ESTIMATION OF SAMPLE SIZE AND POWER FOR QUANTILE REGRESSION

by

ZHENXIAN GONG

A project report submitted to the Department of Mathematics and Statistics in conformity with the requirements for the degree of Master of Science

Queen’s University
Kingston, Ontario, Canada

Aug 2016

Copyright © Zhenxian Gong, 2016
Abstract

Quantile regression (QR) was first introduced by Roger Koenker and Gilbert Bassett in 1978. It is robust to outliers which affect least squares estimator on a large scale in linear regression. Instead of modeling mean of the response, QR provides an alternative way to model the relationship between quantiles of the response and covariates. Therefore, QR can be widely used to solve problems in econometrics, environmental sciences and health sciences.

Sample size is an important factor in the planning stage of experimental design and observational studies. In ordinary linear regression, sample size may be determined based on either precision analysis or power analysis with closed form formulas. There are also methods that calculate sample size based on precision analysis for QR like C.Jennen-Steinmetz and S.Wellek (2005). A method to estimate sample size for QR based on power analysis was proposed by Shao and Wang (2009). In this paper, a new method is proposed to calculate sample size based on power analysis under hypothesis test of covariate effects.

Even though error distribution assumption is not necessary for QR analysis itself, researchers have to make assumptions of error distribution and covariate structure in
the planning stage of a study to obtain a reasonable estimate of sample size. In this project, both parametric and nonparametric methods are provided to estimate error distribution. Since the method proposed can be implemented in R, user is able to choose either parametric distribution or nonparametric kernel density estimation for error distribution. User also needs to specify the covariate structure and effect size to carry out sample size and power calculation.

The performance of the method proposed is further evaluated using numerical simulation. The results suggest that the sample sizes obtained from our method provide empirical powers that are closed to the nominal power level, for example, 80%.
Acknowledgements

It is a great honor to be a student of Prof. Bingshu Chen. It was an exciting moment when I received Prof. Chen’s invitation. Prof. Chen’s kind words and trust in my research potential brought me back to the academic world to pursue my dream. But everything was tough at the beginning for a person who had been away from school for four years. Fortunately, I met great people here who are always open to me and willing to help me. To name a few: Michael Williamson, Justin Slater, Emily Somerset, Yinzheng Gu, Ru Zhang and Siwei Zhang. Without your help, I would not be able to make through the graduate life. And the professors here are knowledgeable and loved by students. They taught me not only knowledge but how to face and solve problems. I also would like to give special thanks to our graduate assistant Ms. Jennifer Read who has always been nice to everyone and working hard to make our life easier.

The end of one thing is the start of another, but the lessons I learned either in class or in a talk with someone will benefit me forever. However, without limitless support from my family, I would not make this far. Thank you for making my life so wonderful.
To my family.
Contents

Abstract i

Acknowledgements iii

Contents v

List of Figures vii

List of Tables ix

Chapter 1: Introduction 1

1.1 Overview ........................................... 1
1.2 Review of Linear Regression Models ......................... 2
1.3 Quantile Regression and its Applications ..................... 4
1.4 Review of Power and Sample Size Calculation .................. 7
1.5 Project Contributions .................................. 11
1.6 Report Outline ....................................... 12

Chapter 2: Sample Size Methods for Quantile Regression 14

2.1 Motivation ........................................... 14
2.2 Asymptotic Results for Quantile Regression ..................... 17
2.3 Power and Sample Size .................................. 18
2.3.1 Sample Size for Test of Single Estimator .................. 18
2.3.2 Sample Size for Test of Multiple Estimators ................. 23
2.4 Examples of Power and Sample Size for QR .................... 25

Chapter 3: An R Package for Quantile Regression Sample Size:
rqsamplesize 29

3.1 Overview ........................................... 29
3.2 rqsamplesize .................................... 30
3.3 Webpage-based calculator-Shiny App .......................... 37
3.4 Comparison of power.rq.test and power.t.test ............... 38
Chapter 4: Numerical Simulations

4.1 Overview .......................................................... 50
4.2 Simulation Methods ................................................. 51
  4.2.1 Set Up Models .............................................. 51
  4.2.2 Generate Data Based on Location-Scale Error Distributions .... 52
  4.2.3 Gamma Distribution: ...................................... 54
  4.2.4 Kernel Density: ............................................ 56
4.3 Results .............................................................. 58

Chapter 5: Conclusions and Future Work

5.1 Summary of Conclusions ......................................... 69
5.2 Future Work ....................................................... 70

Bibliography .......................................................... 72

Appendices ............................................................ 76

Appendix A: R code for rqsamplesize package
  A.1 Kernel smoothing function: kerneldens.R .................... 77
  A.2 Library of supplementary functions: library.R ............. 78
  A.3 Main function: power.rq.test.R .......................... 84
  A.4 Code for figures in Chapter 2 ................................ 86

Appendix B: R code for Shiny app
  B.1 User interface: ui.R ........................................ 88
  B.2 Computation for output: server.R ......................... 89

Appendix C: R code for simulations
  C.1 Simulation for Normal: SimNorm.R ....................... 92
  C.2 Simulation for Cauchy: SimCauchy.R .................... 93
  C.3 Simulation for Gamma: SimGamma.R ..................... 94
  C.4 Simulation for Kernel: SimKernel.R ..................... 95
List of Figures

2.1  Power and Sample size for $\tau = 0.5$ at different values of $\beta$: The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$. .................................................. 25

2.2  Power and $\tau$ for response variable with Normal distribution and $n = 300$ at different values of $\beta$: The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$. 26

2.3  Power and $\tau$ for response variable with Cauchy distribution and $n = 300$ at different values of $\beta$: The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$. 27

3.1  Overall Graphic User Interface of Shiny App ......................... 40

3.2  Covariates Panel .......................................................... 41

3.3  Parameters’ Panel .......................................................... 42

3.4  Results Panel .............................................................. 43

3.5  Limits for uniform distribution .......................................... 44

3.6  Terms and positions for regression coefficients ....................... 45

3.7  Input of $\delta$ values .................................................... 45

3.8  Inputs for kernel smoothing .............................................. 46

3.9  Help file for $rqfun$ .................................................... 47
List of Tables

4.1 Normal Distribution Assumption . . . . . . . . . . . . . . . . . . . . 64
4.2 Cauchy Distribution Assumption . . . . . . . . . . . . . . . . . . . . 65
4.3 Gamma Distribution Assumption . . . . . . . . . . . . . . . . . . . . 66
4.4 Kernel Density . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 67
Chapter 1

Introduction

1.1 Overview

Quantile regression, as a robust alternative to traditional linear regression, models the relationship between response quantile and covariates. There are wide applications of quantile regression in econometrics. Traditional linear regression models used to dominate in the area of health research. Quantile regression is a relatively new method yet becomes more and more popular. Now it takes the same important role as the traditional linear regression and Cox proportional hazards model. Quantile regression offers possibility to study the effects of covariates at different quantiles of response, which usually are the specific the research interests of medical research.

Sample size is an important factor in the planning stage of experimental design and observational studies. In health research, a reasonable sample size brings economic benefits. Depending on the purpose of the study, there are two different criteria that can be used to determine sample size: precision analysis and power analysis. A precision analysis is based on theory of point estimate and width of the corresponding
1.2. REVIEW OF LINEAR REGRESSION MODELS

confidence interval. For a given statistics model, the larger the sample size, the more precise the estimate can be obtained. A power analysis is based on theory of hypothesis test with specific Type I and Type II error rates. A few methods have been developed to calculate sample size using precision analysis for quantile regression, but not many options are available for sample size calculation using power analysis for quantile regression. In this project, we propose an approach to calculate sample size using power analysis for quantile regression. Before we go into details about our approach, we start with a brief review of linear regression models, quantile regression model, and methods power and sample size calculations.

1.2 Review of Linear Regression Models

Traditional linear regression studies the relationship between response variable and a series of predictor variables (covariates). Normally we model the conditional mean of the response as a linear function of the predictor variables. In some cases median and mode can also be modeled to study how response variable performs for fixed values of predictors, but modeling the conditional mean of the response is widely adopted in regression-modeling approaches.

In linear regression models, the response variables are considered as independent random variables at given values of covariates, \( Y_1, Y_2, \ldots, Y_n \), where \( n \) is a positive integer. Let \( F \) be the cumulative distribution function of \( Y_i's, 1 \leq i \leq n \). Linear regression model can be expressed by the following model:

\[
Y_i = x_i' \beta + \epsilon_i, \quad \beta = (\beta_0, \beta_1, \ldots, \beta_{p-1}), \quad x_i = (1, x_{i1}, x_{i2}, \ldots, x_{ip-1}),
\]
where $\epsilon_i$'s are independently identically distributed random variables with mean 0. The least square estimators are obtained by:

$$\hat{\beta} = \arg \min_{\beta} \sum_i (Y_i - x_i'\beta)^2.$$ 

In general, modeling conditional mean of a response variable shows nice statistical properties. The models can be use to briefly describe the relationship between covariates and response. The estimators (effects or coefficients of covariates) obtained using least squares and maximum likelihood methods are easy to calculate and interpret. However, there are several limitations of linear regression models based on conditional mean. First, the estimators are a description of the relationship between the conditional mean of the response and the covariates, and they measure the magnitude of the mean of response at given values of covariates. If the researcher is interested in the non-central locations of the response distribution, the estimators are not able to provide meaningful information. Second, in many real world applications, it is possible that the model assumptions (Independent observations, linearity of conditional means, normality of response variable and homogeneity of error variance) of linear regression models are not met. In this situation, conditional mean is not appropriate any more and the results could be misleading. Third, for response variable that does not follow a normal distribution, it may not be sufficient to use conditional mean to characterize a distribution. There are other potential characteristics that can be considered as candidates to summarize response distribution, such as median, mode, scale or skewness. Due to these limitations, an alternative approach to traditional linear regression is in demand. Quantile regression is a robust approach in situations where the limitations addressed above present for least square estimators.
1.3 Quantile Regression and its Applications

Roger Koenker and Gilbert Bassett (1978) brought up the concept of quantile regression as a robust alternative to traditional linear regression models.

They argued in their paper that “when F is known to be Gaussian (normal), Rao has shown that the least squares estimator, $\hat{\beta}$, is minimum variance in the class of unbiased estimators. Unfortunately, the extreme sensitivity of the least squares estimator to modest amounts of outliers contamination makes it a very poor estimator in many non-Gaussian, especially long-tailed, situations” (Koenker and Bassett 1978). In fact, error distributions with longer tails were observed more commonly. So researchers like Gauss, Laplace and Legendre pointed out that when there are outliers, minimization of absolute deviations might be more suitable than least squares. The earliest proof was done by Laplace in 1818 that “in the simple model of bivariate regression through the origin, this Least Absolute Error (LAE) estimator had smaller asymptotic variance than the least squares estimator if the error law of the model had variance, $\sigma^2$, and density at the median, $f(0)$, satisfying $[2f(0)]^{-1} < \sigma$” (Koenker and Bassett 1978). Lots of research had been devoted to the development of robust alternatives to least squares estimator. The approach suggested by Koenker and Bassett is “a natural extension of the linear regression model” (Hao L and Naiman DQ 2007). Instead of conditional mean, conditional quantiles of response are modeled. Hence, it offers flexibility in choosing points of interest to study the relationship between response and covariates.

Unlike least squares estimator, the estimators for quantile regression can be obtained
by minimizing the sum of asymmetrically weighted absolute residuals using linear programming (Koenker and Bassett 1978). We define \( Q_{Y_i}(\tau|x_i) \) as the conditional \( \tau \)-th quantile of response variable \( Y_i \) given covariate \( x_i \). We have following models:

\[
Q_{Y_i}(\tau|x_i) = x_i'\beta + F_{\epsilon_i}^{-1}(\tau), \quad \beta = (\beta_0, \beta_1, \ldots, \beta_{p-1}), \quad i = 1, 2, \ldots, n,
\]

then

\[
Y_i = Q_{Y_i}(\tau|x_i) + \epsilon_i,
\]

where \( F_{\epsilon_i}^{-1} \) indicates the \( \tau \)-th quantile of the distribution of error term which is \( \epsilon_i \). Normally, we assume \( F_{\epsilon_i}^{-1} \) is 0 and \( \tau \)-th quantile of response variable can be expressed as a linear function of covariates. Then the quantile regression coefficients \( \hat{\beta} \) are given by,

\[
\hat{\beta} = \arg \min_{\beta} \left[ \sum_{i \in \{i: Y_i \geq x_i'\beta\}} \tau|Y_i - x_i'\beta| + \sum_{i \in \{i: Y_i \leq x_i'\beta\}} (1 - \tau)|Y_i - x_i'\beta| \right].
\]

In quantile regression, one is interested in the relationship between quantiles of response variable and the covariates. Therefore, quantile regression coefficient estimates are not affected by outliers or monotone transformations of the response variable. Quantile regression provides a tool to study how the distribution of response variable is affected by covariates and the effect of covariates at different percentile points of the response variable. As a result, more information about the data can be collected and presented. Quantile regression is a semi-parametric method, it makes no assumption about the distribution of error, which determines the distribution of the response variable in quantile regression, so it is robust to model mis-specification. Unlike linear regression models which can be solved by least squared methods, quantile
regression requires more computing power to optimize the least absolute error (LAE) objective function (Hao L and Naiman DQ 2007). This makes quantile regression computationally unpractical until late 1970s with combined computing technology and algorithmic developments.

In recent years, quantile regression is widely used in social sciences research. Buchinsky and Chamberlain (Buchinsky, 1994; Chamberlain, 1994) worked out examples of applying quantile regression to the study of wages, which makes quantile regression a standard analysis tool for wage and income analysis now. Typically in finance, the so called “VAR” (Value At Risk) is actually the quantile of the distribution of the financial return. Since market returns are found to be more skewed than normal distribution, quantile regression works better as a tool to model the distribution of the returns. Bassett and Chen (2002) gave a general discussion for this type of application. Besides the work done by Buchinsky and Chamberlain, quantile regression is also useful to study consumptive markets in economics as the effects of covariates may be different for clients from different consumption groups (for example, people with low, medium and high income). Quantile regression is also central to the hydrology statistics.

Despite its popularity in social sciences, quantile regression has a relative short history in health research. The reason may be that interpretation of quantile regression is unintuitive (Beyerlin 2014). Multiple regression and Cox proportional hazards model are used frequently in health research as investigators need to determine patient and system characteristics in relationship with medical treatment. Peter C. Austin et
al(2005) applied quantile regression in a case study examining gender differences in the timeliness of thrombolytic therapy. The reason to introduce quantile regression into the study is because multiple regression and Cox model can only explore partially how a distribution of delays in treatment or waiting times responds to patient identities. By "partially", it means that linear regression, as discussed earlier, shows only information of the mean of response, therefore, it only provides incomplete information while Cox model focuses on the relative hazard instead of the distribution of time to treatment. So the advantage of quantile regression that can study the effects of covariates at different quantile brings richer information. Koenker and Geling(2001) suggested a way to incorporate quantile regression into derivation of survival and hazard functions by regressing multiple quantiles of survival distribution on subject variables.

1.4 Review of Power and Sample Size Calculation

With the introduction of quantile regression, another option for the researchers in health sciences is now available. When it comes to study the relationship between the change of response variable distribution and the changes of the patient characteristics or system variables, quantile regression has been proven to offer more convincing results than linear regression model or Cox proportional hazards model. With all these tools at hand, it is important to have a proper research design. In clinical trials and epidemiological study, sample size is the key of the study design. The study sample size has to be defined before recruitment of study subjects starts, otherwise, the sample size may be obtained retrospectively in favor of the results. Exceptions
are allowed only under certain guidelines for some particular clinical and epidemiology studies. It is also necessary to control the Type I error or the probability to detect statistical significance, $\alpha$. A reasonable sample size also helps lower both the economic and ethical risks (Röhrig et al 2010). Therefore, the question comes to how to determine a proper sample size for a given study.

Note that the sample size is an estimate of how many subjects to be included in a study. Sample size calculation requires predetermined parameters of the underline probability distribution, such as means and standard deviations. Since data is collected after the sample size is specified, the parameters used in the calculation are basically from either previous similar studies done by the investigator or other researchers. Depending on statistical tests that are going to be used, different parameters need to be set beforehand by professionals. If a $t$-test will be used, then assumptions of means of two population and corresponding standard deviations are required. If a Fisher’s exact test is chosen, then relative proportions or rates of events need to be estimated. The more accurate the parameters are estimated, the better the results will be. The significance level ($\alpha$) has be specified as well. Sample size can be calculated based on these given parameters.

