DISSERTATION

PENALIZED UNIMODAL SPLINE DENSITY ESTIMATE WITH APPLICATION TO M-ESTIMATION

Submitted by

Xin Chen

Department of Statistics

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2020

Doctoral Committee:

Advisor: Mary C. Meyer

Haonan Wang
Piotr Kokoszka
Wen Zhou
Hong Miao

ABSTRACT

PENALIZED UNIMODAL SPLINE DENSITY ESTIMATE WITH APPLICATION TO M-ESTIMATION

This dissertation establishes a novel type of robust estimation, Auto-Adaptive M-estimation (AAME), based on a new density estimation.

The new robust estimation, AAME, is highly data-driven, without the need of priori of the error distribution. It presents improved performance against fat-tailed or highly-contaminated errors over existing M-estimators, by down-weighting influential outliers automatically. It is shown to be root-n consistent, and has an asymptotically normal sampling distribution which provides asymptotic confidence intervals and the basis of robust prediction intervals.

The new density estimation is a penalized unimodal spline density estimation which is established as a basis for AAME. It is constrained to be unimodal, symmetrical, and integrate to 1, and it is penalized to have stabilized derivatives and against over-fitting, overall satisfying the requirements of being applied in AAME. The new density estimation is shown to be consistent, and its optimal asymptotic convergence rate can be obtained when the penalty is asymptotically bounded.

We also extend our AAME to linear models with heavy-tailed and dependent errors. The dependency of errors is modeled by an autoregressive process, and parameters are estimated jointly.

ACKNOWLEDGEMENTS

Firstly, I would like to express my deep gratitude to my research advisor, Dr. Mary C. Meyer. Her earnest, enthusiasm, and motivation for research have deeply inspired me. It was my fortune to work and study under her advice and support. I am extending my sincere thanks to her acceptance and patience during the discussion I had with her on research work and paper preparation.

Secondly, I would like to thank my committee members: Drs. Haonan Wang, Piotr Kokoszka, Wen Zhou, Hong Miao, and my ex-co-advisor Dr. Jean Opsomer, for providing helpful advice, and for serving in my committee.

Thirdly, I appreciate Dr. Mary C. Meyer, for providing me the data set of bodyweight of Georgia school-age children, to be used as an example of application in Chapter 3.

Last but not least, my thanks go to all the people who have supported me to complete the research work directly or indirectly.

DEDICATION

I would like to dedicate this thesis to my mother Ping and my wife Lin.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Motivation

1.1.1 Influential Outliers

It has long been observed that the quality of a regression estimator, assuming normality on the errors, can be massively affected by the introduction of outliers. Possible problems include but are not limited to: firstly, because the variance of the estimates is artificially inflated, outliers can be masked. In many situations, including some areas of earth sciences and medical statistics, true outliers are actually of interest (to recognize abnormal cases). Secondly, least-squares estimation will be inefficient, because the least-squares estimation are sensitive and tends to be dragged towards the outliers. In other words, its standard error will be large, so that would require a larger sample size.

This dissertation establishes a novel robust regression method, to deal with errors that are containing outliers and hence not following a normal distribution. Particularly, the new method has good performance against errors that are potentially (i) contaminated and (ii) heavy-tailed. Contaminated and heavy-tailed errors are discussed widely in different fields and disciplines. Below is an informal definition of them.

Errors with an ϵ-contamination distribution

A frequent cause of errors with outliers is defined as a mixture of two distributions, which may be two distinct sub-populations, indicating the 'expected' noises versus 'unusual' or 'contaminated' error. This is modeled by a mixture model. [Huber, 1964] proposed the $\epsilon-$contamination model, and see [Du et al., 2018] and [Chen et al., 2016] for more recent discussions. In particular, a contaminated normal distribution in which the majority of observations are from a specified normal distribution, but a small proportion are from a normal distribution with much higher variance.

That is, errors have probability $1 - \epsilon$ of coming from a normal distribution with variance σ, and probability ϵ of coming from a normal distribution with variance $c\sigma^2$ for some $c > 1$:

$$e_i \sim (1 - \epsilon)N(0, \sigma^2) + \epsilon N(0, c\sigma^2).$$

Errors with a heavy-tailed distribution

Another frequent case of errors with outliers is described by a heavy-tailed distribution. The distribution of a random variable X with distribution function F is said to have a heavy (right) tail if the moment generating function of X, $M_X(t)$, is infinite for all $t > 0$. That means

$$\int_{-\infty}^{\infty} e^{tx}\, dF(x) = \infty,$$

for all $t > 0$. An implication of this is that

$$\lim_{x \to \infty} e^{tx} \mathbb{P}(X > x) = \infty,$$

for all $t > 0$. In other words, its tail probability $\mathbb{P}(X > x)$ decays more slowly than those of any exponential distribution. See [Foss et al., 2011]. Heavy left tails are defined in a similar way. Common heavy-tailed distributions include but are not limited to: the Pareto distribution; the Log-normal distribution; the Weibull distribution with shape parameter greater than 0 but less than 1; the Fréchet distribution; the Cauchy distribution; and the t-distribution.

Fat-tailed distributions

An important class of heavy-tailed distributions are those with power law behavior, which is said to be fat-tailed. That is, if the tail probability of a random variable X can be expressed as

$$\mathbb{P}(X > x) \sim x^{-\alpha},$$

as $x \to \infty$. Then the distribution is said to have a fat-tail if the positive exponent $\alpha > 0$ is small. Here α is called the tail index which indicates the thickness of the tail probability. See [Haas and Pigorsch, 2009] for more formal definitions and [Cooke et al., 2014] for examples.

From a mathematical and statistical perspective, tail index is the most fascinating aspect of fat-tailed distributions. According to the tail index, a fat-tailed distribution may have its mean, variance, and other measures that describe the shape of the distribution undefined. For instance, if $\alpha \leq 2$, the variance, the skewness, and all higher moments of the tail are mathematically undefined, and hence larger than that in any normal or exponential distribution.

A good example of fat-tailed distributions could be the Pareto distribution, as its tail probability is clearly following the power law. A random variable X is said to have a Pareto distribution with shape parameter $\alpha > 0$ and scale parameter $x_m > 0$ if X takes values in the interval $[x_m, \infty)$ with cumulative distribution

$$\mathbb{P}(X \leq x) = 1 - \left(\frac{x_m}{x}\right)^{\alpha}.$$

It is clear that X is fat-tailed with index α. Moreover, the moment of X is

$$E\left[X^t\right] \begin{cases} \frac{\alpha x_m^t}{\alpha - t} & \text{for } t < \alpha \\ \infty & \text{for } t \geq \alpha, \end{cases}$$

which indicates that X has a finite mean only if $\alpha > 1$, and a finite variance only if $\alpha > 2$. Another well-known example of fat-tailed distributions could be the Cauchy distribution, which is with its mean and all higher moments undefined.

A remark here is that fat-tailed distributions are always heavy-tailed. Some heavy-tailed distributions, however, have a tail which goes to zero slower than an exponential function (meaning they are heavy-tailed), but faster than a power (meaning they are not fat-tailed). An example is the log-normal distribution.

1.1.2 Robust and Adaptive Regression Methods

Robust regression methods have long been of interest in statistical practice. The normal or Gaussian distribution is often not a good representation of the error density, and the error density family is usually unknown. When there is evidence that the convenient assumption of normal errors is incorrect, we would better use "robust" methods where knowledge of the error density family is not required. [Huber, 1964] proposed a solution intermediate to least-squares and median regression, and more generally, proposed M-estimation of regression functions.

Consider observations (x_i, Y_i), and the model

$$Y_i = \mu(x_i) + \varepsilon_i, \tag{1.1}$$

where μ is an unknown regression function and ε_i, $i = 1, \cdots, n$ are iid from a probability distribution f. Classical robust M-estimation of the regression function $\mu(\boldsymbol{x})$ is attained by choosing a function ρ and then finding μ to minimize

$$\sum_{i=1}^{n} \rho(Y_i - \mu(x_i)), \tag{1.2}$$

over the parameter space of μ. M-estimations are described in more details in later sections.

This dissertation proposes a new method for robust regression. We assume only that the error density f in model (1.1) is smooth, unimodal, and symmetric. For the proposed method, the error density function f is estimated using a constrained least-squares penalized spline density estimator, which is then used for determining the shape of the ρ function, to down-weight influential outliers automatically, to be used for further inferences, and possibly also for robust model selection criterion . It is shown that this new method is related to Huber's M-estimation, while more adaptive to the data, without the need of priori of error distribution (besides above mild condition). So we call this the Auto-Adaptive M-Estimator (AAME).

Nonparametric statistics

Note that throughout this dissertation, in the model (1.1), the regression model $\mu(x_i)$ will be considered in a parametric manner, while the error density model f will be considered in a non-parametric manner.

The term "nonparametric statistics" generally refers a statistical method in which models or distributions can not be determined by a small number of parameters. That is to say, it assumes a scenario that we know very little information regarding the form of models or distributions, so that we have to make the prescribed form of models or distributions to be relatively more flexible.

In the form of distribution, the meaning of nonparametric covers techniques that do not rely on data belonging to any particular parametric family of probability distributions, and is therefore not subject to violation of distribution assumptions. As such it is the opposite of parametric statistics.

In the functional form of regression, the meaning of nonparametric covers techniques that do not assume that the structure of a model is fixed. Typically, the model grows in size to accommodate the complexity of the data. These techniques include, among others: kernel regression, regression trees, and regression splines, which are agnostic about the functional form between the outcome and the covariates, and is therefore not subject to misspecification error.

Above is an informal definition of nonparametric statistics. See [Stuart et al., 1999] for more discussions. Because nonparametric techniques do not presume strong assumptions toward models or distributions, hence they generally tend to have lower bias; while tend to have higher variance, due to being highly flexible.

1.1.3 Adapting to the Data

If we know the error density f, or have its estimation $\hat{f}$, we will at least have the following benefits. We can down-weight influential outliers accordingly, we can compute robust prediction interval, and we can do convenient bootstrap inferences for more complicated regression models. Let us see these benefits in details, one-by-one.

Down-weighting influential outliers accordingly

We estimate the error density, then define $\rho = \hat{f}(0) - \hat{f}$, as the ρ function in model (1.2) (traditionally we have $\rho(0) = 0$). This then coincides with an M-estimation. Many of the nice properties of M-estimation can be inherited. Intuitively this is similar to a trimmed mean regression, with the trim points k chosen by the data and "smoothed out" from the right place. Figure 1.1 shows the defined ρ function, as solid curve. Dashed curve shows the heavy-tailed error density function f which has a $t(2)$ distribution. Vertical bars indicate random errors generated by the error density function f. An M-estimation with such a ρ function, can down-weight influential outliers automatically, so that it requires no priori of error distribution, yet has good performance against both the ϵ-contamination type of influential outliers, and very heavy-tailed (fat-tailed) error distributions with a low tail-index and corresponding infinite moments.

Computing robust prediction interval

A prediction interval is an estimate of an interval in which a future observation will fall, with a certain probability, given what has already been observed. Prediction intervals are often used in regression analysis.

In ordinary regression problems, we assume the error term ε follows a normal distribution, then quantify the combined uncertainty in (1.1) by the additive property of normal distribution. However, when the true error density is unknown, constructing a prediction interval will become problematic. Existing robust regression methods did not provide a satisfying solution, which are discussed in the following section.

Note that generally for a distribution, using only the variance (and mean zero) to describe its distribution and make a confidence interval is not adequate. While with the density estimation $\hat{f}$, we will be able to quantify the entire uncertainty which comes from both the sampling error and the future random observation. This is an important advantage that our method has over the others.

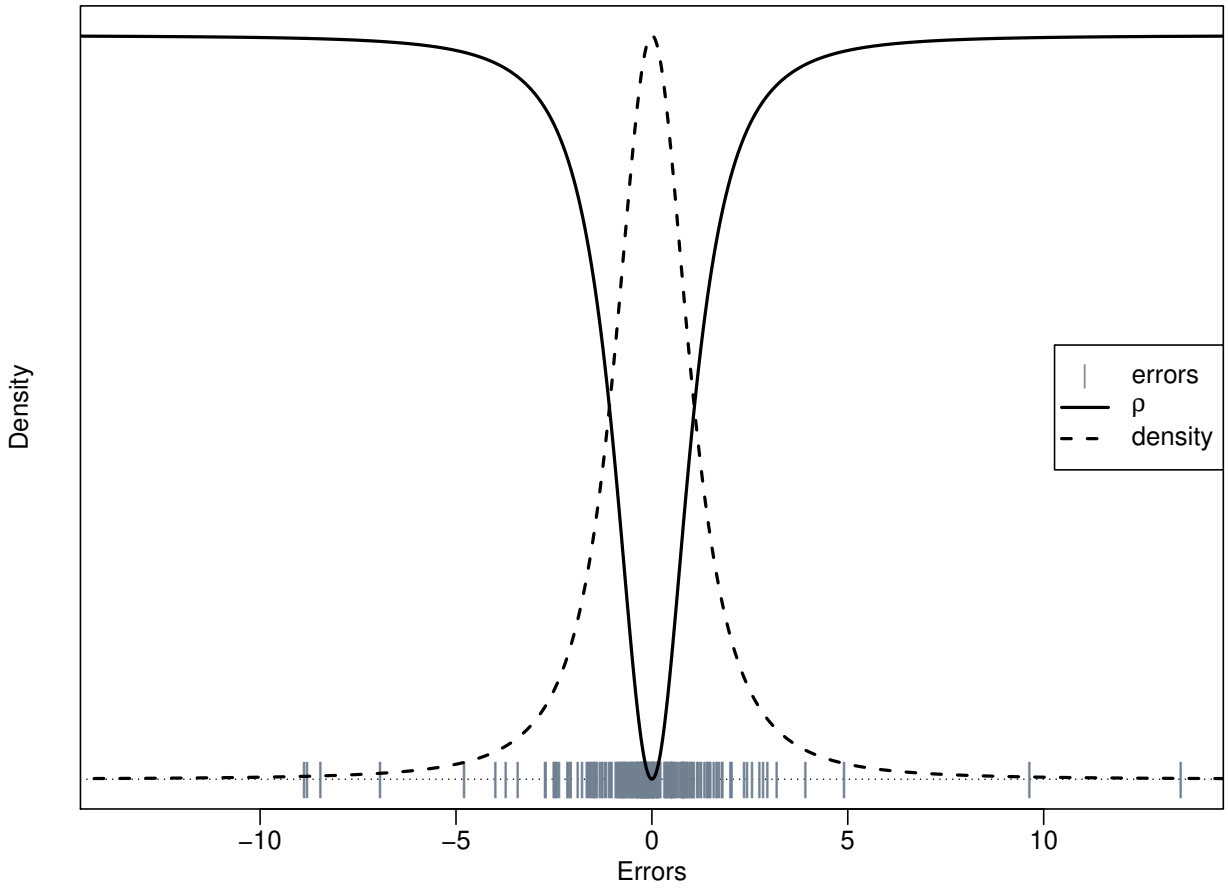

Figure 1.1: Visualized idea of down-weighting influential outliers automatically. The dashed curve shows the heavy-tailed error density function f which is $t(2)$ distribution. Vertical bars indicate random errors generated by error density function f. The solid curve shows the imputed ρ function.

Bootstrap inference for more complicated regression models

When the sample size is small, for example, smaller than $n = 30$, the asymptotic sampling distribution of $\hat{\mu}$ may not be approximated by a normal distribution very well. In this case, we would better consider a bootstrapping confidence interval.

Bootstrapping is any statistical test or metric that uses resampling to randomly sample with replacement, and then estimates the properties of an estimator (such as its variance or percentiles). See [Efron, 1992] for more details. Bootstrap works efficiently when the asymptotic sampling distribution fails due to small sample size, or the sampling distribution being hard to be derived.

We've seen three major ways of doing this: case resampling, parametric bootstrap, and resampling residuals. However, for other applications, we may also need to consider other types of bootstrapping, like block bootstrap for times series. Among these, parametric bootstrap assumes that we know the form of both of the correct functional form of regression, and the error density. It has advantages particularly with small sample sizes, and can be easily conducted even when the functional form of regression or the correlation structure is complicated, while remains valid. With an error density estimation $\hat{f}$, we may still do a convenient parametric bootstrapping even when the true error density is actually unknown.

1.1.4 Penalization: Overfitting versus Underfitting in Density Estimations

From [Claeskens et al., 2008], over-fitting is "the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably". An overfitted model is a statistical model that contains more parameters than can be justified by the data. The essence of overfitting is to have unknowingly extracted some of the residual variation (i.e. the noise) as if that variation represented underlying model structure.

Under-fitting occurs when a statistical model cannot adequately capture the underlying structure of the data. An under-fitted model is a model where some parameters or terms that would appear in a correctly specified model are missing. Under-fitting would occur, for example, when

fitting a linear model to non-linear data. Such a model will tend to have poor predictive performance.

The problems of over-fitting and under-fitting can be further breakdown into a well-known trade-off between bias and variance. Suppose we are using $\hat{f}(x)$ to estimate a target $f(x)$. Omitting the irreducible noise, our mean squared error (MSE) is:

$$\begin{aligned} MSE &= E\left[\left(f(x)-\hat{f}(x)\right)^2\right] \\ &= E\left[f(x)-\hat{f}(x)\right]^2 + Var\left[f(x)-\hat{f}(x)\right] \\ &= \left[Bias\ \hat{f}(x)\right]^2 + Var\left[f(x)-\hat{f}(x)\right]. \end{aligned} \tag{1.3}$$

Generally, a model being too simple will lead to high bias but low variance, while a model being too complex will lead to low bias but high variance. Figure 1.2 shows an example of under-fitting versus over-fitting, and how they are relate to the bias-variance trade-off. In Figure 1.2(a), the data points are generated by a quadratic polynomial model. The dashed line shows an under-fitted estimation with a linear model, the solid line shows an over-fitted estimation with a polynomial model (order 15). In Figure 1.2(b), the higher the order of polynomial, the lower the bias, while the higher the variance.

To overcome the above problem, many methods are proposed. The most popular and efficient way is to use a penalized estimation (as known as shrinkage or regularity method), such as the very well-known Lasso regression ($\mathcal{L}_1$ penalized) and Ridge regression ($\mathcal{L}_2$ penalized).

In our applications of AAME robust regression, the error density will be estimated by a constrained least-squares penalized spline density estimator. See the discussions in later sections for more reasons and considerations of using such density estimation.

Note that traditionally in spline density estimations, placing knots has already long been of interest. In our applications of robust regressions, the problem becomes more challenging: we do not assume the error density belongs to any family of distribution, and the spline density estimation must be flexible enough, to capture any possible shape of the true density function. Too few knots will not give sufficient flexibility (under-fitting) to the density estimate, which is especially needed

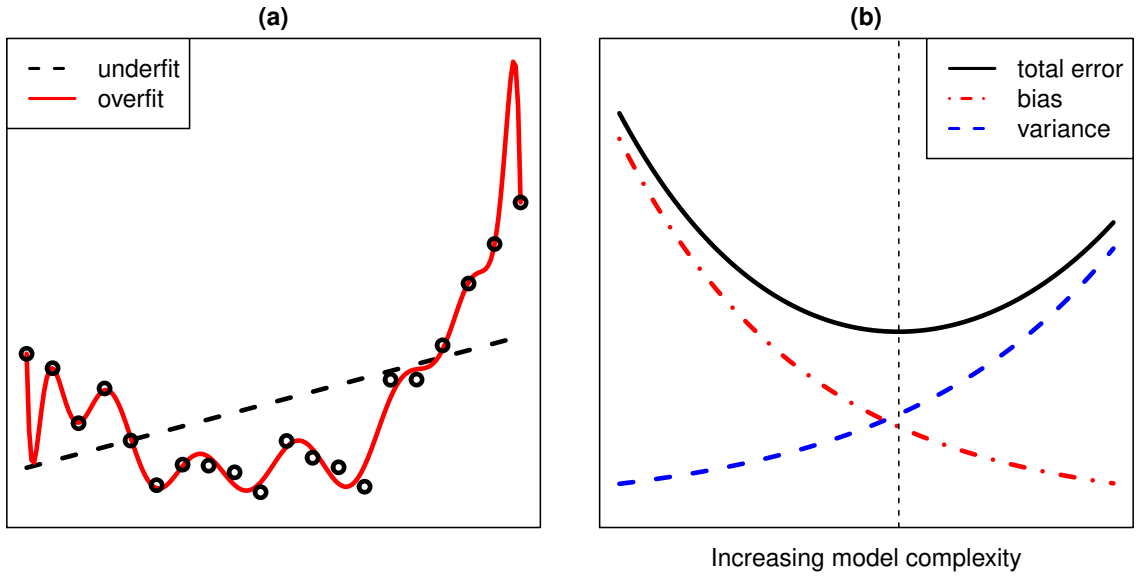

Figure 1.2: Visualized bias-variance trade-off. Left plot: the data points are generated by a quadratic polynomial model. The dashed line shows an under-fitted estimation with a linear model, the solid line shows an over-fitted estimation with a polynomial model (order 15). Right plot: the higher the order of polynomial, the lower the bias, while the higher the variance.

for heavy-tailed data. While too many knots will give too much flexibility (over-fitting), in contrast. Specially, we will see later that for our AAME, we will want to avoid the first two derivatives of the spline density estimation being wildly changing. We therefore, would need a special penalization method, to achieve our requirement.

The penalized density estimator $\hat{f}_\lambda$ minimizes the penalized $\mathcal{L}_2$ loss

$$\mathcal{L}_\lambda(\hat{f}_\lambda) = \int_{-\infty}^{\infty} \left[\hat{f}_\lambda(x) - f_n(x)\right]^2 dx + \lambda P(\hat{f}_\lambda),$$

where $f_n(x)$ is the observed empirical distribution of data. The first component of $\mathcal{L}_\lambda(\hat{f}_\lambda)$ measures the fit (divergence) of density estimation, while the second term measures the complexity of the density estimation, with $\lambda \geq 0$ as the penalty constant. See later sections and chapters for more details of this penalization.

On the other hand, for every penalized estimation, the amount of penalty, λ, always plays a crucial role. We therefore would also need an appropriate process to help to select a good penalty.

1.2 Existing Methods and Results

1.2.1 Robust Statistics and M-estimation

Parametric alternatives

An approach to robust estimation of regression models is to replace the normal distribution with a heavy-tailed distribution, such as a t-distribution with 4–6 degrees of freedom. Parametric alternatives have the advantage that likelihood theory provides an immediate approach to inference. However, an obvious and crucial problem is that such parametric models still assume an underlying error distribution, which is often not true.

Nonparametric alternatives

The simplest method of estimating parameters in a regression model that are less sensitive to outliers than the least-squares estimates, is to use least absolute deviations (LAD). However, LAD is actually the MLE for the double-exponential distribution, so essentially it is assuming an error distribution. Even then, gross outliers can still have a considerable impact on the model, motivating research into even more robust approaches.

Least trimmed squares (LTS), see [Ruppert and Carroll, 1980], is a viable alternative. A so-called trimming constant h has to satisfy $n/2 < h \leq n$. This constant determines the breakdown point of the LTS estimator since that $n - h$ observations with the most outlying values will not affect the estimator. However, choosing the trimming constant becomes a challenge.

M-estimations

In 1964, Huber introduced M-estimation for robust estimation, in [Huber, 1964]. The M in M-estimation stands for "maximum likelihood type". That paper has almost 1500 citations, over 600 of which are in statistics journals, and new versions of robust regression are proposed regularly. One of the weakness of M-estimation is the lack of consideration on the data distribution and not a function of the overall data because only using the median as the weighted value. [Rousseeuw and Yohai, 1984] proposed S-estimation, which is based on residual scale of M estimation, and this

method uses the residual standard deviation to overcome the weaknesses of median. [Yohai, 1987] proposed MM-estimation, which uses S-estimation as a warming-up stage , and MM-estimation aims to obtain estimates that have a high breakdown value and more efficient. [Susanti et al., 2014] has a summary of detailed implements of M, S, and MM-estimations. Each of them actually tries to sort of adapt to the data. Here is how:

Huber's M-estimation: it contains an estimation of the scale parameter σ, and $\hat{\mu}$ is the solution minimizing

$$\sum_{i=1}^{n} \rho\left(\frac{Y_i - \mu(\boldsymbol{x_i})}{\sigma}\right),$$

where the scale σ is estimated by

$$\hat{\sigma} = \frac{median|e_i - median(e_i)|}{0.6745},$$

with e_i the residuals from a warming-up estimation, and the constant 0.6745 ensures $\hat{\sigma}$ unbiased if the error density is normal. Also,

$$\rho(r) = \begin{cases} \frac{1}{2}r^2 & \text{if } |r| \leq k \\ k|r| - \frac{1}{2}k^2 & \text{if } |r| > k \end{cases},$$

where k is a tuning constant ensures Huber's M-estimation has minimax efficiency within a given class of error distributions, and usually $k = 1.345$.

S-estimation: follows the major idea of M-estimation, but with a different form of estimated scale

$$\hat{\sigma}_s = \sqrt{\frac{1}{nK}\sum_{i=1}^{n} w_i e_i^2},$$

where $K = 0.199$,

$$w_i = \frac{\rho(e_i/\hat{\sigma}^2)}{e_i/\hat{\sigma}^2},$$

$\hat{\sigma}^2$ is iterative and with initial estimate

$$\hat{\sigma} = \frac{median|e_i - median(e_i)|}{0.6745}.$$

MM-estimation: follows the major idea of M-estimation and S-estimation, but with a different form of estimated scale, $\hat{\sigma}_{MM}$, which is the standard deviation obtained from the residuals of S-Estimation.

[Hampel et al., 2011b] proposed a smoothed redescending M-estimator which was shown to improve the original in a variety of cases. Redescending M-estimators are M-estimators which have ρ' functions that are (on the right-hand-side of real number line) non-decreasing near the origin, but decreasing toward 0 far from the origin. Their ρ' functions can be chosen to redescend smoothly to zero, so that they usually satisfy $\rho'(r) = 0$ for all x with $|r| > k$, where r is referred to be as the minimum rejection point. Due to these properties, these kinds of estimators are efficient because they completely reject influential outliers, and do not completely ignore moderately large outliers. However, choosing the shape of redescending ρ' functions, and choosing the minimum rejection point k become another challenge.

From above summary, we can see that major and popular existing methods try to achieve a good performance within a pre-specified range of error distributions, by using “magic numbers” determined by given distributions. [Holland and Welsch, 1977] summarized how these were determined.

None of the existing methods involves estimating the unimodal error density, partly because an efficient method for this estimation has only recently become available in [Meyer, 2012]. Contrasting this method with M-estimation, we see that the problem of choosing a ρ function is solved, because this is effectively accomplished through the density estimation. In this way, the user does not have to specify "how heavy" the error density tails might be. Another advantage is that the estimated density can be used for inference about the regression function.

1.2.2 More about M-estimators

The first two derivatives and inferences

In the framework of M-estimations, inferences about the regression function would be based on asymptotic variance of the sampling distribution. [Huber, 1964] derived an expression for the asymptotic variance of $\hat{\mu}$:

$$V(\hat{\mu}) \approx \frac{\int_{-\infty}^{\infty} \rho'(y)^2 f(y) dy}{n \left[\int_{-\infty}^{\infty} \rho''(y) f(y) dy \right]^2}.$$

Implementation of this expression requires knowledge or estimation of the density f. Our new method, with a density estimation $\hat{f}$, provides a reliable solution against influential outliers.

On the other hand, intuitively, we can see that the the first two derivatives of ρ will affect the efficiency of inferences. If ρ' are ρ'' are jumping wildly, a large amount of variation will be induced to the estimated variance $\hat{V}$. Recall that we will be using the estimated density function $\hat{f}$ in the ρ function, and existing proper methods to get $\hat{f}$ present unstable first two derivatives. We hence establish a novel penalized density estimation particularly to address the problems of derivatives. Chapter 3 discusses more details about the penalization.

Prediction interval

[Fisher and Horn, 1994] proposed robust prediction intervals in a regression setting. However, it is in a parametric prediction interval, assuming t-distribution in error term. [Frey, 2013] proposed nonparametric prediction intervals. However, it is an alternative form of order statistics, thus only works for median (one-sample) model, not works for robust linear regression model. [Lei and Wasserman, 2014] proposed a conformal prediction based nonparametric prediction interval. However, it is naturally a (local smoothing) nonparametric regression, which does not mesh well with the most-often used linear models.

To the best of our knowledge, there is no very satisfying solution to do nonparametric robust linear model prediction interval. While ours is straightforward, with uncertainty of new observations Y_{new} fully modeled in a nonparametric manner.

Consistency of M-estimators

Many papers discussed consistency of M-estimators with a convex discrepancy functions, including [Huber, 1964], [Huber, 1973], [Yohai, 1974], [Welsh, 1989], [Bai et al., 1992], [Niemiro, 1992], and [Wu, 2007].

Non-convex discrepancy functions, corresponding to redecending M-estimations, however, are recognized to have good properties against influential outliers. For the one-parameter case, suppose $Y_1, \ldots, Y_n$ are independent random variables; interest is in estimating θ_0. Define

$$M_n(\theta; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho(Y_i - \theta),$$

as the empirical criterion, where $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^\top$. Define

$$M(\theta) = \int_{-\infty}^{\infty} \rho(y - \theta) f(y) dy$$

as the expected criterion. [van der Vaart, 1998], section 5.7, proves convergence of $\hat{\theta}$ under the assumption that

$$\sup_{\theta} |M_n(\theta; \boldsymbol{Y}) - M(\theta)| \xrightarrow{p} 0.$$

[Zhang, 2017] gives a review of the asymptotics of M estimation based on empirical process, and establishes conditions of $M_n(\theta; y)$, for consistency. However, conditions either based on $M_n(\theta; y)$ or $M(\theta)$ require a fully determined error density f to verify, as example cases illustrated in [Zhang, 2017]. This actually contradicts the very spirit of robustness.

In Chapter 2, we present a straight-forward proof of consistency of the redescending M-estimator, requiring only mild conditions on the shape of the distribution, and the shape of ρ function, based on ideas using Glivenko-Cantelli type argument.

1.2.3 Nonparametric Density Estimation

The general expression for Nonparametric Density Estimations (NPDE) is

$$p(x) = \frac{k}{NV},$$

where N is the total number of observations, V is the volume of a region which x belongs to, and k is the number of observations in the same region of x. When applying this result to practical density estimation problems, three basic approaches can be adopted.

Histogram: the simplest form of NPDE. It is with regions to be fixed at equal intervals (bins). One of its biggest problems is discontinuities at the region boundaries.

Kernel Density Estimation (KDE): basically resembles the histogram, with the exception that the bin locations are determined by the data points. Define

$$k = \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right),$$

and plug in back to get

$$p(x) = \frac{1}{Nh} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right),$$

where h is the bandwidth. Use regions with soft edges (smooth kernel, such as gaussian kernel) to avoid discuntinuities.

K Nearest Neighbor (KNN): choose a fixed value of k and determine the corresponding volume V from the data. One of its biggest problems is that it does not integrate to one, violating the important assumption of density functions.

Through the previous discussions, and more results in Chapter 2, we would need a nonparametric density estimation which must to be unimodal, symmetric, integral to one, and ideally smooth with stabilized first two derivatives. To satisfy these features, we therefore establish a penalized unimodal spline density estimation. See [Meyer, 2012] for a review of nonparametric constrained density estimations.

1.3 Organization

This dissertation establishes a novel method of robust estimation, which is highly adaptive to the data, so that without the need of priori of error distribution. We propose a penalized unimodal spline density estimation, as a necessary prerequisite of the new method of robust estimation.

In Chapter 2, we present mild conditions to ensure consistency of M-estimations, in terms of density functions f and the ρ function in an M-estimator. The proof of consistency is based on ideas using Glivenko-Cantelli type argument. Different from existing results, the conditions regarding f and ρ are clear, and do not require the error density to be fully determined or presumed, which conform to the spirit of robustness. The presented conditions also provide a guidance for us to figure out the necessary properties of the new density estimation that will be used in the new method of robust estimation.

In Chapter 3, we first propose the penalized spline density estimation, and show the sufficient conditions to maintain its convergence rate. A cross-validation method is also proposed to select a good penalty constant for the new density estimation. We then apply the new density estimation to establish the new method of robust estimation for one-sample model, with its asymptotic properties derived. The new method of robust estimation is extended to linear regression problem, with a straightforward solution of robust prediction interval. An example of one-sample problem, and an example of simple linear regression are given in this chapter, and we end up this chapter with a simulation study which indicates a good performance of our new method of robust estimation.

In Chapter 4, we further extend our new method of robust estimation to the scenarios of heavy-tailed and dependent errors. The dependency of errors are modeled by AR(1) and more general AR(p) autoregressive models. A two-steps procedure with a warming-up step is proposed, for the sake of finding a good starting point in optimization algorithms. An example of application is presented to demonstrate the usage of the new method, with a data set from earth sciences. A simulation study is presented in this chapter as well.

The detailed algorithm of the penalized unimodal spline density, and the proof of consistency of the new density estimation can be found in the Appendix.

Chapter 2

Consistency of Redescending M-estimators

2.1 Introduction

[Huber, 1964] proposed M-estimation, a broad family of robust estimators. For the one-parameter case, suppose $Y_1, \ldots, Y_n$ are independent random variables with a common distribution f, which is assumed to be symmetric about θ_0; interest is in estimating θ_0. Given a discrepancy function ρ that is symmetric about zero, the M-estimator $\hat{\theta}$ is the value of θ that minimizes the expression

$$M_n(\theta; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho(Y_i - \theta),$$

where $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^\top$. If ρ is a convex function, then M_n is convex and has a unique minimum that can be found with standard gradient-based methods. If ρ is not convex, the M_n function may have many local minima, and finding the global minimum requires a more careful algorithm and a good starting point.

[Huber, 1964] proved consistency with a convex discrepancy function, and for non-convex cases wrote "apparently one has to impose not only local but global conditions on ρ for consistency." [Huber, 1973], [Yohai, 1974], and [Welsh, 1989] provided asymptotics for the case where the number of parameters increases with n. [Bai et al., 1992] and [Niemiro, 1992] further developed the asymptotic theory, establishing a necessary set of conditions for consistency and considering the non-smooth discrepancy functions such as for least-absolute-deviations estimators. [Wu, 2007] studied asymptotic properties of M-estimates of regression parameters in linear models in which errors are dependent. For all of these papers, consistency is established under a convex discrepancy function.

Non-convex loss functions, however, are recognized to have good properties for more extreme distributions. An M-estimator is "redescending" if the discrepancy function flattens out so that $\psi(y) = \rho'(y)$ decreases to zero as y increases. [Hampel et al., 2011b] discussed the efficiency of

redescending M-estimators for very heavy-tailed data. They describe three popular choices of ρ functions, shown in Figure 2.1 along with the well-known convex Huber function.

