BST169: Course Work Project answer

sn0wfree

11/10/2016

Contents

1 BST169: Course Work Project

1.1 Question 1:

1. Consider the model:

$$y_i = \beta_0 + \beta_1 * x_{1,i} + \beta_2 * x_{2,i} + \epsilon_i \ (1)$$

What is the requirement for ϵ_i such that the following test statstics will be valid to test H0: $\beta_1 + \beta_2 = 1$?

- $W = N * (SSR_R - SSR_U)/SSR_U$ (Wald).
- $LM = N * (SSR_R - SSR_U)/SSR_R$ (Lagrange Multiplier),
- $LR = N * ln(SSR_R/SSR_U)$ (Likelihood Ratio)

where SSR_R is the sum of squared residuals obtained from the restricted model, while SSR_R is from the unrestricted model.

1.1.1 answer

$$\beta_1 + \beta_2 = 1$$

$<=>$

$$R * \beta = 1,$$

where $R = \begin{bmatrix} 1 & 1 \end{bmatrix} \ \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$

1. Wald test

$$\text{H0: } \beta_1 + \beta_2 = 1$$
$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon_i$$
$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon_i \text{ with } \beta_1 + \beta_2 = 1$$

0).
$E(x_i \epsilon_i) = 0$, i = 1,2,...,N;
$E(||x_i \epsilon_i||^{2+\delta}) < \Delta < 1$, for $\exists \delta > 0$, k = 1,...,K + 1 and i = 1,2,...,N

1). chi-sq distribution

$$sqrt(N)(R * \tilde{\beta} - 1) \sim N(0, RM_N^{-1} U_N M_N^{-1} R'),$$

where $\tilde{\beta} = (X'X)^{-1} X'y$

$$(1/N)(R\tilde{\beta} - 1)(R(X'X)^{-1}\tilde{U}_N(X'X)^{-1}R')^{-1}(R\tilde{\beta} - 1)' \sim \chi^2,$$

where $X = \begin{bmatrix} X_1 & X_2 \end{bmatrix}$, $\tilde{\beta} = \begin{bmatrix} \tilde{\beta}_1 \\ \tilde{\beta}_2 \end{bmatrix}$

2). homoscedasity

$$\tilde{U}_N = (SSR_U/(N - K - 1))X'X/N$$

is a symmetrical positive definite matrix computed from the constrained regresson such that $\tilde{U}_N - U_N \longrightarrow 0$

   2. Lagrange Multiplier

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon_i + \lambda(\beta_1 + \beta_2 - 1)$$

=> $$y = X\beta + \epsilon_i + \lambda(R * \beta - 1)$$

1). chi-sq distribution

$$(1/N)(R\tilde{\beta} - 1)(R(X'X)^{-1}\tilde{U}_N(X'X)^{-1}R')^{-1}(R\tilde{\beta} - 1)' \sim \chi^2$$

2). homoscedasity

$\tilde{U}_N$ is a symmetrical positive definite matrix computed from the constrained regresson such that $\tilde{U}_N - U_N \longrightarrow 0$

   3. Likelihood Ratio

$$\epsilon_i \sim i.i.d.N(0, \sigma^2)$$

1.2 Question 2

2. For the data set **pbp.csv**, can we use the **three test statistics** mentioned in the previous question to test H0 : $\beta_1 + \beta_2 = 1$? Why? If W and LM are not valid, how can one modify them for the test? What is your conclusion from the valid test?