Different study designs lead to different methods of sample size calculation. As we mentioned at the beginning of this chapter, there are two criteria for sample size calculation: precision analysis and power analysis. Precision analysis for sample size calculation considers the maximum half width of the $(1-\alpha)\times100\%$ confidence interval
of the unknown parameter, which is usually referred to as the maximum error of an estimate of the unknown parameter, \( e \) (Chow et.al 2007). For example, \( Y_1, Y_2, \ldots, Y_n \) are independently identically distributed normal random variables with mean \( \mu \) and variance \( \sigma^2 \). When \( \sigma^2 \) is known, a \((1 - \alpha) \times 100\%\) confidence interval for \( \mu \) is given by:

\[
\bar{Y} \pm z_{1-\alpha/2} \frac{\sigma}{\sqrt{n}},
\]

where \( \bar{Y} \) is the average of \( Y_1, Y_2, \ldots, Y_n \) and \( z_{1-\alpha/2} \) is the upper \((\alpha/2)\)-th quantile of standard normal distribution. Then \( e \) is defined as

\[
e = |\bar{Y} - \mu| = z_{1-\alpha/2} \frac{\sigma}{\sqrt{n}}.
\]

The sample size, hence, can be obtained using

\[
n = \frac{z^2_{1-\alpha/2} \sigma^2}{e^2}.
\]

A nonparametric approach is available by using Chebyshev’s inequality:

\[
P(|\bar{Y} - \mu| \leq e) \geq 1 - \frac{\sigma^2}{ne^2},
\]

we can set

\[
1 - \frac{\sigma^2}{ne^2} = 1 - p,
\]

where \( p = P(|Z| \geq z_{1-\alpha/2}) \). So the sample size is

\[
n = \frac{\sigma^2}{pe^2}.
\]
The sample size formulas based on precision analysis are easy to apply.

Even though Type I error is considered a more important error that one should control, it is also typical to minimize Type II error, \( \beta \), in hypothesis testing while maintaining Type I error at a certain pre-specified level. The power of the test, \((1 - \beta) \times 100\%\), indicates the probability that a significant result will be detect when the alternative hypothesis is true. Sample size calculation under assumptions about power with pre-specified Type I error is called power analysis.

For a two-sample test with normal distribution when the variances are known and sample sizes are equal, the sample size can be calculated by

\[
 n = \frac{(\sigma_1^2 + \sigma_2^2)(z_{1-\beta} + z_{1-\alpha/2})^2}{\delta^2},
\]

(1.3)

where \( \delta \) is the difference between two population means that are tested. Details to derive formula (1.3) will be covered in Chapter 2. If \( \sigma_1^2 \) and \( \sigma_2^2 \) are unknown, sample variances will be used to estimate the population variances. Then the test statistic has a \( t \) distributed. In this situation, one needs an algorithm to solve an equation with \( n \) on both sides. We will have more details on this topic later.

Above we briefly reviewed the two different approaches for sample size calculation. It can be seen that one method focuses on parameter estimation and the corresponding confidence intervals and the other method focuses on hypothesis testing. Regarding which criterion to be used for sample size calculation depends on the purpose of the study. If the objective of the study is to estimate a parameter, then precision analysis
1.5. PROJECT CONTRIBUTIONS

is recommended. If the objective is to test a hypothesis, then power analysis should be considered.

Regression methods are often considered to study the relationship between response variable and covariates, testing the effects of covariates on response variable is a key step of the statistical inference. Since linear regression has long history in health research, there already exist numerous methods to calculate sample size when linear regression model is used. As quantile regression become popular for its advantages over linear regression and Cox proportional hazards model, developing sample size calculation methods for quantile regression brings invaluable benefits. In this research project, we will propose a set of methods for quantile regression based on the power analysis.

1.5 Project Contributions

The main contributions of this research project are as follows:

- Contribution 1
  A sample size calculation method based on power analysis for quantile regression is developed. A hypothesis test is set up to test the effect of one of the covariates. A formula similar to 1.3 is derived to calculate sample size. While Koenker(2005) mentioned that the effect of one of the covariates has asymptotic normal distribution, however, in his quantreg R package, the test that is used is $t$-test. So the sample size formula developed in this project will be derived under $t$-test.
• Contribution 2
There are not many formulas for sample size under the test of multiple effects. So sample size formula for univariate case is extended to multivariate case. A solution is offered for sample size calculation when the interest of the study is to test the effects of multiple covariates.

• Contribution 3
We incorporate kernel smoothing method as an estimate of residual distribution when the response variable’s probability distribution is unknown. Therefore, it is possible to calculate sample size with limited information on residual distribution even limited residual size.

• Contribution 4
We have developed an R package for sample size calculation. It also provides estimated power of a hypothesis test for a given sample size. Furthermore, we also build an online calculator for quantile regression for easy access and it is intuitive to use.

1.6 Report Outline

The remainder of this thesis is organized as follows:

Chapter 2, Methods: In this chapter, theoretical work on derivation of formulas is presented.

Chapter 3, rqsamplesize Introduction: An introduction of key functions of rqsamplesize R package and how to use them.

Chapter 4, Simulation: Simulation results to justify the performance of the tools for quantile regression power and sample size calculations.
Chapter 5, Conclusions and Future Work: Brief conclusion and discussion about this project and possible future work.
Chapter 2

Sample Size Methods for Quantile Regression

2.1 Motivation

In Chapter 1, we have discussed in details the importance of conducting a sample size calculation in the design process of a statistical study. In general, there are two different approaches for sample size calculation: precision analysis and power analysis. In bio-medical research, researchers often use “reference interval” to determine certain measurement (such as bio-chemical level in human blood) is within “normal” range for a patient. For example, when the variable has a normal distribution, the upper limit of the reference interval is often called Upper Limit of Normal (ULN). A 95% reference interval, by definition, is a range of 2.5% and 97.5% quantile of the distribution of a (real valued) measurement (C. Jennen-Steinmetz and S. Wellek 2005). Reference intervals are commonly used “as an initial signal to determine whether further patient testing is indicated” (Koduah et al 2004). Due to the popularity of reference intervals in medical practice, precision-based methods have been well developed. Here we will start a very brief review of precision analysis methods and then will go into details for power analysis methods.
C. Jennen-Steinmetz (2014) had discussed two methods for sample size determination for covariate-independent reference quantile studies. One method was proposed by Linnet (1987) by building the ratio of the width of 90% confidence interval around a reference limit over the width of the reference interval, then sample size is calculated such that the ratio is bounded by a certain limit. Linnet (1987) and Harris and Boyd (1995) gave sample size formulas for parametric method as well as how to calculate sample size when a transformed normal model is assumed. As an extension to symmetric covariate-dependent scenario, the two interval widths are determined for every covariate value.

The other methods developed by Koduah et al (2004) and C. Jennen-Steinmetz and Wellek (2005) are to measure the precision of an estimate of the \( p \)th quantile by the probability that the estimate falls between \( p_1 \)th and \( p_2 \)th quantiles, where \( p_1 < p < p_2 \). The sample size can be calculated such that the probability exceeds a given limit \( \beta \). For example, \( P\{F^{-1}(p_1) \leq \hat{F}^{-1}(p) \leq F^{-1}(p_2)\} > \beta = 0.95 \). This method can also be extended to quantiles for covariate-dependent case.

On the other hand, a limited number of methods have been developed for sample size determination using power analysis. Shao and Wang (2009) proposed an approach to calculate sample size using power analysis for the two sample problem in quantile regression. In their paper, the problem begins with setting up a hypothesis test. Suppose there are reference data generated from random variable \( X \) and monitoring data from random variable \( Y \) with density functions \( f(x) \) and \( g(y) \). Instead
of testing means of $X$ and $Y$, the quantiles of $X$ and $Y$ are tested. For example, let $\mu_\tau$ be the $\tau$-th quantile of $X$ and $\xi_\tau$ be the $\tau$-th quantile of $Y$. A one-sided test of the medians is

$$H_0 : \xi_{0.5} = \mu_{0.5},$$

$$H_a : \xi_{0.5} > \mu_{0.5}.$$  

Let $\delta = \xi_{0.5} - \mu_{0.5}$ under alternative hypothesis, the formula for sample size for monitoring data $Y$ is given by

$$n_Y = \left( \frac{z_{1-\alpha} + z_{1-\beta}}{\delta} \right)^2 \left\{ \frac{1}{4g^2(\xi_{0.5})} + \frac{1}{4Rf^2(\mu_{0.5})} \right\},$$

where $\alpha$ is the Type I error, $1 - \beta$ is the power and $R$ is the sample size ratio of the reference to monitoring. Recall that in Chapter 1 we mentioned $\delta$ is the difference between two population means that are tested. Actually, $\delta$ is defined as difference between two parameters that are tested. So it is difference between two quantiles in quantile regression. This formula is similar to (1.3) with sum of two ratios in curly bracket part instead of sum of two variances.

In this project, we will develop a set of tools to determine sample size based on hypothesis test of the regression coefficients in quantile regression.
2.2 Asymptotic Results for Quantile Regression

Recall the quantile regression model specified in Chapter 1:

\[ Q_{Y_i}(\tau|x_i) = x_i'\beta + F_{\epsilon_i}^{-1}(\tau), \beta = (\beta_0, \beta_1, \ldots, \beta_{p-1}), i = 1, 2, \ldots, n, \]

where \( Q_{Y_i}(\tau|x_i) \) is the same as defined in Chapter 1. Koenker (2005) discusses asymptotic results for quantile regression with independently identically distributed (iid) errors and non-iid errors. In fact, the error in quantile regression has distribution with 0 at specified quantile \( \tau \), i.e. \( F_{\epsilon_i}^{-1}(\tau) = 0 \). Original asymptotic results aim for the joint distribution of multiple numbers of quantiles. Here the formulas are tailored to fit situation with power and sample size for single quantile point.

- **Quantile Regression with iid Errors:**
  
  Assume iid errors \( \epsilon_i \)'s have common cumulative distribution function \( F(\cdot) \) with associated probability density function \( f(\cdot) \) with \( f(F^{-1}(\tau)) > 0 \). Suppose that \( n^{-1}X'X \equiv Q_n \) converges to a positive definite matrix \( Q_0 \), i.e \( E(X'X) \), where \( X \) is an \( n \times p \) covariate matrix \((1, x'_1, x'_2, \ldots, x'_{p-1})\). Koenker and Bassett (1978) showed that the asymptotic distribution for quantile regression estimators \( \hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_{p-1}) \) has the form

  \[
  \sqrt{n}(\hat{\beta} - \beta) \sim N\left(0, \frac{\tau(1-\tau)}{f(F^{-1}(\tau))^2}E(X'X)^{-1}\right),
  \]  

- **Quantile Regression with Non-iid Errors:**

  When the errors do not have common distributions, Koenker (2005) show that the asymptotic distribution of the estimators \( \hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_{p-1}) \) has the
2.3. POWER AND SAMPLE SIZE

form

$$\sqrt{n}(\hat{\beta} - \beta) \sim N(0, \tau(1 - \tau)H^{-1} J_n H^{-1}),$$

(2.2)

where

$$J_n = n^{-1}X'X,$$

and

$$H = \lim_{n \to \infty} n^{-1}X'Xf(\tau), f(\tau) = (f_1(\tau), f_2(\tau), \ldots, f_n(\tau))',$$

$$f_1(\tau), f_2(\tau), \ldots, f_n(\tau)$$ are probability density/mass functions for errors evaluated at \(\tau\)-th quantile for subject \(i = 1, 2, \ldots, n\), respectively. (2.2) can be reduced to (2.1) when errors are iid.

2.3 Power and Sample Size

2.3.1 Sample Size for Test of Single Estimator

We start with power and sample size problem for a simple two-sample hypothesis test setup. Suppose there are two groups of observations, \(x_i, i = 1, 2, \ldots, n_1\), and \(y_j, j = 1, 2, \ldots, n_2\). Assume \(x_i\) and \(y_j\) are independent and normally distributed with means \(\mu_1\) and \(\mu_2\) and variances \(\sigma_1^2\) and \(\sigma_2^2\), respectively. To test the following hypothesis

$$H_0 : \mu_1 = \mu_2,$$

$$H_1 : \mu_1 \neq \mu_2.$$
2.3. POWER AND SAMPLE SIZE

The Type I error of the test is defined as:

\[ \alpha = P(T > c | H_0), \]

where \( T \) is a test statistic and \( c \) is the critical value of rejecting the null hypothesis \( H_0 \). The Type II error is defined as:

\[ \beta = 1 - P(T > c | H_1). \]

Therefore,

\[ power = 1 - \beta = P(T > c | H_1). \]

When \( \sigma_1^2 \) and \( \sigma_2^2 \) are known and \( n_1 = n_2 = n \), then Z-test can be applied in this case. The test statistic is

\[ Z = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{n}}}. \]

Under the null hypothesis, test statistic \( Z \) has standard normal distribution with mean 0 and variance 1. The null hypothesis is rejected when the absolute value of \( Z \) is too large,

\[ |Z| \geq z_{1-\alpha/2}. \]

Under the alternative hypothesis, without loss of generality, \( \mu_1 \) can be expressed as:

\[ \mu_1 = \mu_2 + \delta, \delta > 0. \]

In this case, the test statistic has a normal distribution \( N(\mu^*, 1) \).
with mean value $\mu^* = \frac{\delta}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{n}}} > 0$. The corresponding power is given by

$$P(|Z| > z_{1-\alpha/2}) \approx P(Z > z_{1-\alpha/2}) = P(Z^* > z_{1-\alpha/2} - \mu^*),$$

given that two-tailed probability is small. $Z^* = Z - \mu^*$ and it has standard normal distribution. To achieve desired power of $(1 - \beta)100\%$, set

$$-z_\beta = z_{\alpha/2} - \mu^*.$$

It leads to the sample size formula (1.3). Using the same logic, if $\beta_1$ is the coefficient of interest, suppose a two-sided hypothesis test of $\beta_1$,

$$H_0 : \beta_1 = 0,$$

$$H_a : \beta_1 \neq 0.$$

Here $\delta = \beta_1 + 0 = \beta_1$ under alternative hypothesis and $\mu^* = \frac{\delta}{\sqrt{\frac{\sigma_1^2}{n}}} > 0$. Formula (1.3) can be modified to adapt this test

$$n = \frac{\sigma_{\beta_1}^2 (z_{1-\beta} + z_{1-\alpha/2})^2}{\delta^2}.$$  \hspace{1cm} (2.3)

For the purpose of sample size calculation, the value of $\delta$ is usually assumed to be known. For example, in clinical studies $\delta$ is considered to be a clinically meaningful effect size. It is chosen either based on previous studies or clinical background.
Therefore, the only unknown variable in this equation is $\sigma_{\hat{\beta}_1}^2$. (2.1) and (2.2) give the asymptotic distributions for quantile regression estimators $\hat{\beta}$ when the error distribution is iid and non-iid, respectively. In this project, it is assumed that error distribution is iid, but it can be generalized to non-iid situations. So $\sigma_{\hat{\beta}_1}^2$ can be determined from $\frac{\tau(1-\tau)}{f(F^{-1}(\tau))^2} E(X'X)^{-1}$. For example, the quantile regression model is

$$Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i.$$

We can show that when $x_i$ has a binary distribution with $P(x_i = 1) = 0.5$, then

$$E(X'X) = \begin{pmatrix} 1 & E(x_i) \\ E(x_i) & E(x_i^2) \end{pmatrix}.$$  

so

$$E(X'X) = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}.$$  

The inverse is

$$E(X'X)^{-1} = \begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix},$$

therefore, $\sigma_{\beta_1}^2 = \frac{4\tau(1-\tau)}{f(F^{-1}(\tau))^2}$.

In general, we can test $H_0 : \beta_j = 0$ for any $0 \leq j \leq p - 1$. Here we show the details of testing $H_0 : \beta_1 = 0$ vs $H_a : \beta_1 \neq 0$. One can replace $\beta_1$ with any of $\beta_j, 1 \leq j \leq p - 1$. The variance $\sigma_{\hat{\beta}_j}^2, 1 \leq j \leq p - 1$ is $(j+1)$-th item on the diagonal of $\frac{\tau(1-\tau)}{f(F^{-1}(\tau))^2} E(X'X)^{-1}$ and considered to be unknown (usually intercept $\beta_0$ is not the
2.3. POWER AND SAMPLE SIZE

coefficient of interest). The asymptotic variance

\[ s^2_{\hat{\beta}_j} = \frac{\tau(1-\tau)}{f(F^{-1}(\tau))^2} \left( \frac{1}{n_0}(X'X)_{(j+1)(j+1)} \right) \]

can be used to estimate \( \sigma^2_{\beta_j} \) where \( n_0 \) is the number of rows of \( X \). Therefore, the test statistic for \( H_0 \) to test \( \beta_1 \) is given by

\[ T = \frac{\hat{\beta}_1 - 0}{\sqrt{s^2_{\hat{\beta}_1}/n}} = \frac{\hat{\beta}_1}{\sqrt{s^2_{\hat{\beta}_1}/n}}, \]

which has a \( t \) distribution under null hypothesis. Under the alternative hypothesis, the test statistic has non-central \( t \) distribution with non-central parameter \( \mu^*_1 = \frac{\delta_1}{\sqrt{s^2_{\hat{\beta}_1}/n}} \).

