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 ansewer

$$\beta_1 + \beta_2 = 1$$

<=>

$$R * \beta = 1,$$

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

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). homoscedaticity

if under homoscedasticity,$\tilde{U_N}$ can be estimated as

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

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

and Wald statistic can be simplified as

$$Wald = (SSR_R - SSR_U)/\hat{\sigma}^2$$

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

if under homoscedasity, the LM statistic can be estimated as $LM = N * (SSR_R - SSR_U)/SSR_R$

$\tilde{U_N}$ 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 I 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?

Answer:

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.

if equation has heteroscedasticity, the Wald test and LM test should be generated by the general form:

Wald statistics: $$(1/N)(R\tilde{\beta} - I)'(R(X'X)^{-1}\tilde{U_N}(X'X)^{-1}R')^{-1}(R\tilde{\beta} - 1)$$
LM statistics: $$(1/N)\tilde{\lambda}'\Lambda^{-1}\tilde{\lambda}$$
where $\Lambda = 4(RM_N^{-}1R') - 1RM_N^{-}1U_N M_N^{-}1R'(RM_N^{-}1R')^{-}1$

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

```r
pbp=read.csv("/Users/sn0wfree/Dropbox/PhD_1st_study/BST169_Econometrics/Crousework_Project/pbp.csv")
#head(pbp)
#str(pbp)
signlevel=0.05
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
```

```r
equ1<-lm(y~x1 + x2,data=pbp)
equ2<-lm((y-x2)~(x1-x2), data=pbp)
if (bptest(resid(equ1)^2~pbp$x1*pbp$x2+pbp$x1^2+pbp$x2^2)$p.value<signlevel){
equ1<-lm(y~x1+x2,weights=1/sqrt(x1),data=pbp)
equ2<-lm(I(y-x2)~I(x1-x2),weights=1/sqrt(x1),data=pbp)
}
#Breusch-Pagan-Godfrey Test
bptest(equ1)
```

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

```r
bptest(equ2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  equ2
## BP = 41.139, df = 1, p-value = 1.418e-10
```

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

```
##
##  studentized Breusch-Pagan test
##
## data:  residuals(equ1)^2 ~ x1 + x2 + x1 * x2 + (x1)^2 + (x2)^2
## BP = 12.853, df = 3, p-value = 0.004966
```

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

```
##
##  studentized Breusch-Pagan test
##
## data:  residuals(equ2)^2 ~ (x1 - x2) + (x1 - x2)^2
## BP = 1.2219, df = 1, p-value = 0.269
```

```r
#if heteroscedasticity
N=length(pbp$y)

beta=matrix(equ1$coefficients)
R=cbind(0,1,1)
r=cbind(1,0,0)
X=cbind(rep(1,length(pbp$y)),pbp$x1,pbp$x2)
residual=matrix(resid(equ1))
```

```
#U_N
U_tilde=matrix(0,3,3)
len=length(pbp$y)
for(i in 1:len){U_tilde=U_tilde+residual[i,]^2*X[i,]%*%t(X[i,])}
U_tilde=U_tilde/len
#U_q
U_q=matrix(0,3,3)
ee=rt(len,6)
est_y=equ2$coefficients[1]+equ2$coefficients[2]*(pbp$x1-pbp$x2)+ee+pbp$x2
temp_y=matrix(est_y-mean(est_y))
for(i in 1:len){U_q=U_q+temp_y[i,]^2*X[i,]%*%t(X[i,])}
U_q=U_q/len
#
Wald=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_tilde%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1)
LM=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_q%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1)
```

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 WLS to estimate 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 $x_{2,i}$~$U(0,10)$. 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 $\beta_1 = 0.4$, $\beta_0 = 1$, and a set x_1