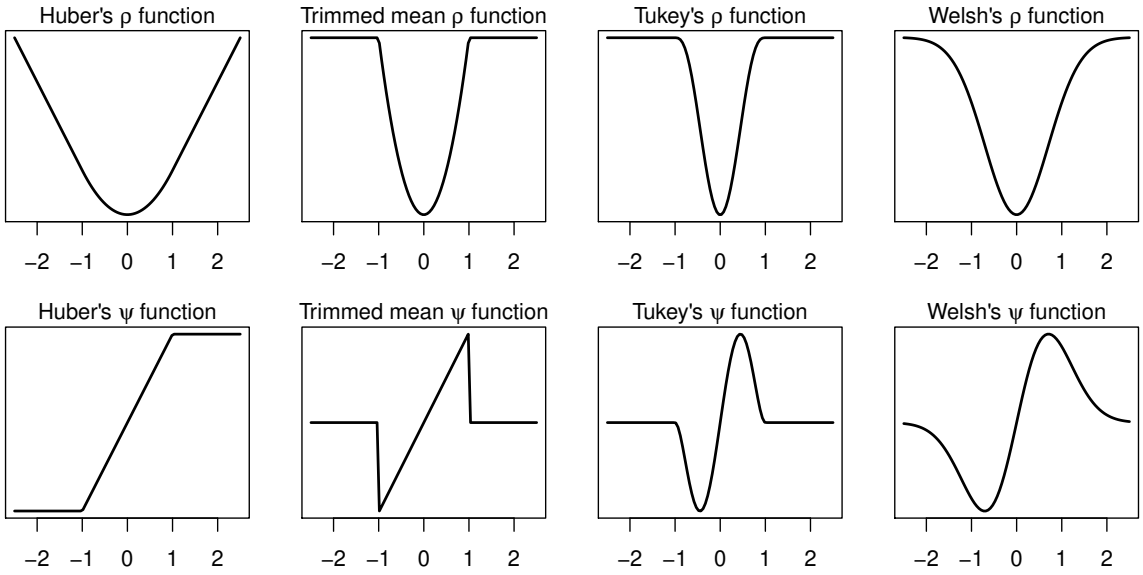

Figure 2.1: The shape of ρ functions and $\psi = \rho'$ functions, for Huber's and redescending M-estimators

Define

$$M(\theta) = \int_{-\infty}^{\infty} \rho(y-\theta) f(y) dy;$$

then for any fixed θ, $M_n(\theta; \boldsymbol{Y}) \overset{a.s.}{\to} M(\theta)$ by the strong law of large numbers. As pointed out in [van der Vaart, 1998], section 5.7, what is needed is functional convergence. They prove convergence of $\hat{\theta}$ under the *assumption* that $\sup_\theta |M_n(\theta; \boldsymbol{Y}) - M(\theta)| \overset{p}{\to} 0$. [Berlinet et al., 2000] presented conditions that are necessary as well as sufficient, for consistency of M estimators in a wide range of conditions, drawing on sophisticated results of general asymptotic theory. [Zhang, 2017] gives a review of the asymptotics of M estimation based on empirical process.

In this chapter, we present a straight-forward proof of consistency of the redescending M-estimator, requiring only the law of large numbers, based on ideas using Glivenko-Cantelli type argument.

2.2 Main Results

We impose the following conditions on the redescending ρ function and on the underlying density f.

A1. $\rho : \mathbb{R} \to [0, \infty)$ is a continuous function increasing on $(0, \infty)$ and $\rho(-y) = \rho(y)$;

A2. ρ is strictly increasing on $(0, C)$, for some $C > 0$;

A3. $f(y)$ is strictly decreasing on (θ_0, ∞) and symmetric about θ_0;

A4. there is a ρ_{max} so that for any $\epsilon > 0$, we can find D_ϵ, such that $0 \leq \rho_{max} - \rho(y) < \epsilon$ for all $y > D_\epsilon$.

Lemma 1. *Under conditions A1-A2, $M(\theta)$ is strictly decreasing on $(-\infty, \theta_0)$ and strictly increasing on (θ_0, ∞).*

Proof. Without loss of generality suppose $\theta_0 = 0$. By symmetry of ρ and f, it follows that M is an even function. Choose any $0 < \theta_1 < \theta_2$; then

$$\begin{aligned} M(\theta_2) - M(\theta_1) &= \int_{-\infty}^{\infty} \Big[\rho(y - \theta_2) - \rho(y - \theta_1)\Big] f(y) dy \\ &= \int_{-\infty}^{(\theta_1+\theta_2)/2} \Big[\rho(y - \theta_2) - \rho(y - \theta_1)\Big] f(y) dy \\ &\quad + \int_{(\theta_1+\theta_2)/2}^{\infty} \Big[\rho(y - \theta_2) - \rho(y - \theta_1)\Big] f(y) dy. \end{aligned}$$

We exploit the fact that $\rho(y - \theta_2) - \rho(y - \theta_1)$ is an odd function about $(\theta_1 + \theta_2)/2$, by a change of variable $z = \theta_1 + \theta_2 - y$ in the first integral. With this change of variable, the above becomes

$$\begin{aligned} &- \int_{(\theta_1+\theta_2)/2}^{\infty} \Big[\rho(\theta_2 - z) - \rho(\theta_1 - z)\Big] f(\theta_1 + \theta_2 - z) dz \\ &+ \int_{(\theta_1+\theta_2)/2}^{\infty} \Big[\rho(y - \theta_2) - \rho(y - \theta_1)\Big] f(y) dy \\ &= \int_{(\theta_1+\theta_2)/2}^{\infty} \Big[\rho(y - \theta_2) - \rho(y - \theta_1)\Big] \Big[f(y) - f(y - (\theta_1 + \theta_2))\Big] dy, \end{aligned}$$

by symmetry of ρ and f. This integral is positive because we have that

$$\rho(y-\theta_2) < \rho(y-\theta_1) \text{ and}$$
$$f(y) < f\Big(y-(\theta_1+\theta_2)\Big), \text{ when } y > \frac{\theta_1+\theta_2}{2}.$$

$\diamondsuit$

Next we show that M_n converges uniformly to M under the assumptions.

Lemma 2. *Under conditions* $A1-A4$, $\sup_{\theta\in\mathbb{R}}\Big|M_n(\theta)-M(\theta)\Big| \overset{a.s.}{\to} 0$.

Proof. Our goal is to show that for any $\epsilon > 0$, there is a finite N such that for all $n > N$, $|M_n(\theta)-M(\theta)| < \epsilon$ holds almost surely for any $\theta \in \mathbb{R}$. We will be first defining a collection of bracket functions which is free of θ and "brackets" $\rho(y-\theta)$, piecewise. We then will be showing that for any $\epsilon > 0$, we can find bracket functions with "gaps" all under ϵ. For any given collection of finite numbers of bracket functions, there is a finite N such that for all $n > N$, each of the bracket functions deviates from its mean within ϵ almost surely. Finally, the deviation of $M_n(\theta)$ from its mean will be bounded almost surely, for any $\theta \in \mathbb{R}$.

We first see how can we define the bracket functions. Let us keep in mind that by the strong law of large numbers, $M_n(\theta)$ converges almost surely to $M(\theta)$ for any fixed θ or any finite collection of θ values. If we choose $-\infty < \theta_1 < \theta_2 < \cdots < \theta_{k-1} < \theta_k < \infty$, we may define a sequence of "bracket functions" as follows,

$$\ell_0(y) = \begin{cases} 0 & \text{for } y \in (-\infty, \theta_1] \\ \rho(y-\theta_1) & \text{for } y \in (\theta_1, \infty); \end{cases}$$

$$u_0(y) = \rho_{max}, \text{ for } y \in (-\infty, \infty);$$

$$\ell_k(y) = \begin{cases} \rho(y-\theta_k) & \text{for } y \in (-\infty, \theta_k) \\ 0 & \text{for } y \in [\theta_k, \infty); \end{cases}$$

$$u_k(y) = \rho_{max}, \text{ for } y \in (-\infty, \infty);$$

and for $j = 1, 2, \cdots, k-1$,

$$\ell_j(y) = \begin{cases} \rho(y-\theta_j) & \text{for } y \in (-\infty, \theta_j) \\ 0 & \text{for } y \in [\theta_j, \theta_{j+1}] \\ \rho(y-\theta_{j+1}) & \text{for } y \in (\theta_{j+1}, \infty); \end{cases}$$

$$u_j(y) = \begin{cases} \rho(y-\theta_{j+1}) & \text{for } y \in \Big(-\infty, \frac{\theta_j+\theta_{j+1}}{2}\Big] \\ \rho(y-\theta_j) & \text{for } y \in \Big(\frac{\theta_j+\theta_{j+1}}{2}, \infty\Big). \end{cases}$$

Note that if we choose $\theta \in [\theta_j, \theta_{j+1}]$, for some $j = 1, 2, \cdots, k-1$, then $\ell_j(y) \le \rho(y-\theta) \le u_j(y)$. In addition, for any $\theta < \theta_1$ we have $\ell_0(y) \le \rho(y-\theta) \le u_0(y)$ and finally, for any $\theta > \theta_k$ we have $\ell_k(y) \le \rho(y-\theta) \le u_k(y)$. Thus, for any $\theta \in \mathbb{R}$, we can find $j \in \{0, \ldots, k\}$ such that $\rho(y-\theta)$ is bracketed by $\ell_j(y)$ and $u_j(y)$.

Now we discuss that for a given $\epsilon > 0$, how can we determine k and a collection of bracket functions with "gaps" all bounded. For any $\epsilon > 0$, by continuity and boundedness of ρ, we can find $\delta > 0$ such that $|\rho(y+\delta) - \rho(y)| \le \epsilon$ for all $y \in \mathbb{R}$. Therefore if $\theta_{j+1} - \theta_j < \delta$ for $j = 1, \ldots, k-1$, we have

$$\int_{-\infty}^{\infty} \Big[u_j(y) - \ell_j(y)\Big] f(y) dy < \epsilon.$$

Let f_p be the pth quantile of f, that is, $\int_{-\infty}^{f_p} f(y)dy = p$. Choose $\theta_1 = f_{\epsilon/(2\rho_{max})} - D_{\epsilon/2}$; then

$$\begin{aligned}\int_{-\infty}^{\infty} \Big[u_0(y) - \ell_0(y)\Big] f(y)dy =& \int_{-\infty}^{f_{\epsilon/(2\rho_{max})}} \Big[u_0(y) - \ell_0(y)\Big] f(y)dy \\ &+ \int_{f_{\epsilon/(2\rho_{max})}}^{\infty} \Big[u_0(y) - \ell_0(y)\Big] f(y)dy \\ \leq& \frac{\epsilon}{2} + \frac{\epsilon}{2} \\ =& \epsilon.\end{aligned}$$

We can choose k to be the integer larger than $-2\theta_1/\delta$ (note that θ_1 is negative), then $\theta_k > -\theta_1$, and by symmetry, $\int_{-\infty}^{\infty}[u_k(y) - \ell_k(y)]f(y)dy \leq \epsilon$.

Then we see that there is a finite N such that each of the bracket functions deviates from its mean within ϵ almost surely. For any $\epsilon > 0$, k is finite, and by the strong law of large numbers, when $y_1, \ldots, y_n \overset{iid}{\sim} f$,

$$\sup_{j=0,\ldots,k} \left[\frac{1}{n}\sum_{i=1}^{n} \ell_j(y_i) - \int_{-\infty}^{\infty} \ell_j(y)f(y)dy\right] \overset{a.s.}{\to} 0.$$

and

$$\sup_{j=0,\ldots,k} \left[\frac{1}{n}\sum_{i=1}^{n} u_j(y_i) - \int_{-\infty}^{\infty} u_j(y)f(y)dy\right] \overset{a.s.}{\to} 0.$$

We finally show the lemma. For any $\theta \in \mathbb{R}$, let $j(\theta) = 1$ if $\theta \leq \theta_1$, let $j(\theta) = k$ if $\theta > \theta_k$, otherwise $j(\theta)$ is such that $\theta_j < \theta \leq \theta_{j+1}$. Then we have

$$\begin{aligned}M_n(\theta) - M(\theta) &\leq \frac{1}{n}\sum_{i=1}^{n} u_{j(\theta)}(y_i) - \int_{\infty}^{\infty} \ell_{j(\theta)}(y)f(y)dy \\ &\leq \left|\frac{1}{n}\sum_{i=1}^{n} u_{j(\theta)}(y_i) - \int_{\infty}^{\infty} u_{j(\theta)}(y)f(y)dy\right| + \epsilon,\end{aligned}$$

and

$$\begin{aligned} M_n(\theta) - M(\theta) &\geq \frac{1}{n}\sum_{i=1}^{n} \ell_{j(\theta)}(y_i) - \int_{\infty}^{\infty} u_{j(\theta)}(y) f(y) dy \\ &\geq -\left| \frac{1}{n}\sum_{i=1}^{n} \ell_{j(\theta)}(y_i) - \int_{\infty}^{\infty} \ell_{j(\theta)}(y) f(y) dy \right| - \epsilon \end{aligned}$$

So for any $\epsilon > 0$, there is a finite N such that for all $n > N$,

$$|M_n(\theta) - M(\theta)| \leq \epsilon + \epsilon$$

holds for any $\theta \in \mathbb{R}$. Then because ϵ may be arbitrarily small (using $\epsilon/2$ to define the bracket functions and N), we have

$$\sup_{\theta \in R} \left| M_n(\theta) - M(\theta) \right| \xrightarrow{a.s.} 0. \tag{2.1}$$

$\diamondsuit$

Theorem 1. *Suppose $\hat{\theta}_n$ minimizes $M_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \rho(Y_i - \theta)$, where Y_i are iid from f. Under conditions $A1 - A4$, $\hat{\theta}_n \xrightarrow{p} \theta_0$.*

Proof. Again assume without loss of generality that $\theta_0 = 0$, and choose some $\epsilon > 0$. When $|\hat{\theta}| > \epsilon$, it must be that $\inf_{|\theta| > \epsilon} M_n(\theta) < M_n(0)$. Let $M(\epsilon) = \eta$; by Lemma 1, $\eta > 0$ and $M(\theta) - M(0) > \eta$

whenever $\theta > \epsilon$. Then we have

$$\begin{aligned}
\mathrm{P}(|\hat{\theta}| > \epsilon) &\leq \mathrm{P}\left(\inf_{|\theta|>\epsilon} M_n(\theta) < M_n(0)\right) \\
&= P\left(\inf_{|\theta|>\epsilon} M_n(\theta) - M(\epsilon) + M(\epsilon) - M(0) + M(0) - M_n(0) < 0\right) \\
&\leq P\left(\inf_{|\theta|>\epsilon} M_n(\theta) - M(\epsilon) + M(0) - \psi(0) < -\eta\right) \\
&\leq P\left(\inf_{|\theta|>\epsilon} M_n(\theta) - M(\epsilon) < -\frac{\eta}{2}\right) + P\left(M(0) - M_n(0) < -\frac{\eta}{2}\right) \\
&\leq P\left(\sup_{|\theta|>\epsilon} |M_n(\theta) - M(\theta)| > \frac{\eta}{2}\right) + P\left(|M(0) - M_n(0)| > \frac{\eta}{2}\right).
\end{aligned}$$

The first term goes to zero by Lemma 2, and the second goes to zero by the law of large numbers. Because ϵ is arbitrarily small, this completes the proof of consistency. $\diamondsuit$

2.3 Numerical Studies

Simulation studies show that the redescending estimators have smaller MSE than the convex-ρ estimators, when the errors are heavy-tailed or very contaminated. Specifically, we compare the MSEs for the mean, median, and Huber, with Tukey's and Welsh's redescending M-estimations, for various types of error distributions. The discrepancy functions of the estimators are described in Table 2.1, along with default tuning parameters designed to give 95% asymptotic efficiency at the normal distribution; see [Holland and Welsch, 1977] for details.

The `R` function `rlm` will compute M estimators for the above discrepancy functions, scaling the data by its standard deviation to create a scale-invariant estimator. We simulate data sets from several error densities:

1. The standard normal distribution: $N(0, 1)$;

2. The t-distribution with two degree of freedom: $t(2)$;

Table 2.1: Discrepancy functions and default tuning parameters.

Name	$\rho(r)$	Default k
Huber	$\begin{cases} \frac{1}{2}r^2 & \text{if } \lvert r\rvert \le k \\ k\lvert r\rvert - \frac{1}{2}k^2 & \text{if } \lvert r\rvert > k \end{cases}$	$k = 1.345$
Tukey	$\begin{cases} 1 - [1 - (r/k)^2]^3 & \text{if } \lvert r\rvert \le k \\ 1 & \text{if } \lvert r\rvert > k \end{cases}$	$k = 4.685$
Welsh	$1 - e^{-(r/k)^2}$	$k = 2.985$

3. The Pareto distribution with shape parameter 2 and scale parameter 5, randomly multiplied by -1 or $+1$, for symmetry: Pareto$(2, 5)$;

4. Various contaminated normal distributions: 20%N(100) indicates a mixture of normals with 80% N$(0, 1)$ and 20%N$(0, 100)$

5. t-contaminated normal: 20%t$(2, 5)$ indicates a mixture with 80% standard normal and 20% t(2) multiplied by 5.

Table 2.2 contains the square root of the mean squared error for five estimators of the mean, with three examples of convex ρ functions, and two redescending estimators. For the latter, we use the default k as well as the "oracle" k value, which is found through preliminary simulations at many values of k for each error density. This is the optimal k if the error density is known.

The redescending M-estimators outperform the convex M-estimators when the errors are heavy-tailed or very contaminated, and are comparable when the errors are normal. The plots in Figures 2.2 and 2.3 show how the mean squared error changes with k, for two example error densities. Although the default k performs well with normal errors, if the errors are known to be heavy-tailed or contaminated, a somewhat smaller tuning parameter might considerably reduce the estimation error.

Table 2.2: Square root of mean squared errors for estimators, computed from 10000 simulated data sets, for each of two sample sizes and nine true distributions. The "oracle" k value is optimal for the error density.

Error Density	n	Mean	Median	Huber	Tukey		Welsh	
					Default	Oracle	Default	Oracle
Std norm	100	.099	.124	.102	.102	.100	.102	.099
	500	.045	.056	.046	.046	.045	.046	.045
$t(2)$	100	2.000	.705	.703	.684	.667	**.681**	.658
	500	.922	.319	.316	.307	.299	**.305**	.295
Pareto$(2,5)$	100	3.362	**.282**	.398	.367	.307	.364	.295
	500	.879	**.118**	.176	.162	.134	.161	.129
$20\%N(100)$	100	.455	.151	.146	**.124**	.121	.125	.122
	500	.201	.068	.065	**.055**	.054	.056	.055
$50\%N(100)$	100	.715	.232	.389	.337	.180	**.336**	.181
	500	.319	.102	.156	**.131**	.077	.132	.078
$80\%N(100)$	100	.895	.481	.858	.860	.412	**.845**	.434
	500	.406	.205	.389	.388	.160	**.382**	.178
$20\%t(2,5)$	100	.838	.148	.139	**.125**	.123	.126	.123
	500	.370	.067	.063	**.057**	.055	**.057**	.056
$50\%t(2,5)$	100	1.4888	.212	.278	.244	.182	**.242**	.181
	500	.677	.096	.120	**.106**	.080	**.106**	.080
$80\%t(2,5)$	100	1.826	.374	.541	.515	.383	**.509**	.372
	500	.845	.165	.242	.229	.160	**.227**	.160

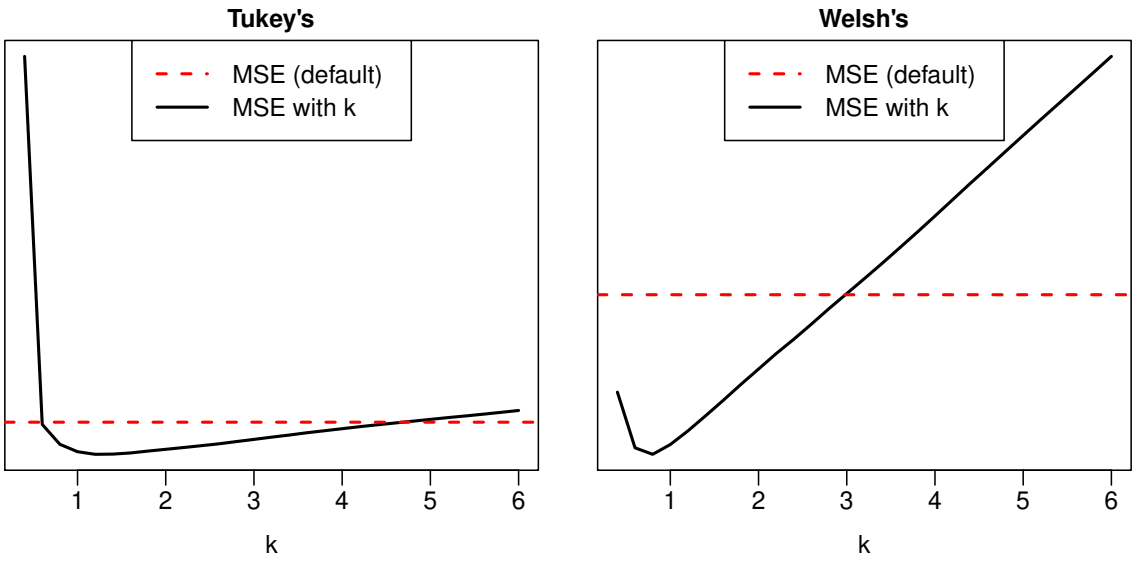

Figure 2.2: Performance of Redescending M-estimations against Pareto(2,5) error.

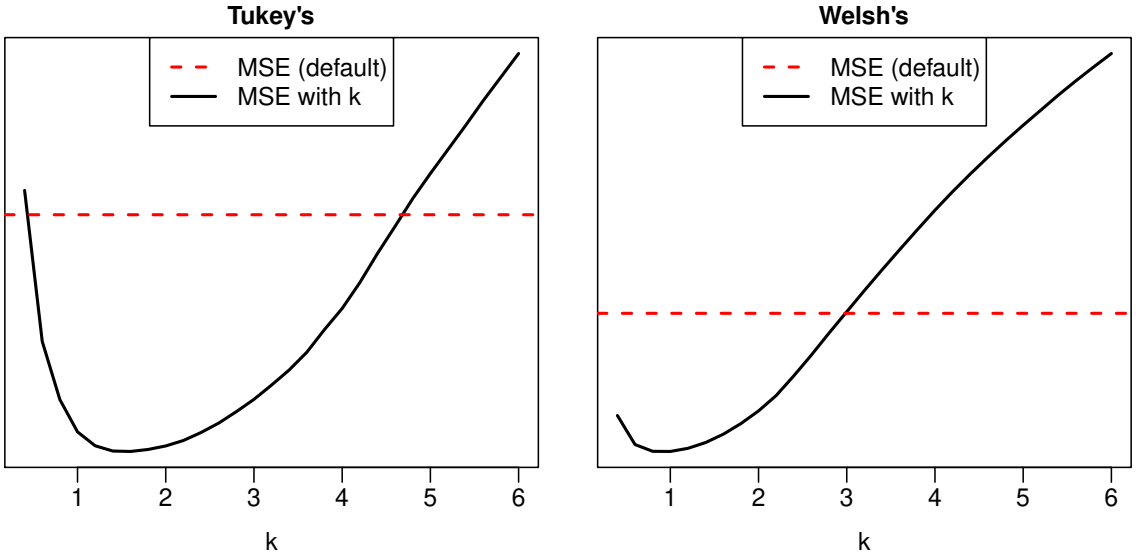

Figure 2.3: Performance of Redescending M-estimations against $50\%N(1, 100)$ error.

2.4 Discussion

This chapter establishes consistency of redescending M-estimators, with a novel proof based on an idea of Glivenko-Cantelli classes, and mild conditions in terms of the error density f and the loss function ρ.

To show the consistency, we first show a lemma that under given mild conditions, the expected criterion function is unimodal and symmetrical, which means that it has a well-defined minimum. We then show another lemma that the empirical criterion function has uniform convergence to the expected criterion function, by defining a set of bracket functions. We finally show the consistency of redescending M-estimators, with the two established lemmas.

Particularly, our proof has mild and explicit conditions. It basically requires the error density f and the loss function ρ to be unimodal and symmetrical, but the error density does not need to be fully determined or following any specific known distribution. This satisfies the very spirit of robust statistics. Moreover, according to the presented conditions, in the next chapter, we establish a penalized unimodal spline density estimation, to satisfy the requirements of our Auto-Adaptive M-estimation (AAME).

On the other hand, the simulation study in Section 2.3 indicates that redescending M-estimators outperform those M-estimators with a convex loss function ρ. However, redescending M-estimations traditionally use tuning parameters not adaptive enough to the data. Their performance can be further improved by using a data-driven tuning parameter.

Chapter 3

Penalized Unimodal Spline Density Estimate with Application to M-Estimation

3.1 Introduction

Suppose we observe $(\boldsymbol{x}_i, Y_i)$, $i = 1, \ldots, n$, and assume $Y_i = \mu(\boldsymbol{x}_i) + \varepsilon_i$, where ε_i are *iid* symmetric mean-zero errors. In the presence of outliers or other evidence that the convenient assumption of normal errors is incorrect, practitioners turn to robust methods where knowledge of the error density family is not required. Classical robust M-estimation of the regression function $\mu(\boldsymbol{x})$ is accomplished by choosing a function ρ and then finding μ to minimize $\sum_{i=1}^{n} \rho(Y_i - \mu(\boldsymbol{x}_i))$ over some parameter or function space. This was first proposed by [Huber, 1964], with further asymptotic results in [Huber, 1973]. [Yohai, 1987] proposed an iterative method for an MM estimator, where the ρ function is adjusted based on the residuals from an initial M estimator. [Hall and Jones, 1990] proposed a robust non-parametric regression estimator combining kernel regression of the regression function with the Huber ρ function. The robust estimator of [Agostinelli and Markatou, 1998] uses weighted regression to down-weight outliers and high leverage points. [Gervini and Yohai, 2002] used the empirical distribution function to obtain a weight function for a robust weighted regression. [Bondell and Stefanski, 2013] proposed an empirical-likelihood framework that down-weights outlying observations by measuring the divergence between the empirical likelihood and the normal error distribution.

In this chapter we lay the foundations of a novel robust regression method, where the error density is estimated with constrained penalized splines. We start with the one-sample problem

$$Y_i = \mu + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.1}$$

where the ε_i are independent, symmetric mean-zero random variables with common density f_0. The M-estimator uses a function ρ and sample values $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, determine $\hat{\mu} \in \mathbb{R}$ to minimize

$$M_n(\mu; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho(Y_i - \mu). \tag{3.2}$$

If the error density family is known, the ρ function may be chosen accordingly. The quadratic function $\rho(y) = y^2$ is optimal for normal errors, and leads to least-squares estimation where the sample mean is the minimizer. Similarly, if the errors are exponential, the best choice is $\rho(y) = |y|$, and the sample median is the minimizer. The function $\rho(y) = y^2$ when $|y| \leq k$ and $\rho(y) = 2k|y| - k^2$ for $t > k$ was suggested by [Huber, 1964] and shown to have nice properties when the errors can be described as contaminated normal. These ρ functions are shown in Figure 3.1. Especially, the tuning parameter k plays a crucial role for the performance of Huber estimator. A few papers established successful methods to make k be adaptive to the tail-thickness of error densities, and have been widely used. For example, see [Wang et al., 2018] and [Sun et al., 2019].

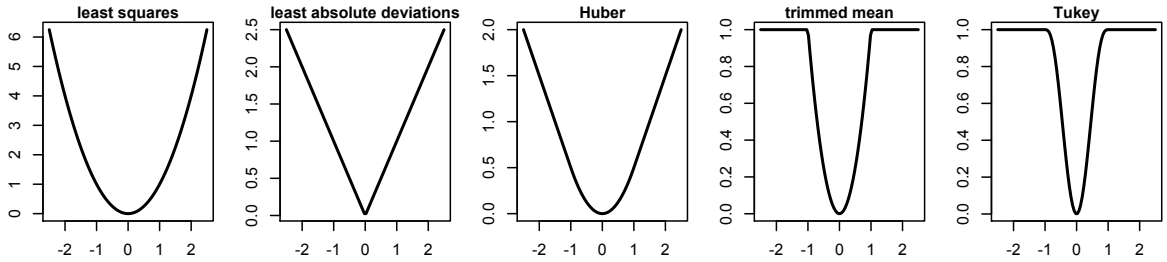

Figure 3.1: Some example ρ functions for Huber's M estimation

The function $\rho(y) = y^2$ for $|y| \leq k$ and $\rho(y) = k^2$ for $|y| > k$ leads to the "trimmed mean," which completely discounts outliers beyond the trim-point k. [Huber, 1964] wrote that although the trimmed mean has nice properties, it "is rather sensitive to the behavior of F at $\pm k$ ... one might avoid it by smoothing ρ at k." This is accomplished by Tukey's biweight ρ function, shown in Figure 3.1 along with the trimmed mean ρ function. [Hampel et al., 2011a] argued that "redescending" ρ functions, where the derivative approaches zero for large y, are effective for heavy-tailed or contaminated errors.

For convex ρ functions, it is straight-forward to show that the minimizer $\hat{\mu}$ of M_n is unique and consistent for μ. An expression for the asymptotic variance of $\hat{\mu}$ can be derived, leading to

$$V(\hat{\mu}) \approx \frac{\int_{-\infty}^{\infty} \rho'(y)^2 f_0(y) dy}{n \left[\int_{-\infty}^{\infty} \rho''(y) f_0(y) dy \right]^2}.$$

Implementation of this expression requires knowledge or estimation of the density f_0.

For the proposed method, f is f_0 truncated to a "large" support $[-S, S]$, and the density f is estimated using a constrained least-squares penalized spline density estimator, which is then used for inference. When the objective function for the least-squares density estimator is minimized over the density and the regression function simultaneously, the estimate of the regression function is then equivalent to an M-estimator, with its ρ function defined as the density estimate "flipped upside down." For computational and theoretical reasons, we estimate the density function using the median as the initial estimator of μ, and use the negative of the density estimate as the ρ function. In practice this ρ function is similar to that for the "smoothed" trimmed mean suggested by Huber and Tukey, with a data-driven choice of trim points. We call this the Auto-Adaptive M-Estimator (AAME). If the estimated density is $\hat{f}$, we may estimate the variance of $\hat{\mu}$ as

$$\hat{V}(\hat{\mu}) = \frac{\int_{-\infty}^{\infty} \hat{f}'(y)^2 \hat{f}(y) dy}{n \left[\int_{-\infty}^{\infty} \hat{f}''(y) \hat{f}(y) dy \right]^2}. \tag{3.3}$$

In the next section, we develop the constrained penalized spline density estimator and derive a rate of convergence. In Section 3.3, the density estimation method is applied to the one-sample problem, and we derive root-n convergence of $\hat{\mu}$ and propose a confidence interval for μ. In Section 3.4, AAME is applied to regression problems, confidence intervals for regression coefficients are presented, and we establish a solution of robust prediction interval. Simulations in Section 3.5 show that AAME out-performs other robust estimators when the true error density results in many large "outliers." We also apply this method to some macroeconomic time series data for one-

sample model in Section 3.3; and to a healthcare study for simple linear regression model in Section 3.4.

3.2 Spline Estimation of a Unimodal Density

3.2.1 Least-Squares Spline Density Estimation

Suppose we have a random sample $Y_1, \ldots, Y_n$ from a density f_0 that is known to be smooth, unimodal, and symmetric about zero. Define g_0 to be the density of $X = |Y|$; we will estimate a smooth decreasing density from $X_1, \ldots, X_n$, the absolute values of the random sample, and convert it to a symmetric unimodal density. We use the quadratic loss function proposed by [Groeneboom et al., 2001]. For a candidate decreasing density g and observations $X_1, \ldots, X_n$, define

$$\mathcal{L}(g) = \int_0^\infty g^2(x)dx - 2\sum_{i=1}^{n} g(X_i).$$

This loss function was used for the (unpenalized) spline density estimates of [Meyer, 2012]; that method is summarized here.

For a quadratic spline basis, there exist necessary and sufficient linear inequality constraints for monotonicity. We consider a support $[0, S]$ and distinct knots $0 = t_1 < t_2 < \cdots < t_{J-1} < t_J = S$ (the choices of S, J, and specific knot placement will be discussed in Section 3.3.1. Because the density is *a priori* unimodal, we increase the spacing of the knots away from the origin; in this way the knots are closer together where there are more observations. The spline basis functions given these knots are as follows. For $j = 1, ..., J-2$,

$$\delta_j(x) = \begin{cases} 1 & 0 \le x < t_j \\ 1 - \frac{(x-t_j)^2}{(t_{j+2}-t_j)(t_{j+1}-t_j)} & t_j \le x < t_{j+1} \\ \frac{(t_{j+2}-x)^2}{(t_{j+2}-t_j)(t_{j+2}-t_{j+1})} & t_{j+1} \le x < t_{j+2} \\ 0 & x \ge t_{j+2} \end{cases}$$

with two additional basis functions

$$\delta_{J-1}(x) = \begin{cases} 1 & 0 \le x < t_{J-1} \\ 1 - \frac{(x-t_{J-1})^2}{(t_J - t_{J-1})^2} & t_{J-1} \le x < t_J \end{cases}$$

and $\delta_J(x) \equiv 1$. These basis functions for $J = 9$ are shown in Figure 3.2, where the knots are shown with tick marks.

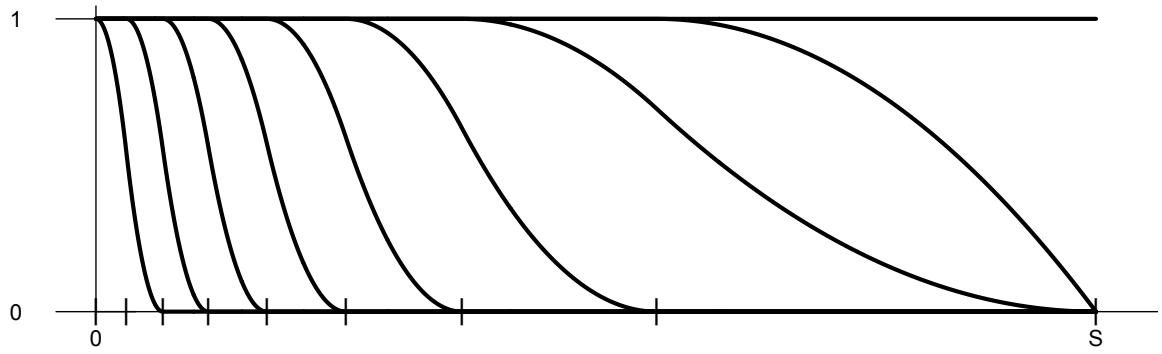

Figure 3.2: Spline basis functions for decreasing density estimation on $[0, S]$.

Consider the class $\mathcal{G}$ of densities of the form $g(x) = \sum_{j=1}^{J} b_j \delta_j(x)$, where $b_j \ge 0$ for $j = 1, \ldots, J$; we estimate f by minimizing $\mathcal{L}(g)$ over this class. Define the $n \times J$ matrix $\boldsymbol{\Delta}$ as $\Delta_{ij} = \delta_j(x_i)$, and further let $H_{j\ell} = \int_0^\infty \delta_j(x)\delta_\ell(x)dx$, and $\boldsymbol{z} = \boldsymbol{\Delta}^\top \mathbf{1}/n$ where the entries of $\mathbf{1} \in \mathbb{R}^n$ are all equal to 1. The last row of $\boldsymbol{H}$ contains the integrals of the basis functions, so that if $\boldsymbol{h}_J$ is the last row, the area under the function $g(x) = \sum_{j=1}^{J} b_j \delta_j(x)$ is $\boldsymbol{h}_J^\top \boldsymbol{b}$. Minimizing $\mathcal{L}(g)$ over the class of decreasing densities is equivalent to minimizing the quadratic loss

$$Q_0(\boldsymbol{b}) = \boldsymbol{b}^\top \boldsymbol{H} \boldsymbol{b} - 2\boldsymbol{z}^\top \boldsymbol{b} \tag{3.4}$$

subject to $\boldsymbol{b} \ge \mathbf{0}$ and $\boldsymbol{h}_J^\top \boldsymbol{b} = 1$.