To calculate sample size, we start from following approximation,

\[ P(|T| > t_{n-p,1-\alpha/2}|H_1) \approx P(T > t_{n-p,1-\alpha/2}|H_1). \tag{2.4} \]

We need \( n \) large enough to achieve desired power of \((1 - \beta) \times 100\%\). Since sample size \( n \) is involved in degrees of freedom and non-central parameter, there is no closed form to solve for \( n \). Instead, it can be solved using numerical method, which will be discussed in Chapter 3. Later in Chapter 4, it will show the difference between using equation (2.3) and this method.

In order to calculate \( \sigma^2_{\beta_j} \), assumptions for error distribution are necessary. Even though, such assumptions are not essential in quantile regression, it is necessary to calculate covariance matrix since quantity \( f(F^{-1}(\tau))^2 \) is required. In this project, 3 parametric distributions are offered: Normal, Cauchy and Gamma when there is no data. In general, normal distribution can be used when the study deals with data whose distribution is symmetric bell shape. For data with heavy tails, Cauchy
distribution is recommended. One can use Gamma distribution for skewed data. But parameters of these distributions need to be defined first. On the other hand, nonparametric kernel estimation is used when a vector of residuals is available from previous study.

2.3.2 Sample Size for Test of Multiple Estimators

If we would like to estimate the sample size of multiple estimators, formula (2.3) no longer applies. Suppose we would like to test a vector of \( m \) estimators and \( m \leq p \), which is a subset of \( \hat{\beta} \), call this vector \( \hat{\beta}_m \). Let \( V \) denote the covariance matrix of \( \hat{\beta}_m \). From asymptotic distribution (2.1), it is a direct result that \( V \) is an \( m \times m \) submatrix of \( \frac{\tau(1-\tau)}{f(F^{-1}(\tau))^2} E(X'X)^{-1} \) with entries \( Cov(\hat{\beta}_i, \hat{\beta}_j), \hat{\beta}_i, \hat{\beta}_j \in \hat{\beta}_m \) and \( \sqrt{n}(\hat{\beta}_m - \beta_m) \sim N(0, V) \). The hypothesis test is

\[
H_0 : \beta_m = 0, \\
H_a : \text{Not every } \beta_j \in \beta_m = 0, 1 \leq j \leq m.
\]

Under \( H_0 \), test statistic is

\[
T_m = n(\hat{\beta}_m' V^{-1} \hat{\beta}_m) \sim \chi^2_m.
\]
2.3. POWER AND SAMPLE SIZE

Under $H_a$, define $\delta_m = 0 + \beta_m$. Then to find power of the test, we need to find the distribution of $T_m$ under $H_a$. Under $H_a$, $\hat{\beta}_m$ has an asymptotic normal distribution

$$\sqrt{n}(\hat{\beta}_m - \delta_m) \sim N(0, V),$$

$$\sqrt{n}\hat{\beta}_m \sim N(\sqrt{n}\delta_m, V),$$

$$\sqrt{n}V^{-\frac{1}{2}}\hat{\beta}_m \sim N(\sqrt{n}V^{-\frac{1}{2}}\delta_m, I_{m \times m}),$$

$$n\hat{\beta}_m'V^{-1}\hat{\beta}_m \sim \chi^2_{m, \lambda},$$

where $\chi^2_{m,\lambda}$ denotes a non-central $\chi^2$ distribution with $m$ degrees of freedom and non-central parameter $\lambda = n\delta_m'V^{-1}\delta_m$. So,

$$T_m = n\hat{\beta}_m'V^{-1}\hat{\beta}_m \sim \chi^2_{m,\lambda}.$$ 

The sample size can be solved by setting $(1 - \alpha) \times 100\%$-th quantile of $\chi^2_m$ equal to $(1 - \beta) \times 100\%$-th quantile of $\chi^2_{m,\lambda}$ to achieve the desired power $(1 - \beta)$ at significance level of $\alpha$. To solve for sample size, we need to use numerical method as sample size $n$ is also in the position of $\lambda$.

If $t$ distribution is assumed instead of using asymptotic normal distribution for $\hat{\beta}_m$, then we can use Hotelling’s $t^2$ statistic. Under $H_0$, $T_m \sim \frac{p(n-1)}{n-p}F_{m,n-p}$. Using the same logic as above for normal case, under $H_a$, $T_m \sim \frac{p(n-1)}{n-p}F_{m,n-p}(\lambda)$, where $F_{m,n-p}(\lambda)$ denotes a non-central $F$ distribution with the first degrees of freedom $m$ and the second degrees of freedom $n - p$ and non-central parameter $\lambda = n\delta_m'V^{-1}\delta_m$. 
2.4 Examples of Power and Sample Size for QR

Figure 2.1: Power and Sample size for $\tau = 0.5$ at different values of $\beta$: The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$.

In this section, we demonstrate several examples on how to use (2.4) to calculate power for quantile regression. **The model considered is**

$$Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i.$$  

**In this case, $\delta = \beta_1$ and it is chosen from 0.3, 0.5 and 0.7.** All examples use a given sample size $n$ and other parameters needed for (2.4), therefore, calculation of sample size is not required.

Figure 2.1 shows the relationship between power and sample size using formula (2.4) at $\tau = 0.5$, where the sample size $n$ takes values between 10 and 1000, the covariate
2.4. EXAMPLES OF POWER AND SAMPLE SIZE FOR QR

Figure 2.2: Power and $\tau$ for response variable with Normal distribution and $n = 300$ at different values of $\beta$: The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$

$x$ has a binary distribution with $\mu = 0.5$ and the error has a normal distribution with mean $= 0$ and variance $= 1$. The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power for $\delta_2 = 0.5$ and the solid line is for $\delta_3 = 0.7$, respectively. The figure shows how the power of test changes with changes of sample size for any given value of $\delta_j$, $j = 1, 2, 3$.

Figure 2.2 shows the relationship between power and quantile points $0.01 \leq \tau \leq 0.99$, where the sample size $n$ takes a fixed value at $n = 300$, the covariate $x$ has a binary distribution with $\mu = 0.5$ and the error has a normal distribution with mean $= 0$ and variance $= 1$. The dotted line is the power for $\delta_1 = 0.3$, the dashed line is for power
2.4. EXAMPLES OF POWER AND SAMPLE SIZE FOR QR

Figure 2.3: Power and \( \tau \) for response variable with Cauchy distribution and \( n = 300 \) at different values of \( \beta \): The dotted line is the power for \( \delta_1 = 0.3 \), the dashed line is for power for \( \delta_2 = 0.5 \) and the solid line is for \( \delta_3 = 0.7 \) for \( \delta_2 = 0.5 \) and the solid line is for \( \delta_3 = 0.7 \), respectively. The figure shows how the power of test changes as quantile point \( \tau \) changes for any given value of \( \delta_j, j = 1, 2, 3 \).

Figure 2.3 shows the relationship between power and quantile points \( 0.01 \leq \tau \leq 0.99 \). All settings are the same as Figure 2.2, except that the error distribution is Cauchy instead of Normal. Due to the heavy tail of Cauchy distribution (i.e., the value of probability density function \( f(x) \) is larger than normal pdf at the same \( x \)), the power for Cauchy error is considerable smaller than Normal error.
We will further explore the relationship of power and sample size after the development of necessary tools in Chapter 3.
Chapter 3

An R Package for Quantile Regression Sample

Size: rqsamplesize

3.1 Overview

As discussed in Chapter 1, quantile regression is more computationally complex than linear regression. Fortunately, some of mainstream statistical analysis softwares, like R and SAS, support quantile regression analysis. Package quantreg (current version 5.26) is an R package for quantile regression written by Koenker et.al. Besides regular R help file, Dr.Koenker also wrote a vignette with more details about the package. In SAS, there is a built-in procedure QUANTREG to compute estimates and related quantiles for quantile regression. Since R is an open source software, its packages can be extended to other softwares like SPSS and Matlab.

Regarding the tools for power and sample size calculation, there are quite a few packages for this purpose based on specific test available in R.

For example, function power.t.test is a build-in function once R is installed. It
computes power of one- or two-sample $t$ test as well as paired $t$ test. It also computes sample size with defined power and returns power with sample size input. Our package $\text{rqsamplesize}$ is modeled from $\text{power.t.test}$ to compute sample size for quantile regression. Our output has the same format as $\text{power.t.test}$ returns. Function $\text{power.anova.test}$ is another build-in R function to compute power of test or determine sample size for target power. Groups, between group variance and within group variance are required. Like $\text{power.t.test}$, one of sample size and power needs to be given. Function $\text{power.prop.test}$ has 5 major parameters: sample size ($n$), probability in one group ($p_1$), probability in other group ($p_2$), significance level ($\text{sig.level} = 0.05$) and power ($\text{power}$). If one of these parameters other than ($\text{sig.level}$) is of interest, at least three of the rest parameters are required ($\text{sig.level}$ can take default 0.05); if $\text{sig.level}$ is the target, then other 4 parameters are all required to get an estimate of significance level. Package $\text{pwr}$ is an R package developed by Stephane Champely. It provides basic functions for power analysis. Besides the 3 build-in functions that have been discussed above, $\text{pwr}$ also has functions for $\text{chi} - \text{squared}$ test, correlation test, linear regression etc.

In this chapter, we will cover the key functions in this package. This package only applies to quantile regression models with location-scale transformation situation.

### 3.2 $\text{rqsamplesize}$

**Main function: $\text{power.rq.test}$**

Similar to default function $\text{power.t.test}$ in R, $\text{power.rq.test}$ is designed to take 11 arguments and returns an object of class $\text{power.htest}$. 
Example 1:
1 delta = 0.5
2 \[x = \text{rqfun}(\mu = 0.5, \text{sd} = 0.5, \text{dist} = \text{"bin"}, \ \text{pos} = 2, \ \text{term} = \text{c}(\text{"1"})\]
3, \ a = \text{NA}, b = \text{NA}, \ \text{method} = \text{"exact"})
4 pw = \text{power.rq.test}(x = x, n = \text{NULL}, \ \text{sig.level} = 0.05, \ \text{power} = 0.8, \ \text{tau} = 0.5, \ \text{delta} = \delta, \ \text{sd} = 3, \ \text{dist} = \text{"Norm"}, \ \text{kernel}\n\ .smooth = \text{"norm"}, \ \text{bw} = \text{NULL}, \ \text{alternative} = \text{"two.sided"})
5 \text{print}(pw)

The output looks like this:

Sample size for quantile regression

\[n = 1778\]
\[delta = 0.5\]
\[pos = 2\]
\[sd = 3\]
\[\text{sig.level} = 0.05\]
\[\text{power} = 0.8\]
\[\text{alternative} = \text{"two.sided"}\]

NOTE: Sample size based on exact method

Even though some arguments can be ignored, the example demonstrates a complete picture of the usage of the function and what to expect in the output.

Example 2: A simpler version:
1 delta = 0.5
2 \[x = \text{rqfun}(\mu = 0.5, \text{sd} = 0.5, \text{dist} = \text{"bin"}, \ \text{pos} = 2, \ \text{term} = \text{c}(\text{"1"}), \ \text{method} = \text{"exact"})\]
3 pw = \text{power.rq.test}(x = x, \ \text{power} = 0.8, \ \text{delta} = \delta, \ \text{sd} = 3, \ \text{alternative} = \text{"two.sided"})
4 \text{print}(pw)

The output is:
Sample size for quantile regression

\[
\begin{align*}
n &= 1778 \\
delta &= 0.5 \\
pos &= 2 \\
\text{sd} &= 3 \\
sig.\text{level} &= 0.05 \\
\text{power} &= 0.8 \\
\text{alternative} &= \text{two.sided}
\end{align*}
\]

NOTE: Sample size based on exact method

To use \textit{power.rq.test}, a \textit{rqfun} object needs to be constructed first as shown in the example. The purpose of constructing a \textit{rqfun} object is to provide a summary of estimates of data when true data is not available to compute asymptotic variance for quantile regression estimators using (2.1). Normally, covariate mean and variance are calculated with given data set. But in the planning stage of a study, sample size needs to be estimated before collecting data. In this case, covariate mean and variance from previous similar studies can be used so that complete historical data is not required.

In \textit{rqfun}, \textit{mu} is the mean of the covariate. \textit{sd} is the standard deviation of the covariate. \textit{dist} is the distribution of the covariate with three options available to choose: Normal (“norm”), Uniform (“unif”) and Binary (“bin”). \textit{pos} points to the estimator
of interest in the regression model. For example, the model is

\[ Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i, \]

and we would like to test the significance of \( \beta_1 \), then \( pos \) is set to 2, which means the second coefficient in the model. \( pos = 1 \) is the intercept which is not so important.

\( term \) indicate number and type of covariates that are used in the model. For example, \( term = c('1', '2', 'sqrt') \) implies that the model is

\[ Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 \sqrt{x_i}. \]

Integers (-3,-2,-1,1,2,3 are currently supported) to be put as \( term \) argument have to be in \textit{char} type, i.e \( ('-3', '-2', '-1', '1', '2', '3') \); \( sqrt, log \) and \( exp \) are also available when there are \( \sqrt{x}, log(x) \) and/or \( e^x \). \( a \) and \( b \) are optional and needed only when uniform distribution is selected.

Last argument in \( rqfun \) is \textit{method}. This argument offers to options: \textit{exact} and \textit{sim}. They are methods used to compute \( E(X'X) \): exact method (\textit{method}='\textit{exact}') and simulation method (\textit{method}='\textit{sim}').

Exact method:
This method applies to covariate with normal and uniform distributions and the model consists of polynomials of univariate with highest power 3 as following:

\[ Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3, \]
\[ E(X'X) = \begin{pmatrix} 1 & \mu & \mu_2 & \mu_3 \\ \mu & \mu_2 & \mu_3 & \mu_4 \\ \mu_2 & \mu_3 & \mu_4 & \mu_5 \\ \mu_3 & \mu_4 & \mu_5 & \mu_6 \end{pmatrix}, \]

where \( \mu_i, 1 \leq i \leq 6 \) is the \( i \)-th raw moment. The raw moments (up to 6th) of normal and uniform distribution are easy to compute since closed form formulas are available. So, exact method is recommended when there are \( x, x^2 \) or \( x^3 \) and the combination of these terms in the model and the covariate \( x \) has a normal or a uniform distribution.

Simulation method:

\( \text{sim} \) is short for “simulation”. This method is used to estimate \( E(X'X) \) when there are terms with odd power, term with logarithm transformation, or exponential term in the model and it is hard to obtain \( E(X'X) \) as nice as above.

The process is:

\textbf{Step 1}: Generate 100000 random numbers from specified distribution (\( \text{dist} \) argument) as well as a vector with the same number of 1, which is to be used as intercept.

\textbf{Step 2}: Apply proper transformation to random numbers based on \( \text{term} \). For example, if \( \text{dist} = \text{“norm”} \) with mean \( \text{mu} = 1 \) and \( \text{sd} = 2 \) and \( \text{term} = c(‘1’, ‘-1’, ‘sqrt’, ‘exp’) \), then a vector of random numbers from normal distribution with mean 1 and standard deviation 2, \( x_i \), will be generated. Transformation as specified in \( \text{term} \) will apply to \( x_i \) accordingly to form a matrix \( X = (1, x_i, \frac{1}{x_i}, \sqrt{x_i}, e^{x_i}) \).

\textbf{Step 3}: Based on Law of Large Numbers, \( E(X'X) \) can be approximated by \( \frac{1}{100000}X'X \).

With \( E(X'X) \) obtained from either of the two method, we are able to compute \( E(X'X)^{-1} \) and get the covariate matrix for estimators, which is \( \frac{\tau(1-\tau)}{\mu(1-\mu)^2} E(X'X)^{-1} \).
with given quantile and error distribution.

After the construction of a covariate object using \texttt{rqfun(\ldots)}; the rest arguments in \texttt{power rq.test} should be defined based on study specification. If sample size estimation is required in the study, then \texttt{power} should be defined instead of taking NULL by default, but \texttt{n} can be left empty. If sample size is given and power estimation is of interest, then \texttt{n} value is required. Parameter \texttt{sig.level} is the significance level (Type I error) of the hypothesis test, which takes value of 0.05 by default. \texttt{tau} is the quantile of interest and default value is 0.5. It means that median regression is used by default. Parameter \texttt{delta} is(are) quantile regression coefficient(s). It can be a number or a vector if multiple regression coefficients are to be tested. Input for this argument may come from other similar studies since data is not available yet. Parameter \texttt{sd} here means the standard deviation of error distribution and \texttt{dist} is error distribution. Don’t confuse these two with those used in constructing a covariate object. Parametric and non-parametric methods are provided to estimate error distribution.