No,the Wald test and LM test may invalid. Becaue there may have heteroscedasticity, the requirements of Wald and LM test(homoscedasity) is not satisfied. Thus, the exteral test–heteroscedasticity test should be used before proceduring the Wald and LM test.

there are two heteroscedasticity test: White test and Breusch-Pagan-Godfrey Test. But the White test is more general for the this linear regression model.

~~~

equ1: $y_i = \beta_0 + \beta_1 * x_{1,i} + \beta_2 * x_{2,i} + \epsilon_i$
equ2: $y_i - x_2 = \beta_0 + \beta_1(x_1 - x_2) + \epsilon_i$

```
pbp=read.csv("/Users/sn0wfree/Dropbox/PhD_1st_study/BST169_Econometrics/Crousework_Project/pbp.csv")
#head(pbp)
#str(pbp)

require(lmtest)
```

```
## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
equ1<-lm(y~x1 + x2,data=pbp)
equ2<-lm((y-x2)~(x1-x2), data=pbp)
```

```
#Breusch-Pagan-Godfrey Test
bptest(equ1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  equ1
## BP = 93, df = 2, p-value < 2.2e-16
```

```
bptest(equ2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  equ2
## BP = 0.81949, df = 1, p-value = 0.3653
```

```
#White test
bptest(residuals(equ1)~x1+x2+x1*x2+x1^2+x2^2,data=pbp)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  residuals(equ1) ~ x1 + x2 + x1 * x2 + x1^2 + x2^2
## BP = 96.036, df = 3, p-value < 2.2e-16
```

```
bptest(residuals(equ2)~(x1-x2)+(x1-x2)^2,data=pbp)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  residuals(equ2) ~ (x1 - x2) + (x1 - x2)^2
## BP = 0.81949, df = 1, p-value = 0.3653
```

From White test and Breusch-Pagan-Godfrey Test, the **equ1** results reject the NULL hypothesis: Homoscedasity, Which means the heteroscedasticity exist. And **equ2** do not reject the NULL hypothesis. thus there exist Homoscedasity

Overall, Wald and LM test is invalid. The original eqution: equ1 exist the heteroscedasticity.

Solutaion: Using Weighted Least Squared method to estimated the targeted regression rather than OLS.

1.3   Question 3

3. Generate y_i from the following model,
$$y_i = \beta_0 + \beta_1 * x_{1,i} + (1 - \beta_1) * x_{2,i} + \sqrt{x_{1,i}} * \epsilon_1 \ (2)$$
where $x_{1,i}$ follows chi-squared distribution with **2** degrees of freedom. Generate ϵ_1 from student t distribution with 6 degrees of freedom and $x_{2,i}{\sim}U(0,10)$. Check whether Wald, LR and LM in Question 1 follow chi-squared distribution by Monte Carlo.(The R command: ks.test(,'pchisq',2) can be used.) If W and LM are not valid, calculate the correct test statistics and also verify them by Monte Carlo. Please consider different sample sizes.

In text, $x_1 \sim \chi^2(2)$ and, Generate $x_2 \sim U(0,10)$ and $\epsilon_i \sim T(6)$.

Here set the sample size with $10 + loop$ and loop 10000 times. and the eqution 2 transform to
$$y_i = \beta_0 + \beta_1 * x_{1,i} + \beta_2 * x_{2,i} + e_i \ (2.1)$$
where

$$\beta_2 = 1 - \beta_1$$
$$e_i = \sqrt{x_{1,i}} * \epsilon_1$$

And I assume that $\beta_1 = 0.5$, $\beta_0 = 24$, and given a fixed value set x_1

from mc1.r mc2.r

lecture Monte Carlo

sample size; estimation:power of test ks.test(x,"pchisq",2)

```r
require(lmtest)
require(MASS)
```

```
## Loading required package: MASS
```

```r
require(stats)
##boost up: translate programme language code into Byte-code.
require(compiler)
```

```
## Loading required package: compiler
```

```r
enableJIT(3)
```

```
## [1] 0
```

```r
##boost up-end for continues
#assumption part

loop=100
#Warning: the loop time cannot be larger any more;please forgive me, this all my Macbook fault. And the

beta_1=0.4
beta_0=1
x1_store=rchisq(80+20, 2)
#initial valueset
original_N=10
signlevel=0.05

W_count=rep(0,loop)
LM_count=rep(0,loop)
LR_count=rep(0,loop)