To convert this into a standard quadratic programming problem, find $\boldsymbol{b}_0$ such that $\boldsymbol{h}_J^\top \boldsymbol{b}_0 = 1$, and define $\boldsymbol{\alpha} = \boldsymbol{b} - \boldsymbol{b}_0$. Then the minimization problem is equivalent to: minimize $\boldsymbol{\alpha}^\top \boldsymbol{H} \boldsymbol{\alpha} - 2(\boldsymbol{z} -$

$\boldsymbol{H}\boldsymbol{b}_0)^\top\boldsymbol{\alpha}$, subject to $\boldsymbol{\alpha} \geq -\boldsymbol{b}_0$ and $\boldsymbol{h}_J^\top\boldsymbol{\alpha} = 0$. Find a $J \times J-1$ matrix $\boldsymbol{W}$ such that the columns span the linear space orthogonal to $\boldsymbol{h}_J$, so that $\boldsymbol{h}_J^\top\boldsymbol{\alpha} = 0$ if and only if $\boldsymbol{\alpha} = \boldsymbol{W}\boldsymbol{\beta}$ for some $\boldsymbol{\beta} \in \mathbb{R}^{J-1}$. The new equivalent problem is: minimize $\boldsymbol{\beta}^\top\boldsymbol{W}^\top\boldsymbol{H}\boldsymbol{W}\boldsymbol{\beta} - 2(\boldsymbol{z} - \boldsymbol{H}\boldsymbol{b}_0)^\top\boldsymbol{W}\boldsymbol{\beta}$, subject to $\boldsymbol{W}\boldsymbol{\beta} \geq -\boldsymbol{b}_0$, and the solution $\hat{\boldsymbol{\beta}}$ is found using the function `solve.QP` in the `R` package `quadprog`. The necessary and sufficient condition for $\hat{\boldsymbol{\beta}}$ being the solution is $(\boldsymbol{Q}\hat{\boldsymbol{\beta}} - \boldsymbol{c})^\top(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \leq 0$, for all $\boldsymbol{\beta}$ such that $\boldsymbol{W}\boldsymbol{\beta} \geq -\boldsymbol{b}_0$ ([Silvapulle and Sen, 2005], Chapter 3). Transforming back, $\hat{\boldsymbol{b}} = \boldsymbol{W}\hat{\boldsymbol{\beta}} + \boldsymbol{b}_0$, and the necessary and sufficient condition becomes

$$(\boldsymbol{H}\hat{\boldsymbol{b}} - \boldsymbol{z})^\top(\hat{\boldsymbol{b}} - \boldsymbol{b}) \leq \boldsymbol{0}, \text{ for all } \boldsymbol{b} \geq \boldsymbol{0} \text{ with } \boldsymbol{h}_J^\top\boldsymbol{b} = 1. \tag{3.5}$$

3.2.2 The Penalized Least-Squares Spline Density Estimator

The second derivative of the spline density estimate will be piecewise constant, with breaks at the knots. Looking forward to our robust regression application, we note that the expression for the asymptotic variance of $\hat{\mu}$ contains the second derivative of the ρ function, which is determined by the density estimate. Over-fitting, or specifying "too many" knots should be avoided as this can result in a wildly varying second derivative function. Too few knots will not give sufficient flexibility to the density estimate, which is especially needed for heavy-tailed data. The solution is a penalized least-squares density estimator: we penalize the consecutive differences of the second derivative function. This stabilizes the estimated density function and its first two derivatives, and allows many knots to be specified without over-fitting.

For a candidate density g, let $\theta_j = g''(x)$ for $x \in (t_j, t_{j+1})$, $j = 1, \ldots, J-1$; the penalized density estimator $\hat{g}_\lambda$ minimizes the penalized quadratic loss over $g \in \mathcal{G}$

$$\mathcal{L}_\lambda(g) = \int_0^S g^2(x)dx - 2\sum_{i=1}^n g(X_i) + \lambda\sum_{j=1}^{J-1}(\theta_{j+1} - \theta_j)^2, \tag{3.6}$$

where $\lambda \geq 0$ is the penalty parameter. To write the penalized loss in vector notation, we note that the second derivative on each knot interval is

$$\theta_j = \begin{cases} \delta_1''(x) b_1 & \text{for } x \in (t_1, t_2), j = 1 \\ \delta_j''(x) b_i + \delta_{j+1}''(x) b_{j+1} & \text{for } x \in (t_j, t_{j+1}), j = 2, ..., J-1 \end{cases}.$$

Because the second derivative of each basis function is piece-wise constant, we have

$$\theta_j = \begin{cases} \frac{-2b_1}{d_1(d_1+d_2)} & \text{for } j = 1 \\ \frac{-2b_i}{d_j(d_j+d_{j+1})} + \frac{2b_{j-1}}{d_j(d_{j-1}+d_j)} & \text{for } j = 2, ..., J-2 \\ \frac{-2b_{J-1}}{d_{J-1}^2} + \frac{2b_{J-2}}{d_{J-1}(d_{J-2}+d_{J-1})} & \text{for } j = J-1. \end{cases},$$

where $d_j = t_{j+1} - t_j$ for $j = 1, \ldots, J-1$. The difference in second derivative between two adjacent intervals is

$$\theta_{j+1} - \theta_j = \begin{cases} \frac{2b_1}{d_1 d_2} - \frac{2b_2}{d_2(d_2+d_3)} & \text{for } j = 1 \\ -\frac{2b_{j-1}}{d_j(d_{j-1}+d_j)} + \frac{2b_j}{d_j d_{j+1}} - \frac{2b_{j+1}}{d_{j+1}(d_{j+1}+d_{j+2})} & \text{for } j = 2, ..., J-3 \\ -\frac{2b_{J-3}}{d_{J-2}(d_{J-3}+d_{J-2})} + \frac{2b_{J-2}}{d_{J-2}d_{J-1}} - \frac{2b_{J-1}}{d_{J-1}^2} & \text{for } j = J-2. \end{cases},$$

Define the $J-2 \times J$ penalty matrix as

$$\boldsymbol{D} = 2 \begin{bmatrix} \frac{1}{d_1 d_2} & \frac{-1}{d_2(d_2+d_3)} & 0 & \cdots & & & 0 \\ \frac{-1}{d_2(d_1+d_2)} & \frac{1}{d_2 d_3} & \frac{-1}{d_3(d_3+d_4)} & 0 & \cdots & & 0 \\ 0 & \frac{-1}{d_3(d_2+d_3)} & \frac{1}{d_3 d_4} & \frac{-1}{d_4(d_4+d_5)} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & \frac{-1}{d_{J-2}(d_{J-3}+d_{J-2})} & \frac{1}{d_{J-2}d_{J-1}} & \frac{-1}{d_{J-1}^2} & 0 \end{bmatrix};$$

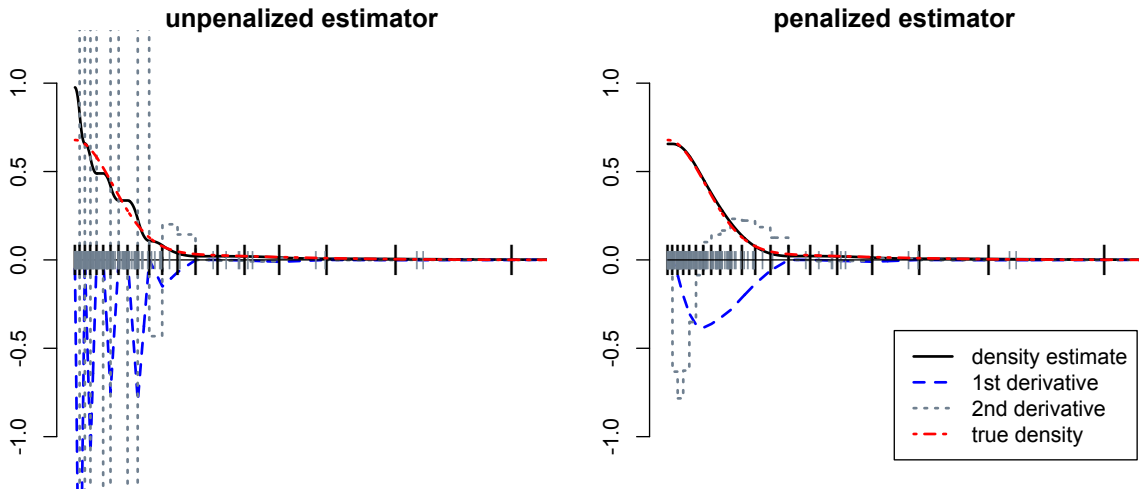

Figure 3.3: Comparing the penalized and unpenalized density estimates for a sample of size $n = 120$ from a mixture of normal densities. The knots are shown by the heavy tick marks and the absolute values of the sample are shown with the lighter tick marks.

so the quadratic loss (3.4), penalized for changes in second derivative, becomes

$$Q_\lambda(\boldsymbol{b}) = \boldsymbol{b}^\top(\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})\boldsymbol{b} - 2\boldsymbol{z}^\top \boldsymbol{b}, \tag{3.7}$$

We find $\hat{\boldsymbol{b}}_\lambda$ to minimize $Q_\lambda(\boldsymbol{b})$ using quadratic programming. The algorithm is the same as outlined in the previous section, with $\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D}$ in place of $\boldsymbol{H}$. Then $\hat{g}_\lambda = \sum_{j=1}^{J} \hat{b}_{\lambda,j}\delta_j$.

One of the effects of the penalty on the density estimate is shown in Figure 3.3. The absolute values of a sample from a mixture of normal densities are shown as small gray tick marks. Density estimates with and without penalty are shown as the black curves. While the density estimates are not that different, their first and second derivatives vary more smoothly with the penalty chosen by cross-validation as described in the next section.

Furthermore, with this penalty, the performance of the density estimation could be maintained against over-fitting. Figure 3.4 shows a comparison of accuracy of unimodel spline density estimations, between penalized and unpenalized situations. In that figure, the empirical risks, by averaging the losses of 500 replicates are presented, across increasing different numbers of knots, with distribution $f = t(2)$ and sample size $n = 100$. Hence we know that in practice we may

feel free to place a lot of knots, then the penalized estimation can still help us to get a good result, automatically, if we select the penalty λ by cross-validation. In contrast, if we use the unpenalized estimation without knowing the infeasible optimal number of knots, its performance gets worse dramatically. Note that even using the infeasible optimal number of knots, the unpenalized estimation still can not outperform the penalized estimation.

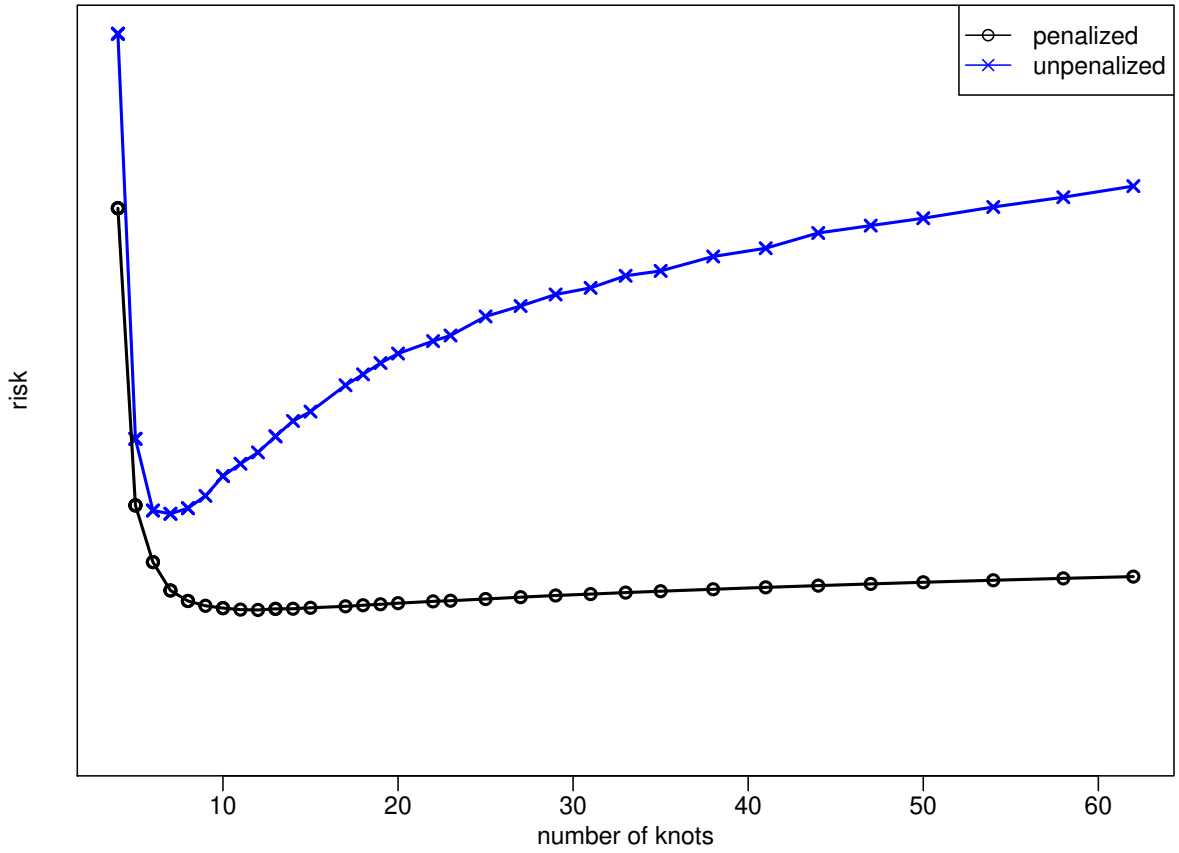

Figure 3.4: Comparison of empirical risk of unimodal spline density estimations, between penalized and unpenalized estimations, with density $f = t(2)$, sample size $n = 100$, and 500 replicates. The lower curve shows the empirical risk of penalized estimation, while the upper curve shows the empirical risk of unpenalized estimation, along with increasing numbers of knots.

3.2.3 Consistency

We obtain a convergence rate for the penalized spline density estimator $\hat{g}_\lambda$, starting with assumptions similar to those in [Meyer, 2012].

(A1) The support of the true density g_0 is $[0, S]$ for some $S \in (0, \infty)$.

(A2) The true density g_0 is twice continuously differentiable and there is a $0 < B < \infty$ such that $-B \leq g_0'(x) \leq 0$ on $(0, S)$.

(A3) Knots are defined in $[0, S]$ according to a scheme that has "bounded mesh ratio," i.e., the ratio of the largest inter-knots intervals to the smallest is bounded for diverging n.

(A4) The number of knots is of order $n^{1/7}$.

For functions on $[0, S]$, define the norm $\|g\|^2 = \int_0^S g(x)^2 dx$. The proof of the following is given in the Appendix.

Theorem 2. *Under (A1)-(A4), if the penalty parameter λ is $O(n^{-12/7})$, then $\|\hat{g}_\lambda - g_0\| = O_p(n^{-3/7})$.*

Theorem 2 tells us that a bounded rate upon the penalty can ensure the consistency of the penalized unimodal spline density estimation. Particularly, at $\lambda = O_p(n^{-12/7})$, the penalized unimodal spline density estimation $\hat{g}_\lambda$ will have the same convergence rate with the unpenalized unimodal spline density estimation $\hat{g}$.

If the theoretical rate is obtained, we should have a log-log relationship showing that

$$log\|\hat{g}_\lambda - \hat{g}\| = log(C) - \frac{3}{7} log(n) + \varepsilon,$$

with the slope being the target convergence rate, $-\frac{3}{7}$. A numerical study verified this fact, which is conducted by regressing the term $log\|\hat{g}_\lambda - \bar{g}\|$ over $log(n)$, with simulated 1000 pairs of norm $\|\hat{g}_\lambda - \bar{g}\|$ across different sample sizes $n \in [50, 10000]$, and distribution standard normal. Table 3.1 shows the simulated slopes (empirical convergence rate) with penalty λ bounded at different rates. The result indicates that $\lambda = O_p(n^{-1})$ is too slow; $\lambda = O_p(n^{-12/7})$ is right above the necessary rate; and $\lambda = O_p(n^{-2})$ is too fast (more than enough). Figure 3.5 shows the simulated pairs of $(log(n), log\|\hat{g}_\lambda - \bar{g}\|)$ and slope (empirical convergence rate) with penalty λ bounded at

rate $\lambda = O_p(n^{-12/7})$, and indicating that the trend of convergence is at an adequate and appropriate rate.

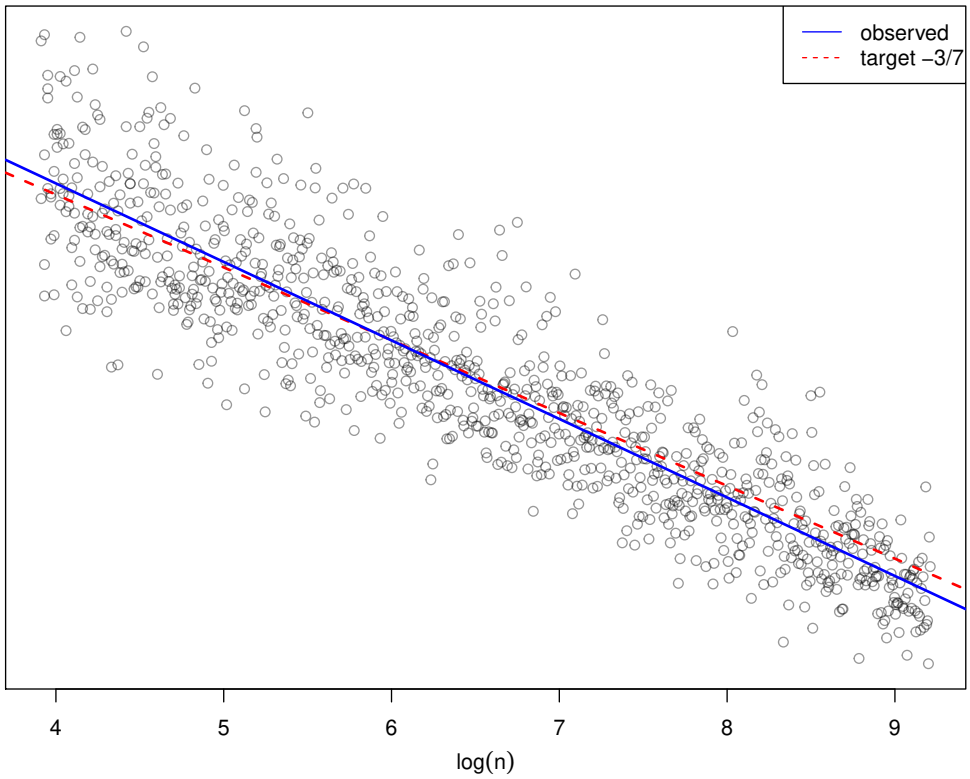

Figure 3.5: Simulated pairs of $(log(n), log\|\hat{g}_\lambda - \bar{g}\|)$ and slope (empirical convergence rate) with penalty λ bounded at rate $\lambda = O_p(n^{-12/7})$. Regressing the term $log\|\hat{g}_\lambda - \bar{g}\|$ over $log(n)$, with simulated 1000 pairs of norm $\|\hat{g}_\lambda - \bar{g}\|$ across different sample sizes $n \in [50, 10000]$, and distribution standard normal.

Table 3.1: Simulated slopes (empirical convergence rate) with penalty λ bounded at different rates. Regressing the term $log\|\hat{g}_\lambda - \bar{g}\|$ over $log(n)$, with simulated 1000 pairs of norm $\|\hat{g}_\lambda - \bar{g}\|$ across different sample sizes $n \in [50, 10000]$, and distribution standard normal.

Penalty	$\lambda = O_p(n^{-1})$	$\lambda = O_p(n^{-12/7})$	$\lambda = O_p(n^{-2})$	Theoretical target
Slope (rate)	-0.368	-0.462	-0.576	-0.429 (-3/7)

Now consider a sequence $\boldsymbol{Y}_n = (Y_{n1}, \ldots, Y_{nn})$, where $Y_{ni} = \mu + \varepsilon_{ni}$ and ε_{ni} are independent draws from a density f_0 with mode zero, let $X_{ni} = |Y_{ni} - \mu|$, and suppose the density g_0 for X_{ni} satisfies (A1)-(A4). Let $\hat{g}_\lambda = \sum_{j=1}^{J} \hat{b}_{\lambda,j}\delta_j$ be the penalized unimodal spline estimator of g_0 using the known mode. Let μ_n be a sequence such that $\mu_n - \mu = O(n^{-3/7})$, and let $\hat{g}_{\lambda,\mu} = \sum_{j=1}^{J} \hat{b}_{\lambda,\mu,j}\delta_j$ be the estimator of g_0 where $\hat{\boldsymbol{b}}_{\lambda,\mu}$ minimizes

$$\boldsymbol{b}^\top(\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})\boldsymbol{b} - 2\boldsymbol{z}_1'\boldsymbol{b}$$

subject to $\boldsymbol{b} \geq \boldsymbol{0}$ and $\boldsymbol{h}_J^\top \boldsymbol{b} = 1$, where $z_{1j} = \sum_{i=1}^{n} \delta_j(|Y_{ni} - \mu_n|)/n$, $j = 1, \ldots, J$; this gives the penalized unimodal least-squares estimator, with mode μ_n.

Theorem 3. *Then $\|\hat{g}_{\lambda,\mu} - \hat{g}_\lambda\| = O_p(n^{-3/7})$, when there is a $S_1 > 0$ such that $f(x)$ is concave for $x \in (-S_1 - \mu, S_1 - \mu)$.*

Proof: Without loss of generality, let $\mu = 0$. Define $W_{ni} = |Y_{ni} - \mu_n|$. We can see that density g_X is density f_Y folded at 0. Also, density g_W is density f_Y shifted to left hand side by μ_n then folded at 0. If $\mu_n > 0$, by the symmetry of f_Y, we have

$$\begin{aligned} g_X(t) &= 2f_Y(t), \\ g_W(t) &= f_Y(t + \mu_n) + f_Y(t - \mu_n), \end{aligned}$$

for $t \in [0, S + \mu_n]$. Then we have

$$
\begin{aligned}
\|g_W - g_X\|^2 &= \int_0^{S+\mu_n} [g_W(t) - g_X(t)]^2 dt \\
&= \int_0^{S+\mu_n} \left[2f_Y(t) - f_Y(t+\mu_n) - f_Y(t-\mu_n)\right]^2 dt \\
&\leq \int_0^{S+\mu_n} \left[f_Y(t) - f_Y(t+\mu_n)\right]^2 dt + \int_0^{S+\mu_n} \left[f_Y(t) - f_Y(t-\mu_n)\right]^2 dt. \\
&= 2\int_0^{S+\mu_n} \left[\mu_n f_Y'(t) + \frac{1}{2}\mu_n^2 f_Y''(t) + \cdots\right]^2 dt, \text{ by making Taylor expansions.} \\
&= 2\mu_n^2 \int_0^{S+\mu_n} \left[f_Y'(t) + \frac{1}{2}\mu_n f_Y''(t) + \cdots\right]^2 dt \\
&= O(\mu_n^2),
\end{aligned}
$$

which is $O(n^{-6/7})$.

Note that $g_W'(t) = f_Y'(t+\mu_n) + f_Y'(t-\mu_n)$. We can see that $g_W'(0) = 0$ and $g_W'(t)$ is non-positive for $t \geq \mu_n$, by the properties of f_Y. Then, since f_Y is concave within S_1, we have that $g_W'(t)$ is non-positive for $t \in [0, S_1)$. Put these together, we have that there is a $N > 0$ such that $g_W'(t)$ is non-positive for $t \in [0, S + \mu_n]$, with all $n > N$.

A similar argument holds for $\mu_n < 0$. Note that we have $\|\hat{g}_{\lambda,\mu} - g_W\| = O_p(n^{-3/7})$, and $\|\hat{g}_\lambda - g_X\| = O_p(n^{-3/7})$, by Theorem 2. Finally, by the triangle rule, we have $\|\hat{g}_{\lambda,\mu} - \hat{g}_\lambda\| = O_p(n^{-3/7})$.

$\diamondsuit$

For applications in robust regression, the assumption of finite known support is not desirable. If the true support is infinite and the support $[0, S_n]$ of the estimator is to include the maximum of the sample, then S_n is stochastically increasing with n. It is argued in [Meyer, 2012] that for S_n growing at the rate of $\log(n)$, for example, the convergence rate will be $O_p(n^{-3/7}\log(n)^{3/2})$. For the robust regression problem, however, we would like to consider densities where the maximum is growing at a much faster rate. Choosing S_n to include the maximum of the sample might result in the support growing at a faster rate than the number of knots, which does not support consistency.

Instead, we choose a "large" S and define g_0 as the true density, truncated to $[0, S]$. Similarly, the density f is defined as f_0 truncated to $[-S, S]$. The value for S can be based on the context of the problem, or chosen based on a high percentile of the data, which will converge to the corresponding percentile of the true density as the sample size grows. In the application of Section 3.3, our data-driven choice of support S_n is taken to be 1.5 times the 98th percentile of the sample $\{X_{ni}\}$. If we define S to be the 98th percentile of the distribution with density g, then $S_n = S + O_p(n^{-1/2})$.

3.2.4 Choosing the Penalty Parameter

If we just need a penalty to have the second-derivatives of $\hat{g}_\lambda$ not jumping too wildly, or just to make $\hat{g}_\lambda$ smoother, we may use an arbitrarily small penalty constant, such as $\lambda = 0.01$. It does work in practice. However, if we hope to optimize the accuracy of $\hat{g}_\lambda$ as well, we need to select a good penalty λ to help minimizing the risk of estimation with $\hat{g}_\lambda$, which is defined as the expected $\mathcal{L}_2$ loss, where the expectation is for everything:

$$\begin{aligned} R(\hat{g}_\lambda) &= E \int_0^S [g(t) - \hat{g}_\lambda(t)]^2 dt \\ &= E \int_0^S \hat{g}_\lambda^2(t) dt - 2E \int_0^S \hat{g}_\lambda(t) g(t) dt + \text{constant}. \end{aligned}$$

Note that here the second term, $2E \int_0^S \hat{g}_\lambda(t) g(t) dt$ contains an unknown distribution g. Therefore, we need to estimate it. To do so, we need to find $\hat{R}(\hat{g}_\lambda)$, a consistent estimator of $R(\hat{g}_\lambda)$. One possible estimator is

$$\hat{R}(\hat{g}_\lambda) = \int_0^S \hat{g}_\lambda^2(t) dt - \frac{2}{n} \sum_{i=1}^n \hat{g}_{\lambda,(-i)}(x_i), \tag{3.8}$$

where $\hat{g}_{\lambda,(-i)}$ is the leave-one-out density estimate; i.e., the estimated density for the sample without x_i. In practice, we specify a suitable range of penalty values, and compute the estimates $\hat{g}_\lambda$, then we compute $\hat{R}(\hat{g}_\lambda)$ for each.

We have below proposition of consistency, which is proved in Appendix A.2.

Proposition 1. *If* $E[f^2(x_i)] < \infty$, *then* $\hat{R}(\hat{g}_\lambda) \xrightarrow{p} R(\hat{g}_\lambda)$, *as* $n \to \infty$.

Therefore, a selected tuning constant λ which minimizes the estimated risk $\hat{R}(\hat{g}_\lambda)$ will asymptotically minimizes the true risk $R(\hat{g}_\lambda)$. Thus we make our selected λ to be

$$\lambda_{CV} = \arg\min_{\lambda} \hat{R}(\hat{g}_\lambda),$$

which is a consistent selection of penalty constant, based on a consistent leave-one-out cross-validation (loocv). See [Scott and Terrell, 1987] as a reference for more discussions about the consistency of cross-validation in density estimations.

When the sample size n is large, the leave-one-out cross-validation procedure may be computationally intensive. Instead, v-fold cross-validation may be used to estimate the risk and give us a selected penalty λ. See [Celisse, 2014] for a reference of doing v-fold cross-validation in density estimations. The original data set is divided to v groups, say $v = 10$, which is a generally acceptable number of folds. Then the expression

$$\hat{R}_v(\hat{g}_\lambda) = \int_0^S \hat{g}_\lambda^2(t)dt - \frac{2}{n}\sum_{i=1}^{v}\sum_{j\in n_i} \hat{g}_{\lambda,(-n_i)}(x_j), \tag{3.9}$$

is minimized, where $\hat{g}_{\lambda,(-n_i)}$ is estimated without the observations in the ith fold of data. The proof for consistency of this v-fold cross-validated risk estimation will be similar to that of Proposition 1. Numerical studies indicate that either the leave-one-out cross-validation based on (3.8) or the v-fold cross-validation based on (3.9) works well in practice. See Figure 3.6 (a) for the side-by-side comparison between the true risk and the estimated risk based on (3.8), averaged by 500 replicates respectively, with density $f = t(2)$ and sample size $n = 100$. See Figure 3.7 (a) for the side-by-side comparison between the true risk and the estimated risk based on (3.9), averaged by 500 replicates respectively, with density $f = t(2)$ and a larger sample size $n = 200$. The plot (a) in each figure shows that the estimated risks match the true risks, therefore providing a good basis of choosing the penalty λ.

On the other hand, if we have lots of sample data sets $\boldsymbol{x}$'s, we might be able to get a quite accurate estimation on the true risk. However in practice, only one data set $\boldsymbol{x}$ is available, so here

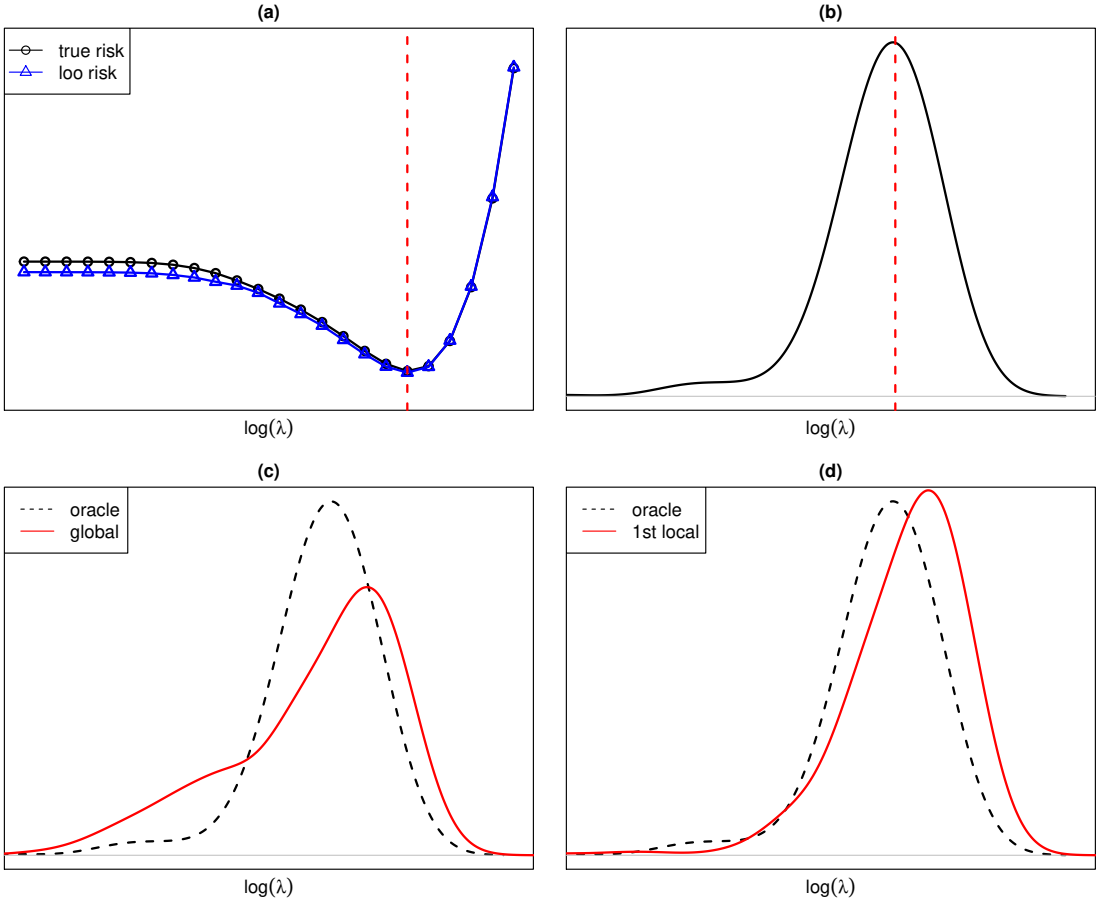

Figure 3.6: Leave-one-out cross-validation, with $f = t(2)$, $n = 100$ and $reps = 500$. Plot (a) compares the true risks and estimated risks, and the dashed vertical line represents the penalty λ minimizing the true risk. Plot (b) shows the frequency of penalty λ that minimizes the empirical true risk. Plot (c) compares the frequencies of cross-validated λ selected as the global minimzer, to the oracle λ selected by the infeasible true risks. Plot (d) compares the frequencies of cross-validated λ selected as the largest local minimzer, to the oracle λ selected by the infeasible true risks.

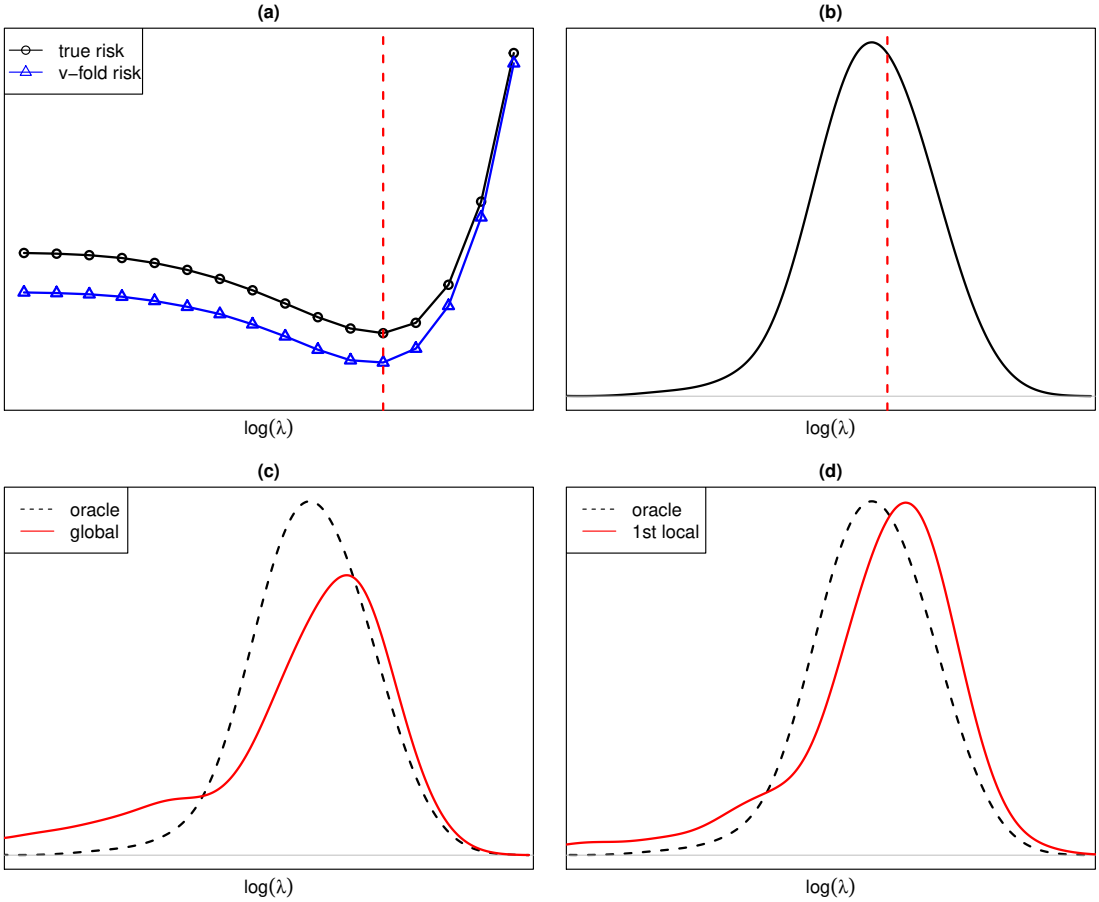

Figure 3.7: V-fold cross-validation, with $f = t(2)$, $n = 200$ and $reps = 1000$. Plot (a) compares the true risks and estimated risks, and the dashed vertical line represents the penalty λ minimizing the true risk. Plot (b) shows the frequency of penalty λ that minimizes the empirical true risk. Plot (c) compares the frequencies of cross-validated λ selected as the global minimzer, to the oracle λ selected by the infeasible true risks. Plot (d) compares the frequencies of cross-validated λ selected as the largest local minimzer, to the oracle λ selected by the infeasible true risks.

we are actually making an estimation on risk, with sample size just one vector. This contains a very large amount of sampling variation. Due to this fact, it happens in some rare cases that the estimations of risk jump a lot, and might be minimized by a bad penalty λ (actually often being too small). Figure 3.8 shows a comparison between two instances of cross-validated risks, with density $f = t(2)$ and sample size $n = 100$. We can see that in the first instance, the cross-validation provides a good selection on λ, while the second instance does not. Though bad situations like the second one is not very often, but when it happens, the selected penalty λ would be too small to overcome the over-fitting, which is not preferred.