- Parametric methods: There are three parametric options for error distribution: Normal (\texttt{dist =‘Norm’}), Cauchy (\texttt{dist =‘Cauchy’}) and Gamma (\texttt{dist =‘Gamma’}). Note that when normal distribution is used for error distribution, \texttt{dist =‘Norm’} in \texttt{power rq.test} while \texttt{dist=‘norm’} in \texttt{rqfun}. This is to differentiate error distribution from covariate distribution. For Normal and Cauchy distributions, location parameters take \(\mu\) and \(\mu_0\), i.e. \(N(\mu, \sigma^2)\) and \(Cauchy(\mu_0, \gamma)\), such that their corresponding \(\tau\)-quantiles are 0. Scale parameters, \(\sigma\) and \(\gamma\) will take value of \texttt{sd}. When Gamma distribution is used, it will be parametrized by a reference gamma distribution. See Chapter 4 Gamma simulation part for details.
3.2. **RQSAMPLESIZE**

- Non-parametric method: If error distribution can not be characterized by the parametric distributions that are offered in `dist`, but part or complete residuals are available, then kernel smoothing method to estimate error distribution is used. In this case, put residuals (a numeric vector) in the place of `dist` i.e `dist = c(...)`. Then choose kernels and bandwidth by defining `kernel.smooth` and `bw` arguments. There are 5 different kernels that can be used: 1,2,3,4 indicating the number of kernels used. `kernel.smooth = 1` is for a uniform variable bounded by -1/2 and 1/2. `kernel.smooth = 2` is for triangle density bounded by -1 and 1. `kernel.smooth = 3` is for three-piece density bounded by -3/2 and 3/2. `kernel.smooth = 4` is for four-piece density bounded by -2 and 2. Standard normal kernel is used by default if it is left as “norm”. Parameter `bw` is the place for bandwidth input. Small bandwidth results in under-smoothing whereas large bandwidth causes over-smoothing. Default is NULL and bandwidth is estimated by normal distribution approximation or Silverman’s (1986) rule of thumb by default.

One can specify `alternative = one.sided` or `alternative = two.sided` for one-sided test or two-sided test.

In Section 2.3, we mentioned that there is no closed form formula for sample size when `t` test is used since the test statistic under the null hypothesis turns out to be a non-central `t` distribution. Since sample size `n` is involved in degrees of freedom and non-central parameter, we use `uniroot` function in R to solve for sample size `n`. 
To sum up, \textit{rqfun} and \textit{power.rq.test} are the most important in this package. Before use \textit{power.rq.test} to calculate sample size or power, user has to use \textit{rqfun} to define a \textit{rqfun} object with summary information of covariate and what method to use to calculate variance of quantile regression estimators. Figure 3.9 to 3.11 show the corresponding help files for this package.

\section*{3.3 Webpage-based calculator-Shiny App}

\textit{rqsamplesize} package is easy to use on computers with R program. For users who are not familiar with the R package, there is a simpler way to use these tools. We build a webpage-based calculator available at http://statapps.tk/rq_samplesize/. This calculator is developed using \textbf{Shiny} by RStudio (http://shiny.rstudio.com/tutorial/). Basically, \textbf{Shiny} is a web application framework for R. Developers are able to transform their R program into web applications so that the applications can be accessed anywhere as long as there is Internet. Our calculator has the user interface as shown in Figure 3.1. The calculator is built with two parts: input and output. Left panel (Figure 3.2) and middle panel (Figure 3.3) are used to take user inputs and summary panel (Figure 3.4) shows the results of calculation. “Covariates” panel is meant to create covariate object using \textit{rqfun}. Figure 3.1 shows how to do \textbf{Example 1} on the calculator. Some variables are conditional. Figure 3.5 shows that when “unif” is selected for \textbf{Distribution}, two more rows show up for user to define lower and upper limits for the uniform distribution. If 2 or larger values are put in \textbf{Covariates}, there are options for user to choose covariate terms used in quantile regression and user can also specify the position(s) of regression coefficients to be tested (Figure 3.6). The options available for covariate terms are enough to approximate an equation for
3.4. COMPARISON OF *POWER.RQ.TEST* AND *POWER.T.TEST*

a smooth relationship between response and covariates (Royston and Altman 1997).

In the middle panel, user can choose if sample size or power estimation is needed by selecting “SampleSize” or “Power” from the drop-down list at the beginning of this panel. Next four variables are straightforward. When 2 or larger values are put in *Covariates* in “Covariates” panel, there are corresponding places to input *delta* (Figure 3.7). Two options are available at *Distribution or Residual*. If “Residual” is selected, then a box will display for user to update residual data and variables for kernel smoothing are visible below the box. A vector of data in .txt file is suggested currently. See Figure 3.8. Right panel of this calculator is a dynamic environment for results summary. User is able to see how results change as any of the variables in the left two panels is modified.

### 3.4 Comparison of *power.rq.test* and *power.t.test*

Below is R output from *power.t.test* using same information from **Example 1**.

```r
power.rq.test(x = x, power = 0.8, delta = 0.5, sd = 3, alternative = "two.sided")
```

The output looks like this:

Two-sample t test power calculation

- \( n = 566.0813 \)
- \( \text{delta} = 0.5 \)
- \( \text{sd} = 3 \)
- \( \text{sig.level} = 0.05 \)
- \( \text{power} = 0.8 \)
- \( \text{alternative} = \text{two.sided} \)
3.4. COMPARISON OF \texttt{POWER.RQ.TEST} AND \texttt{POWER.T.TEST}

\textbf{NOTE:} \(n\) is number in *each* group

Here \(\textit{delta}\) is the difference in mean. In this example, the test is set up to test if the mean difference of two populations, say \(\mu_2 - \mu_1\), is 0 or not and \(\textit{delta}\) is the sample mean difference, \(\beta_1\). Because the linear regression model is \(y = \beta_0 + \beta_1 x + \text{error}\) and

\[
x = \begin{cases} 
0, & \text{Group1} \\
1, & \text{Group2} 
\end{cases}
\]

is an indicator variable. So, \(\mu_1 = \beta_0\) and \(\mu_2 = \beta_0 + \beta_1\). Our function gives estimation of 1778 in total and \texttt{power.t.test} gives an estimate of almost 566 for each sample. Recall that in Chapter 1 Laplace demonstrated that LAE estimator has smaller asymptotic variance when \([2f(0)]^{-1} < \sigma\). In our example, \(\sigma = 3\) and \([2f(0)]^{-1} = 3.759942\), hence sample size for quantile regression is larger than twice the sample size obtained using \texttt{power.t.test}. The ratio \(\frac{1778}{2 \times 566} = \frac{([2f(0)]^{-1})^2}{9}\), where \(f(0)\) is density function of \(N(0,9)\) evaluated at 0. In general, let \(n_q\) denote sample size for quantile regression and \(n_t\) be the sample size estimated using \texttt{power.t.test}, \(\frac{n_q}{2 \times n_t} = \frac{\tau(1-\tau)}{\sigma^2 f(F^{-1}(\tau))^2}\).
3.4. COMPARISON OF POWER.RQ.TEST AND POWER.T.TEST

Figure 3.1: Overall Graphic User Interface of Shiny App
### Covariates

**Mean**

| 0.5 |

**Standard Deviation**

| 0.5 |

**Distribution**

- *bin* 

**Covariates**

| 1 |

**Number of covariates of interest**

| 1 |

**Position of Interest 1**

| 2 |

**Method**

- *exact* 

---

**Figure 3.2: Covariates Panel**
Figure 3.3: Parameters’ Panel
Summary

The covariate X has the following 2 components:
- Intercept
- X

where X has a bin distribution with mean = 0.5 and variance = 0.25

Exact method is used to calculate matrix Q = E(t(X)%%*(X)).

Term(s) with * will be used in sample size calculation[1] 0.5

Sample size for quantile regression

n = 1778
delta = 0.5
pos = 2
sd = 3
sig.level = 0.05
power = 0.8
alternative = two.sided

NOTE: Sample size based on exact method

Figure 3.4: Results Panel
3.4. COMPARISON OF \textit{POWER.RQ.TEST} AND \textit{POWER.T.TEST} 44

Figure 3.5: Limits for uniform distribution
3.4. COMPARISON OF POWER.RQ.TEST AND POWER.T.TEST

Figure 3.6: Terms and positions for regression coefficients

<table>
<thead>
<tr>
<th>Covariates</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
</tr>
</tbody>
</table>

Terms
- 1
- 2
- -1
- -2
- log
- exp
- sqrt

Number of covariates of interest
- 2

Position of Interest 1
- 2

Position of Interest 2
- 3

Figure 3.7: Input of \( \delta \) values

<table>
<thead>
<tr>
<th>( \delta ) 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
</tr>
</tbody>
</table>

| \( \delta \) 2  |

| \( \delta \) 3  |

3.4. COMPARISON OF POWER.RQ.TEST AND POWER.T.TEST

Figure 3.8: Inputs for kernel smoothing

- **Distribution or Residual**: Residual
- **Kernel type for kernel smoothing**: norm
- **Band width for kernel smoothing**
3.4. COMPARISON OF POWER.RQ.TEST AND POWER.T.TEST

rqfun [rqsamplsize]

R Documentation

rqfun object.

Description
Create a rqfun object for power.rq.test.

Usage

x = rqfun(mu=0, sd=1, dist='norm', term=c('1'), pos=2, method='exact', a =NA, b=NA)

Arguments

mu  Mean of independent variable under univariate case.
sd  Standard deviation of independent variable under univariate case.
dist The distribution which takes "norm" as normal and "unif" as uniform.
term It takes a vector of strings of the form of integers or "log", "sqrt" and "exp" eg. term=c("1","-1","log", "sqrt") indicates there are 4 variables: x,1x,log(x) and sqrt(x)
pos  The position of the regression coefficient(s) of interest (position of intercept is 1, but it is not important, so 1 is not recommended). Default is 2.
method "exact" or "sim" which are the methods to compute variance of regression coefficients, "exact" method is recommended when variables are chosen from x,x^2,x^3,...,and x^6
a  If dist is "unif", this is the lower limit of uniform distribution. It is optional if "mu" and "sd" are already specified.
b  If dist is "unif", this is the upper limit of uniform distribution. It is optional if "mu" and "sd" are already specified.

Examples

#Construct x
x = rqfun(mu = 5, sd = 1.5, term = c('1','log','1/sqrt'))

[Package rqsamplsize version 0.1 index]

Figure 3.9: Help file for rqfun
3.4. COMPARISON OF `POWER.RQ.TEST` AND `POWER.T.TEST` 48

```
power.rq.test (rqsample.size)  

Power and sample size for quantile regression.

Description

Compute power and sample size of test under alternative hypothesis to obtain target power (same as `power.anova.test`)

Usage

```
power.rq.test(x, n = NULL, sig.level = 0.05, power = NULL,
  tau = 0.5, beta = 1, sd = 1, dist = "Norm", kernel.smooth = "norm",
  bw = NULL, alternative = c("two.sided", "one.sided"))
```

Arguments

- `x` : A `rqfun` object. See "Details" and "Examples".
- `n` : Given sample size to compute power.
- `sig.level` : Significance level of the test (Type I error probability). The default is 0.05.
- `power` : Power of the test between 0 and 1 (One minus Type II error probability). It is required to compute sample size.
- `tau` : The desired regression quantile between 0 and 1. The default is 0.5.
- `beta` : The desired quantile regression coefficient. It can be a number or a vector.
- `sd` : Standard deviation or scale of error distribution.
- `dist` : The error distribution for the purpose of power/sample size calculation. It takes "Norm", "Cauchy", "Gamma" or a vector of residuals when distribution is unknown and to be estimated.
- `kernel.smooth` : It takes values `1, 2, 3, 4` indicating the number of kernels used. 1 is for a uniform variable bounded by -1/2 and 1/2. 2 is for triangle density bounded by -1 and 1. 3 is for three-piece density bounded by -3/2 and 3/2. 4 is for four-piece density bounded by -2 and 2. Standard normal kernel is used by default if no values are specified.
- `bw` : The bandwidth used in kernel smoothing. Small bandwidth results in under-smoothing whereas large bandwidth causes over-smoothing. Default is NULL and bandwidth is estimated by normal distribution approximation or Silverman’s (1986) rule of thumb by default.
- `alternative` : Choose either "one.sided" for one sided test or "two.sided" for two sided test.

Figure 3.10: Help file for `power.rq.test` Part 1
Details

A rqfun object needs to be defined in univariate regression situation. Here "univariate" means the independent variables are transformation of a single variable. It is defined as following: \( x = \text{rqfun}(\text{mu}=0, \text{sd}=1, \text{dist} = \text{norm}, \text{term} = c(1'), \text{pos} = 2, \text{method} = \text{exact}, a = NA, b=NA) \). It contains the information of the independent variable.

It is assumed that error distribution is independently identically distributed, so the calculation of variance of regression coefficients are based on Section 3.2.2 of Quantile Regression (2005).

Value

Object of class "power.rqtest", a list of the arguments (including the computed one) augmented with method and note elements.

Note

Either \( n \) or power needs to be defined. They can not be NULL at the same time. If estimated sample size is desired, power should be given; if estimated power is desired, actual sample size \( n \) should be given.

unifroot is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Zhenxian Gong

References


See Also

rq, rqfun, power.anova.test, unifroot

Examples