#
# for loop start:Monte Carlo
for(j in 1:loop){
W=rep(0,loop)
LM=rep(0,loop)
LR=rep(0,loop)
N=original_N+j
for (i in 1:loop){
#generation part:data
x1=rchisq(N, 2)
x2=runif(N,0,10)
epsilon=rt(N,6)
y=beta_0+beta_1*x1+(1-beta_1)*x2+sqrt(x1)*epsilon
#generation part:regression
equ1<-lm(y~x1+x2)
equ2<-lm((y-x2)~(x1-x2))
```

4

```
#calculation
##pre-cal:heteroscedasticity
if (bptest(equ1)$p.value<signlevel){
  equ1<-lm(y~x1+x2,weights=(1/x1^0.5))
  equ2<-lm((y-x2)~(x1-x2),weights=(1/x1^0.5))
}
#calc SSR and Wald,LM, and LR
SSRu=sum(residuals(equ1)^2)
SSRr=sum(residuals(equ2)^2)

W=N*((SSRr-SSRu)/(SSRu))
LM=N*((SSRr-SSRu)/(SSRr))
LR=N*(log(SSRr/SSRu))


if(ks.test(W,'pchisq',2)$p.value>signlevel){W_count[j]=W_count[j]+1}
if(ks.test(LM,'pchisq',2)$p.value>signlevel){LM_count[j]=LM_count[j]+1}
if(ks.test(LR,'pchisq',2)$p.value>signlevel){LR_count[j]=LR_count[j]+1}
}
}