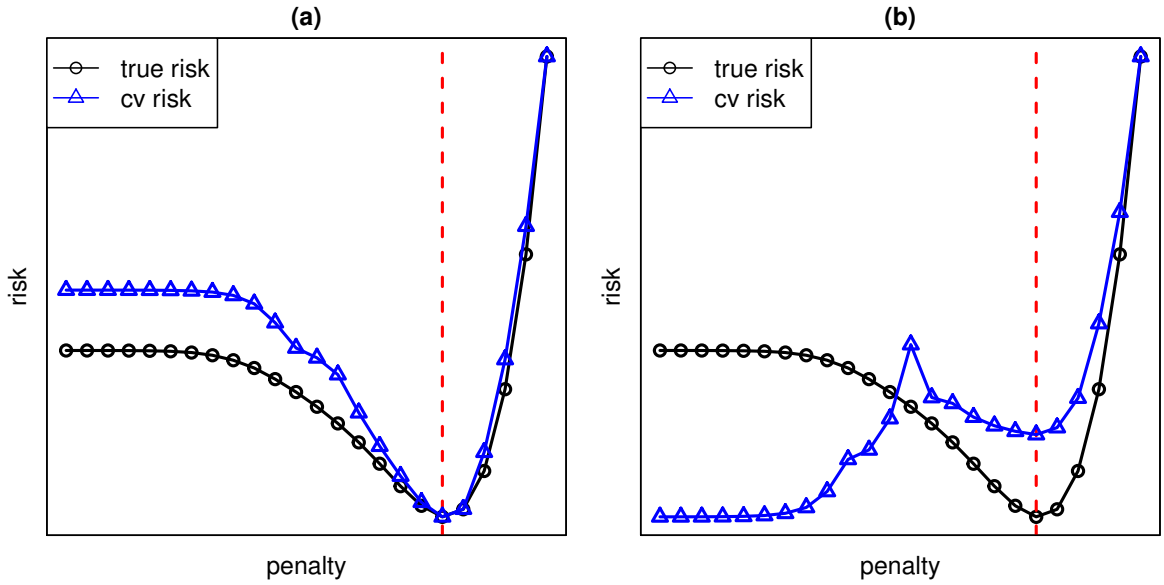

Figure 3.8: The true risk and two instances of cross-validated risk, with density $f = t(2)$ and sample size $n = 100$. The dashed line shows the optimal oracle of penalty λ selected by the infeasible true risk.

As suggested by [Wand and Jones, 1995] (page 64), choosing the largest local minimizer for λ, instead of the global minimizer, may be helpful in solving the problem of variation in cross-validation for density estimations. Numerical studies indicate that the largest local minimizer, as a selection of λ, performs more reliable than the global minimizer.

Figure 3.6 (c) compares the frequencies of leave-one-out cross-validated λ selected as the global minimzer, to the oracle λ selected by the infeasible true risks. Figure 3.6 (d) compares

the frequencies of cross-validated λ selected as the largest local minimzer, to the oracle λ selected by the infeasible true risks. By making the side-by-side comparison of performance between these two figures, we can see that the λ (largest local minimizer) works more reliable, while the λ (largest local minimizer) has a thick left-tail. That is to say, too many small penalty λ are selected during the leave-one-out cross-validation, as the global minimizer. Note that Figure 3.7 (c) and Figure 3.7 (d) also indicate the similar problem of global minimizer, in v-fold cross-validation.

Lastly, numerical studies indicate that the cross-validation described in this subsection works in maintaining the performance of our unimodal spline density estimation, against the over-fitting problem induced by too many knots. Figure 3.9 shows the true risks of unimodal spline density estimation across increasing number of knots, either being penalized or unpenalized, with density $f = t(2)$ and sample size $n = 100$. We can see that the penalized estimation with cross-validated λ (the largest local minimizer) outperformed unpenalized estimation, while chasing the infeasible optimal estimation based on oracle penalty λ.

3.2.5 Algorithm and Details of Implementation

We find $\hat{\boldsymbol{b}}_\lambda$ to minimize $Q_\lambda(\boldsymbol{b})$ using quadratic programming. The algorithm is the same as outlined in Section 3.2.1, with $\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D}$ in place of $\boldsymbol{H}$. Then $\hat{g}_\lambda = \sum_{j=1}^{J} \hat{b}_{\lambda,j} \delta_j$. An example of pseudo algorithm implement is provided in the Appendix A.3 (supplementary material).

3.3 The One-Sample Problem

In this section, we consider the one sample problem with a mean parameter:

$$Y_i = \mu + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.10}$$

where μ is the unknown mean parameter to be estimated, the ε_i are independent, symmetric mean-zero random variables with common density f.

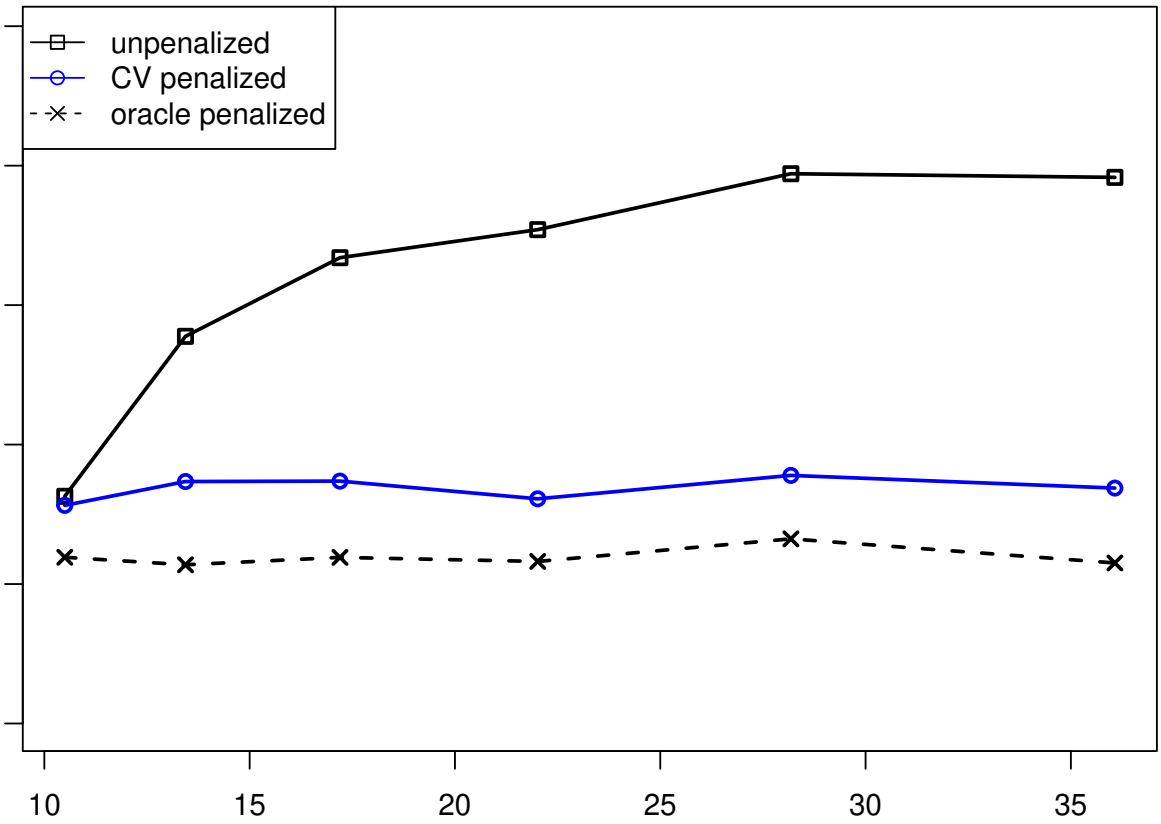

Figure 3.9: True risks of unimodal spline density estimation across increasing number of knots, with density $f = t(2)$ and sample size $n = 100$. The lower curve shows the true risk of penalized estimation, with oracle λ selected by infeasible true risks. The middle curve shows the true risk of penalized estimation, with cross-validated λ selected by estimated risks. The upper curve shows the true risk of unpenalized estimation.

The robust M-estimator uses a function ρ and sample values $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, determine $\hat{\mu} \in \mathbb{R}$ to minimize

$$M_n(\mu; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho\left(Y_i - \mu\right). \tag{3.11}$$

If the error density family is known, the ρ function in (3.11) may be chosen accordingly. Here we let the data itself to determine the shape of ρ function. Therefore, we call this Auto-Adaptive M-Estimator (AAME).

3.3.1 Algorithm and Details of Implementation

Using the sample median μ_0, we first determine the support $[0, S]$ by setting S to be 1.5 times the 98th percentile of the values of $\boldsymbol{x} = |\boldsymbol{y} - \mu_0 \mathbf{1}|$. To place the knots, we use $J = 10n^{1/7}$, with a fixed mesh ratio r (in the simulations in Section 3.5, $r = 10$). The knots are placed in $[0, S]$ with geometrically increasing intervals, as

$$t_j = S\frac{r^{(j-1)/(J-2)} - 1}{r^{(J-1)/(J-2)} - 1},$$

for $j = 1, \cdots, J$. This placement ensures that the knot intervals are increasing with constant increment ratio, that is, $t_{j+2} - t_{j+1} = p(t_{j+1} - t_j)$, $j = 1, \ldots, J-2$, where $p = r^{1/(J-1)}$.

For the penalty parameter, the density is estimated for a range of penalty values, and λ is chosen using 10-fold cross-validation. The range of penalty parameters was determined through extensive simulations with a variety of densities at a fixed scale s_0. The maximum penalty is $e^{10}n^{-12/7}$, and 30 values spaced on a log scale are determined, with the smallest value close to zero. To ensure that the range is invariant to the scale of the observations, we determine the scale s of the observations, and multiply the range of penalty values by $(s/s_0)^5$. To see why, consider a density $g_0(x)$, and $g(x) = g_0(x/s)/s$, so that a sample from g_0, multiplied by s, would result in a sample from g. Then $[g''(x)]^2 = g_0''(x/s)^2/s^6$. A change of scale from s_0 to s affects the first two terms of (3.6) by factor of s_0/s, while the third term changes by a factor of $(s_0/s)^6$.

Given the chosen knots and λ, the density estimate $\hat{f}$ is obtained, and we define $\rho = \hat{f}(0) - \hat{f}$ (traditionally we have $\rho(0) = 0$). Then $\hat{\mu}$ is the minimizer of (3.11). User-friendly code is available at [Chen, 2020].

3.3.2 Asymptotic Properties

We first fix a continuous ρ that is symmetric around zero, and determine conditions for consistency of an estimator of μ. We define $\hat{\mu}$ as the minimizer of $M_n(\mu; \boldsymbol{Y})$, which is a zero of $M_n'(\mu; \boldsymbol{Y})$. If the ρ function is convex, then M_n inherits this convexity, and M_n' has at most one zero; [Huber, 1964] proves consistency of $\hat{\mu}$ in this case.

In our case, however, our $\rho(y)$ function will "flatten out" as $|y|$ gets large, so there is no hope for guaranteeing a single zero of M_n', even when n is very large. Assume that f_0 is symmetric around zero and increasing on $(-\infty, 0)$, while ρ is symmetric around zero, decreasing on $(-\infty, 0)$. Define

$$M(\mu) = \int_{-\infty}^{\infty} \rho(y - \mu) f_0(y) dy;$$

it is easy to see that M' is continuous with $M'(0) = 0$ by symmetry of f and ρ. Further, M' is positive on $(-\infty, 0)$ and negative on $(0, \infty)$, also by symmetry: for $\mu > 0$,

$$\int_{-\infty}^{0} \rho'(y - \mu) f_0(y) dy = -\int_{2\mu}^{\infty} \rho'(y - \mu) f_0(y - 2\mu) dy,$$

so that

$$M'(\mu) = \int_{0}^{2\mu} \rho'(y - \mu) f_0(y) dy + \int_{2\mu}^{\infty} \rho'(y - \mu)[f_0(y) - f_0(y - 2\mu)] dy.$$

For $\mu > 0$, both terms are negative, and a similar argument holds for $\mu < 0$.

For a fixed $\epsilon > 0$, $M_n'(\epsilon; \boldsymbol{Y}) \overset{a.s.}{\to} M'(\epsilon) < 0$ and $M_n'(-\epsilon; \boldsymbol{Y}) \overset{a.s.}{\to} M'(-\epsilon) > 0$ almost surely by the law of large numbers, but we need to show that there are not *many* zeros in any $(-\epsilon, \epsilon)$. Under reasonable conditions on ρ and f, we can hope for a single zero of M_n' "in the middle" of the observations; that is, we can find a $\xi > 0$ so that M_n is convex on $(-\xi, \xi)$, almost surely as n increases. If we then define $\hat{\mu}$ to be the zero of M_n' that is closest to the sample median, we will

have consistency of $\hat{\mu}$. In practice, this works well: for a wide range of simulated data sets and real data sets (see Section 3.5), as we typically find an isolated global minimum near the true value, with some local minima to either side.

To implement these ideas, we make the following assumptions about ρ and f, that are easily implemented in our algorithm.

(A5) The function $\rho(y)$ is a quadratic spline function, that is symmetric about zero and $-\rho$ is unimodal. For some $c_0 > t_2$, the second derivative of ρ is at least $c_1 > 0$ on $(-c_0, c_0)$.

(A6) $f_0(y)$ is a unimodal density, symmetric about zero with $f_0''(y) < -c_1$ on $(-c_0, c_0)$.

Theorem 4. *Under (A1) - (A6), there exists a $\xi > 0$ not depending on n, so that*

$$\inf_{\mu \in (-\xi, \xi)} M_n''(\mu; \boldsymbol{Y}) > 0, \text{ a.s. as } n \to \infty.$$

Proof: For simplicity of exposition, we give the proof for equally spaced knots. We can write

$$M_n''(\mu; \boldsymbol{Y}) = \frac{1}{n} \sum_{i=1}^{n} \rho''(Y_i - \mu) = \frac{1}{n} \sum_{j=1}^{J-1} a_j N_j(\mu),$$

where for $j = 1, \ldots, J-1$, $a_j = \rho''(y)$ for $y \in (t_j, t_{j+1})$, and $N_j(\mu)$ is the number of Y_i such that $Y_i - \mu \in I_j$, and finally $I_j = (-t_{j+1}, -t_j] \cup (t_j, t_{j+1}]$, for $j = 1, \ldots, J-1$. Also,

$$M''(\mu) = \int_{-\infty}^{\infty} \rho''(y - \mu) f_0(y) dy = \int_0^{\infty} \rho''(y) \left[f_0(y - \mu) + f_0(y + \mu) \right] dy = \sum_{j=1}^{J-1} a_j q_j(\mu),$$

where

$$q_j(\mu) = \int_{t_j}^{t_{j+1}} \left[f_0(y - \mu) + f_0(y + \mu) \right] dy.$$

For each μ, $(N_1(\mu), \ldots, N_{J-1}(\mu))$ is a multinomial random vector with mean vector $(q_1(\mu), \ldots, q_{J-1}(\mu))$. Because ρ is increasing on the positive reals, for equally spaced knots we have

$$\sum_{j=1}^{k} a_j \geq 0 \text{ for all } k = 1, \ldots, J-1.$$

Let ℓ be the largest index where t_ℓ is less than $c_0/2$, and recall that $a_j > c_1$ for $j = 1, \ldots, \ell$. For $\mu \in (-c_0/2, c_0/2)$, $f_0(y-\mu) + f_0(y+\mu)$ is symmetric about zero, decreasing for $y > 0$, and concave on $(-c_0/2, c_0/2)$ with second derivative less than $-2c_1$ on this range, by (A6). Therefore for $j = 1, \ldots, \ell - 1$ we have

$$q_j(\mu) - q_{j+1}(\mu) > 2jc_1d^3,$$

where d is the distance between consecutive knots. For $\mu \in (-c_0/2, c_0/2)$,

$$\begin{aligned} M''(\mu) &= \sum_{j=1}^{J-1} a_j q_j(\mu) \\ &= a_1(q_1(\mu) - q_2(\mu)) + (a_1+a_2)(q_2(\mu) - q_3(\mu)) + (a_1+a_2+a_3)(q_3(\mu) - q_2(\mu)) + \cdots \\ &+ (a_1 + \cdots + a_{J-2})(q_{J-2}(\mu) - q_{J-1}(\mu)) + (a_1 + \cdots + a_{J-1})q_{J-1}(\mu). \end{aligned}$$

Each term in the above expression is positive, and for $j \leq \ell$, the jth term $(a_1 + \cdots + a_j)(q_j(\mu) - q_{j+1}(\mu)) \geq 2j^2c_1^2d^3$, so for $\mu \in (-c_0/2, c_0/2)$,

$$M''(\mu) > \sum_{j=1}^{\ell} a_j q_j(\mu) \geq \sum_{j=1}^{\ell} 2j^2c_1^2d^3 = \ell(\ell+1)(2\ell+1)c_1^2d^3/3 \geq c_0^3c_1^2/24,$$

by definition of ℓ and d. Let n_c be the number of times that ρ'' changes sign on the positive reals, and let $\tilde{f}_c = f_0(c_0)$, and set

$$\xi = \min\left(\frac{c_0^3c_1^2}{96n_c\tilde{f}_c}, \frac{d}{2}\right).$$

Define $\tilde{t}_j$, $j = 1, \ldots, J-1$, so that $\tilde{t}_j = t_j$ if a_{j-1} and a_j have the same sign, $\tilde{t}_j = t_j - \xi$ if $a_j \geq 0$ and $a_{j+1} < 0$, and finally $\tilde{t}_j = t_j + \xi$ if $a_j < 0$ and $a_{j+1} \geq 0$. The placement of these alternative knots is illustrated in Figure 3.10, where it is seen that the effect is to widen the intervals over

which ρ'' is negative. For $j = 1, \ldots, J-1$, define

$$\tilde{q}_j = 2 \int_{\tilde{t}_j}^{\tilde{t}_{j+1}} f_0(y) dy$$

and note that for j such that $a_j < 0$, $0 \leq \tilde{q}_j - q_j(\mu) < 2\tilde{f}_c \xi$ when $\mu \in (-\xi, \xi)$. Let $\tilde{N}_j$ be the number of Y_i such that $Y_i \in \tilde{I}_j$, with $\tilde{I}_j = (-\tilde{t}_{j+1}, \tilde{t}_j] \cup (\tilde{t}_j, \tilde{t}_{j+1}]$. Then $(\tilde{N}_1, \ldots, \tilde{N}_{J-1})$ is a multinomial random vector that does not depend on μ, and

$$\inf_{\mu \in (-\xi, \xi)} M_n''(\mu; \boldsymbol{Y}) \geq \frac{1}{n} \sum_{j=1}^{J-1} a_j \tilde{N}_j.$$

Finally, $\sum_{j=1}^{J-1} a_j \tilde{N}_j / n$ has mean

$$\begin{aligned} \sum_{j=1}^{J-1} a_j \tilde{q}_j &= \sum_{j=1}^{J-1} a_j q_j(0) + \sum_{j=1}^{J-1} a_j (\tilde{q}_j - q_j(0)) \\ &\geq \frac{c_0^3 c_1^2}{24} - 2\xi \tilde{f}_c n_c \\ &\geq \frac{c_0^3 c_1^2}{48}, \end{aligned}$$

and the variance of $\sum_{j=1}^{J-1} a_j \tilde{N}_j / n$ is tending to zero. Because ξ does not depend on n, $\inf_{\mu \in (-\xi, \xi)} M_n''(\mu; \boldsymbol{Y}) > 0$ almost surely as n increases.

♢

Corollary 1. *Let $\hat{\mu}_n$ be the zero of M_n' that is closest to the median of the sample. Then $\hat{\mu}_n \xrightarrow{p} \mu$.*

Proof: Again take $\mu = 0$. Let E_n be the event that there is a unique zero of M_n in $(-\xi, \xi)$. By the strong law of large numbers, $M_n'(-\xi) \xrightarrow{a.s.} M'(-\xi) > 0$ and $M_n'(\xi) \xrightarrow{a.s.} M'(\xi) < 0$. Therefore there is a zero of M_n' in $(-\xi, \xi)$ almost surely as n increases, and by the theorem, this zero is almost surely unique, so $\mathrm{P}(E_n) \to 1$. In addition, the median falls in $(-\xi/2, \xi/2)$ almost surely as n increases, so with probability approaching one, $\hat{\mu}_n$ is the unique zero of M_n in $(-\xi, \xi)$. Choose

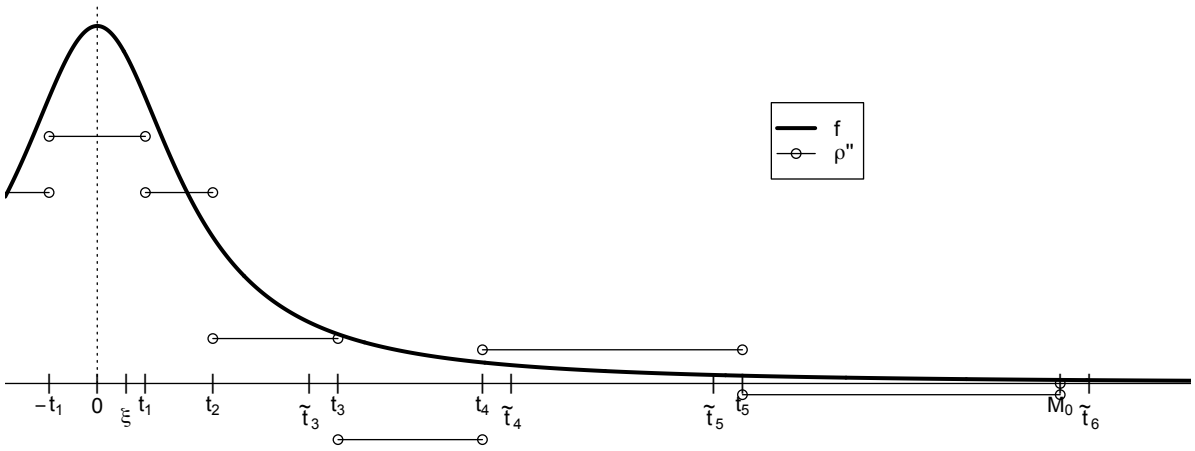

Figure 3.10: Illustration of ideas for proof of consistency of $\hat{\mu}$.

any $\epsilon \in (0, \xi)$. Then

$$\mathrm{P}(\hat{\mu}_n > \epsilon) \leq \mathrm{P}(\{M_n(\epsilon/2) > 0\} \cap E_n) + \mathrm{P}(E_n^c),$$

and both terms on the right go to zero, by the theorem and the law of large numbers. Similarly, $\mathrm{P}(\hat{\mu}_n < -\epsilon)$ goes to zero. ♢

For a fixed ρ function, there are alternative proofs of consistency that require less stringent conditions. For instance, we can use Glivenko-Cantelli ideas to show that M_n converges to M uniformly in μ, see [Chen and Meyer, 2020a]. For our application, however, we will have a different ρ function for each sample, so we need to have a bound on the convexity of M_n around zero, that is uniform over the sequence of ρ functions.

Suppose that we have a sequence of random samples $\boldsymbol{Y}_n = (Y_{n1}, \ldots, Y_{nn})$ from a density f_0 satisfying (A6), and for the nth sample, we use ρ_n to obtain $\hat{\mu}_n$. We require that each ρ_n be a quadratic spline with J_n knots, satisfy (A5), and also satisfy the additional condition

(A7) There is a n_c such that for each n, ρ_n'' changes sign at most n_c times.

Let

$$\Psi_n(\mu; \boldsymbol{Y}_n) = \frac{1}{n} \sum_{i=1}^{n} \rho_n(Y_{ni} - \mu).$$

Theorem 5. *Let $\hat{\mu}_n$ be the zero of Ψ_n' that is closest to the median of the sample. Under (A1)-(A7), $\hat{\mu}_n \xrightarrow{p} \mu$.*

Proof: Using the notation from Theorem 3, for any n we can bound the mean of $\sum_{j=1}^{J} a_j \tilde{N}_j/n$ below by $c_0^3 c_1^2/48$. Then for any $\epsilon > 0$, there is an N such that for all $n > N$, we have

$$\mathrm{P}\left[\Psi_n''(\mu; \boldsymbol{Y}_n) < 0 \text{ for all } \mu \in (-\xi, \xi)\right] > 1 - \epsilon,$$

i.e., $\Psi_n(\mu; \boldsymbol{Y}_n)$ is convex in $(-\xi, \xi)$ with probability approaching one as n increases. Then the steps of the proof of Corollary 1 can be applied. ♢

To show asymptotic normality, we need consistency of the first and second derivatives of $\hat{f}$. The proof of the following lemma is straight forward.

Lemma 3. *If $f_1(x) = a_1 + b_1 x + c_1 x^2$ and $f_2(x) = a_2 + b_2 x + c_2 x^2$, and $|c_1 - c_2| > \epsilon$, then for $d > 0$,*

$$\int_0^d [f_1(x) - f_2(x)]^2 dx \geq \epsilon^2 d^5/180.$$

In Appendix 1, $\bar{f}$ is defined to be the least-squares spline estimate of f given the n quantiles of the true density; this is a spline density that is close approximation to f, and in the proof of Theorem 1 it is shown that $\|\hat{f}_\lambda - \bar{f}\|^2 = O_p(n^{-6/7})$.

Theorem 6. *Let $\hat{\theta}_{nj}$ be $\hat{f}_n''(x)$ for $x \in (t_{j-1}, t_j)$ and let $\bar{\theta}_{nj}$ be $\bar{f}_n''(x)$ in the same interval. Then*

$$\max_j |\hat{\theta}_{nj} - \bar{\theta}_{nj}| \xrightarrow{p} 0.$$

Proof: Choose any $\epsilon > 0$, and suppose there is a sequence $(n_1, j_1), (n_2, j_2), \ldots$, such that $|\hat{\theta}_{n_i, j_i} - \bar{\theta}_{n_i j_i}| \geq \epsilon$, for $i = 1, 2, \ldots,$. Then by Lemma 3, $\|\hat{f}_{n_i} - \bar{f}_{n_i}\|^2 > \epsilon^2 (t_{j_{i+1}} - t_{j_i})^5/180$. By Assumptions (A3) and (A4), the knot intervals shrink on the order of $n^{-1/7}$ so Theorem 2 is contradicted if such a sequence exists. ♢

For asymptotic normality, we follow the ideas of [Huber, 1964].

Theorem 7. *Let $\hat{\mu}_n$ be the zero of Ψ'_n that is closest to the median of the sample, then under (A1)-(A7),*

$$\sqrt{n}(\hat{\mu}_n - \mu) \xrightarrow{\mathcal{D}} N(0, \sigma_\mu^2), \text{ where } \sigma_\mu^2 = \frac{E[f'(Y_{ni} - \mu)^2]}{E[f''(Y_{ni} - \mu)]^2}.$$

Proof: Again, we take $\mu = 0$ for the proof. For each n, we have

$$\sum_{i=1}^{n} \hat{\rho}'_n(Y_{ni} - \hat{\mu}_n) = 0, \tag{3.12}$$

where $\hat{\rho}_n$ is $\hat{f}_n(0) - \hat{f}_n$, and $\hat{f}_n$ is the density estimate for the nth sample. For each i, we find j_{i1} such that $Y_i \in [t_{j_{i1}}, t_{j_{i1}+1})$ and j_{i2} such that $Y_i - \hat{\mu}_n \in [t_{j_{i2}}, t_{j_{i2}+1})$. Define $a_j = \hat{\rho}''_n(y)$ for $y \in (t_j, t_{j+1})$, $j = 1, \ldots, J-1$, as in the proof of Theorem 4. If $j_1 = j_2$, then $\hat{\rho}'_n(Y_{ni} - \hat{\mu}_n) = \hat{\rho}'_n(Y_{ni}) - \hat{\mu}_n a_{j_1}$. If $j_1 = j_2 + 1$, then $\hat{\rho}'_n(Y_{ni} - \hat{\mu}_n) = \hat{\rho}'_n(Y_{ni}) - a_{j_1}(Y_i - t_{j_1}) - a_{j_2}(t_{j_1} - (Y_i - \hat{\mu}_n))$. More generally, for $\hat{\mu}_n > 0$, we can write

$$\hat{\rho}'_n(Y_{ni} - \hat{\mu}_n) = \hat{\rho}'_n(Y_{ni}) + \sum_{j=j_{i1}}^{j_{i2}} r_{i,j} a_j,$$

where $r_{i,j_1} = t_{j_1+1} - Y_i$, $r_{i,j_2} = Y_i - \hat{\mu}_n - t_{j2}$, and for any $j_1 < j < j_2$, $r_{i,j} = t_{j+1} - t_j$, with similar expressions in the case $\hat{\mu} < 0$. We have $\hat{\mu} = \sum_{j=j_1}^{j_2} r_{i,j}$ for each i. Then (3.12) becomes

$$0 = \sum_{i=1}^{n} \left[\hat{\rho}'_n(Y_{ni}) + \sum_{j=j_{i1}}^{j_{i2}} r_{i,j} a_j \right] = \sum_{i=1}^{n} \left[\hat{\rho}'_n(Y_{ni}) + \hat{\mu}_n \frac{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j} a_j}{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}} \right]$$

or

$$\hat{\mu}_n = -\frac{\sum_{i=1}^{n} \hat{\rho}'_n(Y_{ni})}{\sum_{i=1}^{n} \frac{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j} a_j}{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}}}.$$

Let $\bar{\rho}_n = \bar{f}_n(0) - \bar{f}_n$. Then

$$\sqrt{n}\hat{\mu}_n = -\frac{\frac{1}{\sqrt{n}}\sum_{i=1}^{n}\bar{\rho}_n'(Y_{ni}) + \frac{1}{\sqrt{n}}\sum_{i=1}^{n}[\hat{\rho}_n'(Y_{ni}) - \bar{\rho}_n'(Y_{ni})]}{\frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}a_j}{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}}}.$$

The quantity

$$\frac{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}a_j}{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}}$$

is a weighted average of $\hat{\rho}_n''$ values between Y_i and $Y_i - \hat{\mu}_n$. We have $\mathrm{E}[\hat{\rho}_n'(Y_{ni}) - \bar{\rho}_n'(Y_{ni})] = 0$ and $\mathrm{Var}[\hat{\rho}_n'(Y_{ni}) - \bar{\rho}_n'(Y_{ni})] \to 0$, so

$$\frac{1}{\sqrt{n}}\sum_{i=1}^{n}[\hat{\rho}_n'(Y_{ni}) - \bar{\rho}_n'(Y_{ni})] \xrightarrow{p} 0.$$

For every a_j defined between Y_i and $Y_i - \hat{\mu}_n$, we have $\max_j |a_j + \bar{\theta}_{nj}| \xrightarrow{p} 0$, by Theorem 5. Hence the weighted average

$$\left|\frac{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}a_j}{\sum_{j=j_{i1}}^{j_{i2}} r_{i,j}} + \bar{f}_n''(Y_i)\right| \xrightarrow{p} 0.$$

Note that each random variable r_{ij} is an additive term with finite elements Y_i and knots t_j which all have bounded supports. Thus each r_{ij} has a bounded support. As a result, each r_{ij} has finite moments. Then by the central limit theorem and the law of large numbers we have

$$\frac{1}{\sqrt{n}}\sum_{i=1}^{n}\bar{\rho}_n'(Y_{ni}) \xrightarrow{\mathcal{D}} N(0, E[f'(Y_i)^2]), \text{ and}$$
$$\frac{1}{n}\sum_{i=1}^{n}\bar{\rho}_n''(Y_i) \xrightarrow{p} -E[f''(Y_i)].$$

By Slutsky's theorem, and putting everything together, we have

$$\sqrt{n}\hat{\mu}_n \xrightarrow{\mathcal{D}} N(0, \sigma_\mu^2), \text{ where } \sigma_\mu^2 = \frac{E[f'(Y_i)^2]}{E[f''(Y_i)]^2}.$$

♢

3.3.3 Inferences about the Mean

Asymptotic confidence interval

For confidence intervals of μ, based on the asymptotic sampling distribution, we use the results we developed in Section 3.3.2. Note that since the error distribution $f(\varepsilon)$ is unknown, we need to estimate the two expectations in the fraction. We begin with a moment-based estimation which is used widely in robust M-estimations:

$$\tilde{\sigma}_{\mu}^{2} = \frac{\frac{1}{n}\sum_{i=1}^{n}\rho'\left(Y_i - \hat{\mu}\right)^2}{n\left[\frac{1}{n}\sum_{i=1}^{n}\rho''\left(Y_i - \hat{\mu}\right)\right]^2}. \tag{3.13}$$

Alternatively, since we have an estimated error density $\hat{f}$ for the infeasible distribution $f(\varepsilon)$, we may also consider a plug-in estimation which leads to a more reliable estimation of asymptotic variance in our applications of AAME robust estimation:

$$\hat{\sigma}_{\mu}^{2} = \frac{\int_{-\infty}^{\infty}\hat{f}'(\varepsilon)^2\hat{f}(\varepsilon)d\varepsilon}{n\left[\int_{-\infty}^{\infty}\hat{f}''(\varepsilon)\hat{f}(\varepsilon)d\varepsilon\right]^2}. \tag{3.14}$$

Then we may make an asymptotic confidence interval based on quantiles of the asymptotic normal distribution:

$$CI = \left(\hat{\mu} - z_{1-\alpha/2}\hat{\sigma}_{\mu}, \hat{\mu} + z_{1-\alpha/2}\hat{\sigma}_{\mu}\right),$$

where the confidence interval has supposed coverage rate $(1-\alpha)100\%$, and $z_{1-\alpha/2}$ is the $1-\alpha/2$ quantile of the standard normal distribution.

No matter whether the expression of (3.13) or (3.14) will be used, it is obvious that a large amount of variation on first two derivatives will lead to a bad estimation of the numerator, and especially to a bad estimation of denominator. Recall the wildly jumping second-derivative in Figure 3.3. Consequently, the entire estimation on σ_{μ} will be bad. Therefore, we do need the

first two derivatives to be more stable, and the proposed penalized method helps in solving this problem.