```r
# Construct x
x = rqfun(mu = 5, sd = 1.5, dist = "norm", pos = 2, term = c(1'), a=NA, b=NA, method = 'ex')
# Compute estimated sample size when power is 0.8
power.rq.test(x=x, power = 0.8, tau = 0.2, beta = c(1,0.5),
              sd = 10, dist = "Norm", alternative = "two.sided")
```

[Package rqsamplesize version 0.1]

Figure 3.11: Help file for power.rq.test Part II
Chapter 4

Numerical Simulations

4.1 Overview

In this chapter, we conduct numerical simulation to evaluate the performance of the \texttt{power.rq.test} function in the \texttt{rqsamplesize} package. The objective is to show that sample size estimated by this function produces desired statistical power for the corresponding hypothesis testing. In terms of distribution of response variable, we will consider all four different methods used in the \texttt{rqsamplesize} package: Normal distribution, Cauchy distribution, Gamma distribution and Kernel estimation. Usually, when the exact distribution for the response variable is unknown, a Kernel estimation for error density function will be used.

Koenker (2015) states that Silverman’s (1986) kernel smoothing method is used in his R package for quantile regression: \texttt{quantreg}. \textbf{Hence error distribution can be estimated instead of making assumptions for error distribution.} We will investigate the performance of Silverman’s kernel and kernel methods (such as Gaussian kernel) proposed in our sample size package with numerical simulations when
the distribution of response variable is assumed to be unknown. We also study the performance of sample size estimation at different quantile point \( \tau \) and different regression coefficient \( \beta \). Finally, when exact parametric distribution is used for error distribution in formula (2.3), our initial simulation results show sample sizes calculated with asymptotic normal distribution have some deviations from target power in several scenarios. This is the reason that power formula based on \( t \) test (equation 2.4) is used to develop sample size for quantile regression.

In this Chapter, we will provide detailed methods used in numerical simulations and discuss the simulation results.

4.2 Simulation Methods

4.2.1 Set Up Models

For any given covariate \( x_i \) and regression coefficient \( \beta \), we need to generate a random variable \( Y_i \) such that the \( \tau \)-th quantile of \( Y_i \) is \( x_i' \beta \) as we discussed in Chapter 2, error has a distribution with \( \tau \)-th quantile equal to 0. First, by using code for \( rqfun \) object in the \( rqsamplesize \) package, the covariate matrix \( X \) can be generated for covariate that follows a binary, normal, and uniform distribution. Next, as \( \beta \) is already given, a vector of conditional regression quantile, \( Q_Y(\tau|\mathbf{x}) \), can be obtained by \( X\beta \).

Recall that in quantile regression model, the \( \tau \)-th conditional quantile of response
variable $Y_i$ given covariate $x_i$, $Q_{Y_i}(\tau|x_i)$, can be obtained by

$$Q_{Y_i}(\tau|x_i) = x_i'\beta + Q_{\epsilon_i}(\tau),$$

$$Q_{Y_i}(\tau|x_i) = x_i'\beta, i = 1, 2, \ldots, n,$$

where $\beta = (\beta_0, \beta_1, \ldots, \beta_{p-1})$ and $Q_{\epsilon_i}(\tau) = F_{\epsilon_i}^{-1}(\tau) = 0$. When we deal with the simplest case with only one single covariate, the model becomes

$$Q_{Y_i}(\tau|x_i) = \beta_0 + \beta_1 x_i.$$

Next, we will generate response variable $Y_i$ using this conditional quantile depending on different distribution assumption of the error term $\epsilon_i = Y_i - x_i'\beta$. Note that error is a random variable whose $\tau$-th quantile is 0.

### 4.2.2 Generate Data Based on Location-Scale Error Distributions

**Normal Distribution**

Assuming that the error term $\epsilon_i \sim N(\mu, \sigma^2)$, where $\sigma$ is the standard deviation that can be specified by parameter $sd$ in function `power.rq.test`. And its $\tau$-th quantile is 0. Suppose that random variable $Y_i$ has a normal distribution with mean parameter $\mu_i$ and variance $\sigma^2$. In order to generate the response variable $Y_i$ for each individual subject $i$, $\mu_i$ needs to be solved. Steps to solve for $\mu_i$ are as following:
\[ P(Y_i \leq Q_{Y_i}(\tau|x_i)) = \tau, \]
\[ P(Y_i - \mu_i \leq Q_{Y_i}(\tau|x_i) - \mu_i) = \tau, \]
\[ P(Y_i^* \leq Q_{Y_i}(\tau|x_i) - \mu_i) = \tau, \]

Set
\[ Q_{Y_i}(\tau|x_i) - \mu_i = Q_{Y_i^*}(\tau), \]
then
\[ \mu_i = Q_{Y_i}(\tau|x_i) - Q_{Y_i^*}(\tau) \]
\[ = \beta_0 + \beta_1 x_i - Q_{Y_i^*}(\tau), i = 1, 2, \ldots, n. \]

Therefore, for any given \( x_i \) and \( \beta \), response variable \( Y_i \) can be obtained from a normal distribution \( N(\beta_0 + \beta_1 x_i - Q_{Y_i^*}(\tau), \sigma^2) \) and \( Y_i^* \sim N(0, \sigma^2) \).

**Cauchy Distribution:**

Like normal distribution, Cauchy distribution is also a location-scale distribution with location parameter \( \mu_0 \) and scale parameter \( \gamma \) (Cauchy(\( \mu_0, \gamma \))). Since the second moment of Cauchy distribution is infinity, the variance of Cauchy distribution is not defined. Therefore, \( sd \) parameter in `powr.rq.test` is used as scale parameter for Cauchy distribution, and the error distribution is Cauchy(\( \mu_0, \gamma = sd \)) under Cauchy distribution assumption. Suppose that \( Y_i \) has Cauchy distribution with unknown location parameter \( \mu_i \) and known scale parameter \( \gamma \) in this scenario. As we did above, we need to solve for location parameter \( \mu_i \) here as well. Follow the same logic.
4.2. SIMULATION METHODS

and steps, it is easy to find

$$
\mu_i = x_i' \beta - Q_{Y_i^*}(\tau),
$$

$$
= \beta_0 + \beta_1 x_i - \gamma \tan \left[ \pi \left( \tau - \frac{1}{2} \right) \right], i = 1, 2, \ldots, n.
$$

$Y_i^* \sim Cauch(0, \gamma)$. Then the random variable $Y_i$ can be generated in a way similar to Normal distribution.

4.2.3 Gamma Distribution:

Both the Normal and Cauchy distributions are for symmetric distributed response variable. However, in real life application, many data are positive and skewed. Therefore, we will also evaluate how well the power and sample size formula work for skewed data using a Gamma distribution.

In this project report, when a $\text{Gamma}(\alpha, \theta)$ is mentioned, it means a Gamma distribution with shape $\alpha$ and scale $\theta$ by default. If the error variable follows a Gamma distribution, then $Y_i \sim \text{Gamma}(k, \theta_i)$. In the numerical simulations, we let $k = \frac{1}{sd^2}$ and use a reference Gamma random variable $U_i \sim \text{Gamma}(k, \frac{1}{k})$ to resemble scale parameter $\theta_i$ as a function of $U_i$. We start with following equation:

$$
P(Y_i \leq Q_{Y_i}(\tau|x_i)) = P(U_i \leq Q_{U_i}(\tau)) = \tau.
$$

Divide both sides of by $Q_{Y_i}(\tau|x_i)$ inside the first probability gives:

$$
P \left( \frac{Y_i}{Q_{Y_i}(\tau|x_i)} \leq 1 \right) = P(U_i \leq Q_{U_i}(\tau)).
$$
4.2. SIMULATION METHODS

Then, multiply both sides by $Q_{U_i}(\tau)$ inside the first probability:

$$P\left(\frac{Y_i Q_{U_i}(\tau)}{Q_Y(\tau|x_i)} \leq Q_{U_i}(\tau)\right) = P(U_i \leq Q_{U_i}(\tau)).$$

Therefore,

$$U_i = \frac{Y_i Q_{U_i}(\tau)}{Q_Y(\tau|x_i)}.$$ 

Or:

$$Y_i = \frac{U_i Q_Y(\tau|x_i)}{Q_{U_i}(\tau)}.$$ 

Let $C = \frac{Q_{Y_i}(\tau|x_i)}{Q_{U_i}(\tau)} = \frac{x_i' \beta}{Q_{U_i}(\tau)}$, it is not hard to show that $Y_i \sim Gamma(k, \frac{C}{k})$, hence $\theta_i = \frac{C}{k}$. But $C$ is not guaranteed positive due to the value of $-\infty < x_i' \beta < +\infty$, so we use $e^{x_i' \beta}$ instead of $x_i' \beta$. In this way, the model becomes:

$$Q_{ln(Y_i)}(\tau|x) = x_i' \beta + F^{-1}_{e_i'}(\tau), \beta = (\beta_0, \beta_1, \ldots, \beta_{p-1}), i = 1, 2, \ldots, n,$$

which is obtained by taking natural logarithm on both sides of $Y_i = \frac{U_i Q_Y(\tau|x)}{Q_{U_i}(\tau)}$ and $e_i' = ln(U_i) - ln(Q_{U_i}(\tau))$ is a new random error term. The transformation does not affect the distribution of the response.

In the simulation, $Q_{U_i}(\tau)$ is generated first and then $Y_i$ is obtained from $Gamma(k, \frac{e^{\theta_0 + \sum x_i}}{Q_{U_i}(\tau)})$. Finally $ln(Y_i)$ is used in quantile regression.

Note that the density function used in variance calculation for quantile regression estimators under Gamma distribution assumption is not density function of $U_i$, it is actually density function of $e_i'$ and it can be derived using cumulative distribution
4.2. SIMULATION METHODS

function method. At $\tau$-th quantile, the quantity $f(F^{-1}(\tau)) = Q_{U_i}(\tau)f_{U_i}(Q_{U_i}(\tau))$
where $f_{U_i}(Q_{U_i}(\tau))$ is density of $U_i$ evaluated at $\tau$-th quantile of $U_i$.

4.2.4 Kernel Density:

When there is not enough previous information available to make a parametric assumption of the error distribution, researcher often conduct a small pilot study to inform the design of the larger formal study. In this scenario, residual data from the pilot study are available, and kernel smoothing method can be considered to calculate the sample size of the future study.

Let $K(x)$ be kernel cumulative distribution function (kernel cdf) and $k(x)$ be kernel probability distribution function (kernel pdf). Estimated cdf of the error term $F_n(x)$ and estimated pdf of the error term $f_n(x)$ can be expressed as:

$$F_n(x) = \frac{1}{nh} \sum_{i} K(\frac{x - \epsilon_i}{h}),$$
$$f_n(x) = \frac{1}{nh} \sum_{i} k(\frac{x - \epsilon_i}{h}),$$

respectively. Here $\epsilon_i$'s are given residual data points and $h$ is bandwidth.

In Chapter 3, we mentioned that there are five options for kernels available in the `rqsamplesize` package to fit the density function of the residual. Therefore, there will be five pairs of kernel cdf and kernel pdf to choose from. The default option of `kernel.smooth = "norm"` is a standard normal distribution $N(0,1)$. With normal kernel, we have $K(x) = \Phi(x)$ and $k(x) = \phi(x)$, where $\Phi(x)$ and $\phi(x)$ are cdf
and pdf for the standard normal distribution, respectively. There are also other options for kernel function that can be used in qsamplesize package: Rectangle kernels (kernel.smooth = 1), triangle kernels (kernel.smooth = 2), three-piece kernels (kernel.smooth = 3) and four-piece kernels (kernel.smooth = 4) have kernel cdf and kernel pdf of the following format:

\[ K_j(x) = \frac{1}{j!} \sum_{\nu=0}^{j} (-1)^{\nu} \binom{j}{\nu} (x - \nu + \frac{j}{2})^+, \]

\[ k_j(x) = \frac{1}{(j-1)!} \sum_{\nu=0}^{j} (-1)^{\nu} \binom{j}{\nu} (x - \nu + \frac{j}{2})^{j-1}, j = 1, 2, 3, 4. \]

Note that \((g(x))_+\) is the positive part of a given function \(g(x)\). That is,

\[ (x - \nu + \frac{j}{2})_+ = \begin{cases} x - \nu + \frac{j}{2} & \text{if } x - \nu + \frac{j}{2} > 0, \\ 0 & \text{Otherwise.} \end{cases} \]

The simulation process for kernel density is given below:

**Step 1**, we generate a small set of data \((m = 50)\) under a specific type of distribution (e.g. normal).

**Step 2**, a quantile regression model is fitted for this data set the residuals are obtained.

**Step 3**, sample size is calculated based on the choice of kernel (standard normal kernel is used in simulation).

**Step 4**, using sample size from step 3), simulation is conducted under the assumption of the same error distribution and the empirical power is obtained.
4.3 Results

Simulations are done with following set-up: $\beta_0 = 1$, $\beta_1$ is chosen from $0.5, 1, 1.5, 2, 3$ and $\tau$ has options $0.1, 0.3, 0.5, 0.6, 0.8$. $x$ is constructed with $rqfun(mu = 5, sd = 1.5, dist = \text{“norm”}, pos = 2, term = c(’1’), method =’exact’)$. Significance level is set at 0.05 and power is 0.8. For each error distribution assumption, four tables are generated: estimate sample size with asymptotic normal test statistics, empirical power based on sample size with asymptotic normal test statistics, estimate sample size with $t$-test statistics, empirical power based on sample size with $t$-test statistics. For each parametric error distribution assumption, the simulation process is, first, sample size is calculated for each pair of $(\beta_1, \tau)$. We then generate 1000 runs of simulation. At the beginning of each run, random numbers are generated using the methods discussed above as response data and normal random numbers are generated using information provided in $rqfun$ as covariate data. Quantile regression is used to model the relationship between response ($\log$ of response in Gamma case) data and covariate at specified $\tau$. Then we calculate $p-value$ for quantile regression estimator $\hat{\beta}_1$ using $t$ test at the end of each run and store the $p-value$ for each run. Empirical power is the proportion of $p-values$ that are less than defined significance level, i.e how many $\hat{\beta}_1$’s end up being rejected.

**Normal Distribution Assumption:** Error has distribution $N(\mu, 100)$. Results are shown in Table 4.1.1 to 4.1.4.

Table 4.1.1 shows the estimated sample size under each pair of $(\beta_1, \tau)$ using the formula with asymptotic normal distribution. The sample size ranges from $n = 61$
to \( n = 4078 \) for the settings that we considered, with the trend that sample size is small when \( \tau \) is in the middle and \( \beta_1 \) is large, while the sample size is large when \( \tau \) is located in the tail of the distribution and effect \( \beta_1 \) is small.

Table 4.1.2 shows empirical power using formula (2.3), that is
\[
\frac{\sigma^2}{\beta_j} \left( z_{1-\beta} + z_{1-\alpha/2} \right)^2 / \delta^2.
\]
We observed that most of the empirical powers are close to the nominal level of 80%. The empirical power ranges from 61.4\% to 84.8\%, with the average empirical power = 77.66\%. There are several cases when \( n \) is small, the empirical power is a little bit far away from 80%.

Table 4.1.3 shows the sample size with \( t \)-test statistics. The sample size ranges from \( n = 63 \) to \( n = 4080 \) and the pattern is similar to those in table 4.1.1. In general, the sample size obtained using \( t \)-test statistics is larger than the corresponding sample size using asymptotic normal statistics. The difference between these two methods becomes very small when \( n \) gets large.

Table 4.1.4 shows empirical power using sample size from 4.1.3. The empirical power ranges from 73.5\% to 90.6\%, with the average empirical power = 80.5\%.

Overall, the empirical power performs much better than the sample size calculated using asymptotic normal test statistics.

**Cauchy Distribution:** Error has distribution \( Cauchy(\mu_0,5) \). Results are shown in Table 4.2.1 to 4.2.4.
Table 4.2.1 shows the estimated sample size under each pair of $(\beta_1, \tau)$ using the formula with asymptotic normal distribution. The sample size spreads from $n = 24$ to $n = 33982$ for the settings that we considered, with the trend that sample size is small when $\tau$ is in the middle and $\beta_1$ is large, while the sample size is the largest when $\tau$ is located in the tail of the distribution and effect $\beta_1$ is small.

Table 4.2.2 shows empirical power using formula (2.3). We observed that most of the empirical powers are close to the nominal level of 80%. The empirical power ranges from 51.8% to 83.6%, with the average empirical power = 75.7%. Like what we observed in Table 4.1.2, there are several cases when $n$ is small, the empirical power is a little bit far away from 80%.

Table 4.2.3 shows the sample size with $t$-test statistics. The sample size ranges from $n = 26$ to $n = 33984$ and the pattern is similar to those in table 4.2.1. In general, the sample size obtained using $t$-test statistics is larger than the corresponding sample size using asymptotic normal statistics. The difference between these two method becomes very small when $n$ gets large.

Table 4.2.4 shows empirical power using sample size from 4.1.3. The empirical power ranges from 57.2% to 85.8%, with the average empirical power = 77.8%.

Overall, the empirical power performs much better than the sample size calculated using asymptotic normal test statistics.
4.3. RESULTS

**Gamma Distribution:** Reference random variable $U_i$ has distribution $\text{Gamma}(\frac{1}{9}, 9)$. Results are shown in Table 4.3.1 to 4.3.4.

Table 4.3.1 shows the estimated sample size under each pair of $(\beta_1, \tau)$ using the formula with asymptotic normal distribution. The sample size spreads from $n = 10$ to $n = 10173$ for the settings that we considered. The trend of sample size is not the same as it is in previous two cases. Sample size is the largest when the pair $(\beta_1, \tau)$ is the smallest and the largest pair $(\beta_1, \tau)$ leads to the smallest sample size.

Table 4.3.2 shows empirical power using formula (2.3). Most of the empirical powers are close to the nominal level of 80%. The empirical power are in the range from 69.7% to 87.0%, with the average empirical power $= 77.7%$. Unlike what we observed in Table 4.1.2 and Table 4.2.2, there are not cases when $n$ is small, the empirical power is far away from 80%.

Table 4.3.3 shows the sample size with $t$-test statistics. The sample size ranges from $n = 12$ to $n = 10175$ and the pattern is similar to Table 4.3.1. In general, the sample size obtained using $t$-test statistics is larger than the corresponding sample size using asymptotic normal statistics. The difference between these two method becomes very small when $n$ gets large. It can be seen that the sample size calculated using $t$ test is larger than sample size obtained using asymptotic normal distribution by 2, which means extra degrees of freedom for $\hat{\beta}_0$ and $\hat{\beta}_1$ in majority of the cases.
4.3. RESULTS

Table 4.3.4 shows empirical power using sample size from 4.1.3. The empirical power ranges from 58.5\% to 91.0\%, with the average empirical power = 77.6\%.

Overall, the empirical power performs equally well as the sample size calculated using asymptotic normal test statistics on average.

**Kernel Density:** Process for simulation using kernel smoothing method has been discussed above.

Table 4.4.1 shows the estimated sample size under each pair of ($\beta_1, \tau$) using the formula with asymptotic normal distribution. The sample size ranges from $n = 73$ to $n = 4633$. Sample size estimates are the largest at $\tau = 0.1$. At $\tau = 0.3$, sample size estimates are the smallest. When $\beta_1$ gets larger, the sample size becomes smaller.

Table 4.4.2 shows empirical power using formula (2.3). We observed that most of the empirical powers are larger than 85\%. The empirical power ranges from 76.9\% to 98.5\%, with the average empirical power = 87.2\%.

Table 4.4.3 shows the sample size with $t$-test statistics. The sample size ranges from $n = 75$ to $n = 4635$ and the pattern is similar to those in Table 4.4.1 but with one exception that sample size estimates at $\tau = 0.6$ are the smallest. In general, the sample size obtained using $t$-test statistics is larger than the corresponding sample size using asymptotic normal statistics.
Table 4.4.4 shows empirical power using sample size from 4.1.3. The empirical power ranges from 55.4\% to 96.4\%, with the average empirical power = 85.0\%.

Overall, empirical power is larger than target power on average for both methods. So either larger sample size than estimate may be considered or use proper kernels and/or bandwidth in practice.
4.3. RESULTS

Table 4.1: Normal Distribution Assumption

<table>
<thead>
<tr>
<th>β₁</th>
<th>τ=0.1</th>
<th>τ=0.3</th>
<th>τ=0.5</th>
<th>τ=0.6</th>
<th>τ=0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>4078</td>
<td>2424</td>
<td>2192</td>
<td>2244</td>
<td>2849</td>
</tr>
<tr>
<td>1</td>
<td>1020</td>
<td>606</td>
<td>548</td>
<td>561</td>
<td>713</td>
</tr>
<tr>
<td>1.5</td>
<td>454</td>
<td>270</td>
<td>244</td>
<td>250</td>
<td>317</td>
</tr>
<tr>
<td>2</td>
<td>255</td>
<td>152</td>
<td>137</td>
<td>141</td>
<td>179</td>
</tr>
<tr>
<td>3</td>
<td>114</td>
<td>68</td>
<td>61</td>
<td>63</td>
<td>80</td>
</tr>
</tbody>
</table>

Table 4.1.2: Empirical Power Using Formula (2.3)

<table>
<thead>
<tr>
<th>β₁</th>
<th>τ=0.1</th>
<th>τ=0.3</th>
<th>τ=0.5</th>
<th>τ=0.6</th>
<th>τ=0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.801</td>
<td>0.803</td>
<td>0.789</td>
<td>0.797</td>
<td>0.78</td>
</tr>
<tr>
<td>1</td>
<td>0.802</td>
<td>0.838</td>
<td>0.758</td>
<td>0.779</td>
<td>0.788</td>
</tr>
<tr>
<td>1.5</td>
<td>0.749</td>
<td>0.767</td>
<td>0.801</td>
<td>0.832</td>
<td>0.815</td>
</tr>
<tr>
<td>2</td>
<td>0.738</td>
<td>0.846</td>
<td>0.836</td>
<td>0.707</td>
<td>0.71</td>
</tr>
<tr>
<td>3</td>
<td>0.764</td>
<td>0.848</td>
<td>0.749</td>
<td>0.704</td>
<td>0.614</td>
</tr>
</tbody>
</table>

Table 4.1.3: Estimated Sample Size Using t Test

<table>
<thead>
<tr>
<th>β₁</th>
<th>τ=0.1</th>
<th>τ=0.3</th>
<th>τ=0.5</th>
<th>τ=0.6</th>
<th>τ=0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>4080</td>
<td>2426</td>
<td>2194</td>
<td>2246</td>
<td>2851</td>
</tr>
<tr>
<td>1</td>
<td>1022</td>
<td>608</td>
<td>550</td>
<td>563</td>
<td>715</td>
</tr>
<tr>
<td>1.5</td>
<td>455</td>
<td>272</td>
<td>246</td>
<td>252</td>
<td>319</td>
</tr>
<tr>
<td>2</td>
<td>257</td>
<td>154</td>
<td>139</td>
<td>143</td>
<td>180</td>
</tr>
<tr>
<td>3</td>
<td>116</td>
<td>70</td>
<td>63</td>
<td>65</td>
<td>82</td>
</tr>
</tbody>
</table>

Table 4.1.4: Empirical Power Using t Test

<table>
<thead>
<tr>
<th>β₁</th>
<th>τ=0.1</th>
<th>τ=0.3</th>
<th>τ=0.5</th>
<th>τ=0.6</th>
<th>τ=0.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.797</td>
<td>0.809</td>
<td>0.804</td>
<td>0.814</td>
<td>0.804</td>
</tr>
<tr>
<td>1</td>
<td>0.829</td>
<td>0.795</td>
<td>0.796</td>
<td>0.794</td>
<td>0.786</td>
</tr>
<tr>
<td>1.5</td>
<td>0.792</td>
<td>0.771</td>
<td>0.835</td>
<td>0.735</td>
<td>0.814</td>
</tr>
<tr>
<td>2</td>
<td>0.823</td>
<td>0.792</td>
<td>0.824</td>
<td>0.84</td>
<td>0.808</td>
</tr>
<tr>
<td>3</td>
<td>0.752</td>
<td>0.809</td>
<td>0.772</td>
<td>0.906</td>
<td>0.823</td>
</tr>
</tbody>
</table>
### Table 4.2: Cauchy Distribution Assumption

#### Table 4.2.1: Estimated Sample Size Using Formula (2.3)

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>33982</td>
<td>1688</td>
<td>861</td>
<td>1010</td>
<td>4615</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>8496</td>
<td>422</td>
<td>216</td>
<td>253</td>
<td>1154</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>3776</td>
<td>188</td>
<td>96</td>
<td>113</td>
<td>513</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>2124</td>
<td>106</td>
<td>54</td>
<td>64</td>
<td>289</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>944</td>
<td>47</td>
<td>24</td>
<td>29</td>
<td>129</td>
</tr>
</tbody>
</table>

#### Table 4.2.2: Empirical Power Using Formula (2.3)

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>0.805</td>
<td>0.808</td>
<td>0.815</td>
<td>0.815</td>
<td>0.805</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>0.833</td>
<td>0.82</td>
<td>0.716</td>
<td>0.793</td>
<td>0.753</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>0.791</td>
<td>0.731</td>
<td>0.691</td>
<td>0.784</td>
<td>0.828</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>0.768</td>
<td>0.712</td>
<td>0.602</td>
<td>0.835</td>
<td>0.799</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>0.795</td>
<td>0.642</td>
<td>0.518</td>
<td>0.626</td>
<td>0.836</td>
</tr>
</tbody>
</table>

#### Table 4.2.3: Estimated Sample Size Using $t$ Test

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>33984</td>
<td>1690</td>
<td>863</td>
<td>1012</td>
<td>4617</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>8498</td>
<td>424</td>
<td>218</td>
<td>255</td>
<td>1156</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>3778</td>
<td>190</td>
<td>98</td>
<td>115</td>
<td>515</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>2126</td>
<td>108</td>
<td>56</td>
<td>66</td>
<td>291</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>946</td>
<td>49</td>
<td>26</td>
<td>31</td>
<td>131</td>
</tr>
</tbody>
</table>

#### Table 4.2.4: Empirical Power Using $t$ Test

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>0.798</td>
<td>0.794</td>
<td>0.823</td>
<td>0.808</td>
<td>0.788</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>0.802</td>
<td>0.797</td>
<td>0.818</td>
<td>0.815</td>
<td>0.787</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>0.823</td>
<td>0.772</td>
<td>0.69</td>
<td>0.738</td>
<td>0.801</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>0.807</td>
<td>0.799</td>
<td>0.632</td>
<td>0.748</td>
<td>0.813</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>0.78</td>
<td>0.778</td>
<td>0.572</td>
<td>0.858</td>
<td>0.801</td>
</tr>
</tbody>
</table>
Table 4.3: Gamma Distribution Assumption

Table 4.3.1: Estimated Sample Size Using Formula (2.3)

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>10173</td>
<td>2638</td>
<td>1133</td>
<td>762</td>
<td>332</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>2544</td>
<td>660</td>
<td>284</td>
<td>191</td>
<td>83</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>1131</td>
<td>294</td>
<td>126</td>
<td>85</td>
<td>37</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>636</td>
<td>165</td>
<td>71</td>
<td>48</td>
<td>21</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>283</td>
<td>74</td>
<td>32</td>
<td>22</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 4.3.2: Empirical Power Using Formula (2.3)

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>0.812</td>
<td>0.771</td>
<td>0.797</td>
<td>0.749</td>
<td>0.809</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>0.81</td>
<td>0.828</td>
<td>0.772</td>
<td>0.741</td>
<td>0.804</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>0.775</td>
<td>0.793</td>
<td>0.721</td>
<td>0.791</td>
<td>0.697</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>0.777</td>
<td>0.787</td>
<td>0.734</td>
<td>0.714</td>
<td>0.792</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>0.81</td>
<td>0.87</td>
<td>0.752</td>
<td>0.778</td>
<td>0.759</td>
</tr>
</tbody>
</table>

Table 4.3.3: Estimated Sample Size Using $t$ Test

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>10175</td>
<td>2640</td>
<td>1135</td>
<td>764</td>
<td>334</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>2545</td>
<td>662</td>
<td>286</td>
<td>193</td>
<td>85</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>1133</td>
<td>295</td>
<td>128</td>
<td>87</td>
<td>39</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>638</td>
<td>167</td>
<td>73</td>
<td>50</td>
<td>23</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>285</td>
<td>76</td>
<td>34</td>
<td>24</td>
<td>12</td>
</tr>
</tbody>
</table>

Table 4.3.4: Empirical Power Using $t$ Test

<table>
<thead>
<tr>
<th></th>
<th>$\tau=0.1$</th>
<th>$\tau=0.3$</th>
<th>$\tau=0.5$</th>
<th>$\tau=0.6$</th>
<th>$\tau=0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1=0.5$</td>
<td>0.811</td>
<td>0.794</td>
<td>0.818</td>
<td>0.775</td>
<td>0.774</td>
</tr>
<tr>
<td>$\beta_1=1$</td>
<td>0.803</td>
<td>0.747</td>
<td>0.756</td>
<td>0.7</td>
<td>0.789</td>
</tr>
<tr>
<td>$\beta_1=1.5$</td>
<td>0.804</td>
<td>0.802</td>
<td>0.83</td>
<td>0.835</td>
<td>0.833</td>
</tr>
<tr>
<td>$\beta_1=2$</td>
<td>0.813</td>
<td>0.836</td>
<td>0.712</td>
<td>0.818</td>
<td>0.585</td>
</tr>
<tr>
<td>$\beta_1=3$</td>
<td>0.75</td>
<td>0.91</td>
<td>0.665</td>
<td>0.802</td>
<td>0.626</td>
</tr>
</tbody>
</table>
4.3. RESULTS

Table 4.4: Kernel Density

Table 4.4.1: Estimated Sample Size Using Formula (2.3)

<table>
<thead>
<tr>
<th>( \beta_1 )</th>
<th>( \tau = 0.1 )</th>
<th>( \tau = 0.3 )</th>
<th>( \tau = 0.5 )</th>
<th>( \tau = 0.6 )</th>
<th>( \tau = 0.8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>4633</td>
<td>2873</td>
<td>3242</td>
<td>3148</td>
<td>3116</td>
</tr>
<tr>
<td>1</td>
<td>1833</td>
<td>596</td>
<td>712</td>
<td>692</td>
<td>897</td>
</tr>
<tr>
<td>1.5</td>
<td>675</td>
<td>370</td>
<td>265</td>
<td>458</td>
<td>343</td>
</tr>
<tr>
<td>2</td>
<td>627</td>
<td>216</td>
<td>149</td>
<td>229</td>
<td>268</td>
</tr>
<tr>
<td>3</td>
<td>233</td>
<td>73</td>
<td>81</td>
<td>76</td>
<td>95</td>
</tr>
</tbody>
</table>

Table 4.4.2: Empirical Power Using Formula (2.3)

<table>
<thead>
<tr>
<th>( \beta_1 )</th>
<th>( \tau = 0.1 )</th>
<th>( \tau = 0.3 )</th>
<th>( \tau = 0.5 )</th>
<th>( \tau = 0.6 )</th>
<th>( \tau = 0.8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.835</td>
<td>0.844</td>
<td>0.921</td>
<td>0.910</td>
<td>0.815</td>
</tr>
<tr>
<td>1</td>
<td>0.961</td>
<td>0.769</td>
<td>0.873</td>
<td>0.869</td>
<td>0.874</td>
</tr>
<tr>
<td>1.5</td>
<td>0.903</td>
<td>0.903</td>
<td>0.811</td>
<td>0.96</td>
<td>0.801</td>
</tr>
<tr>
<td>2</td>
<td>0.985</td>
<td>0.886</td>
<td>0.788</td>
<td>0.932</td>
<td>0.904</td>
</tr>
<tr>
<td>3</td>
<td>0.959</td>
<td>0.79</td>
<td>0.851</td>
<td>0.826</td>
<td>0.833</td>
</tr>
</tbody>
</table>

Table 4.4.3: Estimated Sample Size Using t Test

<table>
<thead>
<tr>
<th>( \beta_1 )</th>
<th>( \tau = 0.1 )</th>
<th>( \tau = 0.3 )</th>
<th>( \tau = 0.5 )</th>
<th>( \tau = 0.6 )</th>
<th>( \tau = 0.8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>4635</td>
<td>3318</td>
<td>3569</td>
<td>2455</td>
<td>3180</td>
</tr>
<tr>
<td>1</td>
<td>886</td>
<td>738</td>
<td>713</td>
<td>309</td>
<td>1269</td>
</tr>
<tr>
<td>1.5</td>
<td>412</td>
<td>413</td>
<td>472</td>
<td>339</td>
<td>443</td>
</tr>
<tr>
<td>2</td>
<td>299</td>
<td>179</td>
<td>226</td>
<td>166</td>
<td>208</td>
</tr>
<tr>
<td>3</td>
<td>237</td>
<td>108</td>
<td>88</td>
<td>75</td>
<td>77</td>
</tr>
</tbody>
</table>

Table 4.4.4: Empirical Power Using t Test

<table>
<thead>
<tr>
<th>( \beta_1 )</th>
<th>( \tau = 0.1 )</th>
<th>( \tau = 0.3 )</th>
<th>( \tau = 0.5 )</th>
<th>( \tau = 0.6 )</th>
<th>( \tau = 0.8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.833</td>
<td>0.896</td>
<td>0.939</td>
<td>0.815</td>
<td>0.827</td>
</tr>
<tr>
<td>1</td>
<td>0.749</td>
<td>0.849</td>
<td>0.868</td>
<td>0.554</td>
<td>0.964</td>
</tr>
<tr>
<td>1.5</td>
<td>0.723</td>
<td>0.899</td>
<td>0.958</td>
<td>0.87</td>
<td>0.912</td>
</tr>
<tr>
<td>2</td>
<td>0.854</td>
<td>0.847</td>
<td>0.928</td>
<td>0.828</td>
<td>0.833</td>
</tr>
<tr>
<td>3</td>
<td>0.961</td>
<td>0.892</td>
<td>0.884</td>
<td>0.832</td>
<td>0.735</td>
</tr>
</tbody>
</table>
4.3. RESULTS

Comments: As it is mentioned in Chapter 2, $t$ test is used in quantile regression in \textit{quantreg} package, so simulated power when using $t$ distribution for estimators performs better than simulated power using formula (2.3) under Normal, Cauchy and Gamma distributions assumptions. When use kernel smoothing method, type of kernel and/or bandwidth should be chosen carefully. Generally, simulation shows that our method is able to give a practical estimation of sample size for given significance level and power.
Chapter 5

Conclusions and Future Work

5.1 Summary of Conclusions

In this project, we first reviewed linear regression and quantile regression models. Research literature shows the fact that quantile regression model is a robust alternative to linear regression model when data do not follow normal distribution. Quantile regression is broadly applied in econometrics and finance, and is also becoming popular in health research in recent years. Since sample size is an important factor in planning stage of the study design, it is beneficial to develop a set of sample size calculation tools for quantile regression models.

Because there are few methods available for sample size calculation using power analysis for quantile regression, we have developed methods to calculate sample size or power for univariate case in quantile regression. The method is programmed into an R package rqsamplesize to calculate sample size for any given $\tau$ percent quantile of the response variable under the assumption of $iid$ error distribution. It can be easily used on machines with R and user can easily extend the tool to other program
5.2. FUTURE WORK

languages. Our R package uses similar input and output structure as other popular sample size packages in R, such as `pwr` and `power.t.test`.

To make the sample size tools more accessible to a broader user group, an online calculator using graphic user interface is built with the same concept. We make the online tool available for easy access and intuitive operation for users who are not familiar with the R software.

We have shown detailed theoretical work behind our methods. We further conduct numerical simulation to show that our method is able to provide a reliable estimate for given significance level and power. Hence our method can be applied in planning stage of study design where quantile regression is considered.

5.2 Future Work

In the future, the following directions can be considered for the further development of sample size tools for quantile regression models.

- **Incorporate Non-iid Errors**: Our method currently only considers errors with iid distribution. Currently, asymptotic distribution is available for quantile regression estimators when errors are non-iid. However, the derivation of a sample size formula for non-iid case itself is complicated when the dependent structure of the observation is not available. It is worth exploring sample size when errors are non-iid. In the future, both “iid” and “non-iid” may be integrated into existing tools. How to identify non-iid errors when only a set of residual vector is given can be an interesting research topic.
5.2. FUTURE WORK

- **Extend to Multivariate Case:** Even though our method offers a way to calculate sample size for the test of multiple estimators, it still falls under univariate scenario since the information used to construct \( rqfun \) is the summary information of one covariate (only one mean and one standard deviation is allowed). Hence it is possible to extend our method to multivariate case with independent covariates.

- **Extend to Model with Multiple Quantile Points:** Based on the asymptotic theory for the joint distribution of coefficients estimation at multiple quantile points of quantile regression models, it is possible to extend the power and sample size formula for a single quantile (\( \tau \)) to models with multiple quantile points. For example, we can extend the method to provide sample size estimate for studies that need estimate the covariate effects at quantiles of \( \tau_1, \tau_2, ..., \tau_K \) at the same time.

- **Add Analysis of Variance (ANOVA) Test:** It may also be interesting to consider adding ANOVA test to current method. That is, adding one more function so that the method is able estimate the sample size required to compare multiple number of models.
Bibliography


Appendices
Appendix A

R code for rqsamplesize package

A.1 Kernel smoothing function: kerneldens.R

