ldots, N\}$.

### 2.2.2 Computing the HPD modal credibility $1 - \alpha^*$

PM(ECM) labelling has a nice explanation based on regions of highest posterior density (HPD). We know if $c$ is large enough, the modal region $S_c$ is an identifiable subset and thus we can mimic the asymptotic theory and use $S_c$, with its labels, to describe the entire HPD region of the posterior distribution. PM(ECM) will assign all elements of
$S_c$ to the single mode $\hat{\theta}$, and so the modal labels using PM(ECM) within $S_c$ agree with asymptotic labelling.

There remains the practical problem of determining the critical values $c$ for which $S_c$ is identifiable. As we will see, it will not be true for every $c$ because $\Omega$ contains nonregular points that create pathologies. Since the subsets $S_c$ are nested, one can let $c^*$ be the infimum of those $c$ for which $S_c$ is identifiable. We will then say the inference problem has labeling credibility $1 - \alpha^* = \Pr\{p(\theta) \geq c^*\}$.

If the labeling credibility is very high, say 95%, then one might feel comfortable saying that the modal labelling method is providing a close description of the HPD region, and so is “right”. If the labeling credibility is low, say 50%, then one might think any labelling algorithm is somewhat ad hoc in nature, and that sensible methods can reasonably disagree. The artificial separation by any labelling method will cause the bias effect of the parameter estimates. In this situation, the soft labelling, which assigns a probability of labelling for each Gibbs sample, may help reduce the bias effect. This is our future work.

As $c$ decreases, the region $S_c$ ceases to be identifiable either when it first includes a degenerate point $\theta$ with at least two same columns, or when it first intersects $S_\sigma^c$ for some $\sigma$. It is rather easy to determine the former. We just find the maximal posterior probability among all possible degenerate parameter values $\theta$ and call it $c^{**}$. These degenerate modes can be found by modifying the ECM(BM) algorithm. Since the maximal posterior probability increases as the number of components increases, we only need to find the maximal posterior probability among all possible degenerate parameter values $\theta$ with $m - 1$ number of components and use it for $c^{**}$. This degenerate mode can be
found by the ECM(BM) algorithm by setting the parameters of any two components to be equal.

As to the second case, when $c$ reaches a sufficiently small level, the maximal connected subsets $S_c$ and $S^r_c$ will intersect each other. In this case, the corresponding modes can themselves be connected by a continuous path. When the component level parameter is univariate, every such path must pass through a degenerate parameter value $\theta^*$, so in these cases, $c^{**}$ and $c^*$ are necessarily the same.

However, if the component parameters are multivariate, there are continuous paths between $\theta$ and $\theta^r$ that do not pass through a degenerate point. In this case, in order to rigorously determine $c^*$, one should find the continuous path between $\hat{\theta}$ and $\hat{\theta}^r$ whose minimal posterior value along the path is maximal among all paths. This is a difficult analytical question, but there exist pragmatic approximations to this number.

For simple notation, we suppose, temporarily, that the Gibbs values $\theta_j$ are already labelled by PM(ECM) with the corresponding mode $\hat{\theta}$. Plot

$$
    x_j = \frac{(\theta_j - \hat{\theta})^T(\hat{\theta}^r - \hat{\theta})}{||\hat{\theta}^r - \hat{\theta}||^2},
$$

versus $y_j = p(\theta_j)$, for any nonidentity permutation $\sigma$. For simple notation, in formula (2.4), we also use $\theta$ to represent the vector, not the matrix format defined in (1.11), which includes all the parameters. We will know the right format based on the content. For better view of the plot, we might also add the value of $x_j$ for $\theta^r_j$, which is the permuted $\theta_j$ with permutation $\sigma$. Notice that $x_j$ is the normalized projection of $\theta_j$ onto the line from the mode $\hat{\theta}$ and $\hat{\theta}^r$. The region of interest are the values of $x_j$ between 0 and
1, as these correspond to $\theta_j$ values that are intermediate to $\hat{\theta}$ and $\hat{\theta}^\sigma$ (by projection onto the line between them). Any continuous path between $\hat{\theta}$ and $\hat{\theta}^\sigma$ necessarily has $x$ values in the range 0 to 1. Consider continuous paths that pass through the $\theta_j$ points, and consider how one would construct a path whose minimal posterior value along the path was maximal. Pictorially, we would connect by line the points at the top of the scatter plot, from left to right, and the minimum of that line would be the maximal posterior value, denoted by $c^\star_\sigma$, along all paths between $\hat{\theta}$ and $\hat{\theta}^\sigma$. Then $c^\star$ would be the maximum of $c^\star_\sigma$ among all nonidentity permutations $\sigma$. Based on the simulation studies, the maximum of $c^\star_\sigma$ generally occurs for the permutations $\sigma$ which only exchange one pair of the identity permutation of $(1, 2, \ldots, m)$. Hence empirically, we can simply find $c^\star_\sigma$ for the permutations $\sigma$ which only exchange one pair of the identity permutation and choose the largest one. In addition, generally the found $c^\star$ is very close to $c^{**}$. Empirically, we may also use $c^{**}$ to estimate $c^\star$. The labelling credibility level of $1 - \alpha^\star$ is estimated by the proportions of the Gibbs samples with posterior density larger than $c^\star$.

If there are other minor modes except for the $m!$ major modes, the labelling based on the distance criterion in step 2 in Algorithm 2.2.2 works well in general but is not invariant to the scale transformation of the parameters. One solution is to use a loss-based method by restricting that $\theta_t$ has the same label as $m_t$. In some sense, now we are labelling the clusters. The number of labels we need to determine equals the number of different genuine modes of $(m_1, \ldots, m_N)$ and is very small compared to the number of the Gibbs samples. The genuine modes are the modes which do not include the permutated modes. So the labelling process is very fast after we get the modes sequence $(m_1, \ldots, m_N)$. 
From the above description, we can see that PM(ECM) labelling method creates a natural and intuitive partition of the parameter space into labelled regions. Using the new defined labelling credibility, we can see how well the Gibbs samples are labelled. Moreover, unlike the batch algorithm, PM(ECM) labelling method can do labelling along with the MCMC sampling process. Hence the storage requirements are reduced. However, one drawback of PM(ECM) labelling method is that its computation is relatively slow, compared to other labelling methods when \( m \) is not large, due to the slow convergence rate of ECM algorithm. We will use this method as our standard labelling method and use it to check how well the other labelling methods are.