plot(W_count/(original_N+1:(original_N+loop))^2)
```
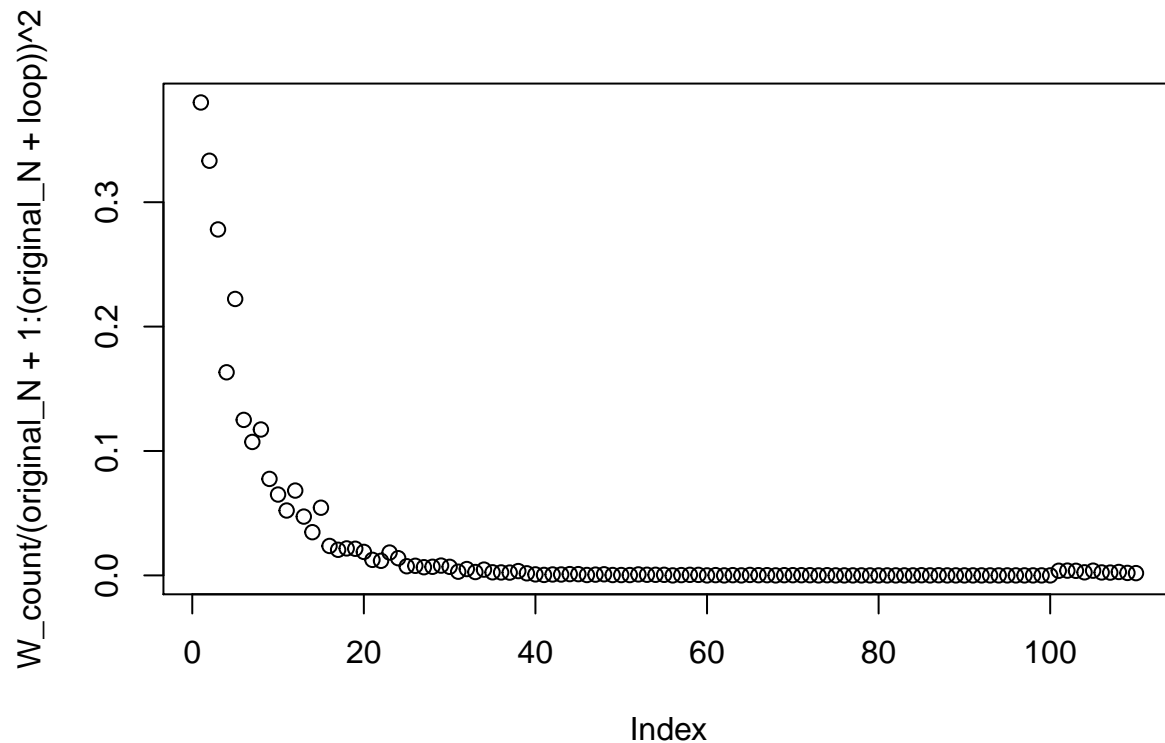
Warning in W_count/(original_N + 1:(original_N + loop))^2: longer object
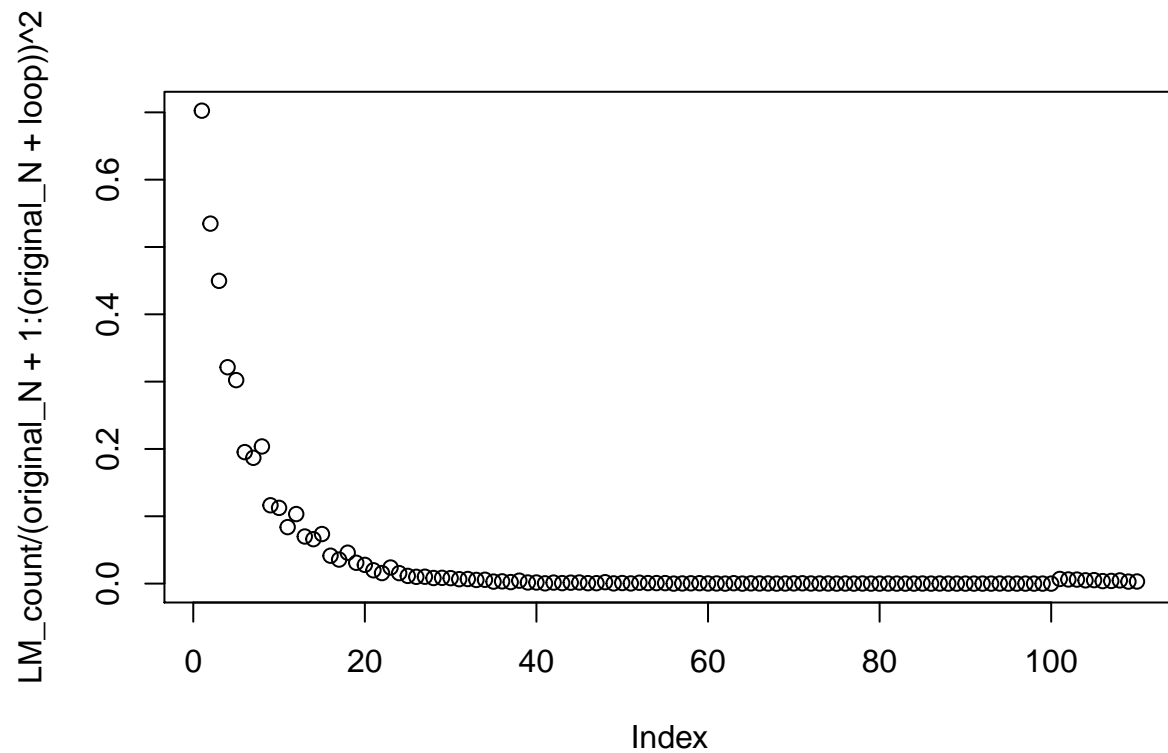## length is not a multiple of shorter object length
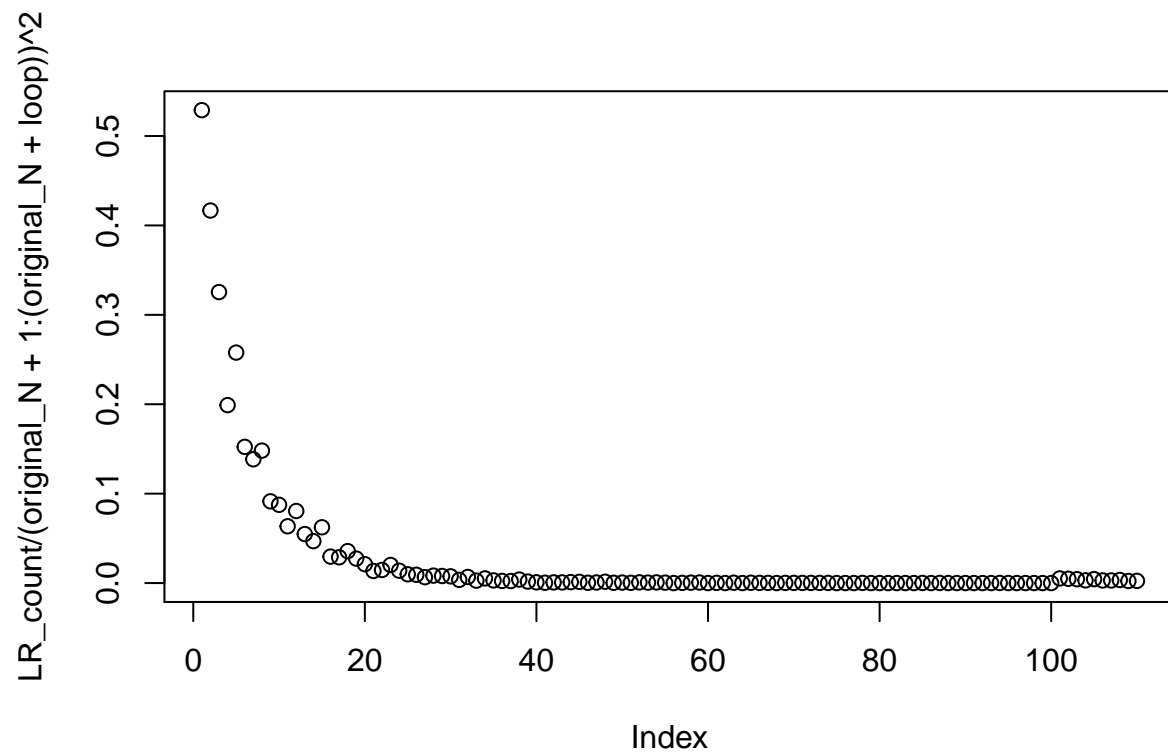