```r
kerneldens <- function (dist, tau, kernel.smooth="norm", bw=NULL)
{
  n = length(dist)
  m = 100
  pt = (1:m)/(m+1)
  s = sd(dist)
  if(is.null(bw)) {
    bw = 1.5*s*n^(-0.2)
  }
  fplus = function(x) (x + abs(x))/2
  # Normal kernel pdf and cdf
  k0 = function(x) dnorm(x)
  K0 = function(x) pnorm(x)
  # Rectangle kernel pdf and cdf
  k1 = function(x) dunif(x,-1/2,1/2) #also dunif(x+1/2)
  K1 = function(x) punif(x,-1/2,1/2) #also punif(x+1/2)
  # Triangle kernel pdf and cdf
  k2 = function(x) ((fplus(x+3/2))^2-3*(fplus(x+1/2)))^2/2
  K2 = function(x) ((fplus(x+3/2))^3-3*(fplus(x+1/2)))^3/6
  # 3-piece kernel pdf and cdf
  k3 = function(x) ((fplus(x+2))^3-4*(fplus(x+1)))^3/6
  K3 = function(x) ((fplus(x+2))^4-4*(fplus(x+1)))^4/24
  # 4-piece kernel pdf and cdf
  k4 = function(x) ((fplus(x+2))^4-4*(fplus(x+1)))^4/24
  K4 = function(x) ((fplus(x+2))^5-5*(fplus(x+1)))^5/120