2.2.3 Labelling Based on Modal Distance

We know the computation of PM(ECM) is slow due to the slow convergence rate of ECM algorithm. In order to avoid using the ECM algorithm starting from each Gibbs sample, we can just simply label Gibbs samples based on the distance to posterior major modes. The algorithm is as follows.

**Algorithm 2.2.3. Labelling Based on the Distance to Modes (PMDIST)**

*Step 1: Find one of the \( m! \) major modes, say \( \hat{\theta} \), by the Algorithm 2.2.1.*

*Step 2: Choose the labels \((\sigma_1, \ldots, \sigma_N)\) by minimizing

\[
L(\sigma) = \sum_{t=1}^{N} (\theta^{\sigma_t} - \hat{\theta})^T (\theta^{\sigma_t} - \hat{\theta}),
\]

where \( \sigma = (\sigma_1, \ldots, \sigma_N) \). Specifically, for any \( t \), we choose \( \sigma_t \) by minimizing \((\theta^{\sigma_t} - \hat{\theta})^T (\theta^{\sigma_t} - \hat{\theta})\). □
Noticing that \( ||\theta^\sigma_t|| \) does not depend on the label \( \sigma_t \), we have

\[
(\theta^\sigma_t - \hat{\theta})^T(\theta^\sigma_t - \hat{\theta}) = ||\theta^\sigma_t|| - 2\hat{\theta}^T\theta^\sigma_t + ||\hat{\theta}||
\]

\[
= ||\theta_t|| - 2\hat{\theta}^T\theta^\sigma_t + ||\hat{\theta}||.
\]

Hence minimizing \((\theta^\sigma_t - \hat{\theta})^T(\theta^\sigma_t - \hat{\theta})\) is equivalent to maximizing \(\hat{\theta}^T\theta^\sigma_t\). We get an alternative way for the above step 2:

\[
\sigma_t = \arg\max_{\sigma} \hat{\theta}^T\theta^\sigma, \ t = 1, \ldots, N. \tag{2.5}
\]

Based on (2.5), we know the algorithm (2.2.3) in fact is to find the label \( \sigma_t \) by minimizing the angle between the vector \( \theta \) and \( \theta^\sigma_t \). This algorithm is very fast, even for large \( m \). Unlike the batch algorithm, it can do labelling along with the sampling process. Hence storage requirements are reduced. One drawback of this method is that it is not invariant to the transformation of the parameters (the scale effect of the parameters).

There are several other methods to estimate the posterior modes. If the sample size is big enough and the MLE is not difficult to calculate (by EM algorithm), we can just use MLE as the estimate of one of the major posterior modes. We can also calculate the posterior likelihood for all the Bayesian samples and choose the one with the highest value as the estimation of the major posterior mode. This is Monte Carlo approximation of the Maximum a Posteriori (MAP) estimator. If using this method to find the major modes, this modes related labelling method will be exactly the same as the labelling method proposed by Marin, Megneresen and Robert (2005).
2.2.4 Labelling Based on Loss Functions

2.2.4.1 The K-means method

Based on asymptotic identifiability, it is reasonable to assume that the “right” labels are those that make the “size” of the cluster consisting of \((\theta_1, \ldots, \theta_N)\) smallest. Different definitions of the size of a cluster lead to different labelling methods. The first proposed method is to define the size of the cluster consisting of \((\theta_1, \ldots, \theta_N)\) by the trace of their covariance matrix. We define the loss function as

\[
L(\theta, \sigma) = \text{Tr} \left( \sum_{t=1}^{N} (\theta^\sigma_t - \theta)(\theta^\sigma_t - \theta)^T \right) = \sum_{t=1}^{N} (\theta^\sigma_t - \theta)^T (\theta^\sigma_t - \theta),
\]

(2.6)

where \(\sigma = (\sigma_1, \ldots, \sigma_N)\) and \(\text{Tr}(A)\) is the trace of \(A\). When the labels \((\sigma_t, t = 1, \ldots, N)\) are fixed, the minimum over \(\theta\) of (2.6) occurs at the sample mean of \(\{\theta_1^\sigma, \ldots, \theta_N^\sigma\}\). When \(\theta\) is fixed, the optimum over \(\sigma_t, t = 1, \ldots, N\) can be done independently for all \(t\).

The algorithm to minimize (2.6) will be as follows.

**Algorithm 2.2.4. Labelling by Trace of Covariance (TRCOV)**

Starting with some initial values for \((\sigma_1, \ldots, \sigma_N)\) (setting them based on the order constraint for example), iterate the following steps until a fixed point is reached.

**Step 1:** Update \(\theta\) by the sample mean based on the current values \(\{\sigma_1, \ldots, \sigma_N\}\),

\[
\theta = \frac{1}{N} \sum_{t=1}^{N} \theta^\sigma_t.
\]
Step 2: Given the current estimated value $m$, $\{\sigma_1, \ldots, \sigma_N\}$ are updated by

$$
\sigma_t = \arg \min_{\sigma} (\theta_t^\sigma - \theta)^T (\theta_t^\sigma - \theta).
$$

The loss function $L(\theta, \sigma)$ defined in (2.6) decreases after each of the above two steps. So this algorithm must converge. Since this algorithm can only guaranteed to converge to a local minimum, we may run this algorithm starting from several initial values. In step 2, we can also update $\theta$ after each change of $\sigma_t$. It may increase the speed of convergence.