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

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

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

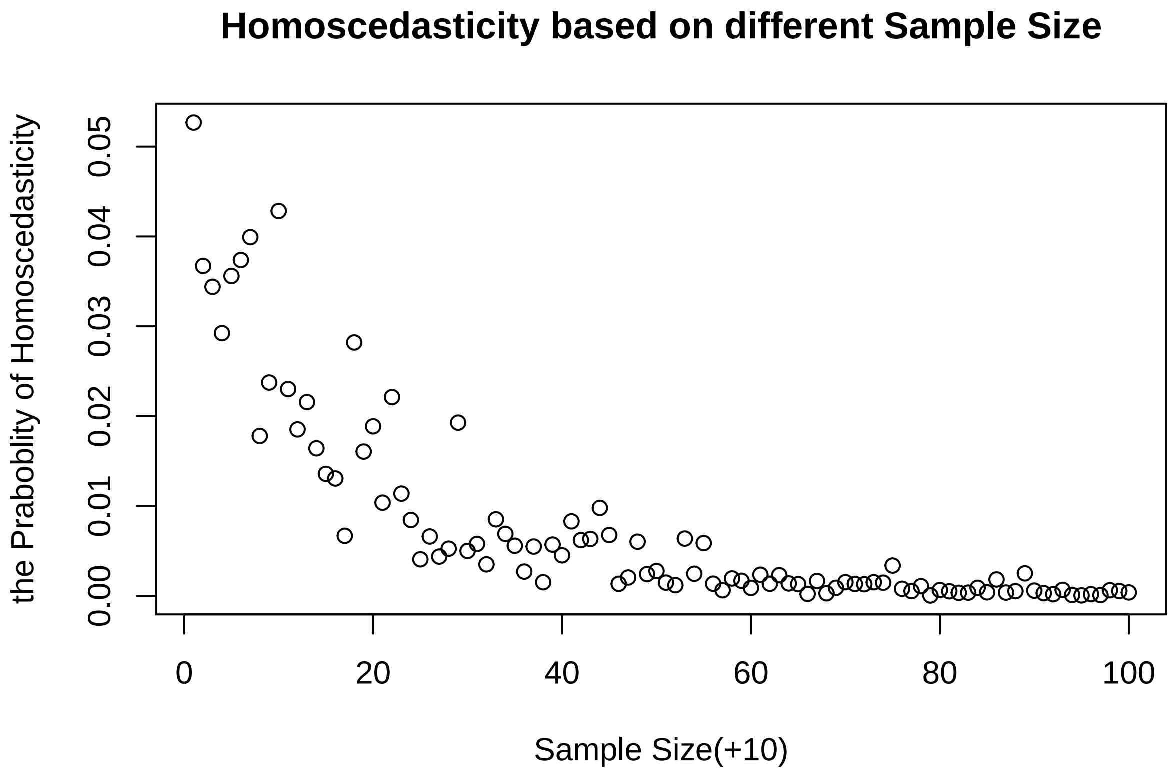
```
## Loading required package: compiler
```

```
enableJIT(3)
```