On the other hand, using "an arbitrarily small penalty $\lambda = 0.01$" is not guaranteed to work well with different incoming error densities, with different shapes and different scales. Thus, finding an appropriate penalty λ is worth doing to improve the performance of inferences. To do so, just use the cross-validation method we discussed in Section 3.2.4. From simulation studies in Section 3.5, we can see that the cross-validation does work in selecting an appropriate penalty λ for a variety of error density and lead to confidence intervals with an appropriate coverage rate so that being almost unbiased.

Bootstrap confidence interval

When the sample size is small, for example, smaller than $n = 30$, the asymptotic distribution of $\hat{\mu}$ may not be approximated by a normal distribution very well. In this case, we would better consider a bootstrap confidence interval by a re-sampling method instead.

If we know the sampling distribution of $\hat{\mu}$, we may use its $\alpha/2$ quantile L and $1-\alpha/2$ quantile U to get a confidence interval $CI = (L, U)$ such that

$$P(L < \beta < U) = 1 - \alpha.$$

In practice, we may use a bootstrap method to estimate the sampling distribution of $\hat{\mu}$. Specially, here we have the estimated error density $\hat{g}_\lambda$, so we may use a special version of bootstrap as following steps:

1 Get the estimates of error density $\hat{g}_\lambda$ and mean $\hat{\mu}$.

2 Sample a vector of error e from $\hat{g}_\lambda$.

3 Make bootstrap data $y_i^* = \hat{\mu} + e_i$, $i = 1, \cdots, n$, and use it to get the bootstrap mean $\hat{\mu}^*$.

4 Loop step 2 and 3, and get a set of $\hat{\mu}^*$.

5 Get the bootstrap confidence interval by the lower and upper quantiles of the set of $\hat{\mu}^*$.

In the step 2, we construct an estimated cumulative distribution function, based on $\hat{g}_\lambda$, to sample e. This step gives a mimic error term. Numerical studies indicate this sampling method works effectively in our AAME, even with heavy-tailed error density. Figure 3.11 shows a random instance of the sampled error term from estimated cumulative density function, with sample size $n = 20$, replicates $reps = 2000$ and error density $f = t(2)$. The dashed line is the true density, the solid line is the estimated density, and the histogram shows the relative frequency of sampled e_i. We can see these three distributions match each other.

Simulation studies indicate that the distribution of our bootstrap $\hat{\mu}^*$ is a good approximation of the sampling distribution of $\hat{\mu}$, especially when we are only looking for the the 2.5% and 97.5% quantiles. Also, our AAME with this bootstrap confidence interval works effectively against heavy-tailed error density and small sample size. See Section 3.5 for the details of the simulation results.

3.3.4 An Example of Application

To illustrate the new method, we consider data from the `R` package `lmtest`, which has several examples of macroeconomic time series data. Shown in Figure 3.12 are first differences of monthly interest rates from 1959 to 1993, with the estimated density and mode at the dotted vertical line. The confidence interval given by AAME is the smallest.

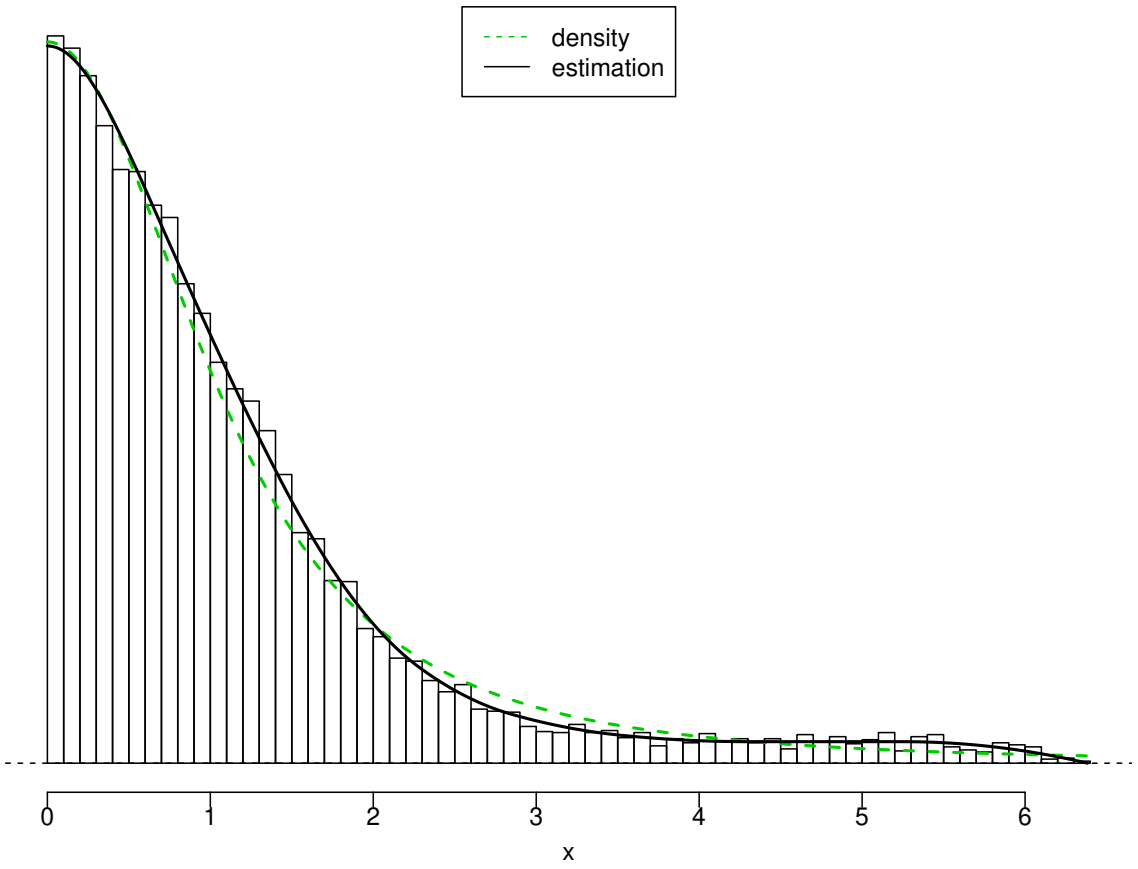

Figure 3.11: Sampled error term from estimated cumulative density function, with $n = 20$, replicates $reps = 2000$, and $f = t(2)$. The dashed line is the true density, the solid line is the estimated density, and the histogram shows the relative frequency of sampled e_i.

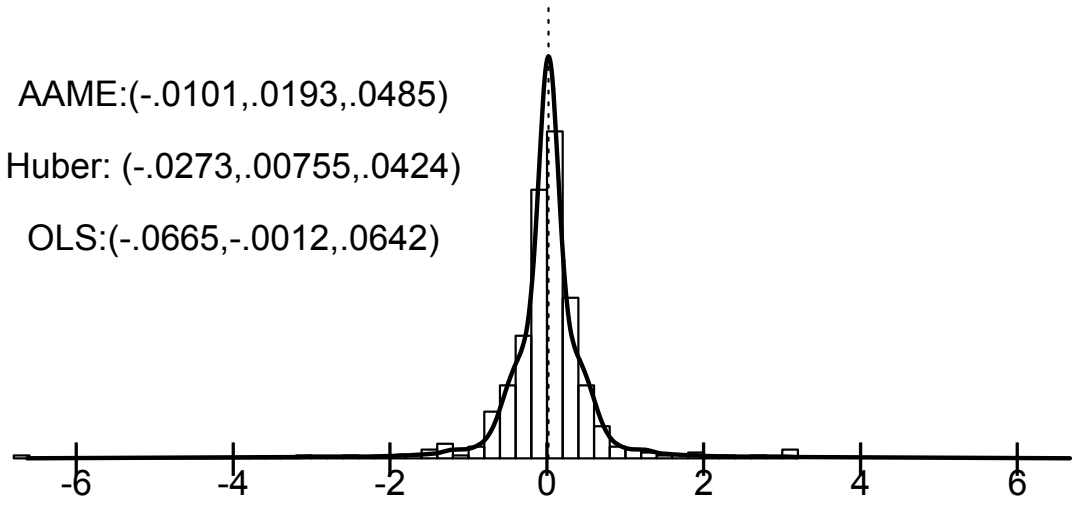

Figure 3.12: Example using data `fyff` of US Macroeconomic Time Series Data from the `R` package `lmtest`. The observations are first differences of monthly interest rates from 1959 to 1993. The estimated density is shown with the histogram of observations, along with three 95% confidence intervals for μ: (left, $\hat{\mu}$, right).

3.4 The Regression Problem

Our AAME can be extended to a parametric linear regression model with the same assumptions on error density:

$$Y_i = \boldsymbol{X}_i^\top \boldsymbol{\beta} + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.15}$$

where Y_i is the response variable, $\boldsymbol{X}_i^\top$ is the $1 \times p$ predictor vector. β is the unknown $p \times 1$ regression parameter vector to be estimated, the ε_i are independent, symmetric mean-zero random variables with common density f.

The robust regression M-estimator uses a function ρ and sample values $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, determine $\hat{\boldsymbol{\beta}} \in \mathbb{R}^p$ to minimize

$$M_n(\boldsymbol{\beta}; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho\left(Y_i - \boldsymbol{X}_i^\top \boldsymbol{\beta}\right). \tag{3.16}$$

Again, if the error density family is known, the ρ function may be chosen accordingly. We follow the idea of AAME which we developed in Section 3.3, i.e. using the data itself to determine the shape of ρ function in (3.16).

3.4.1 Estimating Robust Regression Parameters

Using a good starting point $\boldsymbol{\beta}_0$, such as a least-absolute-deviance estimation, we first determine the support $[0, S]$ by setting S to be 1.5 times the 98th percentile of the values of $\boldsymbol{e} = |\boldsymbol{Y} - \boldsymbol{X}^\top \boldsymbol{\beta}_0|$.

Then place the knots, determine the penalty parameter, and estimate the error density, like the algorithm for one-sample problem which is discussed in Section 3.3.1. Given the density estimate $\hat{f}$ with chosen λ, define $\rho = \hat{f}(0) - \hat{f}$, and $\hat{\boldsymbol{\beta}}$ is the minimizer of (3.16).

Note that here the ρ function is redescending. That is to say, like Tukey's and other well-known redescending M-estimators we introduced in Chapter 2, it may not have a unique solution. Therefore, it is important to make sure that we are using a simple yet robust starting point $\boldsymbol{\beta}_0$, and this is the reason why we use the least absolute deviation estimation to get started.

3.4.2 Confidence Interval for Regression Parameters

For asymptotic normality of the sampling distribution of $\hat{\boldsymbol{\beta}}$, we follow the ideas of [Huber, 2004], section 7.6. In robust regression, M-estimation regression parameter $\hat{\boldsymbol{\beta}}$ which minimizing (3.16) has the following asymptotic distribution:

$$\hat{\boldsymbol{\beta}} \xrightarrow{\mathcal{D}} N(\boldsymbol{\beta}, \Sigma_\beta) \text{ , where } \Sigma_\beta = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \frac{E[\rho'(\varepsilon)^2]}{E[\rho''(\varepsilon)]^2}. \tag{3.17}$$

It is suggested in [Huber, 2004] ,section 7.6, to use the moment-based estimation

$$\tilde{\Sigma}_\beta = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \frac{\frac{1}{n}\sum_{i=1}^n \rho'\left(Y_i - \boldsymbol{X}_i^\top \hat{\boldsymbol{\beta}}\right)^2}{\left[\frac{1}{n}\sum_{i=1}^n \rho''\left(Y_i - \boldsymbol{X}_i^\top \hat{\boldsymbol{\beta}}\right)\right]^2}.$$

Alternatively, since we have an estimated error density $\hat{f}$ for the infeasible distribution $f(\varepsilon)$, we may again consider the plug-in estimation which leads to a more reliable estimation of the covariance matrix in our applications of robust regression:

$$\hat{\Sigma}_\beta = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \frac{\int_{-\infty}^{\infty} \hat{f}'(\varepsilon)^2 \hat{f}(\varepsilon) d\varepsilon}{\left[\int_{-\infty}^{\infty} \hat{f}''(\varepsilon) \hat{f}(\varepsilon) d\varepsilon\right]^2}. \tag{3.18}$$

Then we may make an asymptotic confidence interval based on quantiles of the asymptotic normal distribution, similar to the way we construct confidence intervals that we discussed in Section 3.3. But again, we prefer using the confidence intervals based on the plug-in covariance, because of the benefits we discussed in Section 3.3.

3.4.3 Prediction Interval for New Responses

For a given predictor value $\boldsymbol{X}_{new}^{\top}$, we have the prediction

$$Y_{pred} = \boldsymbol{X}_{new}^{\top}\hat{\boldsymbol{\beta}}. \tag{3.19}$$

Taking the error term into account, we have the full expression of the modeled new response:

$$Y_{new} = \boldsymbol{X}_{new}^{\top}\hat{\boldsymbol{\beta}} + \varepsilon. \tag{3.20}$$

The modeled new response y_{new} contains two different parts of uncertainty, the variation of $\hat{\beta}$ which comes from the sampling distribution, and obviously the variation of error term ε. Note that they are independent in the model, since a new error does not affect the "old" estimation.

If we know the distribution of y_{new}, we may simply use its $\alpha/2$ quantile L and $1-\alpha/2$ quantile U to get a confidence interval $CI = (L, U)$ such that

$$P(L < Y_{new} < U) = 1 - \alpha.$$

In ordinary regression problem, we assume the error term ε follows a normal distribution, then quantify the combined uncertainty in (3.20) by the additive property of normal distribution. Note that generally for a distribution, using only the variance to describe its distribution and make a confidence interval is not adequate. But using a cumulative density function will be.

In our application of AAME robust regression, we have both of the asymptotic sampling distribution of Y_{pred}, and the estimated error density $\hat{f}_{\lambda}$ which is not necessarily a normal distribution. To get the asymptotic distribution of their summation, by their independence, we may use a con-

volution:

$$f_{new}(z) = (f_{pred} * \hat{f}_{\varepsilon})(z),$$

and

$$F_{new}(z) = \int_{-\infty}^{\infty} F_{pred}(z-\varepsilon)\hat{f}_{\varepsilon}(\varepsilon)d\varepsilon,$$

where f_{new} and f_{pred} are the probability density functions of Y_{new} and Y_{pred}. F_{new} and F_{pred} are the cumulative density functions of Y_{new} and Y_{pred}. Note that here

$$Y_{pred} \xrightarrow{\mathcal{D}} N\left(\mu_{pred}, \sigma^2_{pred}\right),$$

where $\mu_{pred} = \boldsymbol{X}^{\top}_{new}\boldsymbol{\beta}$, and $\sigma_{pred} = \boldsymbol{X}^{\top}_{new}\Sigma_{\beta}\boldsymbol{X}_{new}$, by (3.17) and (3.19).

By monotone and symmetric assumptions on error density f and asymptotic normality of $\hat{\boldsymbol{\beta}}$, f_{new} is also monotone and symmetric. Then it is straightforward to consider using the center interval as the shortest confidence interval. So we can find the 97.5% percentile C by doing a grid search on F_{new} such that $F^{-1}_{new}(0.975) = C$, then $(2Y_{pred} - C, C)$ is the 95% confidence interval and that

$$\mathbb{P}(2Y_{pred} - C \leq Y_{new} \leq C) = 0.95$$

Furthermore, we may do a repeated grid search to improve the accuracy of prediction intervals. Figure 3.13 shows an example of estimated cumulative density function F_{new} for Y_{new}, and the corresponding prediction interval based on F_{new}, with sample size $n = 100$, and error distribution $f = t(2)$. The solid line shows the estimated F_{new}. The two vertical dashed lines show the prediction interval corresponding to the 2.5% and 97.5% quantiles of F_{new}. Simulation studies in Section 3.5 indicates that this methodology of robust prediction interval works appropriately in our AAME.

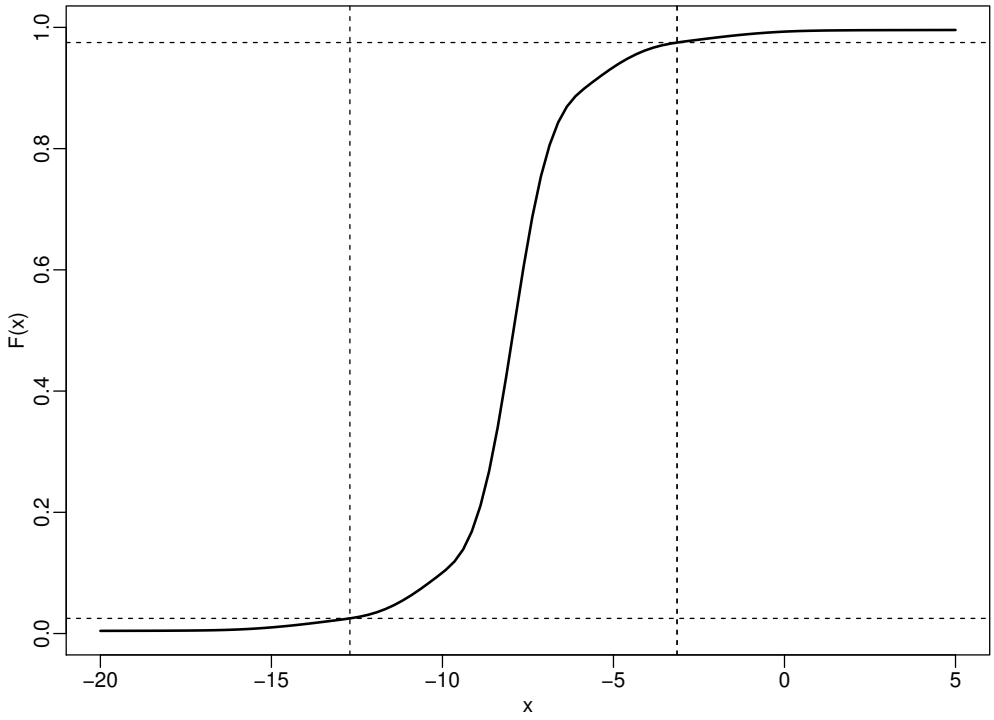

Figure 3.13: An example of estimated cumulative density function F_{new} for Y_{new}, and the corresponding prediction interval based on F_{new}, with sample size $n = 100$, error distribution $f = t(2)$. The solid line shows the estimated F_{new}. The dashed lines show the prediction interval corresponding to the 2.5% and 97.5% quantiles of F_{new}.

3.4.4 An Example of Application

To give an example of a data analysis where the error density is very heavy-tailed, consider the data set in [Lewis et al., 2006]. The purpose of the study was to determine the percent of school-age children in the state of Georgia who are overweight or obese, and to attempt to find predictors for overweight status. Researchers went to the schools to measure the heights and weights of children, and in addition, a questionnaire was given to ask the children about diet and lifestyle. In the questionnaire, children were asked to report their heights and weights, before being measured.

Fit the data with simple linear regression model

$$Y_i = \beta_0 + x_i\beta_1 + \varepsilon_i, \quad i = 1, \ldots, n, \tag{3.21}$$

where x_i are the measured weight (kg), Y_i are the reported weight, β_0 and β_1 are the unknown regression intercept and slope to be estimated.

Figure 3.14 shows the reported weights for the $n = 1540$ 8th and 11th grade students, against the actual measured weights. While the majority of the children had a good idea of their weights, many were quite far from the target. The dashed line shows the diagonal; points on this line represent children who knew their weight exactly. The least-squares line has a slope smaller than one, and is shown as the red dotted (lowest) line in the plot. The middle solid line is the AAME line. This is closer to the diagonal, because it is more robust to the many outliers.

Table 3.2 shows the estimated coefficients and confidence intervals, by ordinary least-squares (OLS), Huber's M-estimation, and our AAME. We can see that AAME has the estimated slope the closest to diagonal, and that AAME has the shortest confidence intervals among all three estimators.

The estimated error density for the reported weights against the measured weights is shown on Figure 3.15, with the residuals from the line estimated with AAME. The estimated density is quite peaked, so that the points close to the line are given the most weight in determining the fit, and many outliers are virtually ignored.

Table 3.2: The estimated coefficients and confidence intervals, by ordinary least-squares (OLS), Huber's M-estimation, and AAME.

Estimator	Coefficients		Confidence Interval	
	$\hat{\beta}_0$	$\hat{\beta}_1$	β_0	β_1
OLS	8.83	.841	$(7.366, 10.301)$	$(.819, .863)$
Huber	3.70	.926	$(3.037, 4.363)$	$(.916, .936)$
AAME	.819	.978	$(.159, 1.478)$	$(.969, .988)$

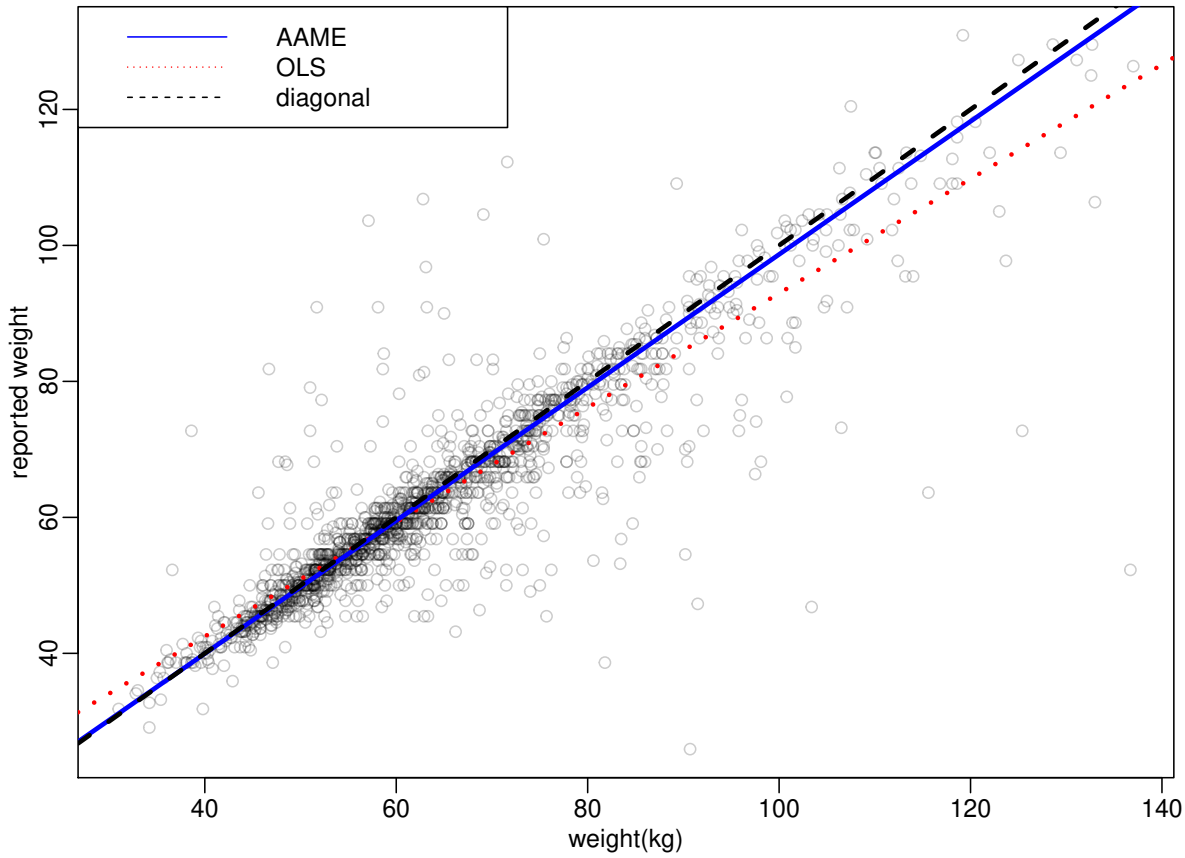

Figure 3.14: The reported weights for the $n = 1540$ 8th and 11th grade students, against the actual measured weights. The dashed line shows the diagonal. The least-squares line is shown as the red dotted (lowest) line. The middle solid line is the AAME line.

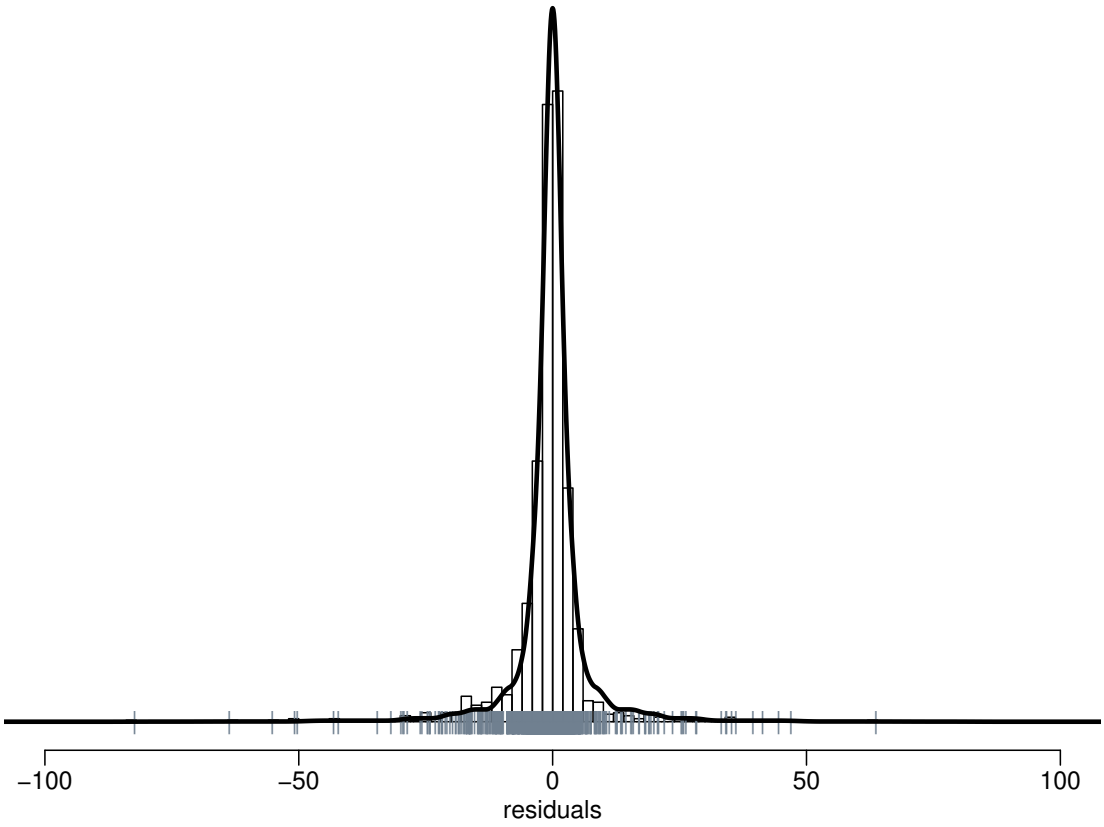

Figure 3.15: The estimated error density for the reported weights against the measured weights, and the residuals from the line estimated with AAME.

3.5 Simulations

In this section, we generate data from the one-sample model (3.10) and the simple linear regression model (3.21). We first check the performance of bootstrap confidence interval against non-normal distribution with small sample size. We then compare the performance of our AAME and other estimators in one sample problem. Lastly, we check the performance of our AAME in robust regression problem, and especially, with robust prediction interval checked.

3.5.1 Bootstrap Confidence Interval for Small Sample Size

A simulation study is conducted to verify that the distribution of our bootstrap $\hat{\mu}^*$ is a good approximation of the sampling distribution of $\hat{\mu}$, especially when we are only looking for the the 2.5% and 97.5% quantiles, with heavy-tailed error density and small sample size.

We generate data from one-sample model (3.10) with errors from a heavy-tailed distribution, the $t(2)$. There are 1000 replicates, bootstrap size 1000 in each replicate. Two different sample sizes, $n = 20$ and $n = 100$ are considered.

The distribution of $\hat{\mu}$ is centered at μ, describing the variation of $\hat{\mu}$ about its center. The distribution of $\hat{\mu}^*$ is centered at each $\hat{\mu}$, describing the variation of $\hat{\mu}^*$ about its center. So even if the centers are quite different the two variations about the centers is supposed to be approximated equal. Figure 3.16 shows how the bootstrap cumulative distribution function (CDF) approximate the empirical cumulative distribution function, with sample size $n = 20$ and error distribution $f = t(2)$. The figure shows the bootstrap CDF as a red line with points, and the empirical CDF as a black line. Dashed lines are the 0.025 and 0.975 percentiles, corresponding to the target of confidence intervals. We can see the bootstrap method work effectively.

Additionally, Table 3.3 shows the comparison between asymptotic confidence interval and bootstrap interval, against small sample size and heavy-tailed error distribution. We can see that with a reasonably large sample size, $n = 100$, both CI works good. However, with small sample size, bootstrap CI works more reliably. Also note that, these two CI's have similar average widths to each other.

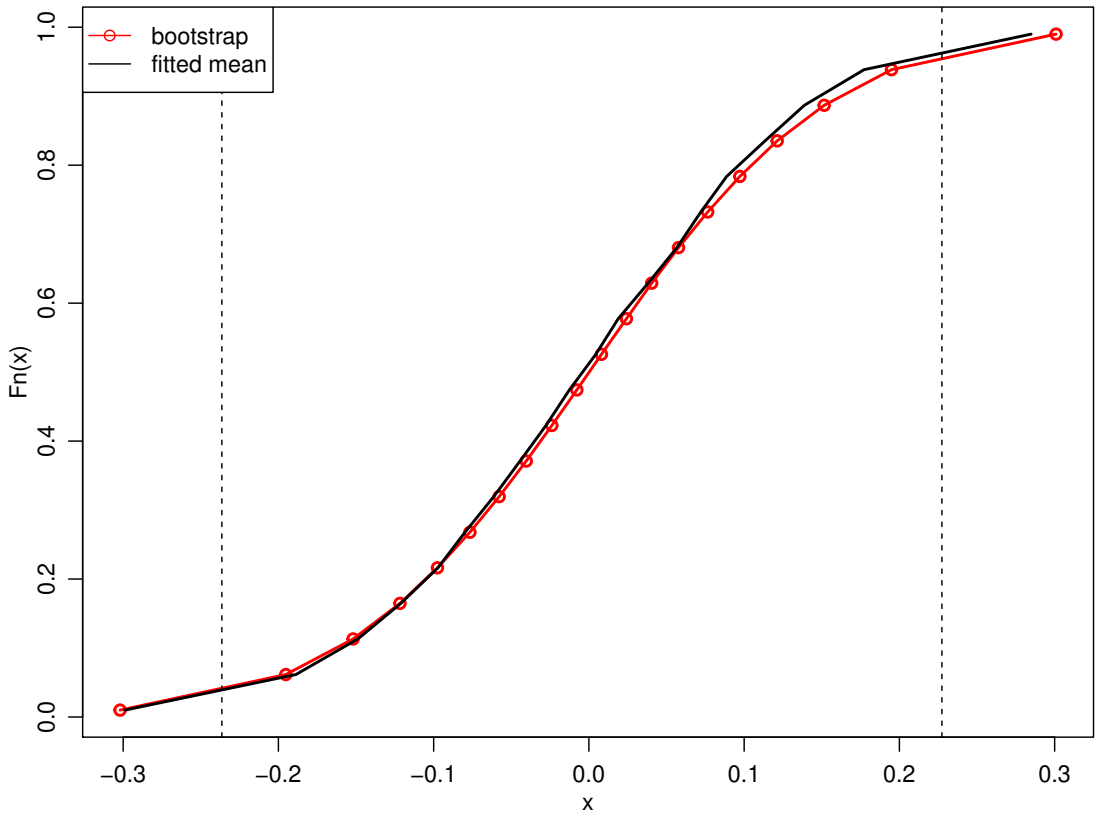

Figure 3.16: Simulation settings: $n = 20$, $f = t(2), reps = 1000$, bootstrap size $= 1000$ in each rep. λ is selected by cross validation. Two dashed lines are the 2.5% and 97.5% percentile, which relate to CI's.

Table 3.3: Simulation results comparing asymptotic and bootstrap confidence intervals.

Error		Coverage rate		Average CI width	
Density	n	Asymptotic	Bootstrap	Asymptotic	Bootstrap
$t(2)$	20	.937	.946	1.354	1.391
$t(2)$	100	.952	.957	.576	.583

3.5.2 One-Sample Problem

We generate data from one-sample model (3.10) with errors from various distributions: the standard normal distribution $N(0,1)$, the t-distribution with 2 degrees of freedom, a symmetric unimodal Pareto distribution with shape parameter 2, a standard normal distribution contaminated by Gaussian noise, a standard normal contaminated by $t(2)$ noise, and the Cauchy distribution. For each of 3000 samples, we estimate μ using the AAME method, and use (3.3) to compute a 95% confidence interval for μ.

The square root of mean square error,

$$SMSE = \sqrt{\frac{1}{3000}\sum_{i=1}^{3000}(\hat{\mu}_i - \mu)^2},$$

confidence coverage rate, and average width of confidence interval are reported in Table 3.4. The notation $20\%N(0,10^2)$ represents $80\%N(0,1) + 20\%N(0,10^2)$ contamination, and $20\%t(2,8)$ represents $80\%N(0,1) + 20\%t(2,8)$ contamination, where $t(2,8)$ is with two degrees of freedom and scale 8. Pareto(2) uses shape 2, multiplied randomly by -1 or 1 for symmetry. Our method is compared to least-squares (LS), Huber's M-estimator (H), MM-estimator [in [Yohai, 1987]] with Tukey's biweight (T), and the median (M). The AAME estimator performs the best for the most heavy-tailed (Cauchy, Pareto) or most contaminated (50% or more) error distributions. For other distributions, the performance is similar to that of the median or the Tukey method.

Note that H and T estimation are computed by `R` function `rlm`, with default settings for their tuning parameters. Also, Figure 3.17 and Figure 3.18 show the comparison among estimators H, T and A. The figures show the empirical quantiles of absolute deviation, $|\hat{\mu} - \mu|$, for different estimators, with sample size $n = 200$. We again can see AAME presents good performance against fat-tailed and/or highly-contaminated data.

Table 3.4: Comparison of the AAME with least-squares (LS), Huber (H), Tukey (T), and the median (M), by square root of mean squared error, as well as the coverage rates and confidence interval widths, for a 95% target coverage.