Notice that if $\theta$ only contains $m$ parameters (one for each component), say the $m$ component means for one dimension data, then this labelling method will be exactly the same as the labelling by putting an order constraint on the component means. Because of this, this method can be considered as an extension of the order constraint labelling method. Moreover, this algorithm will produce exactly the same labelling results as “$K$-means Clustering Algorithm” for the samples that include all the permutated values of $(\theta_t, t = 1, \ldots, N)$.

2.2.4.2 The Determinant Based Loss

A drawback of TRCOV method is that the objective function (2.6) is not invariant to the scale transformation of the parameters. For example, if the scale of one component specific parameter is too large compared to the other parameters, the labelling will be mainly dominated by this component parameter and nearly close to the order constraint labelling based on this parameter.
To solve the scale effect of the parameters, we propose another way to define the size of a cluster by the determinant of covariance matrix. We find the labels on the samples, along with the mean, that minimize the determinant of covariance matrix

\[ L(\theta, \sigma) = \det \left( \sum_{t=1}^{N} (\theta^{\sigma_t} - \theta)(\theta^{\sigma_t} - \theta)^T \right), \]  

(2.7)

where \( \sigma = (\sigma_1, \ldots, \sigma_N) \) and \( \det(A) \) is the determinant of matrix \( A \). This determinant criterion is invariant to all permutation invariant linear transformations of the parameters (changing both component means by a scale factor, both variances by a different one, for example). Based on the next theorem, we can know when \( (\sigma_1, \ldots, \sigma_N) \) are fixed, the minimum of (2.7) over \( \theta \) occurs at the sample mean of \( \{\theta_1, \ldots, \sigma_N\} \), denoted by \( \bar{\theta} \).

**Theorem 2.2.1.** When \( (\sigma_1, \ldots, \sigma_N) \) are fixed, the sample mean of \( \{\theta_1, \ldots, \theta_N\} \), denoted by \( \bar{\theta} \), minimizes (2.7) over \( \theta \).

**Proof:** Notice that

\[
\det \left( \sum_{t=1}^{N} (\theta^{\sigma_t} - \theta)(\theta^{\sigma_t} - \theta)^T \right) = \det \left( \sum_{t=1}^{N} (\theta^{\sigma_t} - \bar{\theta})(\theta^{\sigma_t} - \bar{\theta})^T + N(\bar{\theta} - \theta)(\bar{\theta} - \theta)^T \right) \\
\geq \det \left( \sum_{t=1}^{N} (\theta^{\sigma_t} - \bar{\theta})(\theta^{\sigma_t} - \bar{\theta})^T \right).
\]

So the minimum of (2.7) over \( \theta \) occurs at the sample mean of \( \{\theta_1, \ldots, \theta_N\} \). \( \square \)

Unlike the trace of covariance case, the optimum over \( \sigma_t, t = 1, \ldots, N \) can not truly be done independently for all \( t \). Rather we need to optimize over \( \sigma_t \) one \( t \) at a time while holding all the other fixed. The algorithm to minimize (2.7) will be as follows.
Algorithm 2.2.5. Labelling by Determinant of Covariance (DETCOV)

Starting with some initial values for \((\sigma_1, \ldots, \sigma_N)\) (setting them based on the order constraint or trace of covariance loss labelling, for example), iterate the following two steps until a fixed point is reached.

**Step 1:** Update \(\theta\) by the sample mean based on the current values \(\{\sigma_1, \ldots, \sigma_N\}\),

\[
\theta = \frac{1}{N} \sum_{t=1}^{N} \theta^t_i.
\]

**Step 2:** For \(b = 1, \ldots, N\), given the current estimated value \(\theta\), and \((\sigma_t, t \neq b)\), define

\[
C_{<b>} = \sum_{t \neq b} (\sigma_t^b - \theta)(\sigma_t^b - \theta)^T,
\]

then \(\sigma_b\) are updated by

\[
\sigma_b = \arg \min_{\sigma} \det \left( C_{<b>} + (\sigma_b^b - \theta)(\sigma_b^b - \theta)^T \right). \quad \Box
\]

Notice that the objection function \(L(\theta, \sigma)\) in (2.7) is

\[
L(\theta, \sigma) = \det \left( C_{<b>} + (\theta_b^b - \theta)(\theta_b^b - \theta)^T \right)
\]

\[
= \det(C_{<b>}) \det \left[ I + \frac{1}{2} (\theta_b^b - \theta)(\theta_b^b - \theta)^T C_{<b>}^{-1} \right]
\]

\[
= \det(C_{<b>}) \left[ 1 + (\theta_b^b - \theta)^T C_{<b>}^{-1} (\theta_b^b - \theta) \right].
\]

Thus to optimize over \(\sigma_b\) for a particular \(b\), other terms fixed, we just optimize

\[
(\theta_b^b - \theta)^T C_{<b>}^{-1} (\theta_b^b - \theta).
\]
Let \( C = \sum_{t=1}^{N} (\theta_{t} - \theta)(\theta_{t} - \theta)^T \). The term \( C_{<b>}^{-1} \) can be determined as a rank one correction to \( C^{-1} \). Note that

\[
C_{<b>} = C - (\theta_{b} - \theta)(\theta_{b} - \theta)^T = C^{1/2}(I - uu^T)C^{1/2},
\]

where \( u = C^{-1/2}(\theta_{b} - \theta) \). Notice that

\[
u^T u = (\theta_{b} - \theta)^T C^{-1}(\theta_{b} - \theta).
\] (2.9)

So

\[
C_{<b>}^{-1} = C^{-1/2}(I + \frac{1}{1 - uu^T} uu^T)C^{-1/2}.
\]

Since the above expression cannot depend on the label \( \sigma_{b} \), we can use any \( \sigma_{b} \) in \( C \) and \( u \). Most simply, we use the old (current) \( \sigma_{b} \) to calculate \( C \) and \( u \).