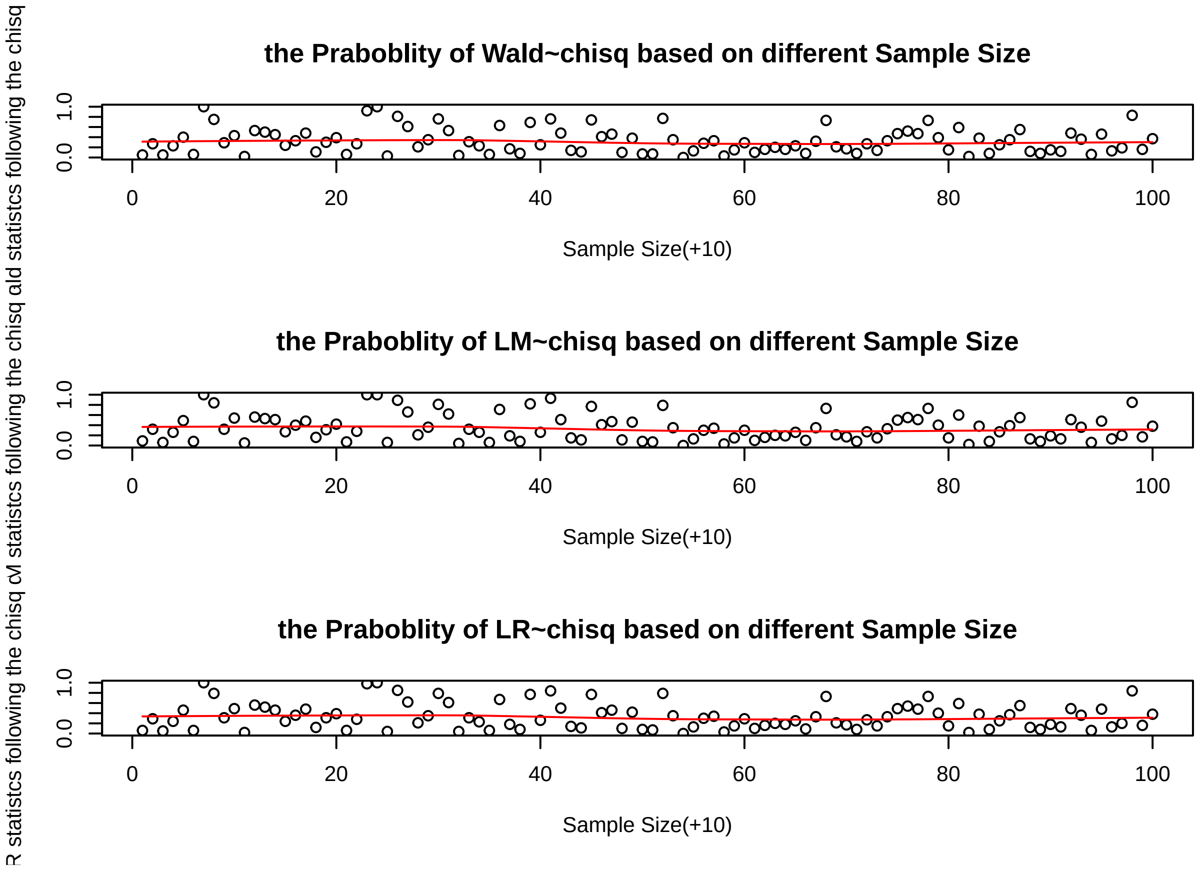
```
## [1] 0
```

```r
##boost up-end for continues
#set seed
#set.seed(2112)
#assumption part
loop=100
#I have a multiplication factor:1, which means when you set loop=N,
#It will generate N different (increased) sample size, and for each sample will do N*10 times Monte Car
#be careful your settings, your computer may explode.
#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

#initial container for Wald, LM, LR
W_count=rep(0,loop)
LM_count=rep(0,loop)
LR_count=rep(0,loop)
P.value_homo_container=rep(0,loop)
P.value_W_chisq_container=rep(0,loop)
P.value_LM_chisq_container=rep(0,loop)
# for loop start:Monte Carlo
for(j in 1:loop){#first for-loop for generating multi-sample
W=0
LM=0
LR=0
N=original_N+j
#generation part:data
x1=rchisq(N, 2)
x2=runif(N,0,10)
for (i in 1:loop){# second for-loop: the main Monte Carlo code
e=rt(N,6)
U_q=matrix(0,3,3)
U_tilde=matrix(0,3,3)
R=cbind(0,1,1)
r=cbind(1,0,0)
X=cbind(rep(1,N),x1,x2)
y=beta_0+beta_1*x1+(1-beta_1)*x2+sqrt(x1)*e
#generation part:regression
equ1<-lm(y~x1+x2)
equ2<-lm(I(y-x2)~I(x1-x2))
#calculate beta and residual
#beta=matrix(equ1$coefficients)
#residual=matrix(resid(equ1))
#calc SSR and Wald,LM, and LR
#U_N
#
#for(i in 1:N){U_tilde=U_tilde+residual[i,]^2*X[i,]%*%t(X[i,])}
#U_tilde=U_tilde/N
```

```
#U_q
#est_y=equ2$coefficients[1]+equ2$coefficients[2]*(x1-x2)+sqrt(x1)*e+x2
#temp_y=matrix(est_y-mean(est_y))
#for(i in 1:N){U_q=U_q+temp_y[i,]^2*X[i,]%*%t(X[i,])}
#U_q=U_q/N
#calculate Wald and LM
SSRu=sum(residuals(equ1)^2)
SSRr=sum(residuals(equ2)^2)
#W[j]=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_tilde%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1
#LM[j]=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_q%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1)
W[i]=N*((SSRr-SSRu)/(SSRu))
LM[i]=N*((SSRr-SSRu)/(SSRr))
LR[i]=N*(log(SSRr/SSRu))

#if (bptest(equ1,studentize = 0)$p.value<signlevel){P.value_homo_container[j]=P.value_homo_container[j]
P.value_homo_container[j]=P.value_homo_container[j]+bptest(equ1,studentize = 0)$p.value
P.value_W_chisq_container[j]=P.value_W_chisq_container[j]+ks.test(W,'pchisq',1)$p.value
#P.value_LM_chisq_container[j]=P.value_LM_chisq_container[j]+ks.test(LM,'pchisq',1)$p.value
if(ks.test(W,'pchisq',1)$p.value>signlevel){W_count[j]=W_count[j]+1}
if(ks.test(LM,'pchisq',1)$p.value>signlevel){LM_count[j]=LM_count[j]+1}
if(ks.test(LR,'pchisq',1)$p.value>signlevel){LR_count[j]=LR_count[j]+1}

}

}
plot(P.value_homo_container/(loop*10), xlab = "Sample Size(+10)",ylab = "the Praboblity of Homoscedasti
```



```
#plot(P.value_W_chisq_container/(loop*10), xlab = "Sample Size(+10)",ylab = "the P-value of Wald statis
#plot(P.value_LM_chisq_container/(loop*10),xlab = "Sample Size(+10)",ylab = "the P-value of LM statistc
par(mfrow=c(3,1))
plot(W_count/(loop),xlab = "Sample Size(+10)",ylab = "the Praboblity of Wald statistcs following the chi
points(lowess(W_count/(loop)),type="l",col="red")
plot(LM_count/(loop),xlab = "Sample Size(+10)",ylab = "the Praboblity of LM statistcs following the chi
points(lowess(LM_count/(loop)),type="l",col="red")
plot(LR_count/(loop),xlab = "Sample Size(+10)",ylab = "the Praboblity of LR statistcs following the chi
points(lowess(LR_count/(loop)),type="l",col="red")
```