  # fsmooth is estimated density function
  # Fsmooth is estimated cumulative distribution function
```
A.2 Library of supplementary functions: library.R

1 ######################################################################
2 Define a Class of rqfun for quantile regression
3 ######################################################################
4 # example of usage:
5 #
6 x = rqfun(mu = 3, sd = 5)
7 #
8 source("./kerneldens.R")
9 rqfun = function(x, ...) UseMethod("rqfun")
10
11 ######################################################################
12 Initiate regression function
13 ######################################################################
14 rqfun.default = function(mu=0, sd=1, dist='norm', term=c('1'), pos=2, method='exact',
15 a =NA, b=NA) {
16 x = list(mu=mu, sd=sd, dist=dist, term=term, pos=pos, method=method)
17 if (x$dist == 'bin') {
18 if (x$mu<0|x$mu>1)
19 stop("Probability of binary distribution is between 0 and 1.")
20 x$sd = sqrt(x$mu*(1-x$mu))
21 }
if (x$dist == 'unif') {
  if (is.na(a) | is.na(b)) {
    a = mu - sqrt(3)*sd
    b = mu + sqrt(3)*sd
  } else {
    x$mu = (a+b)/2
    x$sd = sqrt(1/12)*(b-a)
  }
  x$a = a
  x$b = b
} else {
  x$a = a
  x$b = b
}

class(x) = 'rqfun'
return(x)

# example of usage:
x = rqfun(a = 1, b = 10, dist = 'unif')
print(x)

print.rqfun = function(x, ...) {
  x.dim = length(x$term)
  pos = x$pos
  cat('The covariate X has the following ', x.dim+1, ' components:
  
  Intercept
  
  x
  
  1/x
  
  sqrt(x)
  
  x^2
  
  1/x^2
  
  x^3
  
  1/x^3
  
  log(x)
  
  exp(x)
  
  for (j in 1:length(pos)) {
    if ((1+1) == pos[j]) cat(' *')
  }
  cat('
')
}
if (x$dist == 'unif') {
    cat('where x has a uniform distribution with a = ', x$a, ' and b = ', x$b, 'n')
} else {
    cat('where x has a ', x$dist, ' distribution with mean = ', x$mu, ' and variance = ', (x$sd)^2, 'n')
}

if (x$method == 'exact') {
    cat('Exact')
} else {
    cat('Simulation')
}

cat('method is used to calculate matrix Q = E(t(X) %*% (X)).n')

cat('Term(s) with * will be used in sample size calculation')

# example of usage:

# x = rqfun(mu = 3, sd = 5, term = c('sqrt', '1', '2'))
# plot(x)

plot.rqfun = function(x, beta=NULL, B = 100, ...) {
    x.dim = length(x$term)
    if(is.null(beta))
        beta = rep(1, (x.dim+1))
    Qs = getQs(x, B = B, X.return = TRUE)
    X = Qs$X
    plot(X[, 1], X[, -1] %*% beta, type = 'l', xlab = 'x', ylab = 'g(x)')
    bc = paste(beta[1])
    for(i in 1:x.dim)
        bc = paste(bc, ' ', beta[i+1])
    title(paste('Quantile regression function for nbeta = (', bc, ').'))
}

getQs = function(x, B = 100000, X.return=FALSE) {
    if(x$dist == 'norm')
        xi = rnorm(B, x$mu, x$sd)
    if(x$dist == 'bin')
        xi = rbinom(B, 1, x$mu)
    if(x$dist == 'unif')
        xi = runif(B, x$a, x$b)
    x.dim = length(x$term)
    X = matrix(0, B, x.dim+1)
    x.lab = rep(' ', x.dim+1)
    X[, 1] = rep(1, B)
    x.lab[1] = 'Intercept'
    for (i in 1:x.dim) {
        j = i + 1
        if (x$term[i] == '-2') {
A.2. LIBRARY OF SUPPLEMENTARY FUNCTIONS: LIBRARY.R

```r
if (min(x[i])==0)
  stop("Error: 0 is in denominator."
117  X[, j] = 1/x[i]^2
118  x.lab[j] = '1/x^2'
119 }
120 if (x$term[i] == '-1') {
121    if (min(x[i])==0)
122      stop("Error: 0 is in denominator."
123    X[, j] = 1/x[i]
124    x.lab[j] = '1/x'
125  }
126 if (x$term[i] == '1/sqrt') {
127    if (min(x[i])==0)
128      stop("Error: 0 is in denominator."
129    else if (min(x[i])<0)
130      stop("Error: There is negative value in sqrt term."
131    X[, j] = 1/x[i]^0.5
132    x.lab[j] = '1/sqrt(x)'
133  }
134 if (x$term[i] == '-3') {
135    if (min(x[i])==0)
136      stop("Error: 0 is in denominator."
137    X[, j] = x[i](-3)
138    x.lab[j] = '1/x^3'
139  }
140 if (x$term[i] == 'sqrt') {
141    if (min(x[i])<0)
142      stop("Error: There is negative value in sqrt term."
143    X[, j] = x[i]^0.5
144    x.lab[j] = 'sqrt(x)'
145  }
146 if (x$term[i] == '1') {
147    X[, j] = x[i]
148    x.lab[j] = 'x'
149  }
150 if (x$term[i] == '2') {
151    X[, j] = x[i]^2
152    x.lab[j] = 'x^2'
153  }
154 if (x$term[i] == '3') {
155    X[, j] = x[i]^3
156    x.lab[j] = 'x^3'
157  }
158 if (x$term[i] == 'exp') {
159    X[, j] = exp(x[i])
160    x.lab[j] = 'e^x'
161  }
162 if (x$term[i] == 'log') {
163    if (min(x[i])<0)
164      stop("Error: There is negative value in sqrt term."
165    X[, j] = log(x[i])
166  }
```
x.lab[j] = 'log(x)'

Q = t(X)%*%X/B

dimnames(Q) = list(x.lab, x.lab)

if(X.return)
    XB = cbind(xi, X)
    tmp = sort(XB[, 1], index.return=TRUE)
    XB = XB[tmp$ix,
}

return(list(Q-Q, X=XB))
else {
    return(Q)
}

getQe = function(x){
    # Matrix for polynomial of x
    if(x$dist=='norm'){
        u1 = x$mu
        u2 = x$mu^2 + x$sd^2
        u3 = x$mu^3 + 3*x$mu*x$sd^2
        u4 = x$mu^4 + 6*x$mu^2*x$sd^2 + 3*x$sd^4
        u5 = x$mu^5 + 10*x$mu^3*x$sd^2 + 15*x$mu*x$sd^4
        u6 = x$mu^6 + 15*x$mu^4*x$sd^2 + 45*x$mu^2*x$sd^4 + 15*x$sd^6
    } else if (x$dist=='unif'){
        if(is.na(x$a) & is.na(x$b)){
            a = x$mu-sqrt(3)*x$sd
            b = x$mu+sqrt(3)*x$sd
        } else {
            a=x$a
            b=x$b
        }
        u1 = (a+b)/2
        u2 = (a^2+a*b+b^2)/3
        u3 = (a+b)*(a^2+b^2)/4
        u4 = (a^4+a^3*b+a^2*b^2+a*b^3+b^4)/5
        u5 = (b^6-a^6)/(6*(b-a))
        u6 = (b^7-a^7)/(7*(b-a))
    } else {
        stop("ERROR: x shall be either normal or uniform distribution.")
    }
    Ex = c(1,u1,u2,u3,u4,u5,u6)
    d=length(x$term)+1
    X.power = matrix(c(0,as.numeric(x$term)), d, d)
    index = X.power+t(X.power)+1
    Q=matrix(Ex[index], d, d)
}

182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
A.2. LIBRARY OF SUPPLEMENTARY FUNCTIONS: LIBRARY.R

# Calculate the asymptotic variance matrix of quantile regression beta hat

qrV <- function(x, sd, tau, dist = "Norm", kernel.smooth = bw, subint) {
  # Covariate
  # sd of error distribution
  # dist distribution of error: Norm, Cauchy, Gamma or a vector of residual
  x.dim = length(x$term)
  if (x.dim == 1) {
    Q = matrix(c(1, x$name, x$name, x$name^2 + x$sd^2), nrow = 2, ncol = 2)
  }
  if (x.dim > 1) {
    # use simulation method to find Q
    if (x$method == 'sim')
      Q = getQs(x)
    else # use exact method to find Q
      Q = getQe(x)
  }
  Q = solve(Q)
  if (is.null(dist) || is.character(dist) == TRUE) {
    if (is.null(dist) || dist == "Norm") {
      if (length(sd) > 1)
        stop("Need a single number for standard deviation.")
      df2 = dnorm(qnorm(tau, 0, sd), 0, sd)^2
    } else if (dist == "Cauchy") {
      if (length(sd) > 1)
        stop("Need a single number for Cauchy scale parameter.")
      df2 = dcauchy(qcauchy(tau, 0, sd), 0, sd)^2
    } else if (dist == "Gamma") {
      # Error Ui has Gamma distribution with Gamma(k, 1/k), Var(Ui) = 1/k = s^2
      # log(Yi) = x%*%beta + log(Ui) - log(U_tau)
      # e_i = logUi - logU_tau, e_i has density function U_tau*exp(e_i)*f_Ui(U_tau*exp(e_i))
      # e_i_tau = 0
      df2 = ggamma(tau, k, scale = 1/k)
    } else if (is.numeric(dist) == TRUE) {
      if (is.numeric(dist) == TRUE) {
        df2 = kerneldens(dist, tau, kernel.smooth, bw)$ftau
      }
    df2 = df2^2
  }
  Var = tau*(1-tau)/df2 * Q
  return(Var)
A.3 Main function: power.rq.test.R

```r
source("./library.R")
```

```r
def power.rq.test = function (x, n = NULL, sig.level = 0.05, power = NULL, tau = 0.5, delta = 1, sd = 1, dist = "Norm", kernel.smooth = "norm", bw = NULL, alternative = c("two.sided", "one.sided")) {
  # x is data in vector or matrix form
  # sig.level is Type I error and power is Type II error
  # tau is the desired quantile i.e. 0.1, 0.25, 0.5, 0.75, 0.9
  # delta is a vector of all regression coefficients in quantile regression
  # sd standard deviation of error
  # dist is the distribution of error term: it can be Normal ('norm'), Cauchy ('cauchy')
  # or Gamma ('Gamma') or a vector of residuals
  # Note that since we test the coefficient is 0 under null hypothesis, beta[pos] is also the difference, delta in Section 1.3.3 of Sample Size Calculations in Clinical Research 2008
```
A.3. MAIN FUNCTION: `POWER.RQ.TEST.R`

```r
if (sum(sapply(list(n, power), is.null)) != 1)
  stop("exactly one of n, power must be NULL")
if (sig.level <= 0 || sig.level > 1)
  stop("Type I error is between 0 and 1.")
if (!is.null(power)) {
  if (power < 0 || power > 1)
    stop("Type II error is between 0 and 1.")
  beta = c(1, delta)
  alternative = match.arg(alternative)
  tside = switch(alternative, one.sided = 1, two.sided = 2)
  NOTE = paste('Sample size based on', x$method, 'method')
  METHOD = 'Sample size for quantile regression'
  pos = x$pos
  b = beta[pos]
  Vq = qrV(x, sd, tau, dist, kernel.smooth, bw)[pos, pos]
  p = length(beta)
  df = length(pos)
  # Sample size and power for test of multiple coefficients
  if (df == 1) {
    if (is.null(n)) {
      p.body = quote({
        1 - pt(qt(1 - sig.level / tside, n - 2), ncp = b / (sqrt(Vq / n)), df = n - 2)
      })
      n = uniroot(function(n) eval(p.body) - power, c(3, 1e+06))$root
    }
  } else if (is.null(power)) {
    power = 1 - pt(qt(1 - sig.level / tside, n - 2), ncp = b / (sqrt(Vq / n)), df = n - 2)
  }
  # Sample size and power for test of multiple coefficients
  if (df > 1) {
    lambda = t(b) %*% solve(Vq) %*% b
    p.body = quote({
      c = p * (n - 1) / (n - p)
      1 - c * pf(c * qf(1 - sig.level, df1 = df, df2 = n - p), ncp = n * lambda, df1 = df, df2 = n - p)
    })
    if (is.null(n)) {
      n = uniroot(function(n) eval(p.body) - power, c(p + 1, 1e+06))$root
    } else if (is.null(power)) {
      power = eval(p.body)
    }
  }
  n = ceiling(n)
```

A.4. CODE FOR FIGURES IN CHAPTER 2

84 \texttt{structure(list(n = n, delta = delta, pos = pos, sd = sd, sig.level = sig
.level,}
85 \texttt{power = power, alternative = alternative, note = NOTE,}
86 \texttt{method = METHOD), class = "powerctest")}
87 \}

A.4 Code for figures in Chapter 2

1 \#Code to generate figure 2.1:
2
3 \texttt{source("./power.rq.test.R")}
4 \texttt{x = rqfun(mu = 0.5, dist = "bin", pos = 2, term = c(’1’), method = ’
exact’)}
5 \texttt{tau = seq(0.1, 0.9, 0.01)}
6 \texttt{n = seq(10, 1000, 10)}
7 \texttt{pw = n}
8 \textbf{for} (i \textbf{in} 1:length(n)){
9 \texttt{pw[i] = power.rq.test(x, n = n[i], tau =0.5, delta = 0.7,}
10 \texttt{sd = 1, dist = ”Norm”, alternative = ”two.sided”)$power}
11 \}
12 \texttt{plot(n, pw, type = ’n’, ylab = ’Power’, ylim = c(0, 1))}
13 \texttt{lines(n, pw)}
14
15 \textbf{for} (i \textbf{in} 1:length(n)){
16 \texttt{pw[i] = power.rq.test(x, n = n[i], tau =0.5, delta = 0.5,}
17 \texttt{sd = 1, dist = ”Norm”, alternative = ”two.sided”)$power}
18 \}
19 \texttt{lines(n, pw, lty = 2)}
20
21 \textbf{for} (i \textbf{in} 1:length(n)){
22 \texttt{pw[i] = power.rq.test(x, n = n[i], tau =0.5, delta = 0.3,}
23 \texttt{sd = 1, dist = ”Norm”, alternative = ”two.sided”)$power}
24 \}
25 \texttt{lines(n, pw, lty = 3)}
26
27 \#Code to generate figure 2.2:
28
29 \texttt{source("./power.rq.test.R")}
30 \texttt{x = rqfun(mu = 0.5, dist = ”bin”, pos = 2, term = c(’1’), method = ’
exact’)}
31 \texttt{tau = seq(0.01, 0.99, 0.01)}
32 \texttt{n = seq(10, 1000, 10)}
33 \texttt{pw = tau}
34 \textbf{for} (i \textbf{in} 1:length(tau)){
35 \texttt{pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.7,}
36 \texttt{sd = 1, dist = ”Norm”, alternative = ”two.sided”)$power}
37 \}
38 \texttt{plot(tau, pw, type = ’n’, ylab = ’Power’, ylim = c(0, 1))}
39 \texttt{lines(tau, pw)}
40
for (i in 1:length(tau))
    pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.5, sd = 1, dist = "Norm", alternative = 'two.sided')$power
lines(tau, pw, lty = 2)

for (i in 1:length(tau))
    pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.3, sd = 1, dist = "Norm", alternative = 'two.sided')$power
lines(tau, pw, lty = 3)

#Code to generate figure 2.3:
source("./power.rq.test.R")
x = rqfun(mu = 0.5, dist = "bin", pos = 2, term = c('1'), method = 'exact')
tau = seq(0.01, 0.99, 0.01)
n = seq(10, 1000, 10)
pw = tau
for (i in 1:length(tau))
    pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.7, dist = "Cauchy", alternative = 'two.sided')$power
plot(tau, pw, type = 'n', ylab = 'Power', ylim = c(0, 1))
lines(tau, pw)

for (i in 1:length(tau))
    pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.5, dist = "Cauchy", alternative = 'two.sided')$power
lines(tau, pw, lty = 2)

for (i in 1:length(tau))
    pw[i] = power.rq.test(x, n = 300, tau = tau[i], delta = 0.3, dist = "Cauchy", alternative = 'two.sided')$power
lines(tau, pw, lty = 3)
Appendix B

R code for Shiny app

B.1 User interface: ui.R