```
plot(LM_count/(original_N+1:(original_N+loop))^2)
```

Warning in LM_count/(original_N + 1:(original_N + loop))^2: longer object
length is not a multiple of shorter object length
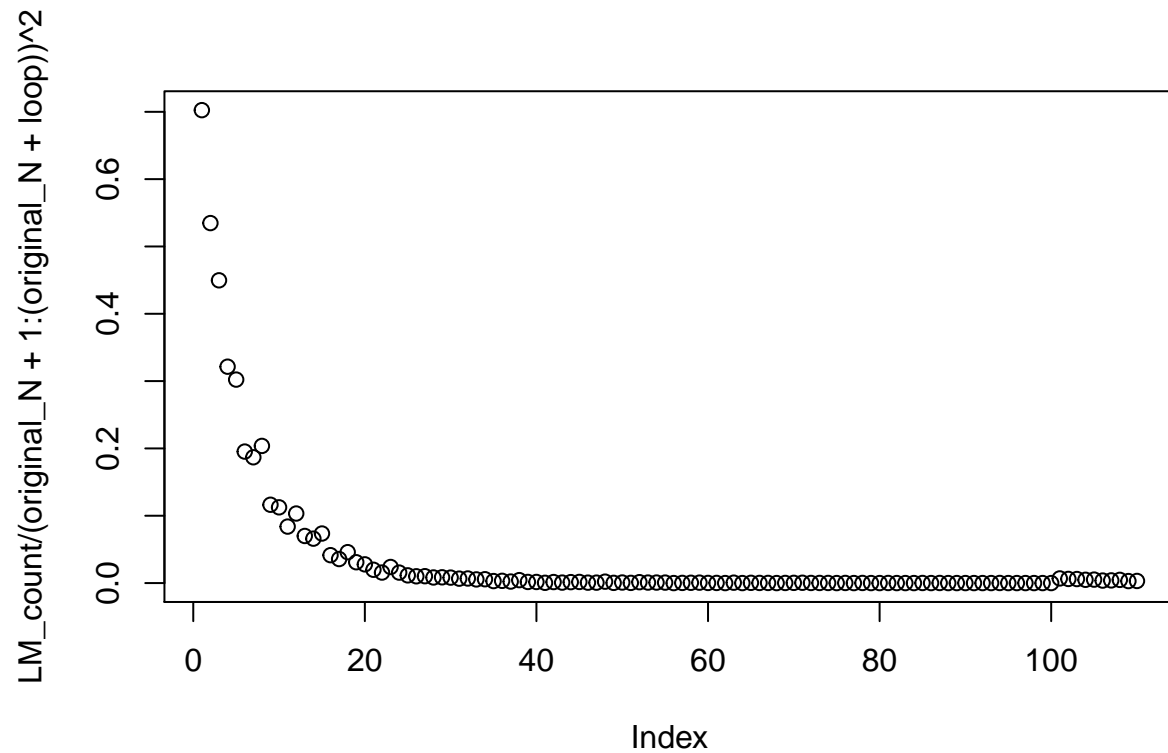
```
plot(LR_count/(original_N+1:(original_N+loop))^2)
```