Error		SMSE					Coverage rate					Average CI width				
Density	n	LS	H	T	M	A	LS	H	T	M	A	LS	H	T	M	A
$N(0,1)$	200	**.072**	.074	.074	.089	.088	.947	.948	.947	.947	.940	**.277**	.284	.284	.352	.345
	500	**.045**	.046	.046	.056	.056	.952	.949	.949	.949	.946	**.175**	.180	.180	.221	.219
$t(2)$	200	.343	.100	**.098**	.101	.102	.958	.948	.949	.953	.942	.844	.387	**.379**	.402	.401
	500	.173	.063	**.061**	.063	.064	.965	.947	.943	.956	.942	.539	.245	**.239**	.250	.253
Cauchy	200	33.5	.136	.122	.111	**.106**	.978	.951	.945	.950	.957	19.1	.524	.475	.453	**.449**
	500	346	.084	.076	.070	**.069**	.977	.947	.951	.949	.946	59.6	.329	.298	.279	**.274**
Pareto(2)	200	.252	.056	.053	**.039**	**.039**	.957	.950	.946	.951	.948	.697	.219	.206	.161	**.160**
	500	.157	.035	.032	**.023**	**.023**	.961	.953	.952	.952	.949	.471	.137	.129	.097	**.096**
$20\%N(0,10^2)$	200	.328	.104	**.089**	.108	.100	.944	.948	.946	.949	.942	1.26	.405	**.346**	.430	.392
	500	.199	.064	**.056**	.068	.063	.955	.955	.949	.953	.946	.795	.255	**.218**	.269	.247
$50\%N(0,10^2)$	200	.510	.259	.242	.163	**.132**	.950	.952	.956	.954	.947	1.97	.975	.928	.650	**.534**
	500	.317	.154	.144	.101	**.084**	.953	.958	.952	.957	.949	1.24	.605	.573	.405	**.334**
$80\%N(0,10^2)$	200	.638	.614	.614	.333	**.283**	.943	.949	.948	.948	.941	2.48	2.38	2.38	1.38	**1.13**
	500	.402	.385	.384	.202	**.157**	.951	.949	.944	.952	.949	1.57	1.51	1.51	.815	**.641**
$20\%t(2,8)$	200	.887	.103	**.089**	.107	.100	.964	.952	.945	.948	.944	2.57	.402	**.345**	.430	.394
	500	.663	.064	**.055**	.068	.063	.959	.953	.951	.950	.946	1.73	.254	**.218**	.268	.248
$50\%t(2,8)$	200	1.71	.251	.220	.161	**.133**	.959	.946	.946	.956	.949	4.41	.937	.833	.644	**.547**
	500	.931	.147	.130	.099	**.083**	.962	.958	.955	.951	.951	2.88	.583	.517	.400	**.335**
$80\%t(2,8)$	200	2.59	.618	.588	.320	**.265**	.957	.947	.946	.946	.950	5.96	2.38	2.27	1.33	**1.13**
	500	1.23	.388	.369	.197	**.158**	.962	.941	.945	.951	.955	3.80	1.50	1.43	.785	**.639**

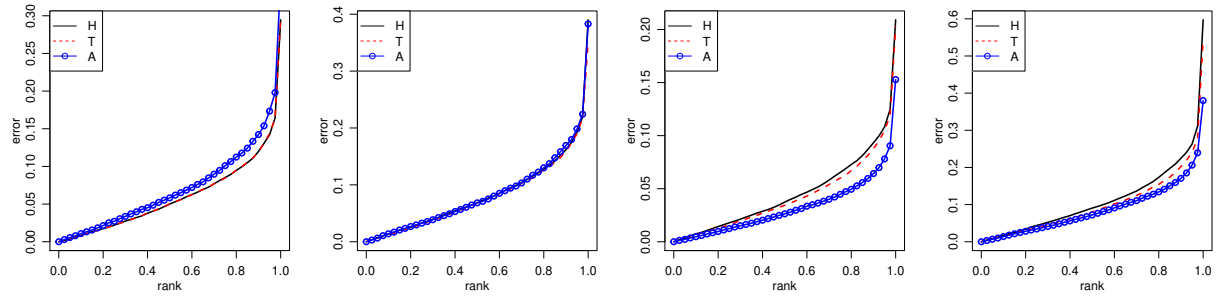

Figure 3.17: Distributions (left to right): $N(0,1)$, $t(2)$, Pareto(2), and Cauchy, with $n = 200$.

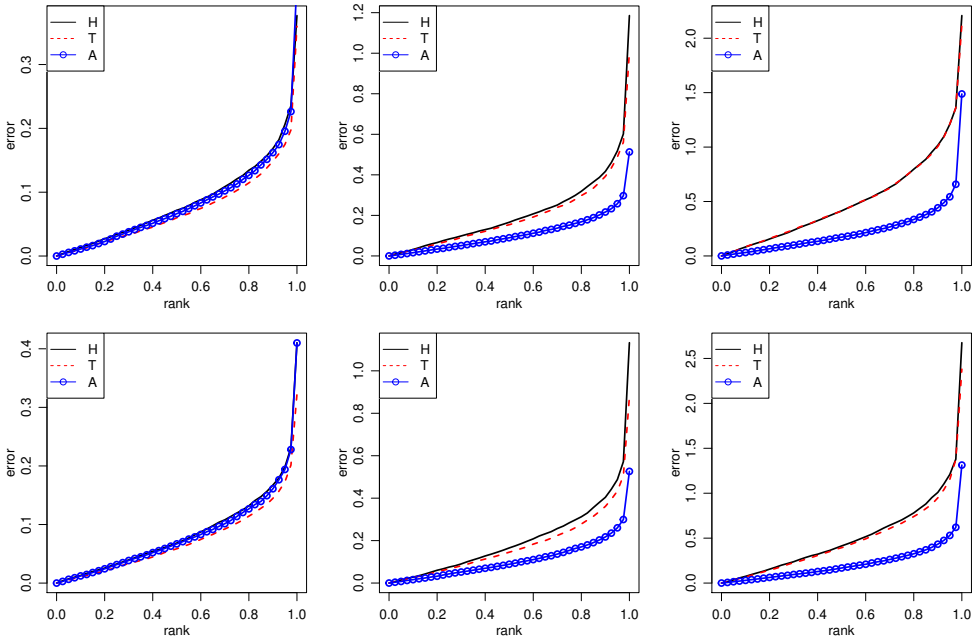

Figure 3.18: First row (left to right): $20\%N(0, 10^2)$, $50\%N(0, 10^2)$, $80\%N(0, 10^2)$. Second row (left to right): $20\%t(2, 8)$, $50\%t(2, 8)$, and $80\%t(2, 8)$.

3.5.3 Robust Prediction Interval

We generate data from model (3.21) with errors from various distributions (same to the simulation of one-sample problem). For each of 3000 samples, we estimate β_0 and β_1 using the AAME method, use (3.18) to compute a 95% confidence interval for β_1, and use the method discussed in Section 3.4.3 to compute a 95% robust prediction interval for a new observation with new predictor x_{new}.

The square root of mean square error,

$$SMSE = \sqrt{\frac{1}{3000}\sum_{i=1}^{3000}(\hat{\beta}_{1,(i)} - \beta_1)^2},$$

confidence coverage rate, average width of confidence interval, prediction coverage rate, average width of prediction interval are reported in Table 3.5. The same notations in simulation of one-sample problem are used to represent different error distributions. Our method is compared to least-squares (LS) and Huber's M-estimator (H).

The AAME estimator performs the best for the most heavy-tailed (Cauchy, Pareto) or most contaminated (50% or more) error distributions. Moreover, our AAME provides a robust prediction interval more reliable than the LS estimator, which is unavailable for other robust estimators.

Table 3.5: Comparison of the AAME with least-squares (LS) and Huber (H), by square root of mean squared error, as well as the coverage rates and confidence interval widths, for a 95% target coverage. Robust prediction intervals are also reported. Sample size $n = 200$.

Error	SMSE			CI Coverage			CI width			PI coverage		PI width	
Density	LS	H	A	LS	H	A	LS	H	A	LS	A	LS	A
$N(0,1)$	**.012**	.013	.015	.940	.944	.952	**.048**	.049	.059	.945	.947	3.94	3.98
$t(2)$	.050	**.017**	**.017**	.953	.947	.953	.139	**.067**	.070	.957	.954	36.5	32.7
Cauchy	211.3	.023	**.019**	.968	.951	.947	11.8	.091	**.084**	.970	.952	978	44.7
Pareto(2)	.059	.010	**.007**	.954	.943	.944	.122	.038	**.029**	.965	.961	10.1	8.85
$20\%N(0,10^2)$	.055	.018	**.017**	.952	.952	.953	.216	.070	**.068**	.929	.951	17.8	22.4
$50\%N(0,10^2)$	.085	.044	**.023**	.954	.950	.957	.338	.170	**.098**	.921	.948	27.9	32.5
$80\%N(0,10^2)$	.108	.104	**.051**	.949	.947	.936	.426	.409	**.197**	.935	.948	35.2	36.9
$20\%t(2,8)$	.172	.018	**.017**	.953	.947	.955	.442	.069	**.068**	.957	.954	36.5	32.7
$50\%t(2,8)$	.285	.044	**.023**	.952	.947	.957	.753	.163	**.098**	.953	.958	62.3	56.7
$80\%t(2,8)$	.359	.104	**.053**	.950	.951	.960	.977	.410	**.245**	.956	.961	79.2	74.1

3.6 Discussion

This chapter establishes a penalized constrained nonparametric method of density estimation for error distribution. The new density estimation is unimodal, has stabilized first two derivatives, and integrate to 1, so that satisfies the requirements of being used as the ρ function in M-estimation and in further inferences. A theorem is shown to ensure the consistency of the new density estimation. Based on the new density estimation, this chapter then establishes a new robust estimation, AAME, which is highly adaptive to the data. Consistency and asymptotic sampling distribution are established for AAME, and it shows a good performance against influential outliers. Based on estimated CDF resampling, a convenient bootstrap method is established for inferences with a small sample size. Another significant advantage of AAME is its robust prediction interval which is more reliable over existing methods.

Bias with penalized estimations

For penalized methods, bias is almost always a problem that is worth looking through. Even though we used a penalization to make the error density estimation smooth, the good thing here is

that our parametric regression $\mu(x_i)$ will not be biased. Because the penalty we have put in AAME is not on the $\mathcal{L}_1$, $\mathcal{L}_2$, or any other scale of the regression parameters in $\mu(x_i)$. The penalty is on the "consecutive differences of the second derivative" of the ρ function, and the ρ function will always remain symmetrical.

Fitting AAME model in an iterative manner

Because the computational burden for current AAME is fairly low, it is suitable to do an iterative form of AAME. That is

---

1 Using the sample median μ_0, determine the support $[0, S]$.

2 Select a penalty value λ by cross-validation.

3 Get $\hat{f}_\lambda(x)$ with $\boldsymbol{x} = |\boldsymbol{y} - \mu_0 \mathbf{1}|$.

4 Get $\hat{\mu}$ by using $\hat{f}_\lambda(x)$.

5 Loop steps 3 and 4 until converge.

With both of $\hat{\mu}$ and $\hat{f}$ being updated, we may get an overall better AAME result.

Heavy-tailed and dependent errors

In practice, we sometimes would need to deal with errors being not only heavy-tailed, but also dependent. In that case, we would better take the dependence into account, and fit the additional parameters with a AAME type of estimation. Next chapter establishes this method.

Chapter 4

Auto-Adaptive M-estimation for Linear Model with Heavy-Tailed and Autoregressive Errors

4.1 Introduction

Ordinary regression requires assumptions of *iid* normality on the error term. When any of the assumptions is violated, we may need to consider alternative procedures. Suppose we observe longitudinal data $(\boldsymbol{x}_t, Y_t)$ at time points $t = 1, \ldots, T$, and assume

$$\begin{aligned} y_t &= \mu(x_t) + \varepsilon_t; \\ \varepsilon_t &= \sum_{i=1}^{p} \phi_i \varepsilon_{t-i} + e_t, \end{aligned} \tag{4.1}$$

where $\mu(x_t)$ is a linear model, ε_t are autoregressive AR(p) errors with order p, and e_i are *iid* symmetric mean-zero innovations (or say white noises). Model (4.1) may be used to represent lots of applications in different fields like finance, economics and earth sciences. [Cochrane and Orcutt, 1949] proposed a very well-known modification of ordinary least-squares regression (Cochrane-Orcutt procedure) which takes the autoregression of errors into account, by estimating the linear model and the AR(1) coefficient iteratively.

However, Cochrane-Orcutt procedure assumes normal distribution on the innovation e_t, which is often violated in reality. In the presence of outliers or other evidence that the convenient assumption of normal innovations is incorrect, practitioners turn to robust methods where knowledge of the error density family is not required. Classical robust M-estimation of the regression function $\mu(\boldsymbol{x})$ is accomplished by choosing a function ρ and then finding μ to minimize $\sum_{i=1}^{n} \rho\left(Y_i - \mu(\boldsymbol{x_i})\right)$ over some parameter or function space. This was first proposed by [Huber, 1964], with further asymptotic results in [Huber, 1973]. [Yohai, 1987] proposed an iterative method for an MM estimator, where the ρ function is adjusted based on the residuals from an initial M estimator. Many other

papers regarding improvements on robust M-estimations have been proposed. Lots of literature studied the asymptotic behaviors of robust estimations against heavy-tailed and dependent errors, when we do not take into account the dependency effects. For example, [Gastwirth and Rubin, 1975] studied the efficiency of robust estimations against AR(1) and heavy-tailed errors, [Koul, 1977] studied the efficiency of M-estimations when errors are strictly stationary and strongly mixing. [Wu, 2007] derived asymptotic properties of M-estimation for linear models with errors being a form of general short-range dependence and heavy-tailed linear processes.

But if the form of dependency on the error term is sort of known, we would better consider taking both the dependence and heavy-tail of error into account, to improve the efficiency of estimation. [Portnoy, 1977] and [Portnoy, 1979] gave a solution for MA errors, based on Huber's M-estimation. [Lee and Martin, 1986] established a proper M-estimation which takes the AR(1) dependency into account. [Coursey and Nyquist, 1986] established a Cochrane-Orcutt type M-estimation which significantly improves the estimation efficiency of a linear model with AR(1) and heavy-tailed errors.

However, there is still remaining unsatisfaction. On one hand, no matter M-estimation or MM-estimation, both require an important tuning constant and a few stages to determine an estimated scale to maintain their high efficiency – see [Susanti et al., 2014] for a summary. To the best of our knowledge, these is no available algorithm implement of well-tuned M-estimation or MM-estimation for fitting model (4.1). On the other hand, new challenges in longitudinal data analysis have raised. [Ling, 2005] points out that infinite variance autoregressive (IVAR) models have been having wider and wider usage. See also [Davis et al., 1992], [Hill, 2013] and [Wu and Cui, 2018] for examples of relevant methodologies and applications. Besides, [Mikosch et al., 1995] proposed well-known Whittle estimation for infinite variance ARMA models, and derived asymptotic properties.

[Chen and Meyer, 2020b] proposed Auto-Adaptive M-Estimation (AAME) which inherits asymptotic properties of traditional M-estimators, and has its own nice features. It adapts to the data and down-weights influential outliers automatically, so that requires no priori of innovation

distribution, yet presents high breakdown-point against the ϵ-contamination type of influential outliers – raised by [Huber, 1964]. AAME also presents improved performance against fat-tailed errors – refers to those very heavy-tailed distributions with a low tail-index and corresponding infinite moments, see [Haas and Pigorsch, 2009] for more formal definitions and [Cooke et al., 2014] for examples.

In this chapter, we propose a novel AAME solution of fitting model (4.1), by estimating the linear model and the AR process jointly, while the innovation density is not necessarily normal and would be estimated with constrained penalized splines. We start with the one-sample problem with AR(1) errors, then extend to one-sample problem with AR(p) errors. We refer [Xiao, 2004] as a literature review for recent major results on robust regression with dependent errors. That paper establishes a partially adaptive and proper M-estimation which takes the dependency into account, but only adapts within the range of t-distributions with different tail-thickness.

This chapter is organized as follows: in the next section, we review the Auto-Adaptive M-estimation (AAME) as a prerequisite, then apply AAME to one-sample model with heavy-tailed and AR(1) dependent errors. In Section 4.3, AAME is applied to one-sample model with heavy-tailed and AR(p) dependent errors. Section 4.4 shows the results of simulation that we conducted, to compare our methods to other estimators.

4.2 AAME with AR(1)

4.2.1 Background of AAME

The M-estimator uses a function ρ and sample values $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, determine $\hat{\boldsymbol{Y}} = (\hat{Y}_1, \ldots, \hat{Y}_n) \in \mathbb{R}^n$ to minimize

$$M_n(\hat{\boldsymbol{Y}}; \boldsymbol{Y}) = \sum_{i=1}^{n} \rho(Y_i - \hat{Y}_i). \tag{4.2}$$

If the noise density family is known, then the ρ function may be chosen accordingly. Choices include least-square, least-absolute-deviation, [Huber, 1964]'s, and many others. The function

$\rho(y) = y^2$ for $|y| \leq k$ and $\rho(y) = k^2$ for $|y| > k$ leads to the "trimmed mean," which completely discounts outliers beyond the trim-point k. [Hampel et al., 2011a] argued that "redescending" ρ functions, where the derivative approaches zero for large y, are effective for heavy-tailed or contaminated errors. These popular choices of ρ functions are shown in Figure 2.1.

The general idea here is that for heavy-tailed or highly contaminated errors, down-weighting those influential outliers would decrease their impacts so that it will improve the overall performance of the M-estimator. For the proposed method, the noise density f is estimated using a constrained least-squares penalized spline density estimator, and then we use the negative of the density estimate $\hat{f}$ as the ρ function. We define $\rho = \hat{f}(0) - \hat{f}$ (traditionally we have $\rho(0) = 0$). Then $\hat{\boldsymbol{Y}}$ is the minimizer of (4.2). In practice this ρ function is similar to that for the "smoothed" trimmed mean suggested by Huber and Tukey, with a data-driven choice of trim points. We call this the Auto-Adaptive M-Estimator (AAME). Note that under mild conditions and a bounded rate of penalty, the constrained least-squares penalized spline density estimator achieves convergence rate $O_p(n^{-3/7})$. See [Meyer, 2012] for the the smoothness and other benefits of a constrained least-squares spline density estimator, and [Chen and Meyer, 2020b] discusses penalizing the consecutive differences of the second derivative function, which stabilizes the estimated density function and its first two derivatives, and allows many knots to be specified without over-fitting, as Figure 3.3 shown. The absolute values of a sample from a mixture of normal densities are shown as small gray tick marks. Density estimates with and without penalty are shown as the black curves. While the density estimates are not that different, their first and second derivatives vary more smoothly with the penalty. See [Chen and Meyer, 2020b] also for more details about AAME.

4.2.2 Combining AAME with AR(1)

Now let's consider applying AAME for estimating the coefficients below model of longitudinal data. At time points $t = 1, \ldots, T$, suppose the model of one-sample with AR(1) errors

$$\begin{aligned} y_t &= \mu + \varepsilon_t; \\ \varepsilon_t &= \phi_1 \varepsilon_{t-1} + e_t, \end{aligned} \tag{4.3}$$

where the serial y_t is the dependent variable (of major interest), μ (mean) is the linear model coefficient, ε_t is the error generated by the AR(1) process with autoregressive coefficient ϕ_1. The innovation term e_t is assumed to be independent and identically distributed, with unknown symmetric unimodal density function f. The parameter vector $(\mu, \phi_1) \in U = \mathbb{R} \times [-1, 1]$ is to be estimated jointly, after the innovation density function f is estimated by the constrained penalized spline density estimator.

For error term lagged by one period, plug $\varepsilon_{t-1} = y_{t-1} - \mu$ into (4.3), we have

$$y_t - \mu = \phi_1(y_{t-1} - \mu) + e_t,$$

$t = 2, \cdots, T$, or

$$e_t = y_t - \phi_1 y_{t-1} - (1 - \phi_1)\mu.$$

With a ρ function optimizing the discrepancy, we then have the objective function:

$$M_n(\mu, \phi_1) = \sum_{t=2}^{T} \rho\Big(y_t - \phi_1 y_{t-1} - (1 - \phi_1)\mu\Big), \tag{4.4}$$

where $\rho = \hat{f}(0) - \hat{f}$, and $\hat{f}$ is the constrained penalized spline density estimation which is obtained from a warming-up step. The following subsections will be discussing the algorithm implement in more details.

With such data-driven ρ function, the estimated coefficients $(\hat{\mu}, \hat{\phi}_1)$ obtained by minimizing (4.4) adapts to the data and down-weights influential outliers automatically, so that it presents high breakdown-point against the ϵ-contamination type of influential outliers and/or fat-tailed noises.

4.2.3 Choosing the Starting Point

Note that model (4.3) is a nonlinear model, with parameters not being additive. Hence the objective function (4.4) has a surface which needs to be treated carefully – needs to find a good starting point.

Figure (4.1) shows an example of response contour of $M_n(\mu, \phi_1)$ across the inputs (μ, ϕ_1). The random data of the example is generated from a $t-$distribution with degree of freedom 2, and the true coefficients $(\mu, \phi_1) = (100, 0.6)$. We can see that the gradients of the contour are fairly good around a Y-shaped neighbor of the optimal point $(100, 0.6)$, while they are almost completely "flatten out" outside of the Y-shaped neighbor.

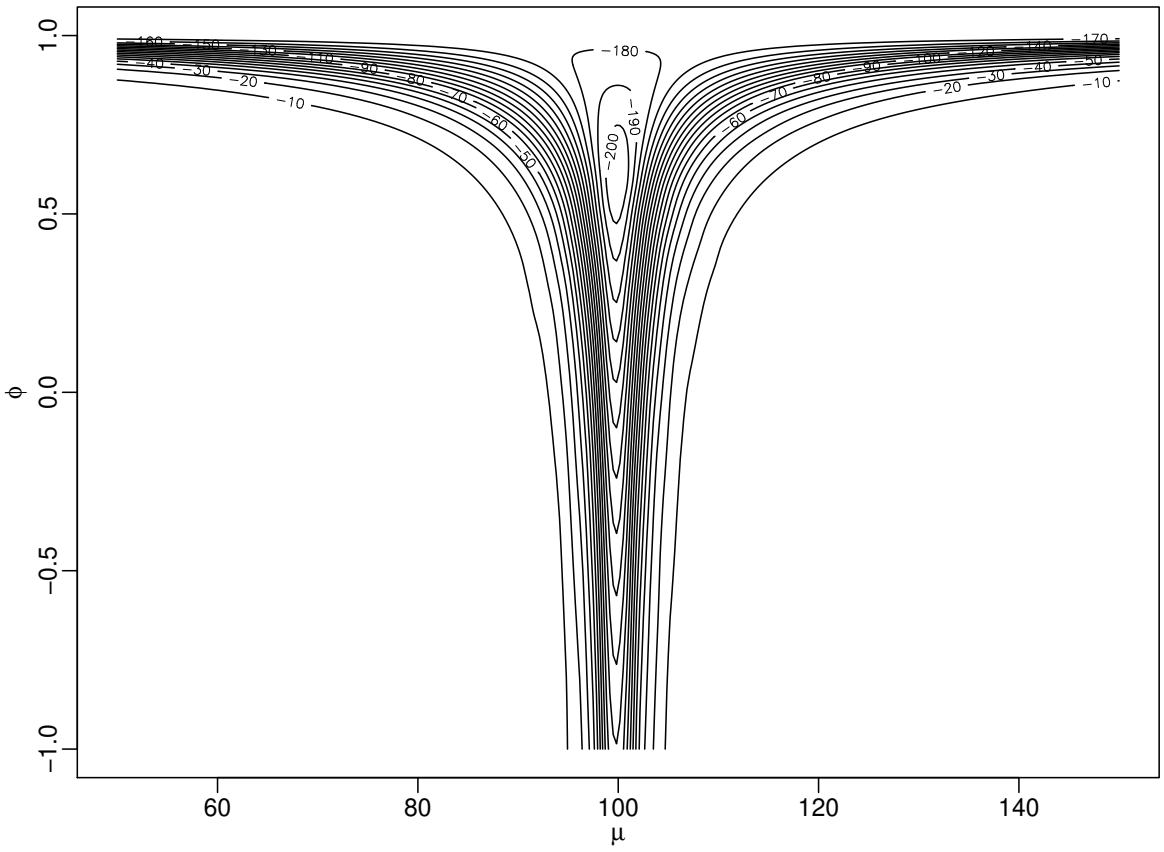

Figure 4.1: Contour of the profile objective function (4.4), across μ and ϕ. The random data $\boldsymbol{Y}$ is generated from $\mu = 100$, $\phi = 0.6$, and $t(2)$ innovation.

Therefore, finding a fairly good starting-point would play a very important role in the estimation. Otherwise, the algorithm doing optimization would stop too early before entering the Y-shaped neighbor and approaching a valid gradient. In empirical studies, with a bad starting point, it does happen to generates crazy estimated coefficients $(\hat{\mu}, \hat{\phi}_1)$, no matter whether the innovation term e_t is heavy-tailed or not.

4.2.4 Algorithm Implementation

We first estimate μ and ϕ partially, by a Cochrane-Orcutt type iteration of least-absolute-deviance estimation, as a warming-up stage to find a good starting point and to estimate the noise density function (constrained penalized spline density estimation). We then do a joint estimation to finish up fitting the model. The algorithm implement is summarized as below. User-friendly code is available at [Chen, 2020].

Input: observations $\boldsymbol{y}$

Output: estimated model coefficients $(\hat{\mu}, \hat{\phi}_1)$, and estimated noise density function $\hat{f}$

1 $\hat{\mu} \leftarrow median(y)$

2 $\hat{\phi}_1 \leftarrow \arg\min_{\phi_1} \sum_{t=2}^{T} \left| y_t - \phi_1 y_{t-1} - (1-\phi_1)\hat{\mu} \right|$

3 $\hat{\mu} \leftarrow \arg\min_{\mu} \sum_{t=2}^{T} \left| y_t - \hat{\phi}_1 y_{t-1} - (1-\hat{\phi}_1)\mu \right|$

4 Loop steps 2 and 3 until converge

5 $e_t \leftarrow y_t - \hat{\phi}_1 y_{t-1} - (1-\hat{\phi}_1)\hat{\mu}$, for $t = 2, \cdots, T$

6 $\hat{f} \leftarrow$ constrained penalized spline density estimation, from vector $\{e_t\}_{t=2}^{T}$

7 $(\hat{\mu}, \hat{\phi}_1) \leftarrow \arg\min_{\mu,\phi_1} \sum_{t=2}^{T} \rho\Big(y_t - \phi_1 y_{t-1} - (1-\phi_1)\mu\Big)$, where $\rho = \hat{f}(0) - \hat{f}$

4.2.5 Confidence Intervals

An asymptotic solution

For the linear model with dependent errors, modeled by (4.3), we may take the dependence into account then derive an asymptotic confidence interval of μ. Consider starting from the asymptotic variance in independent errors model, then make a correction with the estimated autoregressive parameter $\hat{\phi}_1$:

$$\hat{V}(\hat{\mu}) = \frac{\int_{-\infty}^{\infty} \hat{f}'(\varepsilon)^2 \hat{f}(\varepsilon) d\varepsilon}{(T-1)(1-\hat{\phi}_1)^2 \left[\int_{-\infty}^{\infty} \hat{f}''(\varepsilon)\hat{f}(\varepsilon) d\varepsilon\right]^2}, \tag{4.5}$$

where $\hat{\phi}_1$ needs to be bounded at $-1 < \hat{\phi}_1 < 1$, to be stationary, and avoid an infinity in the denominator. Then compute the confidence intervals based on the quantiles of the asymptotic normality, with estimated variance $\hat{V}(\hat{\mu})$.

A bootstrap resampling solution

Alternatively, we may do a resampling on the estimated innovation density and then generate a bootstrap confidence interval for the parameters. The algorithm is as below:

---

1 Get the estimates of error density $\hat{g}_\lambda$ and parameters $(\hat{\mu}, \hat{\phi}_1)$.

2 Sample a vector of innovation $\boldsymbol{e}$ from $\hat{g}_\lambda$.

3 Make bootstrap data $\boldsymbol{y}^*$, by the model (4.3) with $(\hat{\mu}, \hat{\phi}_1)$ plugged-in.

4 Use $\boldsymbol{y}^*$ to get the corresponding bootstrap parameter estimation $(\hat{\mu}^*, \hat{\phi}_1^*)$.

5 Loop step 2 to 4, for an enough number of bootstrap estimation $(\hat{\mu}^*, \hat{\phi}_1^*)$.

6 Get the bootstrap confidence interval by the quantiles of the set of $(\hat{\mu}^*, \hat{\phi}_1^*)$.

Note that this resampling algorithm is based on parametric bootstrap, which requires that the assumed model form (4.3) is correct for the data.

4.3 AAME with AR(p)

AAME longitudinal models can also deal with higher order of autoregressive errors. At time points $t = 1, \ldots, T$, suppose the model of one-sample with AR(p) errors

$$\begin{aligned} y_t &= \mu + \varepsilon_t; \\ \varepsilon_t &= \sum_{i=1}^{p} \phi_i \varepsilon_{t-i} + e_t, \end{aligned} \tag{4.6}$$

where the serial y_t is the dependent variable (of major interest), μ (mean) is the linear model coefficient, ε_t is the error generated by the AR(p) process with autoregressive coefficients $(\phi_1, \cdots, \phi_p)$. The innovation term e_t is assumed to be independent and identically distributed, with unknown symmetric unimodal innovation density function f. The parameter vector $(\mu, \phi_1, \cdots, \phi_p) \in U = \mathbb{R} \times [-1, 1]^p$ is to be estimated jointly, after the density function f is estimated by the constrained penalized spline density estimator.

For errors lagged by $1, \cdots, p$ period, plug $\varepsilon_{t-i} = y_{t-i} - \mu$ into (4.6), we have

$$y_t - \mu = \sum_{i=1}^{p} \phi_i (y_{t-i} - \mu) + e_t,$$

$t = p + 1, \cdots, T$, or

$$e_t = y_t - \phi_1 y_{t-1} - \cdots - \phi_p y_{t-p} - \mu(1 - \phi_1 - \cdots - \phi_p).$$

With a ρ function optimizing the discrepancy, we then have the objective function:

$$M_n(\mu, \boldsymbol{\phi}) = \sum_{t=p+1}^{T} \rho\Big(y_t - \phi_1 y_{t-1} - \cdots - \phi_p y_{t-p} - \mu(1 - \phi_1 - \cdots - \phi_p)\Big), \tag{4.7}$$

where $\rho = \hat{f}(0) - \hat{f}$, and $\hat{f}$ is the constrained penalized spline density estimation which is obtained from a warming-up step.

Note that parameters in (4.6) are not additive, so that again we would need a good starting point. We first estimate μ and $(\phi_1, \cdots, \phi_p)$ partially, by a Cochrane-Orcutt type iteration of least-absolute-deviance estimation, as a warming-up stage to find a good starting point and to estimate the noise density function (constrained penalized spline density estimation). We then do a joint estimation to finish up fitting the model. The algorithm implement is summarized as below. User-friendly code is available at [Chen, 2020].

Input: observations $\boldsymbol{y}$, and order of autoregressive p (must-have)

Output: estimated model coefficients $(\hat{\mu}, \hat{\phi}_1, \cdots, \hat{\phi}_p)$, and estimated noise density function $\hat{f}$

1 $\hat{\mu} \leftarrow median(y)$

2 $(\hat{\phi}_1, \cdots, \hat{\phi}_p) \leftarrow \underset{\phi_1, \cdots, \phi_p}{\arg\min} \sum_{t=p+1}^{T} \left| y_t - \phi_1 y_{t-1} - \cdots - \phi_p y_{t-p} - \hat{\mu}(1 - \phi_1 - \cdots - \phi_p) \right|$

3 $\hat{\mu} \leftarrow \underset{\mu}{\arg\min} \sum_{t=p+1}^{T} \left| y_t - \hat{\phi}_1 y_{t-1} - \cdots - \hat{\phi}_p y_{t-p} - \mu(1 - \hat{\phi}_1 - \cdots - \hat{\phi}_p) \right|$

4 Loop steps 3 and 4 until converge

5 $e_t \leftarrow y_t - \hat{\phi}_1 y_{t-1} - \cdots - \hat{\phi}_p y_{t-p} - \hat{\mu}(1 - \hat{\phi}_1 - \cdots - \hat{\phi}_p)$, for $t = p+1, \cdots, T$

6 $\hat{f} \leftarrow$ constrained penalized spline density estimation, from vector $\{e_t\}_{t=p+1}^{T}$

7 $(\hat{\mu}, \hat{\phi}_1, \cdots, \hat{\phi}_p) \leftarrow \underset{\mu, \phi_1, \cdots, \phi_p}{\arg\min} \sum_{t=p+1}^{T} \rho\Big(y_t - \phi_1 y_{t-1} - \cdots - \phi_p y_{t-p} - \mu(1 - \phi_1 - \cdots - \phi_p) \Big)$, where $\rho = \hat{f}(0) - \hat{f}$

4.4 Simulation

4.4.1 Confidence Intervals for the Model with AR(1) error

In this simulation study, we generate data by model (4.3), with innovations from various distributions: the standard normal distribution $N(0, 1)$, the t-distribution with 2 degrees of freedom, a symmetric unimodal Pareto distribution with shape parameter 2, a standard normal distribution contaminated by Gaussian noise, and a standard normal distribution contaminated by $t(2)$ noise.

The notation $20\%N(0, 10^2)$ represents $80\%N(0, 1)$+$20\%N(0, 10^2)$ contamination, and $20\%t(2, 8)$ represents $80\%N(0, 1) + 20\%t(2, 8)$ contamination, where $t(2, 8)$ is with two degrees of freedom and scale 8. Pareto(2) uses shape 2, multiplied randomly by -1 or 1 for symmetry.

We do different levels of autoregression, $\phi_1 = 0, 0.3, 0.6$, or 0.9, and compute the average length of the confidence intervals (CI) for μ, by the asymptotic solution (A) and the bootstrap solution (B). The result is reported in Table 4.1, with the CI coverage rate reported as well. We can see solution A and B both work, though the solution B has better coverage rates against high level of autoregression.

Table 4.1: Results of confidence intervals for μ in AR(1) model, by the asymptotic solution (A) and the bootstrap solution (B), with $T = 200$ and $reps = 1000$.

Noise Density	Solution	CI length (coverage)			
		$\phi_1 = 0$	$\phi_1 = .3$	$\phi_1 = 0.6$	$\phi_1 = 0.9$
$N(0, 1)$	A	.345(.939)	.494(.937)	.861(.929)	3.42(.903)
	B	.351(.943)	.498(.938)	.870(.934)	3.53(.925)
$t(2)$	A	.400(.936)	.570(.934)	1.00(.941)	4.02(.928)
	B	.409(.936)	.586(.947)	1.03(.942)	4.06(.943)
Pareto(2)	A	.158(.947)	.225(.947)	.394(.936)	1.60(.928)
	B	.167(.945)	.239(.951)	.425(.958)	1.73(.936)
$20\%N(0, 10^2)$	A	.392(.943)	.562(.934)	.985(.948)	3.91(.934)
	B	.396(.942)	.568(.953)	1.00(.944)	4.05(.937)
$50\%N(0, 10^2)$	A	.535(.940)	.766(.947)	1.34(.937)	5.33(.929)
	B	.550(.949)	.787(.958)	1.38(.941)	5.72(.938)
$80\%N(0, 10^2)$	A	1.14(.930)	1.60(.932)	2.83(.925)	11.1(.917)
	B	1.23(.949)	1.79(.949)	3.10(.937)	12.9(.928)
$20\%t(2, 8)$	A	.395(.944)	.565(.939)	.988(.944)	3.93(.931)
	B	.397(.939)	.570(.938)	1.00(.939)	4.05(.934)
$50\%t(2, 8)$	A	.547(.947)	.779(.945)	1.38(.940)	5.47(.926)
	B	.557(.961)	.806(.952)	1.41(.949)	5.75(.933)
$80\%t(2, 8)$	A	1.15(.937)	1.61(.947)	2.83(.950)	11.7(.926)
	B	1.22(.961)	1.74(.955)	3.05(.950)	12.6(.947)

4.4.2 Simulation Comparison for AR(1)

In this simulation study, we generate data by model (4.3), with innovations from various distributions which are the same to the last simulation study. The true mean $\mu = 100$ and we do different levels of autoregression, $\phi_1 = 0, 0.3, 0.6$, or 0.9. For each of 3000 samples, we estimate μ and ϕ_1 using the AAME method.