Using the above technique, we can use the inverse of the initial matrix \( C \) to find all other matrix inversions. Suppose that we have optimized over \( b = 1 \), and we changed the permutation involved. We move to \( b = 2 \) but technically we need to update the covariance matrix for the change in \( b = 1 \) if we are to guarantee that we are monotonely decreasing the objective function. We can do so in a way that we do not need to recalculate inverses. First, we can remove the original \( b = 1 \) term as above:

\[
C_{<1>}^{-1} = C^{-1/2}(I + \frac{1}{1 - uu^T} uu^T)C^{-1/2}
\]
where \( u = C^{-1/2}(\theta^{\text{old}}_1 - \theta) \). Then when the new \((\theta^{\text{new}}_1 - \theta)(\theta^{\text{new}}_1 - \theta)^T\) is added back in, let \( w = C^{-1/2}_{<1>}(\theta^{\text{new}}_1 - \theta) \),

\[
C_{\text{new}} = C_{<1>} + (\theta^{\text{new}}_1 - \theta)(\theta^{\text{new}}_1 - \theta)^T = C^{1/2}_{<1>}(I + w w^T)C^{1/2}_{<1>}
\]

so

\[
C^{-1}_{\text{new}} = C^{-1/2}_{<1>}(I - \frac{1}{1 + w^T w} w w^T)C^{-1/2}_{<1>}. \tag{2.10}
\]

Thus we can carry out all these calculations with only the initial matrix inversion of \( C \).

**2.2.4.3 The Classification MLE Method**

From the discussion of the asymptotic labelling for Bayesian mixtures, we know that when sample size is large enough the “right” labelled MCMC samples should, approximately, follow the normal distribution \(\text{Le Cam, 1953,1986}\). Based on this property, we propose another method to do labelling by maximizing the following normal likelihood

\[
LR(\theta, C, \sigma) = |C|^{-N/2} \prod_{t=1}^{N} \exp\left\{-\frac{1}{2}(\theta_{\sigma t} - \theta)^T C^{-1}(\theta_{\sigma t} - \theta)\right\}, \tag{2.11}
\]

or, equivalently, minimizing the loss function

\[
L(\theta, C, \sigma) = -2 \log LR(\theta, C, \sigma) = \sum_{t=1}^{N} (\theta_{\sigma t} - \theta)^T C^{-1}(\theta_{\sigma t} - \theta) + N \log |C|, \tag{2.12}
\]

where \( \theta \) is the center value for the normal distribution, \( C \) is the covariance structure, and \( \sigma = (\sigma_1, \ldots, \sigma_N) \). This corresponds to applying the classification MLE to the full

By maximizing (2.11), in fact, we are choosing the labels $\sigma$ so that the maximum of the normal likelihood is greatest. Hence it is reasonable to assume that $(\theta, C, \sigma)$ reach the maxima of (2.11) at the same time. This method is affine transformation invariant, like the determinant loss, but is much easier to calculate.

The whole computation is very easy. Given the labels $\sigma$, maximizing (2.11) is equivalent to finding the MLE of a regular normal likelihood, i.e.

$$
\theta = \frac{1}{N} \sum_{t=1}^{N} \theta_{\sigma^t}^t,
$$

$$
C = \frac{1}{N} \sum_{t=1}^{N} (\theta_{\sigma^t}^t - \theta)(\theta_{\sigma^t}^t - \theta)^T.
$$

Given the center $\theta$ and covariance structure $C$ of the normal distribution, in order to maximize (2.11), the label $\sigma_t$ is chosen by minimizing

$$(\theta_{\sigma^t}^t - \theta)^T C^{-1} (\theta_{\sigma^t}^t - \theta),$$

which is a weighted distance between $\theta_{\sigma^t}^t$ and $\theta$. The weight $C$ will make this labelling method invariant to the affine transformation of the component parameters. So the algorithm to find the labels by maximizing (2.11) is as follows.

Algorithm 2.2.6. Labelling by Normal Likelihood (NORMLH)

Starting with some initial values for $(\sigma_1, \ldots, \sigma_N)$ (setting them based on the order constraint or trace of covariance loss labelling, for example), iterate the following two steps
until a fixed point is reached.

**Step 1:** Update $\theta$ by the sample mean based on the current values \{\(\sigma_1, \ldots, \sigma_N\)\},

\[
\theta = \frac{1}{N} \sum_{t=1}^{N} \theta_t^\sigma t .
\]

**Step 2:** Given the current value $\theta$, let $C = \frac{1}{N} \sum_{t=1}^{N} (\theta_t^\sigma - \theta)(\theta_t^\sigma - \theta)^T$. For $t = 1, \ldots, N$, update $\sigma_t$ by

\[
\sigma_t = \arg \min_\sigma (\theta_t^\sigma - \theta)^T C^{-1}(\theta_t^\sigma - \theta) .
\]

□

Since in each step of the above algorithm, the loss function (2.12) decreases, this algorithm must converge. In step 2, if there is any change of $\sigma_t$, we can also update $\theta$ and $C$, which may increase the speed of convergence. When $\theta$ is stable, \{\(\sigma_t, t = 1 \ldots, N\)\} are not expected to change a lot and so we need not update $C$ very often. So the Algorithm 2.2.6 is much faster than the Algorithm 2.2.5. If $N$ is big enough, which is generally the case for MCMC samples, $C_{<b>}^{-1}$ in (2.8) is approximately equal to $C^{-1}$. Hence, generally, NORMLH labelling method produces similar labelling results to DETCOV method.

**2.2.5 Comparing the Proposed Labelling Methods**

Now we have proposed five labelling methods. We think PM(ECM) method, labelling based on ECM(BM) algorithm, is the most reasonable and intuitive method and use it as a benchmark for comparison, as it can reproduce the HPD labels when they exist. Another advantage of PM(ECM) is that it can do labelling along the process...
of MCMC sampling, unlike batch algorithm. In addition, the PM(ECM) method does not require to compare \(m!\) permutations, which saves lot of time when \(m\) is large.