```
## Warning in LR_count/(original_N + 1:(original_N + loop))^2: longer object
## length is not a multiple of shorter object length
```


if(ks.test(W,'pchisq',2)$p.value < signlevel)print($Wald\,do\,not\,follow\,chi-squre\,distribution$)$else$print($Wald\,do\,follow\,chi-squre\,distribution$)if(ks.test(LM,'pchisq',2)$p.value
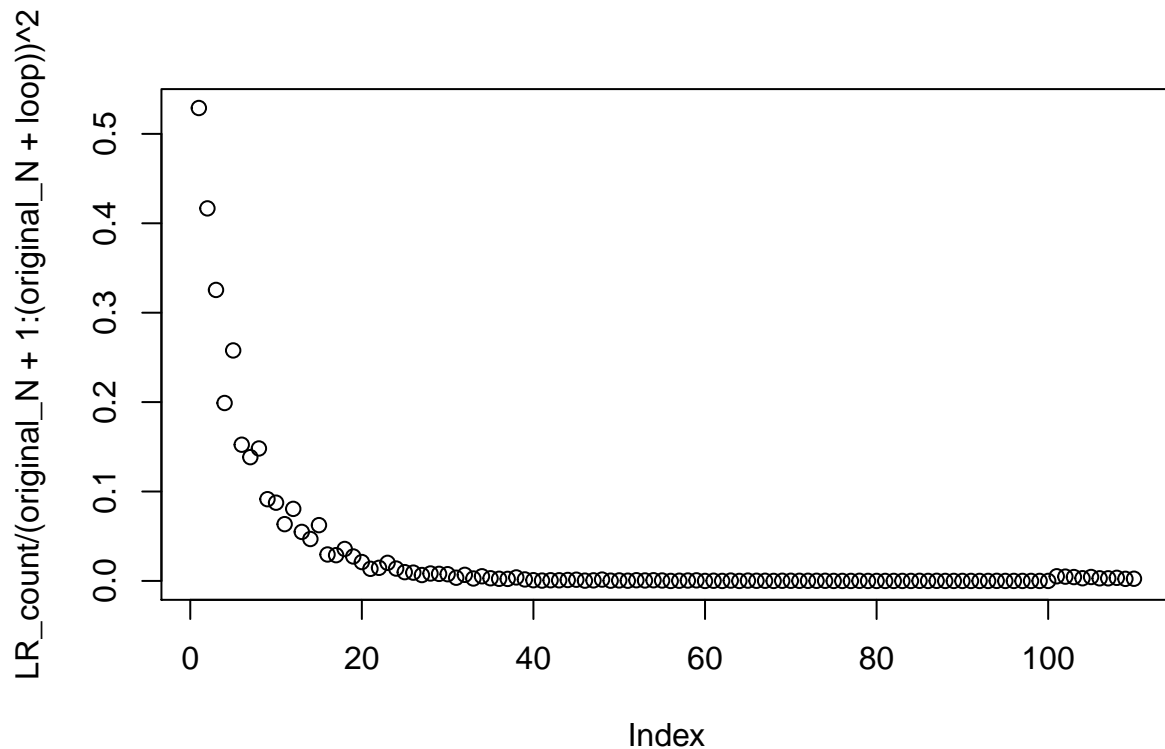
```
plot(LM_count/(original_N+1:(original_N+loop))^2)
```

Warning in LM_count/(original_N + 1:(original_N + loop))^2: longer object
length is not a multiple of shorter object length