```r
library(shiny)
source('./library.R')
source('./power.rq.test.R')
ui<-fluidPage(
  titlePanel("Sample size and power calculator for quantile regression"),
  fluidRow(
    column(3,
      h1("Covariates"),
      numericInput("xmean", label = "Mean", value = 0),
      numericInput("xsd", label = "Standard Deviation", value = 1),
      selectInput("xdist", label = "Distribution"),
      choices = list("norm", "unif", "bin"),
      selected = "norm"),
    numericInput("covariates", label = "Covariates", value = 1),
    conditionalPanel("input.covariates>1"),
    checkboxGroupInput("xterm", label = "Terms"),
    choices = list("1", "2", "-1", "-2", "log", "exp", "sqrt"),
    selected = "1")
  )
)
```

```r
numericInput("xpos", label = "Number of covariates of interest", value = 1),
numericInput("xpos1", label = "Position of Interest 1", value = 2),
conditionalPanel("input.xpos>1"),
uiOutput("Positions")
```

```r
selectInput("xmth", label = "Method"),
choices = list("exact", "sim"),
selected = "exact"),
conditionalPanel("input.xdist=='unif'"),
```
B.2 Computation for output: server.R

```r
numericInput("xa", label = "Lower limit of uniform distribution (only when unif is selected above)", value = NA),
numericInput("xb", label = "Upper limit of uniform distribution (only when unif is selected above)", value = NA)
```

```
#h1("Define other parameters"),
selectInput("pw_or_n", label = "Power or Sample size", choices = list("Power", "SampleSize"), selected = "SampleSize"),
conditionalPanel("input.pw_or_n=='Power'", numericInput("n", label = "Sample size", value = 100)),
sliderInput("sig.level", label = "Significance level", value = 0.05, min = 0.005, max = 0.1),
conditionalPanel("input.pw_or_n=='SampleSize'",
sliderInput("pwr", label = "Power of test", value = 0.8, min = 0.5, max = 1),
sliderInput("tau", label = "Quantile", value = 0.5, min = 0, max = 1),
numericInput("sd", label = "Error standard deviation", value = 1),
numericInput("c1", label = "delta_1", value = 0.5),
conditionalPanel("input.covariates>1"),
uiOutput("Additional_coefficients"),
selectInput("dist_or_resid", label = "Distribution or Residual", choices = list("Distribution", "Residual"), selected = "Distribution"),
conditionalPanel("input.dist_or_resid=='Distribution'",
selectInput("dist", label = ""),
choices = list("Norm", "Cauchy", "Gamma"),
selected = "Norm"),
conditionalPanel("input.dist_or_resid=='Residual'",
fileInput("file", label = ""),
selectInput("kernel.smooth", label = "Kernel type for kernel smoothing", choices = list("norm","1"="1","2"="2","3"="3","4"="4"), selected = "0"),
numericInput("bw", label = "Bandwidth for kernel smoothing", value = NULL)
)
```

```r
selectInput("alternative", label = "Test type", choices = list("two.sided", "one.sided"), selected = "two.sided"),
```

```
column(6,
mainPanel(
  h3("Summary"),
  verbatimTextOutput("summary")
))
```
B.2. COMPUTATION FOR OUTPUT: SERVER.R

```r
server<--function(input, output){
  output$Additional_coefficients<-renderUI({
    lapply(2:input$coefficients, function(i) {
      numericInput(inputId = paste0("c", i), label = paste("delta", i), value = NULL)
    })
  })
  output$Positions<-renderUI({
    lapply(2:input$xpos, function(i) {
      numericInput(inputId = paste0("xpos", i), label = paste("Position_of
Interest", i), value = i+1)
    })
  })
  output$summary<-renderPrint({
    poss=input$xpos1
    for (1 in 2:input$xpos){
      poss=c(poss, input[[paste0("xpos", 1)]]))
    }
  })
}

x=rqfun(input$xmean, input$xsd, input$xdist, input$xterm, pos = poss, 
         input$xmth)
print(x)

if(input$pw_or_n == 'Power') {
  n = input$n
  pwr = NULL
} else {
  n = NULL
  pwr = input$pwr
}

daelta=input$c1
if (input$coefficients >1){
  for (j in 2:input$coefficients){
    delta=c(delta, input[[paste0("c", j)]]))
  }
}
print(delta)

if(is.numeric(input$file)==TRUE){
  res=read.table(input$file)
  pwn =power.rq.test(x, input$n, input$sig.level, input$pwr, input$tau, input$sd, 
                     delta=delta, dist=res, input$kernel.smooth, input$bw, input$alternative)
}
else
  pwn = try(power.rq.test(x, n = n, sig.level = input$sig.level, power = pwr 
                       , tau = input$tau, sd = input$sd, 
                       delta = delta, dist = input$dist, alternative = input$alternative))
if(class(pwn)=='try-error') cat('Please input a set of validate parameters...\n')
```
46 else print(pwn)
47 }
48 }
Appendix C

R code for simulations

C.1 Simulation for Normal: SimNorm.R

1 # Table 4.1
2 library(quantreg)
3 rm(list = ls())
4 source("./library.R")
5 source("./samplesize_revised20160614.R")
6 set.seed(101)
7 sig.level = 0.05
8 power = 0.8
9 sd = 10
10 tau = c(0.1, 0.3, 0.5, 0.6, 0.8)
11 b1 = c(0.5, 1, 1.5, 2, 3)
12 n.Normal = matrix(0, nrow = length(b1), ncol = length(tau))
13 x = rqfun(mu = 5, sd = 1.5, dist = "norm", pos = 2, term = c('1'), a=NA, b =NA, method = 'exact')
14 Pwr = matrix(0, nrow = length(b1), ncol = length(tau))
15 pos = x$pos
16 N = 1000
17 # Table 4.1.1 and Table 4.1.3
18 for (i in 1:length(b1)) {
19   b0 = b1[i]
20   for (j in 1:length(tau)) {
21     pw = power.rq.test(x=x, power = power, tau = tau[j], delta = b0,
22                       sd = sd, dist = "Norm", alternative = 'two.sided')
23     n.Normal[i,j] = pw$n
24   }
25 }
26 # Table 4.1.2 and Table 4.1.4
27 n = n.Normal[i,j]
28 z0 = rep(1, n)
29 z1 = rnorm(n, x$mu, x$sd)
30 Z = cbind(z0, z1)
31 Qy = Z%*%c(1, b0)
C.2 Simulation for Cauchy: SimCauchy.R

```{r}
# Table 4.2
library(quantreg)
rm(list = ls())
source("./library.R")
source("./samplesize_revised20160614.R")
set.seed(101)
sig.level = 0.05
power = 0.8
sd = 5
tau = c(0.1, 0.3, 0.5, 0.6, 0.8)
b1 = c(0.5, 1, 1.5, 2, 3)
n.Cauchy = matrix(0, nrow = length(b1), ncol = length(tau))
x = rqfun(mu = 5, sd = 1.5, dist = "norm", pos = 2, term = c('1'), a=NA, b =NA, method = 'exact')
Pwr=matrix(0, nrow = length(b1), ncol = length(tau))
pos = x$pos
N = 1000
# Table 4.2.1 and Table 4.2.3
for (i in 1:length(b1)) {
  b0 = b1[i]
  for (j in 1:length(tau)) {
    pw = power.rq.test(x=x, power = power, tau = tau[j], delta = b0,
    sd = sd, dist = "Cauchy", alternative = 'two.sided')
    n.Cauchy[i, j] = pw$n
  }
  # Table 4.2.2 and Table 4.2.4
  n = n.Cauchy[i,]
  z0 = rep(1,n)
z1 = rnorm(n, x$mu, x$sd)
Z = cbind(z0, z1)
Qy = Z%*%c(1,b1[i])
yloc = Qy-sd*tan(pi*(tau[j]-0.5))
pv = rep(0, N)
for (l in 1:N)
  ymean = Qy-qnorm(tau[j], 0, sd)
  for (l in 1:N)
    Yl = rnorm(n, ymean, sd)
    qfit = rq(Yl~z1, tau = tau[j])
    b = qfit$coefficients[pos]
    sbeta = summary(qfit, se = "iid", covariance = TRUE)$cov[pos, pos]
    tstar = b/sqrt(sbeta)
    pv[l] = 2*(1-pt(abs(tstar), n-length(b0)))
  }
Pwr[i, j]=mean(pv <= sig.level)
}
C.3 Simulation for Gamma: SimGamma.R

```r
# Table 4.3
library(quantreg)
rm(list = ls())
source("./library.R")
source("./samplesize_revised20160614.R")
set.seed(101)
sig.level = 0.05
power = 0.8
sd = 3
k = 1/sd^2
tau = c(0.1,0.3,0.5,0.6,0.8)
b1 = c(0.5,1,1.5,2,3)
n.Gamma = matrix(0,nrow = length(b1),ncol = length(tau))
x = rqfun(mu = 5,sd = 1.5,dist = "norm",pos = 2,term = c('1'),a=NA,b =NA,method = 'exact')
Pwr = matrix(0,nrow = length(b1),ncol = length(tau))
pos = x$pos
N = 1000
# Table 4.3.2 and Table 4.3.4
for (i in 1:length(b1)) {
  b0 = b1[i]
  for (j in 1:length(tau)) {
    pw = power.rq.test(x=x,power = power,tau = tau[j],delta = b0,
    sd = sd, dist = "Gamma", alternative = 'two.sided')
    n.Gamma[i,j] = pw$n
  }
}
# Table 4.3.2 and Table 4.3.4
n = n.Gamma[i,j]
z0 = rep(1,n)
z1 = rnorm(n,x$mu,x$sd)
Z = cbind(z0,z1)
Qy = Z%*%c(1,b1[i])
Utau = gamma(tau[k],k, scale = 1/k)
pv = rep(0,N)
for (l in 1:N) {
  #cat(i, "\n")
  Yl = rgamma(n,k,scale = exp(Qy)/(Utau*k))
  lgy = log(Yl)
}
```

Yl = rcauchy(n, yloc, sd)
qfit = rq(Yl~z1,tau = tau[j])
b = qfit$coefficients[pos]
sbeta = summary(qfit,se = "iid",covariance = TRUE)$cov[pos, pos]
tstar = b/sqrt(sbeta)
pv[l] = 2*(1 - pt(abs(tstar),n-length(b0)))
Pwr[i,j] = mean(pv <= sig.level)
```
C.4 Simulation for Kernel: SimKernel.R

```r
# Table 4.4
library(quantreg)
rm(list = ls())
source("./library.R")
source("./samplesize_revised20160614.R")
set.seed(101)
sig.level = 0.05
power = 0.8
sd = 10
tau = c(0.1, 0.3, 0.5, 0.6, 0.8)
b1 = c(0.5, 1, 1.5, 2, 3)
n.Normal = matrix(0, nrow = length(b1), ncol = length(tau))
kernel.Normal = matrix(0, nrow = length(b1), ncol = length(tau))
x = rqfun(mu = 5, sd = 1.5, dist = "norm", pos = 2, term = c('1'), a=NA, b =NA, method = 'exact')

Pwr = matrix(0, nrow = length(b1), ncol = length(tau))
pos = x$pos
N = 1000
pv = rep(0, N)

# Table 4.4.1 and Table 4.4.3
for (i in 1:length(b1)) {
  b0 = b1[i]
  for (j in 1:length(tau)) {
    n = 50
    z0 = rep(1, n)
    z1 = rnorm(n, x$mu, x$sd)
    Z = cbind(z0, z1)
    Qy = Z%*%c(1, b0)
ymean = Qy-qnorm(tau[j], 0, sd)
    Yl = rnorm(n, ymean, sd)
    qfit = rq(Yl*z1, tau = tau[j])
    res = qfit$residuals
    kernel.Normal[i,j] = power.rq.test(x=x, power = power, tau = tau[j], delta = b0, sd = sd, dist = res, alternative = 'two.sided')$p
  }
}
```

C.4 Simulation for Kernel: SimKernel.R

```r
# Table 4.4
library(quantreg)
rm(list = ls())
source("./library.R")
source("./samplesize_revised20160614.R")
set.seed(101)
sig.level = 0.05
power = 0.8
sd = 10
tau = c(0.1, 0.3, 0.5, 0.6, 0.8)
b1 = c(0.5, 1, 1.5, 2, 3)
n.Normal = matrix(0, nrow = length(b1), ncol = length(tau))
kernel.Normal = matrix(0, nrow = length(b1), ncol = length(tau))
x = rqfun(mu = 5, sd = 1.5, dist = "norm", pos = 2, term = c('1'), a=NA, b =NA, method = 'exact')

Pwr = matrix(0, nrow = length(b1), ncol = length(tau))
pos = x$pos
N = 1000
pv = rep(0, N)

# Table 4.4.1 and Table 4.4.3
for (i in 1:length(b1)) {
  b0 = b1[i]
  for (j in 1:length(tau)) {
    n = 50
    z0 = rep(1, n)
    z1 = rnorm(n, x$mu, x$sd)
    Z = cbind(z0, z1)
    Qy = Z%*%c(1, b0)
ymean = Qy-qnorm(tau[j], 0, sd)
    Yl = rnorm(n, ymean, sd)
    qfit = rq(Yl*z1, tau = tau[j])
    res = qfit$residuals
    kernel.Normal[i,j] = power.rq.test(x=x, power = power, tau = tau[j], delta = b0, sd = sd, dist = res, alternative = 'two.sided')$p
  }
}
C.4. SIMULATION FOR KERNEL: SIMKERNEL.R

```r
37 z0.kernel = rep(1,n.kernel)
38 z1.kernel = rnorm(n.kernel, x$mu, x$sd)
39 Z.kernel = cbind(z0.kernel, z1.kernel)
40 Qy.kernel = Z.kernel%*%c(1,b0)
41 ymean.new=Qy.kernel-qnorm(tau[j], 0, sd)
42
43 Y.kernel=rnorm(n.kernel, ymean.new, sd)
44 qfit.kernel = rq(Y.kernel~z1.kernel, tau = tau[j])
45 b.kernel = qfit.kernel$coefficients[pos]
46 sbeta = summary(qfit.kernel, se = "iid", covariance = TRUE)$cov[pos, pos]
47 tstar = b.kernel/sqrt(sbeta)
48 pv[1] = 2*(1-pt(abs(tstar),n-length(b0)))
49 }
50 Pwr[i,j]=mean(pv <= sig.level)
51 }
52 }
```