The PMDIST, labelling based on modal distance, is an approximation to ECM(BM) that avoids iteration. One drawback of this method is that it is not invariant to the scale transformation of the component parameters. In addition, it needs to compare \(m!\) permutations for each MCMC sample.

All the other three methods are based on certain loss function and use the batch algorithm, which is similar to the relabelling algorithm (Stephens, 2000b and Celeux, 1998) and the labelling method using label invariant loss functions (Celeux, Hurn and Robert, 2000 and Hurn, Justel and Robert, 2003). For these kind of labelling methods, we must do labelling for all the MCMC samples simultaneously, which demands high storage requirements. In addition, we need to compare \(m!\) permutations for each MCMC sample in every iteration. When \(m\) is large, it demands huge computation and may even be slower than the PM(ECM) method.

The TRCOV method, labelling based on the trace of covariance loss, is the fastest method among the three loss based labelling methods. However, TRCOV is not invariant to the scale transformation of the parameters. Both the DETCOV (labelling based on the determinant of covariance loss) and NORMLH methods (labelling based on the normal likelihood) are scale invariant. This property is very important if we want to use different scale parameters of every component to do labelling. For example, for a normal mixture, if we want to do labelling using all the information including component means, component variance, and component proportions, the transformation invariant labelling method will delete the scale effect of the means, variance, and proportions. Though the
NORMLH method produce similar labels as DETCOV, NORMLH is much faster than DETCOV. Moreover, the NORMLH method gives us a nice explanation based on the normal likelihood and the asymptotic labelling. As we will see, it is also excellent at reproducing HPD labels. Thus in practice, we recommend using the NORMLH method, among the other four labelling methods except for the benchmark method PM(ECM).

2.3 Simulation Studies and Real Data Application

In this section, we will use several examples to compare different labelling methods. We will mainly compare our proposed labelling method based on normal likelihood (NORMLH) and the method based on the posterior modes and ECM(BM) algorithm (PM(ECM)), with the previously labelling methods based on the order constraint of component means (OC) and Stephens’s KL algorithm (KL). In each simulation example, we will report the labelling credibility level to check how difficulty the labelling is for the Gibbs samples. To compare different labelling methods, we will also report the posterior means of labelled Gibbs samples and the number of different labels between PM(ECM) and other labelling methods.

For simplicity we will use the two-component normal mixtures to compare the different labelling methods. All the MCMC samples are generated by Gibbs sampler with the prior given by Richardson and Green (1997).

Example 1. Generate 400 data from $0.8N(0,1)+0.2N(1.4,1)$. Based on this sample, generate 3000 MCMC samples of component means, component proportions, and the common component variance.
In this example, both of the component means and the proportions provide label information about the Gibbs samples. The labelling credibility level for the Gibbs samples is 84.27%. The numbers of different labels between (OC, KL, NORMLH) and PM(ECM) are: 191, 0, and 0 respectively. Hence, in this example, KL, NORMLH, and PM(ECM) had the same labelling results.

The running time for OC, KL, NORMLH, and PM(ECM) are 0.11, 10.62, 1.28, and 99.70 seconds respectively using the personal desktop with Pentium 4 CPU 2.40GHz. In this example 94% of MCMC samples converged to the maximal modes and thus easily can be labelled by the converged modes directly. However there are around 6% of samples that converged to a flat posterior region. For those points, we used the NORMLH method to label them.

Figure 2.3 is the plot of posterior density versus projection of labelled Gibbs samples and their permutations for different labelling methods. The blue points are the projection of original labelled Gibbs samples. The green points are the projection of the permuted labelled Gibbs samples. From this plot, we can see that both PM(ECM) and OC reproduced the HPD labels well. The difference of the labelling between PM(ECM) and OC all occurred for lower posterior density area. The red star in the PM(ECM) plot is the point we are interested in to find the $c^*$ value. In this example, the $c^{**}$ value, the posterior density for degenerate mode, is even bigger than $c^*$. This happens because $c^{**}$ is the theoretic posterior density of the degenerate mode found by ECM(BM) algorithm but $c^*$ is the approximated value based on Gibbs samples. Thus in this example $c^*$ is the same as $c^{**}$. The labelling credibility level is estimated by the proportions of Gibbs samples with posterior density larger than $c^*$. 
Table 2.1 are the average of 3000 Gibbs samples labelled by OC, KL, NORMLH, and PM(ECM). For comparison, we also included the parameter values (labelled as True) of the normal mixture model from which the data was generated and the posterior mode (labelled as Mode). The mu1 refers to the first component mean, mu2 refers to the second component mean, sigma refers to the common component standard deviation, and pi1 refers to the proportion of the first component. These notations will also be used in the next few examples. In this example, both the component proportions and the component means give us the label information. Since the order constraint method only focused on the order of component means, it created bigger estimation bias of component proportion than the other three labelling methods.

Table 2.1.
Sample means of labelled Gibbs samples for different labelling methods.

<table>
<thead>
<tr>
<th></th>
<th>mu1</th>
<th>mu2</th>
<th>sigma</th>
<th>pi1</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>0</td>
<td>1.4</td>
<td>1</td>
<td>0.8</td>
</tr>
<tr>
<td>OC</td>
<td>-0.0640</td>
<td>1.4455</td>
<td>0.9220</td>
<td>0.7506</td>
</tr>
<tr>
<td>KL</td>
<td>-0.0323</td>
<td>1.4138</td>
<td>0.9220</td>
<td>0.7873</td>
</tr>
<tr>
<td>NORMLH</td>
<td>-0.0323</td>
<td>1.4138</td>
<td>0.9220</td>
<td>0.7873</td>
</tr>
<tr>
<td>PM(ECM)</td>
<td>-0.0323</td>
<td>1.4138</td>
<td>0.9220</td>
<td>0.7873</td>
</tr>
<tr>
<td>Mode</td>
<td>-0.0827</td>
<td>1.5798</td>
<td>0.8840</td>
<td>0.7900</td>
</tr>
</tbody>
</table>

Figure 2.4 are the plots of mu1−mu2 vs. pi1 of the 3000 Gibbs samples and their permutated values labelled by OC and PM(ECM). The red hexagram points are the posterior modes. From the graph, we can see PM(ECM) labelling separated the parameter space well for the Gibbs samples. However the order constraint method did
not work well in this example (mainly for the Gibbs samples with low posterior density), since it did not use the component proportion information.