The square root of mean square error,

$$SMSE(\mu) = \sqrt{\frac{1}{3000}\sum_{i=1}^{3000}(\hat{\mu}_i - \mu)^2},$$

are reported in Table 4.2 with sample size $T = 200$, and in Table 4.3 with sample size $T = 500$. Also,

$$SMSE(\phi_1) = \sqrt{\frac{1}{3000}\sum_{i=1}^{3000}(\hat{\phi}_{1(i)} - \phi_1)^2},$$

are reported in Table 4.4. Our method is compared to improper estimations (only estimate a partial model without taking the dependence into account): least-squares (LS*), Huber's M-estimator (H*), least-absolute-deviance (LAD*). It is also compared to Cochrane-Orcutt type of joint estimation: least-squares (LS), Huber's M-estimator (H), least-absolute-deviance (LAD) and whittle's (W). Note that here H is computed by the algorithm based on [Coursey and Nyquist, 1986], with the ρ function being [Huber, 1964]'s loss function. `R` function `rlm` is used for H, and function `artfima` is used for W.

The AAME estimator performs comparably to the best estimator (LAD) for the fat-tailed Pareto distribution, and outperforms all others for contaminated noise distributions. For other distributions, the performance is similar to that of the LAD. On the other hand, the performance of an over-fitted model (AAME AR(p) using $p = 2$ or $p = 3$) is also reported, as a reference. We can see that over-fitting (in order of p) does not worsen the performance of $\hat{\mu}$ a lot.

Table 4.2: Results of AR(1) model. MSE of $\hat{\mu}$, with $T = 200$ and $reps = 3000$.

Noise	ϕ_1	Partial			Joint					Overfit	
Density		LS*	H*	LAD*	LS	H	LAD	W	AAME	$p=2$	$p=3$
$N(0,1)$	0	.071	.072	.088	.071	.073	.088	.071	.089	.090	.092
	.3	.101	.102	.115	.101	.103	.124	.101	.126	.128	.130
	.6	.177	.179	.191	.179	.185	.226	.177	.228	.231	.233
	.9	.688	.694	.719	.712	.736	.897	.688	.918	.926	.928
$t(2)$	0	.256	**.098**	.099	.255	.099	.100	.256	.104	.105	.107
	.3	.435	.161	.155	.437	**.143**	.145	.435	.147	.149	.152
	.6	.614	.318	.304	.621	**.247**	.253	.614	.257	.260	.264
	.9	2.21	1.57	1.48	2.31	**1.02**	1.05	2.21	1.08	1.11	1.12
Pareto(2)	0	.220	.056	**.038**	.220	.057	.039	.220	.040	.041	.042
	.3	.358	.105	.084	.360	.080	**.057**	.358	.058	.058	.060
	.6	.602	.234	.202	.928	.144	**.102**	.602	**.102**	.105	.107
	.9	2.78	1.34	1.18	2.85	.585	**.417**	2.78	.424	.431	.447
$20\%N(0,10^2)$	0	.331	.104	.108	.332	.105	.108	.331	**.100**	.102	.105
	.3	.461	.221	.190	.463	.148	.153	.461	**.144**	.147	.149
	.6	.796	.554	.438	.803	.263	.272	.796	**.255**	.260	.265
	.9	3.07	2.92	2.78	3.18	1.10	1.12	3.07	**1.05**	1.07	1.10
$50\%N(0,10^2)$	0	.510	.257	.163	.511	.262	.164	.510	**.138**	.141	.144
	.3	.714	.552	.384	.717	.378	.236	.714	**.193**	.197	.202
	.6	1.27	1.16	1.01	1.27	.654	.418	1.27	**.347**	.352	.361
	.9	4.89	4.85	4.90	5.01	2.75	1.75	4.89	**1.44**	1.48	1.51
$80\%N(0,10^2)$	0	.632	.606	.329	.633	.610	.341	.632	**.293**	.317	.340
	.3	.892	.864	.758	.896	.861	.498	.892	**.423**	.442	.476
	.6	1.57	1.55	1.58	1.59	1.51	.868	1.57	**.739**	.779	.833
	.9	6.06	6.07	6.25	6.28	6.10	3.62	6.05	**3.10**	3.30	3.52
$20\%t(2,8)$	0	1.02	.103	.107	1.03	.104	.108	.102	**.100**	.102	.105
	.3	2.33	.225	.191	2.34	.148	.153	2.33	**.145**	.147	.149
	.6	2.22	.638	.472	2.23	.267	.280	2.22	**.257**	.261	.265
	.9	7.49	4.49	3.75	7.80	1.10	1.15	7.49	**1.06**	1.07	1.08
$50\%t(2,8)$	0	1.57	.246	.158	1.58	.252	.161	1.57	**.138**	.139	.143
	.3	2.85	.619	.400	2.86	.355	.233	2.85	**.193**	.197	.203
	.6	3.34	1.48	1.16	3.37	.627	.419	3.34	**.348**	.353	.360
	.9	12.8	8.07	7.29	13.3	2.65	1.79	12.8	**1.53**	1.56	1.57
$80\%t(2,8)$	0	1.76	.608	.317	1.75	.612	.327	1.76	**.278**	.290	.300
	.3	2.63	1.07	.825	2.64	.897	.477	2.63	**.389**	.413	.430
	.6	4.46	2.18	1.97	4.52	1.55	.825	4.46	**.691**	.734	.766
	.9	16.1	10.9	10.1	16.9	6.30	3.51	16.1	**3.04**	3.16	3.38

Table 4.3: Results of AR(1) model. MSE of $\hat{\mu}$, with $T = 500$ and $reps = 3000$.

Noise	ϕ_1	Partial			Joint					Overfit	
Density		LS*	H*	LAD*	LS	H	LAD	W	AAME	$p=2$	$p=3$
$N(0,1)$	0	.044	.046	.056	.044	.046	.056	.044	.056	.056	.057
	.3	.064	.065	.074	.064	.066	.081	.064	.080	.080	.080
	.6	.113	.113	.122	.113	.116	.140	.113	.141	.142	.143
	.9	.447	.449	.462	.454	.467	.562	.447	.573	.576	.584
$t(2)$	0	.169	**.060**	.062	.169	.061	.062	.169	.065	.066	.066
	.3	.238	.102	.098	.239	**.089**	.090	.238	.092	.093	.093
	.6	.414	.199	.191	.416	**.154**	.155	.414	.159	.161	.162
	.9	1.65	1.01	.964	1.67	**.633**	.641	1.65	.663	.666	.672
Pareto(2)	0	.157	.035	**.024**	.157	.035	**.024**	.157	.025	.025	.025
	.3	.211	.065	.052	.212	.050	**.033**	.211	.035	.035	.036
	.6	.358	.143	.125	.358	.090	**.062**	.358	.063	.063	.064
	.9	1.56	.799	.733	1.58	.362	**.248**	1.56	.251	.253	.254
$20\%N(0,10^2)$	0	.209	.065	.068	.209	.065	.068	.209	**.064**	.064	.065
	.3	.287	.137	.119	.287	.093	.097	.287	**.088**	.088	.089
	.6	.515	.355	.283	.516	.167	.171	.515	**.157**	.158	.160
	.9	2.06	1.96	1.86	2.10	.684	.707	2.06	**.659**	.663	.665
$50\%N(0,10^2)$	0	.319	.156	.102	.319	.157	.101	.319	**.086**	.087	.088
	.3	.449	.339	.235	.449	.220	.144	.449	**.122**	.123	.125
	.6	.777	.715	.616	.779	.386	.259	.777	**.215**	.218	.221
	.9	3.15	3.13	3.19	3.20	1.63	1.07	3.15	**.891**	.895	.918
$80\%N(0,10^2)$	0	.395	.379	.200	.396	.379	.202	.395	**.159**	.161	.168
	.3	.574	.553	.484	.575	.550	.297	.574	**.236**	.239	.246
	.6	1.00	.991	1.02	1.00	.968	.521	1.00	**.408**	.414	.426
	.9	3.92	3.93	4.04	3.99	3.83	2.11	3.92	**1.68**	1.71	1.76
$20\%t(2,8)$	0	.801	.064	.068	.803	**.064**	.068	.801	**.064**	.064	.065
	.3	.875	.147	.126	.875	.092	.098	.875	**.089**	.090	.091
	.6	1.75	.403	.299	1.76	.166	.172	1.75	**.158**	.159	.161
	.9	5.18	2.71	2.31	5.26	.669	.704	5.18	**.661**	.664	.667
$50\%t(2,8)$	0	1.07	.150	.100	1.07	.151	.101	1.07	**.085**	.085	.086
	.3	1.43	.391	.255	1.44	.215	.148	1.43	**.122**	.124	.125
	.6	2.35	.935	.719	2.36	0.377	.255	2.35	**.213**	.214	.218
	.9	8.65	5.09	4.72	8.74	1.59	1.06	8.65	**.908**	.935	.944
$80\%t(2,8)$	0	1.09	.375	.194	1.09	.376	.198	1.09	**.161**	.165	.168
	.3	1.70	.655	.503	1.70	.541	.282	1.70	**.229**	.235	.239
	.6	2.87	1.35	1.23	2.88	.941	.498	2.87	**.409**	.416	.425
	.9	12.2	6.94	6.61	12.3	3.88	2.03	12.2	**1.69**	1.75	1.78

Table 4.4: Results of AR(1) model. MSE of $\hat{\phi_1}$, with $T = 200$ or $T = 500$ and $reps = 3000$.

Noise Density	ϕ_1	$T = 200$					$T = 500$				
		LS	H	LAD	W	AAME	LS	H	LAD	W	AAME
$N(0,1)$	0	.070	.072	.086	.071	.088	.045	.046	.055	.045	.055
	.3	.067	.068	.082	.068	.082	.043	.044	.053	.043	.053
	.6	.059	.060	.073	.061	.073	.036	.037	.045	.037	.044
	.9	.040	.041	.046	.043	.046	.022	.022	.027	.023	.027
$t(2)$	0	.064	.043	.044	.065	.048	.041	.025	.026	.041	.027
	.3	.061	.041	.041	.062	.044	.039	.023	.024	.040	.026
	.6	.056	.035	.036	.060	.037	.034	.020	.020	.037	.021
	.9	.038	.022	.022	.047	.022	.021	.011	.011	.023	.012
Pareto(2)	0	.064	.032	.027	.065	.028	.040	.018	.014	.040	.016
	.3	.060	.030	.024	.063	.026	.037	.016	.013	.039	.014
	.6	.055	.025	.019	.061	.019	.032	.014	.010	.034	.011
	.9	.036	.015	.011	.045	.011	.020	.007	.005	.023	.005
$20\%N(0,10^2)$	0	.069	.024	.025	.069	.024	.044	.015	.015	.044	.014
	.3	.067	.024	.024	.069	.022	.044	.014	.015	.045	.013
	.6	.059	.020	.020	.062	.019	.036	.012	.012	.037	.011
	.9	.042	.013	.013	.046	.012	.022	.007	.007	.023	.006
$50\%N(0,10^2)$	0	.070	.037	.024	.071	.020	.045	.022	.014	.045	.012
	.3	.067	.036	.024	.068	.019	.043	.021	.014	.044	.012
	.6	.059	.031	.019	.061	.016	.036	.018	.012	.036	.010
	.9	.042	.022	.013	.046	.010	.022	.011	.007	.023	.006
$80\%N(0,10^2)$	0	.071	.068	.041	.072	.036	.044	.042	.023	.044	.019
	.3	.069	.066	.041	.070	.035	.044	.042	.023	.044	.018
	.6	.058	.055	.033	.062	.028	.037	.035	.019	.042	.015
	.9	.040	.038	.022	.044	.017	.022	.021	.011	.024	.008
$20\%t(2,8)$	0	.063	.018	.018	.064	.017	.041	.009	.009	.041	.009
	.3	.060	.017	.017	.062	.016	.040	.009	.009	.040	.008
	.6	.054	.013	.013	.060	.013	.033	.007	.007	.036	.007
	.9	.039	.008	.008	.048	.007	.020	.004	.004	.023	.004
$50\%t(2,8)$	0	.062	.024	.018	.063	.014	.040	.013	.009	.041	.007
	.3	.059	.022	.016	.061	.013	.040	.012	.008	.040	.007
	.6	.056	.018	.012	.060	.010	.034	.009	.006	.037	.005
	.9	.038	.011	.007	.046	.006	.021	.005	.004	.023	.003
$80\%t(2,8)$	0	.065	.039	.027	.066	.026	.042	.023	.014	.042	.012
	.3	.061	.037	.025	.062	.023	.039	.021	.013	.040	.011
	.6	.056	.031	.020	.060	.017	.034	.018	.010	.038	.009
	.9	.037	.019	.011	.045	.009	.021	.010	.005	.024	.005

4.4.3 AR(p)

In this simulation study, we generate data from model (4.6) with order of autoregression $p = 4$, and various innovation distributions which are the same to the other simulation studies in this chapter. The true mean $\mu = 100$, and $\phi_1 = 0.6, \phi_2 = -0.6, \phi_3 = 0.6$, and $\phi_4 = -0.6$. Our method is compared to improper estimations: LS*, H*, and LAD*. It is also compared to Cochrane-Orcutt type of joint estimation: LAD and W. Note that there is no available algorithm implementation for H with AR(4) errors. All other settings are the same to the simulation of AR(1) cases.

The AAME estimator performs comparably to the best estimator(LAD) for the fat-tailed (Pareto), and outperforms all others for contaminated noises distributions. For other distributions, the performance is similar to that of the LAD.

4.5 Discussion

This chapter illustrates that Auto-Adaptive M-estimation (AAME) method can be applied to linear model with heavy-tailed and dependent errors. We started with one-sample model with heavy-tailed and autoregressive errors.

Two different cases, AR(1) and AR(p) errors, with heavy-tailed innovation are modeled. Confidence intervals for the case of AR(1) are established, by either asymptotic solution or bootstrap re-sampling solution.

The conducted simulation indicates that AAME contributes to improving the performance of models against fat-tailed/contaminated and dependent errors. The confidence interval is demonstrated valid as well.

On the other hand, though the simulation indicates that selecting different (wrong) orders of p for the autoregressive AR(p) model does not significantly affect the performance of the model, it might still worth to establish a robust model selection criterion for AR(p) AAME model, to select an appropriate order p.

Table 4.5: Results of AR(p) model. MSE of $\hat{\mu}$ and $\hat{\phi}s$, with $T = 200$ or $T = 500$ and $reps = 3000$.

Noise Density		$T = 200$						$T = 500$					
		LS*	H*	LAD*	W	LAD	AAME	LS*	H*	LAD*	W	LAD	AAME
$N(0,1)$	μ	**.072**	.075	.102	**.072**	.087	.093	.045	.047	.064	.045	.056	.057
	ϕ_1				**.062**	.073	.075				.037	.044	.045
	ϕ_2				**.062**	.076	.078				.037	.046	.046
	ϕ_3				**.066**	.076	.078				.039	.047	.047
	ϕ_4				**.065**	.073	.076				.038	.046	.045
$t(2)$	μ	.251	.123	.161	.251	**.104**	.110	.172	.078	.101	.172	.065	.068
	ϕ_1				.060	**.036**	.038				.035	.020	.022
	ϕ_2				.062	**.036**	.039				.035	.020	.022
	ϕ_3				.065	**.036**	.039				.037	.021	.022
	ϕ_4				.065	**.036**	.039				.037	.020	.022
Pareto(2)	μ	.236	.089	.104	.236	**.043**	**.043**	.155	.056	.065	.155	.026	.026
	ϕ_1				.058	**.020**	.022				.036	.011	.011
	ϕ_2				.059	**.020**	.021				.035	.010	.011
	ϕ_3				.065	**.021**	.022				.038	.010	.011
	ϕ_4				.066	**.020**	.021				.038	.010	.011
$20\%N(0,100)$	μ	.327	.217	.229	.327	.110	**.108**	.204	.135	.144	.204	.069	.065
	ϕ_1				.062	**.020**	**.020**				.038	.012	**.011**
	ϕ_2				.059	**.020**	**.020**				.036	**.012**	**.012**
	ϕ_3				.064	**.020**	**.020**				.038	**.012**	**.012**
	ϕ_4				.063	**.020**	**.020**				.038	**.012**	**.012**
$50\%N(0,100)$	μ	.504	.465	.535	.504	.171	**.150**	.319	.297	.342	.319	.105	.088
	ϕ_1				.062	.020	**.018**				.040	.012	.010
	ϕ_2				.063	.020	**.017**				.039	.012	.010
	ϕ_3				.064	.020	**.017**				.040	.012	.010
	ϕ_4				.065	.020	**.018**				.042	.012	.010
$80\%N(0,100)$	μ	.643	.655	.861	.643	.407	**.362**	.399	.406	.539	.399	.218	.175
	ϕ_1				.062	.038	**.035**				.037	.020	.016
	ϕ_2				.062	.039	**.035**				.037	.020	.016
	ϕ_3				.065	.039	**.036**				.038	.020	.016
	ϕ_4				.063	.037	**.034**				.039	.020	.016
$20\%t(2,8)$	μ	1.11	.246	.252	1.11	.111	**.106**	.822	.151	.158	.822	.068	.066
	ϕ_1				.063	**.013**	**.013**				.036	.007	.007
	ϕ_2				.062	**.013**	**.013**				.037	.007	.007
	ϕ_3				.067	.014	**.013**				.039	.007	.007
	ϕ_4				.067	**.013**	**.013**				.040	.007	.007
$50\%t(2,8)$	μ	1.52	.575	.628	1.52	.173	**.145**	1.08	.360	.400	1.08	.103	.087
	ϕ_1				.059	.012	**.011**				.038	.007	.005
	ϕ_2				.059	.012	**.011**				.037	.006	.006
	ϕ_3				.066	.012	**.011**				.040	.006	.006
	ϕ_4				.064	.012	**.011**				.039	.006	.005
$80\%t(2,8)$	μ	1.67	.843	1.04	1.67	.379	**.335**	1.10	.537	.654	1.10	.213	.174
	ϕ_1				.061	.021	**.020**				.035	.011	.009
	ϕ_2				.062	.021	**.019**				.035	.011	.009
	ϕ_3				.067	.021	**.020**				.038	.011	.009
	ϕ_4				.067	.022	**.020**				.037	.011	.009

Chapter 5

Conclusion and Future Works

This dissertation establishes a novel method of robust estimation (AAME), which is highly adaptive to the data, so that it is without the need of priori of the error distribution. It has improved performance against fat-tailed or contaminated errors by down-weighting influential outliers, according to a fully data-driven discrepancy function, therefore no need to be limited within a prespecified range of error distributions.

AAME is shown to be root-n consistent. Its asymptotic confidence interval and robust prediction interval are derived. A straightforward bootstrap confidence interval for a small sample size is presented as well. AAME draws information from the data, by applying a penalized constrained density distribution, which is shown to be consistent, and the optimal asymptotic convergence rate can be maintained if the penalty is bounded.

The new density estimation is penalized to have stabilized derivatives, and being tolerant across increasing numbers of knots, while maintaining its accuracy against over-fitting. It is constrained to be unimodal, symmetrical, and integrate to 1, so that it is suitable to be used as the ρ function in an M-estimation, and for further inferences.

We also present mild and explicit conditions to ensure consistency of M-estimations, in terms of density functions f and the ρ function in an M-estimator. The error density does not need to be fully determined or presumed, which conform to the spirit of robustness. This provides us a guideline for the shape constraints we have put on our new density estimation.

We further extend our new method of robust estimation to the scenarios of heavy-tailed and dependent errors. The dependency of errors is modeled by AR(1) and more general AR(p) autoregressive models.

Our AAME can be applied in economics, finance, earth sciences, and other fields that have data with influential outliers, to improve the efficiency of estimation so that it reduce the required

sample size. The algorithm implements of AAME which we proposed in this dissertation can be found in [Chen, 2020].

5.1 Future Works

5.1.1 Robust AIC or BIC for AAME Model Selection

The Akaike information criterion (AIC) is an estimator of out-of-sample prediction error and thereby can be a representative of the quality of statistical models for a given data set. Suppose that we have a statistical model of a data. Then the AIC value of the model, in [Akaike, 1974], is defined as

$$AIC = 2K - 2ln(\mathcal{L}),$$

where K is the number of estimated parameters in the model, and $\mathcal{L}$ is the maximum value of the likelihood function for the model. The lower AIC value a model presents, the higher the quality of that model has. AIC is based on Kullback–Leibler divergence. Given a collection of models for the data, AIC therefore estimates the quality of each model, relatively to each of the other models, hence providing a model selection criterion.

Another popular alternative of model selection criterion is the BIC:

$$BIC = Kln(n) - 2ln(\mathcal{L}),$$

where K is the number of estimated parameters in the model, and $\mathcal{L}$ is the maximum value of the likelihood function for the model.

Both BIC and AIC try to measure the fit of the model to the data while penalizing complexity; the penalty term is larger in BIC than in AIC. Obviously, either AIC or BIC, would need the error density function to compute the log-likelihood. We may try to derive a robust AIC or BIC, based on the estimated error density, therefore no need of the priori of the error distribution.

Bibliography

[Agostinelli and Markatou, 1998] Agostinelli, C. and Markatou, M. (1998). A one-step robust estimator for regression based on the weighted likelihood reweighting scheme. *Statistics & Probability Letters*, 37(4):341–350.

[Akaike, 1974] Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.

[Bai et al., 1992] Bai, Z., Rao, C. R., and Wu, Y. (1992). M-estimation of multivariate linear regression parameters under a convex discrepancy function. *Statistica Sinica*, pages 237–254.

[Berlinet et al., 2000] Berlinet, A., Liese, F., and Vajda, I. (2000). Necessary and sufficient conditions for consistency of m-estimates in regression models with general errors. *Journal of statistical planning and inference*, 89(1-2):243–267.

[Bondell and Stefanski, 2013] Bondell, H. D. and Stefanski, L. A. (2013). Efficient robust regression via two-stage generalized empirical likelihood. *Journal of the American Statistical Association*, 108(502):644–655.

[Celisse, 2014] Celisse, A. (2014). Optimal cross-validation in density estimation with the $\mathcal{L}_2$-loss. *The Annals of Statistics*, 42(5):1879–1910.

[Chen et al., 2016] Chen, M., Gao, C., and Ren, Z. (2016). A general decision theory for huber's ϵ-contamination model. *Electron. J. Statist.*, 10(2):3752–3774.

[Chen, 2020] Chen, X. (2020). Github webpage. http://chenxinstat.github.io.

[Chen and Meyer, 2020a] Chen, X. and Meyer, M. C. (2020a). Consistency of redescending M estimators. *submitted*.

[Chen and Meyer, 2020b] Chen, X. and Meyer, M. C. (2020b). Penalized unimodal spline density estimation with application to m-estimation. *submitted*.

[Claeskens et al., 2008] Claeskens, G., Hjort, N. L., et al. (2008). Model selection and model averaging. *Cambridge Books*.

[Cochrane and Orcutt, 1949] Cochrane, D. and Orcutt, G. H. (1949). Application of least squares regression to relationships containing auto-correlated error terms. *Journal of the American statistical association*, 44(245):32–61.

[Cooke et al., 2014] Cooke, R. M., Nieboer, D., and Misiewicz, J. (2014). *Fat-tailed distributions: Data, diagnostics and dependence*, volume 1. John Wiley & Sons.

[Coursey and Nyquist, 1986] Coursey, D. and Nyquist, H. (1986). A Procedure For Obtaining M-Estimates In Regression Models With Serially Dependent Errors. *Journal of Time Series Analysis*, 7(4):255–267.

[Davis et al., 1992] Davis, R. A., Knight, K., and Liu, J. (1992). M-estimation for autoregressions with infinite variance. *Stochastic Processes and Their Applications*, 40(1):145–180.

[Du et al., 2018] Du, S. S., Wang, Y., Balakrishnan, S., Ravikumar, P., and Singh, A. (2018). Robust nonparametric regression under huber's ϵ-contamination model.

[Efron, 1992] Efron, B. (1992). Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer.

[Fisher and Horn, 1994] Fisher, A. and Horn, P. S. (1994). Robust prediction intervals in a regression setting. *Computational statistics & data analysis*, 17(2):129–140.

[Foss et al., 2011] Foss, S., Korshunov, D., Zachary, S., et al. (2011). *An introduction to heavy-tailed and subexponential distributions*, volume 6. Springer.

[Frey, 2013] Frey, J. (2013). Data-driven nonparametric prediction intervals. *Journal of Statistical Planning and Inference*, 143(6):1039–1048.

[Gastwirth and Rubin, 1975] Gastwirth, J. L. and Rubin, H. (1975). The behavior of robust estimators on dependent data. *The Annals of Statistics*, pages 1070–1100.

[Gervini and Yohai, 2002] Gervini, D. and Yohai, V. J. (2002). A class of robust and fully efficient regression estimators. *The Annals of Statistics*, 30(2):583–616.

[Groeneboom et al., 2001] Groeneboom, P., Jongbloed, G., and Wellner, J. (2001). Estimation of a convex function: Characterizations and asymptotic theory. *Annals of Statistics*, 29(6):1653–1698.

[Haas and Pigorsch, 2009] Haas, M. and Pigorsch, C. (2009). Financial economics, fat-tailed distributions. *Encyclopedia of Complexity and Systems Science*, 4:3404–3435.

[Hall and Jones, 1990] Hall, P. and Jones, M. C. (1990). Adaptive M-estimation in nonparametric regression. *The Annals of Statistics*, 18(4):1712–1728.

[Hampel et al., 2011a] Hampel, F., Hennig, C., and Ronchetti, E. (2011a). A smoothing principle for the huber and other location m-estimators. *Computational Statistics and Data Analysis*, 55:324–337.

[Hampel et al., 2011b] Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., and Stahel, W. A. (2011b). *Robust statistics: the approach based on influence functions*, volume 196. John Wiley & Sons.

[Hill, 2013] Hill, J. B. (2013). Least tail-trimmed squares for infinite variance autoregressions. *Journal of Time Series Analysis*, 34(2):168–186.

[Holland and Welsch, 1977] Holland, P. W. and Welsch, R. E. (1977). Robust regression using iteratively reweighted least-squares. *Communications in Statistics-theory and Methods*, 6(9):813–827.

[Huber, 1964] Huber, P. J. (1964). Robust estimation of a location parameter. *Ann. Math. Statist.*, 35(1):73–101.

[Huber, 1973] Huber, P. J. (1973). Robust regression: Asymptotics, conjectures and monte carlo. *Ann. Statist.*, 1(5):799–821.

[Huber, 2004] Huber, P. J. (2004). *Robust statistics*, volume 523. John Wiley & Sons.

[Koul, 1977] Koul, H. L. (1977). Behavior of robust estimators in the regression model with dependent errors. *The Annals of Statistics*, pages 681–699.

[Lee and Martin, 1986] Lee, C.-H. and Martin, R. D. (1986). Ordinary and proper location M-estimates for autoregressive-moving average models. *Biometrika*, 73(3):679–686.

[Lei and Wasserman, 2014] Lei, J. and Wasserman, L. (2014). Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 71–96.

[Lewis et al., 2006] Lewis, R., Meyer, M., Lehman, S., Trowbridge, F., Bason, J. J., Yurman, K., and Yin, Z. (2006). Prevalence and degree of childhood and adolescent overweight in rural, urban, and suburban georgia. *Journal of School Health*, 67(4):126–132.

[Ling, 2005] Ling, S. (2005). Self-weighted least absolute deviation estimation for infinite variance autoregressive models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):381–393.

[Meyer, 2012] Meyer, M. C. (2012). Nonparametric estimation of a smooth density with shape restrictions. *Statistica Sinica*, 22(2):681–701.

[Mikosch et al., 1995] Mikosch, T., Gadrich, T., Kluppelberg, C., and Adler, R. J. (1995). Parameter estimation for arma models with infinite variance innovations. *The Annals of Statistics*, pages 305–326.

[Niemiro, 1992] Niemiro, W. (1992). Asymptotics for m-estimators defined by convex minimization. *Ann. Statist.*, 20(3):1514–1533.

[Portnoy, 1977] Portnoy, S. L. (1977). Robust estimation in dependent situations. *The Annals of Statistics*, pages 22–43.

[Portnoy, 1979] Portnoy, S. L. (1979). Further remarks on robust estimation in dependent situations. *The Annals of Statistics*, pages 224–231.

[Rousseeuw and Yohai, 1984] Rousseeuw, P. and Yohai, V. (1984). Robust regression by means of s-estimators. In *Robust and nonlinear time series analysis*, pages 256–272. Springer.

[Ruppert and Carroll, 1980] Ruppert, D. and Carroll, R. J. (1980). Trimmed least squares estimation in the linear model. *Journal of the American Statistical Association*, 75(372):828–838.

[Scott and Terrell, 1987] Scott, D. W. and Terrell, G. R. (1987). Biased and unbiased cross-validation in density estimation. *Journal of the american Statistical association*, 82(400):1131–1146.

[Silvapulle and Sen, 2005] Silvapulle, M. J. and Sen, P. (2005). *Constrained Statistical Inference*. John Wiley & Sons.

[Stuart et al., 1999] Stuart, A., Ord, J., and Arnold, S. (1999). Kendall's advanced theory of statistics, volume 2a: Classification inference and the linear model.

[Sun et al., 2019] Sun, Q., Zhou, W.-X., and Fan, J. (2019). Adaptive Huber regression. *Journal of the American Statistical Association*, pages 1–24.

[Susanti et al., 2014] Susanti, Y., Pratiwi, H., et al. (2014). M estimation, s estimation, and mm estimation in robust regression. *International Journal of Pure and Applied Mathematics*, 91(3):349–360.

[Trench, 1985] Trench, W. F. (1985). On the eigenvalue problem for toeplitz band matrices. *Linear Algebra and its Applications*, 64:199–214.

[van der Vaart, 1998] van der Vaart, A. W. (1998). *M- and Z-Estimators*, pages 41–84. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

[Wand and Jones, 1995] Wand, M. P. and Jones, M. C. (1995). *Kernel Smoothing*. Springer.

[Wang et al., 2018] Wang, L., Zheng, C., Zhou, W., and Zhou, W.-X. (2018). A new principle for tuning-free huber regression. *Preprint*.

[Welsh, 1989] Welsh, A. H. (1989). On m-processes and m-estimation. *Ann. Statist.*, 17(1):337–361.

[Wu and Cui, 2018] Wu, R. and Cui, Y. (2018). Least tail-trimmed absolute deviation estimation for autoregressions with infinite/finite variance. *Electron. J. Statist.*, 12(1):941–959.

[Wu, 2007] Wu, W. B. (2007). M -estimation of linear models with dependent errors. *Ann. Statist.*, 35(2):495–521.

[Xiao, 2004] Xiao, Z. (2004). Estimating average economic growth in time series data with persistency. *Journal of Macroeconomics*, 26(4):699–724.

[Yohai, 1974] Yohai, V. J. (1974). Robust estimation in the linear model. *Ann. Statist.*, 2(3):562–567.

[Yohai, 1987] Yohai, V. J. (1987). High breakdown-point and high efficiency robust estimates for regression. *The Annals of Statistics*, 15(2):642–656.

[Zhang, 2017] Zhang, J. (2017). Consistency of mle, lse and m-estimation under mild conditions. *Statistical Papers*, pages 1–11.

Appendix A

A.1 Proofs of Convergence Rate in Chapter 3

The convergence rate for the unpenalized decreasing quadratic spline density estimator with finite support was given by [Meyer, 2012] as $\|\hat{g} - g\| = O_p(n^{-3/7})$. Here, we determine the rate of the tuning parameter λ so that the convergence rate of the penalized constrained density estimate is maintained.

Define the feasible set as $\mathcal{B}$ for all $\boldsymbol{b} \geq \mathbf{0}$ with $\boldsymbol{h}_J^\top \boldsymbol{b} = 1$, and let $\hat{\boldsymbol{b}}_\lambda$ minimize the penalized criterion (3.7) over $\mathcal{B}$. Let $\hat{\boldsymbol{b}}$ be the minimizer of the unpenalized criterion (3.4), again over $\mathcal{B}$. Let $\tilde{\boldsymbol{b}}_\lambda$ minimize the penalized criterion (3.7), without constraints. Let $\tilde{\boldsymbol{b}}$ be the unconstrained minimizer of criterion (3.4), and let $\bar{\boldsymbol{b}}$ be the minimizer of $Q_0(\boldsymbol{b}; \boldsymbol{q})$, where $q_1 < \cdots < q_n$ are the quantiles of the density g.

Define $\bar{g} = \sum_{j=1}^{J} \bar{b}_j \delta_j$; $\|\bar{g} - g\|$ is called the "approximation error." [Meyer, 2012] showed that for quadratic splines, when the number of knots grows as $n^{1/7}$, we will have $\|\tilde{g} - \bar{g}\| = \|\bar{g} - g\| = O_p(n^{-3/7})$. Our goal is to determine the size of the tuning parameter λ which allows $\|\tilde{g}_\lambda - \bar{g}\| = O_p(n^{-3/7})$, so that by the triangle inequality, $\tilde{g}_\lambda$ attains the optimal rate. Finally, we show that the constrained estimator must attain the same rate as the unconstrained estimator.

Let $\nu_{i,X}$ denote the ith largest eigenvalue of matrix $\boldsymbol{X}$. The following lemmas are proved at the end of this appendix.