From Homoscedasticity plot , I can find when the size of sample increases, the probablity of the Homoscedasticity of equ1 will decrease.

Thus, there exist the heteroscedacity issue to make the Wald and LM test invalid.

From these Praboblity of Wald$_{\text{chisq/LM}}$chisq plots, I can find the distribution of p-value of ks.test of each statistcs. And they both own decreasing trend depend on sample size. and more importantly, the p-value of Wald and LM statistics are almost less than the 5% of signiifcant level, which means the Wald and LM statistics are invalid. Thus, I should correct the model.

I choose WLS to elimate the heteroscedacity. However, I should seek a appropriate weigths to procedure the WLS. normally choose the Inverse of independent variable with m power as weights. But here, i choose $1/sqrt(x1)$

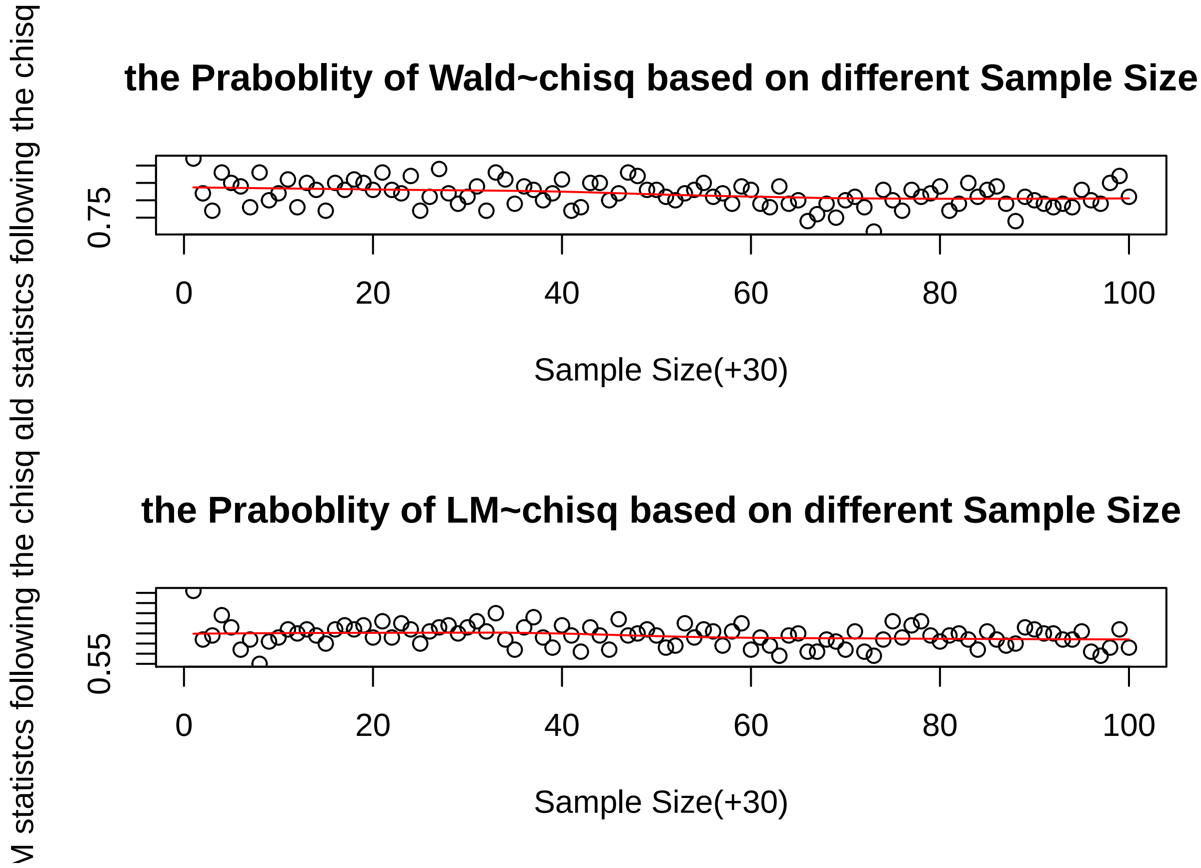
```
require(lmtest)
require(MASS)



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