Figure 2.5 are the marginal density plots of \( \mu_1, \mu_2, \sigma, \) and \( \pi_1 \). The first two columns are based on the Gibbs samples labelled by OC and PM(ECM). The third column is based on the HPD Gibbs samples labelled by PM(ECM). The marginal density plots of \( \sigma \) are the same for the two labelling methods, since the variance are assumed to be equal for the two components. Based on the marginal density plots of \( \mu_1, \mu_2, \) and \( \pi_1 \), it appears that PM(ECM) successfully removed the label switching problem for the Gibbs samples but the order constraint method did not. The marginal density plot of the HPD Gibbs samples labelled by PM(ECM) shows clearly that the PM(ECM) method created two well labelled modal regions.

**Example 2.** Generating 400 data from \( 0.3N(0,1)+0.7N(0.5,2) \). Based on this sample, generate 3000 MCMC samples of component means, component proportions, and the unequal component variance.

In this example, all of the component parameters, including component means, component proportions, and the unequal component variance, provide label information.

The labelling Credibility level for the Gibbs samples is 80.6%. The numbers of different labels between (OC, KL, NORMLH) and PM(ECM) are: 361, 149, and 10 respectively. Thus NORMLH had almost the same labelling results as PM(ECM). The order constraint method had more than 10% of the Gibbs samples labelled different from PM(ECM). The KL algorithm had about 5% of the Gibbs samples labelled different PM(ECM).
The running time for OC, KL, NORMLH, and PM(ECM) are 0.06, 22.90, 3.91, and 83.12 seconds respectively using the personal desktop with Pentium 4 CPU 2.40GHz. In this example, all the samples converged to the maximal modes. So all the samples can be labelled directly by the corresponding converged modes.

Figure 2.6 is the plot of posterior density versus projection of labelled Gibbs samples and their permutations for different labelling methods. From this plot, we can see NORMLH and PM(ECM) labelled the Gibbs samples better than OC and KL. In this example, the $c^{**}$ value is bigger than $\hat{c}^*$, the posterior density of the red star. Thus $c^*$ is the same as $c^{**}$. From this plot, we can see both NORMLH and PM(ECM) recovered the HPD labels well but OC and KL did not.

Table 2.2 are the average of Gibbs samples labelled by OC, KL, NORMLH and PM(ECM). Figure 2.7 are the plots of $\sigma_1 - \sigma_2$ vs. $\pi_1$ for Gibbs samples labelled by different methods. Figure 2.8 are the plots of $\sigma_1 - \sigma_2$ vs. $\mu_1 - \mu_2$. Figure 2.9 are the plots of $\mu_1 - \mu_2$ vs. $\pi_1$. From the above plots we can see that the order constraint method did not separate the two symmetric modal regions well. KL labelling worked a little better than order constraint but still did not remove the label switching completely, based on the Figure 2.7 and Figure 2.8. NORMLH and PM(ECM) worked almost the same and both solved the label switching well.

Figure 2.10 – 2.12 are the marginal density plots of $\mu$, $\sigma$, and $\pi$ of Gibbs samples labelled by different methods. The above marginal density plots reconfirm that NORMLH and PM(ECM) worked well. Figure 2.13 are the marginal density plots of $\mu$, $\sigma$, and $\pi$ of HPD Gibbs samples labelled by PM(ECM). From the plot, we can
Table 2.2.
Sample means of labelled Gibbs samples for different labelling method.

<table>
<thead>
<tr>
<th></th>
<th>$\mu_1$</th>
<th>$\mu_2$</th>
<th>$\sigma_1$</th>
<th>$\sigma_2$</th>
<th>$\pi_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>0.3</td>
</tr>
<tr>
<td>OC</td>
<td>0.0510</td>
<td>0.5288</td>
<td>1.1466</td>
<td>1.9538</td>
<td>0.3082</td>
</tr>
<tr>
<td>KL</td>
<td>0.0876</td>
<td>0.4923</td>
<td>1.0678</td>
<td>2.0327</td>
<td>0.2320</td>
</tr>
<tr>
<td>NORMHL</td>
<td>0.0752</td>
<td>0.5047</td>
<td>1.0336</td>
<td>2.0669</td>
<td>0.2500</td>
</tr>
<tr>
<td>PM(ECM)</td>
<td>0.0761</td>
<td>0.5037</td>
<td>1.0326</td>
<td>2.0679</td>
<td>0.2515</td>
</tr>
<tr>
<td>Mode</td>
<td>0.0205</td>
<td>0.5092</td>
<td>0.9620</td>
<td>2.0757</td>
<td>0.2351</td>
</tr>
</tbody>
</table>