```
plot(LR_count/(original_N+1:(original_N+loop))^2)
```

Warning in LR_count/(original_N + 1:(original_N + loop))^2: longer object
length is not a multiple of shorter object length

1.4 Question 5

For the data set pbp.csv, suppose Equation (2) is the true model. Use proper bootstrapped errors from the true model to study whether different test statistics for H0 : $\beta_1 + \beta_2 = 1$ in the previous questions follow chi-squared distribution. Explain your results.

---

reject null, make type I error

bootstrap,

homoscedacity two types of bootstrap robust test bootstrap for ϵ_i different performance

t-distribution ˆ2 => F distribution single β->t-test

$y_i = \beta_0 + \beta_1(x_1 - x_2) + \theta x_2 + \epsilon_i$ $H_0 : \theta = \beta_1 + \beta_2$ use t test for θ is as same with f-test with β_1, β_2

```r
require(lmtest)
require(MASS)
require(stats)
##boost up: translate programme language code into Byte-code.
require(compiler)
enableJIT(3)
```

```
## [1] 3
```

```r
##boost up-end for continues
#assumption part

loop=50
#Warning: the loop time cannot be larger any more;please forgive me, this all my Macbook fault. And the
```

```r
#initial valueset
original_N=10
signlevel=0.05

theta_count=rep(0,loop)

#
# for loop start:Monte Carlo
for(j in 1:loop){
theta=rep(0,loop)
N=original_N+j
for (i in 1:loop){
#generation part:data
s=sample(1:length(pbp$y),N,replace = 1)
question5.y<-pbp$y[s]
question5.x1<-pbp$x1[s]
question5.x2<-pbp$x2[s]
ee=rnorm(N,mean=0,sd=sqrt(var(question5.y)))
#epsilon=ee/sqrt(x1)
#y=beta_0+beta_1*x1+(1-beta_1)*x2+sqrt(x1)*epsilon
#generation part:regression
question5.equ1<-lm(question5.y~question5.x1+question5.x2)
#question5.equ2<-lm((question5.y-question5.x2)~(question5.x1-question5.x2))
#calculation
##pre-cal:heteroscedasticity
if (bptest(question5.equ1)$p.value<signlevel){
  question5.equ1<-lm(question5.y~question5.x1+question5.x2,weights=(1/question5.x1^2))
  #question5.equ2<-lm((question5.y-question5.x2)~(question5.x1-question5.x2),weights=(1/question5.x1^0..
}
#calc

theta[i]=question5.equ1$coefficients[2]+question5.equ1$coefficients[3]

}
if(t.test(theta,rep(1,loop))$p.value<signlevel){theta_count[j]=theta_count[j]+1}
}

plot(theta_count/(N+1:(N+loop)^2))
```