Lemma 4. *With $\boldsymbol{H}$ and $\boldsymbol{z}$ the matrix and vector defined in Section 3.2.2, we have*

$$\nu_{1,H} = O(1); \nu_{1,H^{-1}} \asymp J; \|\boldsymbol{z}\|^2 = O_p(J).$$

Lemma 5. *With $\boldsymbol{D}$ the penalty matrix defined Section 3.2.2, we have*

$$\nu_{1,D^\top D} = O(J^4).$$

The unconstrained coefficient estimates are $\tilde{\boldsymbol{b}} = \boldsymbol{H}^{-1}\boldsymbol{z}$ and $\tilde{\boldsymbol{b}}_\lambda = (\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})^{-1}\boldsymbol{z}$. By the Woodbury matrix identity, we have $(\boldsymbol{H}+\lambda \boldsymbol{D}^\top \boldsymbol{D})^{-1} = \boldsymbol{H}^{-1}+\lambda \boldsymbol{H}^{-1}\boldsymbol{D}^\top(\boldsymbol{I}+\lambda \boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top)^{-1}\boldsymbol{D}\boldsymbol{H}^{-1}$. Thus,

$$\begin{aligned}
\tilde{\boldsymbol{b}} - \tilde{\boldsymbol{b}}_\lambda &= \boldsymbol{H}^{-1}\boldsymbol{z} - (\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})^{-1}\boldsymbol{z} \\
&= \boldsymbol{H}^{-1}\boldsymbol{z} - \boldsymbol{H}^{-1}\boldsymbol{z} + \lambda \boldsymbol{H}^{-1}\boldsymbol{D}^\top(\boldsymbol{I} + \lambda \boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top)^{-1}\boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{z} \\
&= \lambda \boldsymbol{H}^{-1}\boldsymbol{D}^\top(\boldsymbol{I} + \lambda \boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top)^{-1}\boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{z}.
\end{aligned}$$

The matrix $\boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top$ is positive definite, because $\boldsymbol{H}$ is positive definite. Then by Lemmas 4 and 5, and by Gelfand corollary: $\nu_{1,AB} \leq \nu_{1,A}\nu_{1,B}$, we have

$$\begin{aligned}
\|\tilde{\boldsymbol{b}} - \tilde{\boldsymbol{b}}_\lambda\|^2 &= \lambda^2 \boldsymbol{z}^\top[\boldsymbol{H}^{-1}\boldsymbol{D}^\top(\boldsymbol{I} + \lambda \boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top)^{-1}\boldsymbol{D}\boldsymbol{H}^{-1}]^2\boldsymbol{z} \\
&\leq \lambda^2\|\boldsymbol{z}\|^2\nu^2_{1,H^{-1}D^\top(I+\lambda D^\top H^{-1}D)^{-1}DH^{-1}}, \text{ by properties of spectral radius} \\
&\leq \lambda^2\|\boldsymbol{z}\|^2\nu^4_{1,H^{-1}}\nu^2_{1,D^\top D}\nu^2_{1,(I+\lambda D^\top H^{-1}D)^{-1}} \\
&= \lambda^2\|\boldsymbol{z}\|^2\nu^4_{1,H^{-1}}\nu^2_{1,D^\top D}\nu^{-2}_{J,I+\lambda D^\top H^{-1}D} \\
&\leq \lambda^2\|\boldsymbol{z}\|^2\nu^4_{1,H^{-1}}\nu^2_{1,D^\top D}\nu^{-2}_{J,I}, \text{ since } \boldsymbol{D}\boldsymbol{H}^{-1}\boldsymbol{D}^\top \text{ is positive definite} \\
&= \lambda^2 O_p(J)O(J^4)O(J^8) \\
&= \lambda^2 O_p(J^{13}),
\end{aligned}$$

which is $\lambda^2 O_p(n^{13/7})$. Then

$$\begin{aligned}
\|\tilde{g} - \tilde{g}_\lambda\|^2 &= \int_0^S [\tilde{g}(x) - \tilde{g}_\lambda(x)]^2 dx \\
&= \int_0^S \left[\sum_{i=1}^J (\tilde{b}_i - \tilde{b}_{\lambda,i})\delta_i(x)\right]^2 dx \\
&= \int_0^S \sum_{i=1}^J \sum_{j=1}^J (\tilde{b}_i - \tilde{b}_{\lambda,i})(\tilde{b}_j - \tilde{b}_{\lambda,j})\delta_i(x)\delta_j(x) dx \\
&= \sum_{i=1}^J \sum_{j=1}^J (\tilde{b}_i - \tilde{b}_{\lambda,i})(\tilde{b}_j - \tilde{b}_{\lambda,j}) \int_0^S \delta_i(x)\delta_j(x) dx \\
&= (\tilde{\boldsymbol{b}} - \tilde{\boldsymbol{b}}_\lambda)\boldsymbol{H}(\tilde{\boldsymbol{b}} - \tilde{\boldsymbol{b}}_\lambda) \\
&\leq \|\tilde{\boldsymbol{b}} - \tilde{\boldsymbol{b}}_\lambda\|^2 \nu_{1,H} \\
&= \lambda^2 O_p(J^{13}),
\end{aligned}$$

which is $\lambda^2 O_p(n^{13/7})$. So we have

$$\begin{aligned}
\|\tilde{g}_\lambda - \bar{g}\| &\leq \|\tilde{g}_\lambda - \tilde{g}\| + \|\tilde{g} - \bar{g}\| \\
&\leq \|\tilde{g} - \bar{g}\| + \lambda O_p(n^{13/14}).
\end{aligned} \tag{A.1}$$

Next, we want to determine the rate of $\|\hat{g}_\lambda - \bar{g}\|$. We assume that g satisfies the shape assumption and $\bar{\boldsymbol{b}} \in \mathcal{B}$. By (3.5), the conditions for unpenalized constrained estimation, the necessary and sufficient conditions for $\hat{\boldsymbol{b}}_\lambda$ to minimize $Q_\lambda(\boldsymbol{b})$ over $\boldsymbol{b} \in \mathcal{B}$ are

$$[(\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})\hat{\boldsymbol{b}}_\lambda - \boldsymbol{z}]^\top (\hat{\boldsymbol{b}}_\lambda - \boldsymbol{b}) \leq \boldsymbol{0}, \text{ for all } \boldsymbol{b} \in \mathcal{B},$$

therefore $(\boldsymbol{H}\hat{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \le -\lambda\hat{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})$. The conditions for the unconstrained $\tilde{\boldsymbol{b}}_\lambda$ are simply

$$\left[(\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})\tilde{\boldsymbol{b}}_\lambda - \boldsymbol{z}\right]^\top \boldsymbol{b} = 0, \text{ for all } \boldsymbol{b} \in \mathbb{R}^J,$$

so $(\boldsymbol{H}\tilde{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top \bar{\boldsymbol{b}} = -\lambda \tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}\bar{\boldsymbol{b}}$, and $(\boldsymbol{H}\tilde{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top \hat{\boldsymbol{b}}_\lambda = -\lambda \tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}\hat{\boldsymbol{b}}_\lambda$. Therefore, $(\boldsymbol{H}\tilde{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) = -\lambda \tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})$. Then we have

$$\begin{aligned}
\|\tilde{g}_\lambda - \bar{g}\|^2 &= (\tilde{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})^\top \boldsymbol{H}(\tilde{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&= (\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda + \hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})^\top \boldsymbol{H}(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda + \hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&= (\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda)^\top \boldsymbol{H}(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda) + (\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) + 2(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda)^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&= \|\tilde{g}_\lambda - \hat{g}_\lambda\|^2 + \|\hat{g}_\lambda - \bar{g}\|^2 + 2\tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) - 2\hat{\boldsymbol{b}}_\lambda^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&\ge \|\hat{g}_\lambda - \bar{g}\|^2 + 2\tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) - 2\hat{\boldsymbol{b}}_\lambda^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&= \|\hat{g}_\lambda - \bar{g}\|^2 + 2(\boldsymbol{H}\tilde{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) - 2(\boldsymbol{H}\hat{\boldsymbol{b}}_\lambda - \boldsymbol{z})^\top \boldsymbol{H}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&\ge \|\hat{g}_\lambda - \bar{g}\|^2 + 2\lambda\hat{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) - 2\lambda\tilde{\boldsymbol{b}}_\lambda^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) \\
&= \|\hat{g}_\lambda - \bar{g}\|^2 - 2\lambda(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda)^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}).
\end{aligned}$$

The following lemma is proved at the end of this appendix.

Lemma 6. *For any $\boldsymbol{b} \in \mathcal{B}$, we have $\|\boldsymbol{b}\| = O_p(J)$.*

Then we have

$$\begin{aligned}
\lambda(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda)^\top \boldsymbol{D}^\top \boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}) &\leq \lambda \|\boldsymbol{D}(\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda)\| \|\boldsymbol{D}(\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}})\| \\
&\leq \lambda \nu_{1,D^\top D} \|\tilde{\boldsymbol{b}}_\lambda - \hat{\boldsymbol{b}}_\lambda\| \|\hat{\boldsymbol{b}}_\lambda - \bar{\boldsymbol{b}}\|, \text{ by properties of spectral radius} \\
&\leq \lambda O(J^4)(\|\tilde{\boldsymbol{b}}_\lambda\| + \|\hat{\boldsymbol{b}}_\lambda\|)(\|\hat{\boldsymbol{b}}_\lambda\| + \|\bar{\boldsymbol{b}}\|) \\
&= \lambda O(J^4)[O_p(J) + O_p(J)][O_p(J) + O_p(J)] \\
&= \lambda O_p(J^6) \\
&= \lambda O_p(n^{6/7}).
\end{aligned}$$

Therefore,

$$\|\hat{g}_\lambda - \bar{g}\|^2 \leq \|\tilde{g}_\lambda - \bar{g}\|^2 + \lambda O_p(n^{6/7}). \tag{A.2}$$

Combining (A.1), (A.2) and the approximation error, we have

$$\begin{aligned}
\|\hat{g}_\lambda - g\| &\leq \|\hat{g}_\lambda - \bar{g}\| + \|\bar{g} - g\| \\
&\leq \|\tilde{g}_\lambda - \bar{g}\| + \lambda^{1/2} O_p(n^{3/7}) + \|\bar{g} - g\| \\
&\leq \|\tilde{g} - \bar{g}\| + \|\bar{g} - g\| + \lambda^{1/2} O_p(n^{3/7}) + \lambda O_p(n^{13/14}).
\end{aligned}$$

Finally, when $\lambda = O_p(n^{-12/7})$ and under assumptions (A1)-(A4) stated in Section 3.2.2, we have $\|\hat{g}_\lambda - g\| = O_p(n^{-3/7})$. $\diamondsuit$

A.1.1 Proof of Lemma 4 in Appendix A.1

Proof: Recall that $\nu_{J,H} = \min(\boldsymbol{u}^\top \boldsymbol{H}\boldsymbol{u})$, where u is a unit vector such that $\boldsymbol{u}^\top \boldsymbol{u} = 1$. We have $\boldsymbol{u}^\top \boldsymbol{H}\boldsymbol{u} = \|g_u\|$, where $g_u = \sum_{j=1}^{J} u_j \delta_j$, and δ_j are the basis functions. Also, our knots are placed by:

$$t_j = S\frac{r^{(j-1)/(J-2)} - 1}{r^{(J-1)/(J-2)} - 1},$$

where r is the mesh ratio, and $j = 1, 2, \cdots, J$. By the definition of basis functions, we can easily check that

$$\begin{aligned}
\nu_{J,H} &= \min(\boldsymbol{u}^\top \boldsymbol{H}\boldsymbol{u}) \\
&= \int_0^S \delta_1^2(x)dx \\
&= \int_0^{t_1} \left[1 - \frac{x^2}{t_1 t_2}\right]^2 dx + \int_{t_1}^{t_2} \left[\frac{(x-t_2)^2}{t_2(t_1-t_2)}\right]^2 dx \\
&= \frac{2(1 - r^{\frac{1}{J-2}})}{5(1 - r^{\frac{J}{J-2}})} + \frac{(1 - r^{\frac{2}{J-2}})}{5(1 - r^{\frac{J}{J-2}})} + \frac{(1 - r^{\frac{1}{J-2}})^2}{15(1 - r^{\frac{J}{J-2}})(1 - r^{\frac{2}{J-2}})}.
\end{aligned}$$

The power series expansion of $\nu_{J,H}^{-1}/J$ at $J = \infty$ is

$$\frac{6(r-1)}{5log(r)} + O(J^{-1}),$$

so we have

$$\lim_{J\to\infty} \frac{\nu_{J,H}^{-1}}{J} = \frac{6(r-1)}{5log(r)}.$$

As a result, $\nu_{1,H^{-1}} = 1/\nu_{J,H} \asymp J$. Secondly, we have

$$\begin{aligned}\nu_{1,H} &= \max(\boldsymbol{u}^\top \boldsymbol{H}\boldsymbol{u}) \\ &= \int_0^S \delta_j^2(x)dx \\ &= S,\end{aligned}$$

which is $= O(1)$. Lastly, the vector $\boldsymbol{z}$ has elements $z_i = 1/n \sum_{j=1}^n \delta_i(x_j)$, which is bounded with

$$\begin{aligned}\|\boldsymbol{z}\|^2 &= \frac{1}{n^2}\sum_{i=1}^J \left[\sum_{j=1}^n \delta_i(x_j)\right]^2 \\ &\leq \frac{1}{n^2}\sum_{i=1}^J \left[\sum_{j=1}^n 1\right]^2, \text{ as each } \delta_i(x_j) \text{ is between } 0 \text{ and } 1 \\ &= J.\end{aligned}$$

Hence we have $\|\boldsymbol{z}\|^2 = O_p(J)$. ♢

A.1.2 Proof of Lemma 5 in Appendix A.1 with Geometric Knots Intervals

Proof: Follow by the definition of $J - 2 \times J$ penalty matrix $\boldsymbol{D}$ in Section 3.2.2,

$$\boldsymbol{D} = 2\begin{bmatrix} D_{11} & D_{12} & 0 & \cdots & & & 0 \\ D_{21} & D_{22} & D_{23} & 0 & \cdots & & 0 \\ 0 & D_{32} & D_{33} & D_{34} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & D_{(J-2)(J-3)} & D_{(J-2)(J-2)} & D_{(J-2)(J-1)} & 0 \end{bmatrix},$$

where

$$D_{(j)(j+1)} = \begin{cases} -\frac{1}{d_{j+1}(d_{j+1}+d_{j+2})} & \text{for } j = 1, ..., J-3 \\ -\frac{1}{d_{J-1}^2} & \text{for } j = J-2 \end{cases},$$

$$D_{jj} = \frac{1}{d_j d_{j+1}} \text{ for } j = 1, ..., J-2,$$

$$D_{(j)(j-1)} = -\frac{1}{d_j(d_{j-1}+d_j)} \text{ for } j = 2, ..., J-2.$$

Denote $p = d_{j+1}/d_j$, for $j = 1, \ldots, J-2$, such that the knots are defined as in Section 3.3.1. Therefore the differences become,

$$d_j = d_1 p^{j-1} \text{ for } j = 1, \cdots, J-1.$$

We can rewrite

$$\boldsymbol{D} = \frac{2}{d_1^2} \begin{bmatrix} D_{11} & D_{12} & 0 & \cdots & & & 0 \\ D_{21} & D_{22} & D_{23} & 0 & \cdots & & 0 \\ 0 & D_{32} & D_{33} & D_{34} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & D_{(J-2)(J-3)} & D_{(J-2)(J-2)} & D_{(J-2)(J-1)} & 0 \end{bmatrix},$$

where

$$D_{(j)(j+1)} = \begin{cases} -\frac{1}{p^{2j}(1+p)} & \text{for } j = 1, ..., J-3 \\ -\frac{1}{p^{2J-4}} & \text{for } j = J-2 \end{cases},$$

$$D_{jj} = \frac{1}{p^{2j-1}} \text{ for } j = 1, ..., J-2,$$

$$D_{(j)(j-1)} = -\frac{1}{p^{2j-3}(1+p)} \text{ for } j = 2, ..., J-2.$$

Then we have

$$\boldsymbol{D}^\top \boldsymbol{D} = \frac{4}{d_1^4} \begin{bmatrix} DD_{11} & DD_{12} & DD_{13} & & & & & 0 \\ DD_{21} & DD_{22} & DD_{23} & DD_{24} & & & & \\ DD_{31} & DD_{32} & DD_{33} & DD_{34} & DD_{35} & & & \\ & DD_{42} & DD_{43} & DD_{44} & DD_{45} & DD_{46} & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & & & & & & \cdots & 0 \end{bmatrix},$$

which is a $J \times J$ banded matrix, and where

$$\begin{aligned} DD_{(j)(j+2)} &= \frac{1}{p^{4j+1}(1+p)^2} \text{ for } i = 1, 2, \cdots \\ DD_{(j)(j+1)} &= -\frac{1}{p^{4j}} \text{ for } i = 1, 2, \cdots \\ DD_{(j)(j)} &= \frac{p^2 + 2p + 2}{p^{4j-2}(1+p)^2} \text{ for } i = 1, 2, \cdots \\ DD_{(j)(j-1)} &= -\frac{1}{p^{4j-4}} \text{ for } i = 2, 3, \cdots \\ DD_{(j)(j-2)} &= \frac{1}{p^{4j-7}(1+p)^2} \text{ for } i = 3, 4, \cdots . \end{aligned}$$

Later we will see the top left corner of $\boldsymbol{D}^\top \boldsymbol{D}$ is dominating, so the right bottom would be trivial to calculate. On the other hand, the last row and last column are all $0's$, so that $\boldsymbol{D}^\top \boldsymbol{D}$ is singular. Define a trimmed $J-1 \times J-1$ matrix $\boldsymbol{T}$ as $\boldsymbol{D}^\top \boldsymbol{D}$ with last row and last column removed, that

is to say,

$$\boldsymbol{T} = \begin{bmatrix} T_{11} & t_{12} & T_{13} & & & & & & \\ T_{21} & T_{22} & T_{23} & T_{24} & & & & & \\ T_{31} & T_{32} & T_{33} & T_{34} & T_{35} & & & & \\ & T_{42} & T_{43} & T_{44} & T_{45} & T_{46} & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & & & T_{(J)(J-2)} & T_{(J)(J-1)} & T_{JJ} \end{bmatrix},$$

where

$$\begin{aligned} T_{(j)(j+2)} &= DD_{(j)(j+2)} = \frac{1}{p^{4j+1}(1+p)^2} \text{ for } i = 1, 2, \cdots \\ T_{(j)(j+1)} &= DD_{(j)(j+1)} = -\frac{1}{p^{4j}} \text{ for } i = 1, 2, \cdots \\ T_{(j)(j)} &= DD_{(j)(j)} = \frac{p^2+2p+2}{p^{4j-2}(1+p)^2} \text{ for } i = 1, 2, \cdots \\ T_{(j)(j-1)} &= DD_{(j)(j-1)} = -\frac{1}{p^{4j-4}} \text{ for } i = 2, 3, \cdots \\ T_{(j)(j-2)} &= DD_{(j)(j-2)} = \frac{1}{p^{4j-7}(1+p)^2} \text{ for } i = 3, 4, \cdots . \end{aligned}$$

So the eigenvalues of $\boldsymbol{T}$ are those non-zero eigenvalues of $\boldsymbol{D}^\top \boldsymbol{D}$. Doing an LU decomposition to $\boldsymbol{T}$, we have:

$$
\begin{aligned}
\boldsymbol{T} &= \frac{4}{d_1^4}\boldsymbol{L}\boldsymbol{U} \\
&= \frac{4}{d_1^4}\begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & l_{(J-1)(J-3)} & l_{(J-1)(J-2)} & 1 \end{bmatrix}\begin{bmatrix} u_{11} & u_{12} & u_{13} & & \\ & u_{22} & u_{23} & \ddots & \\ & & u_{33} & \ddots & u_{(J-3)(J-1)} \\ & & & \ddots & u_{(J-2)(J-1)} \\ & & & & u_{(J-1)(J-1)} \end{bmatrix},
\end{aligned}
$$

where $\boldsymbol{L}$ and $\boldsymbol{U}$ are both $J-1 \times J-1$ triangle matrix. We can easily check that such $\boldsymbol{L}\boldsymbol{U}$ is unique, and according to the Gaussian elimination approach,

$$
\begin{aligned}
u_{11} &= \boldsymbol{T}_{11} = O(p^{-2}) \\
u_{22} &= O(\boldsymbol{T}_{12}) = O(p^{-4}) \\
u_{33} &= O(\boldsymbol{T}_{13}) = O(p^{-7}) \\
&\dots
\end{aligned}
$$

Hence, u_{11} is dominating the diagonal elements in $\boldsymbol{U}$. Also, note that the eigenvalues of $\boldsymbol{L}$ and $\boldsymbol{U}$ are their diagonal elements. Therefore, we have

$$
\begin{aligned}
\nu_{1,T} &\leq \frac{4}{d_1^4}\nu_{1,L}\nu_{1,U}, \text{ by Gelfand corollary} \\
&= \frac{4}{d_1^4}O(p^{-2}) \\
&= O(d_1^{-4}p^{-2}).
\end{aligned}
$$

Note that

$$d_1^{-4}p^{-2} = S^{-4}\Big(\frac{r^{\frac{1}{J-2}}-1}{r^{\frac{J}{J-2}}-1}\Big)^{-4} r^{-\frac{2}{J-1}},$$

and the power series expansion of $d_1^{-4}p^{-2}/J^4$ at $J=\infty$ is

$$S^{-4}\frac{log^4(r)}{(r-1)^4} + O(J^{-1}).$$

So we have

$$\lim_{J\to\infty}\frac{d_1^{-4}p^{-2}}{J^4} = S^{-4}\frac{log^4(r)}{(r-1)^4}.$$

As a result, $\nu_{1,D^\top D} = \nu_{1,T} = O(J^4)$. ♢

A.1.3 Proof of Lemma 5 in Appendix A.1 with Equal Knots Intervals

Proof. Because the second derivative of each basis function is piece-wise constant, we have

$$\theta_j = \begin{cases} -\frac{2b_1}{d_1(d_1+d_2)} & \text{for } j=1 \\ -\frac{2b_i}{d_j(d_j+d_{j+1})} + \frac{2b_{j-1}}{d_j(d_{j-1}+d_j)} & \text{for } j=2,...,J-2 \\ -\frac{2b_{J-1}}{d_{J-1}^2} + \frac{2b_{J-2}}{d_{J-1}(d_{J-2}+d_{J-1})} & \text{for } j=J-1. \end{cases},$$

where $d_j = t_{j+1} - t_j$ for $j = 1,\ldots,J-1$. Assuming equal intervals, the length between two adjacent knots should be $\frac{S}{J-1}$, where S is the largest knot, and J is the number of knots. Then we have $d_j = \frac{S}{J-1}$ for $j = 1,\ldots,J-1$. The difference in second derivative between two adjacent

intervals is

$$\theta_{j+1} - \theta_j = \begin{cases} \frac{(J-1)^2}{S^2}(2b_1 - b_2) & \text{for } j = 1, \\ \frac{(J-1)^2}{S^2}(-b_{j-1} + 2b_j - b_{j+1}) & \text{for } j = 2, ..., J-3 \\ \frac{(J-1)^2}{S^2}(-b_{J-3} + 2b_{J-2} - 2b_{J-1}) & \text{for } j = J-2. \end{cases},$$

Then we have the penalty matrix:

$$\boldsymbol{D} = \frac{(J-1)^2}{S^2}\boldsymbol{D}^*, \text{ where } \boldsymbol{D}^* = \begin{bmatrix} 2 & -1 & 0 & \cdots & & & & 0 \\ -1 & 2 & -1 & 0 & \cdots & & & 0 \\ 0 & -1 & 2 & -1 & 0 & \cdots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & \dots & & 0 & -1 & 2 & -2 & 0 \end{bmatrix}.$$

Let $\boldsymbol{T}$ be a tridiagonal Toeplitz matrix such that

$$\boldsymbol{T} = \begin{bmatrix} 2 & -1 & 0 & \cdots & & & 0 \\ -1 & 2 & -1 & 0 & \cdots & & 0 \\ 0 & -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & & 0 & -1 & 2 & -1 \end{bmatrix}.$$

[Trench, 1985] indicates that tridiagonal matrix, like $\boldsymbol{T}$, has a closed form of eigenvalues:

$$\begin{aligned}\lambda_T &= C_0 + 2\sqrt{C_{-1}C_1}cos\left(\frac{q\pi}{n+1}\right) \\ &= 2 + 2cos\left(\frac{q\pi}{n+1}\right),\end{aligned}$$

with $C_0 = 2$, $C_{-1} = C_1 = -1$. So we have the largest eigenvalue of $\boldsymbol{T}$, $\nu_{1,T} \leq 4$, and $\nu_{1,T^\top T} \leq 16$. Note that $\boldsymbol{D}^*$ is asymptotically equivalent to $\boldsymbol{T}$, so we have

$$\nu_{1,D^\top D} \leq 16\frac{(J-1)^2}{S^2}.$$

Numerical results also verified it was bounded by $16\frac{(J-1)^2}{S^2}$, as Figure A.1 shows. Then we have $\nu_{1,D^\top D} = \frac{(J-1)^2}{S^2}O(1) = O(J^4)$, since S is not related to n. ♢

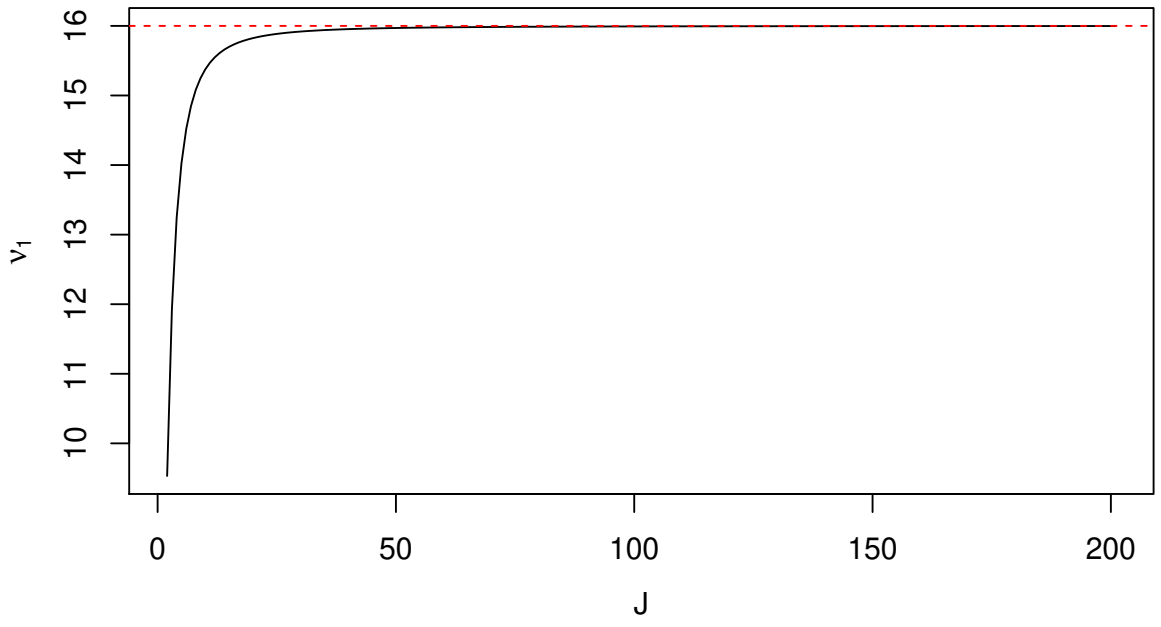

Figure A.1: The largest eigenvalue of matrix $\boldsymbol{D}^\top\boldsymbol{D}$, after scaling.

A.1.4 Proof of Lemma 6 in Appendix A.1

Proof: Note that for any $\boldsymbol{b} \in \mathcal{B}$, there is a non-negative constraint for monotone, and an area constraint of integral one. To find the maximum of $\|\boldsymbol{b}\|$ satisfying area and monotone constraints is to find

$$\begin{aligned} \underset{\boldsymbol{b}}{\text{Maximize}} \quad & \|\boldsymbol{b}\| \\ \text{subject to} \quad & \sum_{i=1}^{J} b_i \int_0^S \delta_i(x)dx = 1; \\ & b_j \geq 0,\ i = 1, \ldots, J. \end{aligned}$$

By definition of the basis functions, the maximum is at $b_1 = 1/\int_0^S \delta_1(x)dx$, and $b_2 = b_3 = \cdots = b_J = 0$. Thus,

$$\begin{aligned} \max(\|\boldsymbol{b}\|) &= \frac{1}{\int_0^S \delta_1(x)dx} \\ &= \left[t_1 + \frac{(t_1 - t_2)^2}{3t_2} - \frac{t_1^2}{3t_2} \right]^{-1} \\ &= \left[\frac{1 - r^{\frac{1}{J-2}}}{1 - r^{\frac{J}{J-2}}} + \frac{(r^{\frac{2}{J-2}} - r^{\frac{1}{J-2}})^2}{3(1 - r^{\frac{2}{J-2}})(1 - r^{\frac{J}{J-2}})} - \frac{(1 - r^{\frac{1}{J-2}})^2}{3(1 - r^{\frac{2}{J-2}})(1 - r^{\frac{J}{J-2}})} \right]^{-1}. \end{aligned}$$

The power series expansion of $\max(\|\boldsymbol{b}\|)/J$ at $J = \infty$ is

$$\frac{r-1}{\log(r)} + O(J^{-1}),$$

so we have

$$\lim_{J \to \infty} \frac{\max(\|\boldsymbol{b}\|)}{J} = \frac{r-1}{\log(r)},$$

and finally $\|\boldsymbol{b}\| = O_p(J)$. ♢

A.2 Proof of Proposition 1 in Chapter 3

Proof. We want to estimate $\zeta = E\int \hat{f}_\lambda(t)f(t)dt$. Define that

$$\hat{\zeta} = \frac{1}{n}\sum_{i=1}^{n} \hat{f}_{\lambda(-i)}(x_i),$$

matrix $\boldsymbol{Q} = \boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D}$, and vector $\boldsymbol{\delta}(x_i) = \Big[\delta_1(x_i), \cdots, \delta_J(x_i)\Big]^\top$. Note that

$$\begin{aligned}
\tilde{\boldsymbol{b}}_\lambda &= (\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})^{-1}\boldsymbol{z} \\
&= \frac{1}{n}\boldsymbol{Q}^{-1}\boldsymbol{\Delta}^\top \mathbf{1};
\end{aligned}$$

$$\begin{aligned}
\tilde{\boldsymbol{b}}_{\lambda(-i)} &= \frac{1}{n}\boldsymbol{Q}^{-1}\boldsymbol{\Delta}_{(-i)}^\top \mathbf{1} \\
&= \frac{1}{n}\boldsymbol{Q}^{-1}\Big[\boldsymbol{\Delta}^\top \mathbf{1} - \boldsymbol{\delta}(x_i)\Big] \\
&= \tilde{\boldsymbol{b}}_\lambda - \frac{1}{n}\boldsymbol{Q}^{-1}\boldsymbol{\delta}(x_i);
\end{aligned}$$

$$\begin{aligned}
\tilde{f}_{\lambda(-i)}(x_i) &= \boldsymbol{\delta}^\top(x_i)\tilde{\boldsymbol{b}}_{\lambda(-i)} \\
&= \boldsymbol{\delta}^\top(x_i)\left[\tilde{\boldsymbol{b}}_\lambda - \frac{1}{n}\boldsymbol{Q}^{-1}\boldsymbol{\delta}(x_i)\right] \\
&= \tilde{f}_\lambda(x_i) - \frac{1}{n}\boldsymbol{\delta}^\top(X_i)\boldsymbol{Q}^{-1}\boldsymbol{\delta}(x_i).
\end{aligned}$$

Then we have

$$
\begin{aligned}
E\Big[\hat{f}_{\lambda(-i)}(x_i)\Big] &= E\Big[\tilde{f}_{\lambda(-i)}(x_i)\Big] + E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i)\Big] \\
&= E\int \tilde{f}_{\lambda(-i)}(t)dt + E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i)\Big] \\
&= E\int \tilde{f}_{\lambda}(t)f(t)dt + E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i)\Big] - r \\
&= E\int \hat{f}_{\lambda}(t)f(t)dt + E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i) + \tilde{f}_{\lambda}(x_i) - \hat{f}_{\lambda}(x_i)\Big] - r \\
&= \zeta + E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i) + \tilde{f}_{\lambda}(x_i) - \hat{f}_{\lambda}(x_i)\Big] - r,
\end{aligned}
$$

where $r = 1/nE\int \boldsymbol{\delta}^\top(t)\boldsymbol{Q}^{-1}\boldsymbol{\delta}(t)f(t)dt$. By the result of Appendix A.1, we have

$$
\begin{aligned}
E\Big[\hat{f}_{\lambda(-i)}(x_i) - \tilde{f}_{\lambda(-i)}(x_i) + \tilde{f}_{\lambda}(x_i) - \hat{f}_{\lambda}(x_i)\Big] \to & E\Big[f(x_i)\Big] - E\Big[f(x_i)\Big] \\
& + E\Big[f(x_i)\Big] - E\Big[f(x_i)\Big] \\
= & 0.
\end{aligned}
$$

Also, by Lemma 4 in Appendix A.1, we have

$$
\begin{aligned}
r &= \frac{1}{n}E\int \boldsymbol{\delta}^\top(t)\boldsymbol{Q}^{-1}\boldsymbol{\delta}(t)f(t)dt \\
&= \frac{1}{n}E\boldsymbol{\delta}^\top(x_i)\boldsymbol{Q}^{-1}\boldsymbol{\delta}(x_i) \\
&\leq \frac{1}{n}\nu^2_{1,Q^{-1}}E\Big[\|\boldsymbol{\delta}(x_i)\|^2\Big] \\
&= \frac{1}{n}O(J^2)E\left[\sum_{j=1}^{J}\boldsymbol{\delta}_j^2(x_i)\right] \\
&\leq \frac{1}{n}O(J^2)J \\
&= O(n^{-4/7}) \\
&\to 0.
\end{aligned}
$$

Thus $E[\hat{\zeta}] \to \zeta$, which says $\hat{\zeta}$ is asymptotically unbiased. In addition, we have the variance

$$\begin{aligned} Var(\hat{\zeta}) &= \frac{1}{n} Var\Big[\hat{f}_{\lambda(-i)}(x_i)\Big] \\ &\leq \frac{1}{n} E\Big[\hat{f}^2_{\lambda(-i)}(x_i)\Big] \\ &= \frac{1}{n} E\Big[f^2(X_i)\Big] \\ &\to 0 \end{aligned}$$

So we have $\hat{\zeta} \xrightarrow{p} \zeta$, the consistency. $\diamondsuit$

A.3 Supplementary Materials

A.3.1 Implement of Penalized Unimodal Spline Density Estimation

Input: observations $\boldsymbol{x}$, mesh ratio r, and penalty λ (optional)

Output: estimated spline coefficients $\hat{\boldsymbol{b}}_\lambda$

Function `Estimate(`$\boldsymbol{x}$, $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$, λ`)`**:**

- $\boldsymbol{H}_\lambda = (\boldsymbol{H} + \lambda \boldsymbol{D}^\top \boldsymbol{D})$
- $\boldsymbol{h}_J$ = last column of $\boldsymbol{H}_\lambda$
- determine $\boldsymbol{b}_0$ and $\boldsymbol{W}$ `// see next page for the detail of these`
- $\boldsymbol{Q} = \boldsymbol{W}^\top \boldsymbol{H}_\lambda \boldsymbol{W}$
- $\boldsymbol{c} = \boldsymbol{W}^\top (\boldsymbol{z} - \boldsymbol{H}_\lambda \boldsymbol{b}_0)$
- $\hat{\boldsymbol{\beta}}$ = QP($\boldsymbol{Q}, \boldsymbol{c}, \boldsymbol{W}^\top, -\boldsymbol{b}_0$)
- $\hat{\boldsymbol{b}}_\lambda = \boldsymbol{W}\hat{\boldsymbol{\beta}} + \boldsymbol{b}_0$
- **return** $\hat{\boldsymbol{b}}_\lambda$

Function `CV(`$\boldsymbol{x}$, $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$, λ*'s*`)`**:**

- **for** $i \leftarrow 1$ **to** *number of* λ*'s* **do**
  - λ = ith λ's
  - **for** $j \leftarrow 1$ **to** 10 **do**
    - $\boldsymbol{x}_{(-j)}$ = $\boldsymbol{x}$ with jth fold removed
    - $\hat{\boldsymbol{b}}_{\lambda,(-j)}$ = Estimate($\boldsymbol{x}_{(-j)}$, $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$ and λ)
    - compute $\hat{g}_{\lambda,(-j)}$
  - compute ith $\hat{R}(\hat{g}_\lambda)$ for ith λ's
- λ = which minimizes $\hat{R}(\hat{g}_\lambda)$
- **return** λ

Function `Main`**:**

- determine knots by $\boldsymbol{x}$ and r
- determine $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$, λ's
- **if** λ *is not given* **then**
  - λ = CV($\boldsymbol{x}$, $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$, λ's)
- $\hat{\boldsymbol{b}}_\lambda$ = Estimate($\boldsymbol{x}$, $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{z}$, λ)
- **return** $\hat{\boldsymbol{b}}_\lambda$

Note that $\boldsymbol{b}_0 = \mathbf{1}/(\mathbf{1}^\top \boldsymbol{h_J})$ and $\boldsymbol{W}$ could be:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{h}_{\lambda,2,J} & \boldsymbol{h}_{\lambda,3,J} & \cdots & & \boldsymbol{h}_{\lambda,J,J} \\ \boldsymbol{h}_{\lambda,1,J} & 0 & \cdots & & 0 \\ 0 & \boldsymbol{h}_{\lambda,1,J} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & & 0 & \boldsymbol{h}_{\lambda,1,J} \end{bmatrix},$$

where $\boldsymbol{h}_{\lambda,i,J}$ is the ith row and Jth column element in matrix $\boldsymbol{H}_\lambda$, for $i = 1, 2, \cdots, J$. To make the algorithm more stable, we can normalize the matrix $\boldsymbol{W}$ to let each of its column as a different unit vector.