see the component parameters are all unimodal and thus PM(ECM) removed the label switching well in the HPD area.
Fig. 2.1. Plot of the two component means of 1000 Gibbs samples for the observations with sample size of 500 generated from $0.5N(0,1)+0.5N(1.5,1)$. 
Fig. 2.2. Plot of the two component means of 1000 Gibbs samples for the observations with sample size of 100 generated from $0.5\text{N}(0,1)+0.5\text{N}(1.5,1)$. 
Fig. 2.3. Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 1.
Fig. 2.4. Plots of $\mu_1 - \mu_2$ vs. $\pi_1$ for different labelling methods in example 1. The red hexagram points is the posterior modes.
Fig. 2.5. Marginal density plot of parameters of the Gibbs samples labelled by OC and PM(ECM) for example 1.
Fig. 2.6. Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 2.
Fig. 2.7. Plots of $\sigma_1-\sigma_2$ vs. $\pi_1$ for different labelling methods in example 2.
Fig. 2.8. Plots of $\sigma_1-\sigma_2$ vs. $\mu_1-\mu_2$ for different labelling methods in example 2.
Fig. 2.9. Plots of $\mu_1-\mu_2$ vs. $\pi_1$ for different labelling methods in example 2.
Fig. 2.10. Marginal density plots of component means for different labelling methods in example 2.
Fig. 2.11. Marginal density plots of component standard deviation for different labelling methods in example 2.
Fig. 2.12. Marginal density plots of component proportions for different labelling methods in example 2.
Fig. 2.13. Marginal density plots of the parameters for HPD Gibbs samples labelled by PM(ECM) for example 2.
**Example 3.** Generating 400 data from $0.3N(0,1)+0.7N(0.5,2)$. Based on this sample, generate 5000 MCMC samples of component means, component proportions, and the unequal component variance.

The data are generated from the same model as example 2 but this data set provide us other useful information about different labeling methods.

The labeling credibility is 93.3%. The numbers of different labels between (OC, KL, NORMLH) and PM(ECM) are: 86, 337, and 3 respectively. Hence NORMLH and PM(ECM) had almost the same labels. In this example, KL had larger number of different labels compared to PM(ECM) than OC method.

The running time for OC, KL, NORMLH, and PM(ECM) are 0.26, 22.60, 1.19, and 131.89 seconds respectively using the personal desktop with Pentium 4 CPU 2.40GHz. In this example, all the samples converged to the maximal modes. So all the samples can be labelled directly by the corresponding converged modes. Figure 2.14 is the plot of posterior density versus projection of labelled Gibbs samples and their permutations for different labeling methods. From this plot, we can see NORMLH and PM(ECM) labelled the Gibbs samples better than OC and KL. In this example, the $c^{**}$ value, the posterior density for degenerate mode, is bigger than the posterior density of the red hexagram point, thus $c^*$ is the same as $c^{**}$. From this plot, we can also see both NORMLH and PM(ECM) recovered the HPD labels well but OC and KL did not.

Table 2.3 are the average of Gibbs samples labelled by OC, KL, NORMLH and PM(ECM). Figure 2.15 are the plots of $\sigma_1 - \sigma_2$ vs. $\pi_1$ for different labelling methods. Figure 2.16 are the plots of $\sigma_1 - \sigma_2$ vs. $\mu_1 - \mu_2$. Figure 2.17 are the plots of $\mu_1 - \mu_2$ vs. $\pi_1$. From above plots, we can clearly see KL did not separate
the parameter space well. It seems that KL focused more on component proportions based on the Figure 2.15 and 2.17. The order constraint method worked a little better than KL for this data set. NORMLH and PM(ECM) separated the parameter space quite well from all the three plots (especially from Figure 2.16).

Table 2.3.
Sample means of labelled Gibbs samples for different labelling method.

<table>
<thead>
<tr>
<th></th>
<th>mu1</th>
<th>mu2</th>
<th>sigma1</th>
<th>sigma2</th>
<th>pi1</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>0.3</td>
</tr>
<tr>
<td>OC</td>
<td>-0.0404</td>
<td>0.7088</td>
<td>0.9815</td>
<td>1.8851</td>
<td>0.3311</td>
</tr>
<tr>
<td>KL</td>
<td>0.0290</td>
<td>0.6394</td>
<td>1.0101</td>
<td>1.8565</td>
<td>0.2894</td>
</tr>
<tr>
<td>NORMLH</td>
<td>-0.0355</td>
<td>0.7039</td>
<td>0.9679</td>
<td>1.8987</td>
<td>0.3237</td>
</tr>
<tr>
<td>PM(ECM)</td>
<td>-0.0357</td>
<td>0.7041</td>
<td>0.9680</td>
<td>1.8987</td>
<td>0.3238</td>
</tr>
<tr>
<td>Mode</td>
<td>-0.0081</td>
<td>0.6947</td>
<td>0.9747</td>
<td>1.9352</td>
<td>0.3442</td>
</tr>
</tbody>
</table>

Figure 2.18 – 2.20 are the marginal density plots of mu, sigma, and pi of Gibbs samples labelled by different methods. Figure 2.21 are the marginal density plots of mu, sigma, and pi of HPD Gibbs samples labelled by PM(ECM). From the plot, we can see the component parameters are all unimodal and thus PM(ECM) removed the label switching well in the HPD area.
**Real Data Application:** Now let us apply our new labelling method to the Galaxy data introduced in section 1.2. We post processed the 10000 Gibbs samples by PM(ECM).

Figure 2.22 are the trace plot of component means of Gibbs samples labelled by PM(ECM). Figure 2.23 are the plot of marginal posterior densities of component means based on the Gibbs samples labelled by PM(ECM). Compared to the Figure 1.2 and 1.3, we can see the label switching in evidence in the raw output of the Gibbs sampler has been successfully removed. Most of the multimodality of the marginal posterior densities of the means has been removed. Base on the labelled Gibbs samples, we can perform inference for any quantity of interest.
Fig. 2.14. Plot of posterior density versus projection of Gibbs samples for different labelling methods in example 3.
Fig. 2.15. Plots of $\sigma_1-\sigma_2$ vs. $\pi_1$ for different labelling methods in example 3.
Fig. 2.16. Plots of $\sigma_1-\sigma_2$ vs. $\mu_1-\mu_2$ for different labelling methods in example 3.
Fig. 2.17. Plots of $\mu_1-\mu_2$ vs. $\pi_1$ for different labelling methods in example 3.
Fig. 2.18. Marginal density plots of component means for different labelling methods in example 3.
Fig. 2.19. Marginal density plots of component standard deviation for different labelling methods in example 3.
Fig. 2.20. Marginal density plots of component proportions for different labelling methods in example 3.
Fig. 2.21. Marginal density plots of the parameters for HPD Gibbs samples labelled by PM(ECM) for example 3.
Fig. 2.22. Plot of Gibbs samples of component means labelled by PM(ECM) for Galaxy data.
Fig. 2.23. Plot of marginal posterior densities of component means based on the Gibbs samples labelled by PM(ECM).
Chapter 3

Summary and Future Research

Suppose \( \mathbf{x} = (x_1, \ldots, x_n) \) are i.i.d. observations from a mixture density with \( m \) components. The likelihood for \( \mathbf{x} \) is

\[
L(\theta; \mathbf{x}) = \prod_{i=1}^{n} \left\{ \pi_1 f(x_i; \lambda_1) + \pi_2 f(x_i; \lambda_2) + \cdots + \pi_m f(x_i; \lambda_m) \right\}
\]

where

\[
\theta = \left[ \begin{array}{c} (\pi_1) \, \lambda_1, \ldots, (\pi_m) \, \lambda_m \end{array} \right].
\]

The likelihood function of mixture distribution is symmetric for all the component parameters. Thus any permutated component parameters give the same likelihood. Given a sequence of the parameter estimates \( \hat{\theta}_1, \ldots, \hat{\theta}_n \), we need to decide which permutation of \( \hat{\theta}_j \), \( j = 1 \ldots, n \) is the right permutation for the true parameter \( \theta \) if we want to estimate the uncertainty of the estimators based on \( (\hat{\theta}_1, \ldots, \hat{\theta}_n) \). That is how label switching problem occurs in frequentist case.

For Bayesian mixtures, if we use noninformative prior, i.e. the symmetric prior for each component, the posterior distribution will also be symmetric for each component parameters. Any quantities relating to individual components of the mixture, such as predictive component densities, marginal classification probabilities, are the same. This is the label switching problem for Bayesian mixtures. Label switching is one of the most
fundamental problems for Bayesian mixtures. The easiest way to solve the labelling is to put an explicit parameter constraint so that only one permutation can satisfy it. This method is initially used by Diebolt and Robert (1994) and Dellaportas, et al (1996). A generally used constraint is $\lambda_1 < \lambda_2 < \ldots < \lambda_m$ in the univariate case. We can also put left-right ordering on $\pi$ if we have the information that the proportions of each component are different. Artificial identifiability constraint can also be used to solve labelling switching in frequentist analysis. Celeux (1997), Celeux, Hurn and Robert (2000) and Stephens (1997a, 1997b, 2000) all expressed their concerns about imposing an identifiability constraint. One problem with identifiability constraint is the choice of constraint, especially for multivariate problems. Different ordering may generate markedly different results as demonstrated by Celeux, Hurn and Robert (2000) and generally it is difficult to anticipate the overall effect. Moreover, many choices of identifiability constraint can not completely remove the symmetry in the posterior distribution. As a result, label switching problem may remain after imposing an identifiability constraint. See the example by Stephens (2000b).

We proposed five new methods to solve the label switching problem for Bayesian mixtures. The five methods label the Gibbs samples by the posterior modes and the ECM(BM) algorithm (PM(ECM)), the distance to the posterior modes (DISTRPM), the trace of covariance loss (TRCOV), the determinant of covariance loss (DETCOV), and the normal likelihood (NORMLH), respectively. The ECM(BM) algorithm is the one we proposed to find the posterior modes for Bayesian mixtures by using the idea of ECM (Meng and Rubin, 1993) and Gibbs sampler. The first method of PM(ECM) is the most reasonable labelling methods among all the five methods we proposed. The PM(ECM)
method can reproduce the HPD labels and create a natural and intuitive partition of the parameter space into labelled regions. We consider this labelling method as the standard one and use it as a benchmark for comparison. Associated with PM(ECM), we also defined a new term “labelling credibility level”. It can be used to evaluate how separate the permutated modal regions are and how well the labelling can be. The second method of labelling, based on the distance to the posterior modes (DISTPM), is a very fast labelling method. However it is not invariant to the transformation of the parameters. The last three labelling methods are based on loss functions. The TRCOV method is the fastest method among the three loss based labelling methods. However, TRCOV is also not invariant to the scale transformation of the parameters. The scale invariance property of DETCOV and NORMLH, is important if we want to use different scale parameters of every component to do labelling. The NORMLH method is similar to DETCOV, but NORMLH is much faster than DETCOV.

In practice, when \( m \) (the number of components) is small, we recommend using NORMLH due to both its fast computation and transformation invariance. In addition, it is also excellent at reproducing HPD labels based on our simulations. Since all the three loss based methods and DISTPM need to compare \( m! \) permutations but PM(ECM) need not, when \( m \) is large, we recommend directly using the PM(ECM) method. From the simulation study we can see our new proposed method works quite well and generally works better than order constraints and the KL algorithm.

Currently, our proposed labelling methods use hard labels. In the future we will try to utilize the soft labelling, which gives a probability for each label. This method seems more reasonable especially for the situation when the labelling credibility level is
small. It not only gives us the label of each MCMC sample but also the measure of certainty of each label.
Bibliography


Weixin Yao was born in Jiang Su in People’s Republic of China on December 13, 1979. In 2002 he received the B.S. degree in Statistics, from the University of Science and Technology of China. In 2005, he married Jian Guo. In 2002 he enrolled in the Ph. D. program in Statistics at the Pennsylvania State University. From 2002 to 2004 he has been employed in the Statistics Department of the Pennsylvania State University as a teaching assistant. From 2005 to 2007 he has been employed in the Statistics Department of the Pennsylvania State University as a research assistant supported by Dr. Bruce G. Lindsay and Dr. Runze Li.

Weixin Yao is a member of the American Statistical Association and Institute of Mathematical Statistics.