```r
enableJIT(3)
```

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

```r
##boost up-end for continues
#set seed
set.seed(2112)
#assumption part

loop=100
m=1#I have a multiplication factor:1, which means when you set loop=N,
#It will generate N different (increased) sample size, and for each sample will do N*10 times Monte Car
#be careful your settings, your computer may explode.
#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=30
signlevel=0.05

#initial container for Wald, LM, LR
W_count=rep(0,loop)
LM_count=rep(0,loop)
LR_count=rep(0,loop)
P.value_homo_container=rep(0,loop)
P.value_W_chisq_container=rep(0,loop)
P.value_LM_chisq_container=rep(0,loop)

# for loop start:Monte Carlo

for(j in 1:loop){#first for-loop for generating multi-sample
W=0
LM=0
LR=0
N=original_N+j

#generation part:data
x1=rchisq(N, 2)
x2=runif(N,0,10)

for (i in 1:loop*m){# second for-loop: the main Monte Carlo code
e=rt(N,6)
U_q=matrix(0,3,3)
U_tilde=matrix(0,3,3)
R=cbind(0,1,1)
r=cbind(1,0,0)
X=cbind(rep(1,N),x1,x2)
y=beta_0+beta_1*x1+(1-beta_1)*x2+sqrt(x1)*e
#generation part:regression
equ1<-lm(y~x1+x2,weights=1/(x1^.5))
equ2<-lm(I(y-x2)~I(x1-x2),weights=1/(x1^.5))
```

```r
#heter
if (bptest(resid(equ1)^2~x1*x2+x1^2+x2^2)$p.value<signlevel){
equ1<-lm(y~x1+x2,weights=1/sqrt(x1))
equ2<-lm(I(y-x2)~I(x1-x2),weights=1/sqrt(x1))
}
#calculate beta and residual
beta=matrix(equ1$coefficients)
residual=matrix(resid(equ1))
#calc SSR and Wald,LM, and LR
#U_N
#
for(i in 1:N){U_tilde=U_tilde+residual[i,]^2*X[i,]%*%t(X[i,])}
U_tilde=U_tilde/N
#U_q
est_y=equ2$coefficients[1]+equ2$coefficients[2]*(x1-x2)+sqrt(x1)*e+x2
temp_y=matrix(est_y-mean(est_y))
for(i in 1:N){U_q=U_q+temp_y[i,]^2*X[i,]%*%t(X[i,])}
U_q=U_q/N
#calculate Wald and LM
W[j]=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_tilde%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1)
LM[j]=(1/N)*t(R%*%beta-1)%*%solve(R%*%solve(t(X)%*%X)%*%U_q%*%solve(t(X)%*%X)%*%t(R))%*%(R%*%beta-1)
#SSRu=sum(residuals(equ1)^2)
#SSRr=sum(residuals(equ2)^2)

#W[i]=N*((SSRr-SSRu)/(SSRu))
#LM[i]=N*((SSRr-SSRu)/(SSRr))
#LR[i]=N*(log(SSRr/SSRu))


#if (bptest(equ1,studentize = 0)$p.value<signlevel){P.value_homo_container[j]=P.value_homo_container[j]
P.value_homo_container[j]=P.value_homo_container[j]+bptest(equ1,studentize = 0)$p.value
P.value_W_chisq_container[j]=P.value_W_chisq_container[j]+ks.test(W,'pchisq',1)$p.value
P.value_LM_chisq_container[j]=P.value_LM_chisq_container[j]+ks.test(LM,'pchisq',1)$p.value
if(ks.test(W,'pchisq',1)$p.value>signlevel){W_count[j]=W_count[j]+1}
if(ks.test(LM,'pchisq',1)$p.value>signlevel){LM_count[j]=LM_count[j]+1}
#if(ks.test(LR,'pchisq',1)$p.value>signlevel){LR_count[j]=LR_count[j]+1}
gc()
}

}

#plot(P.value_homo_container/(loop*m), xlab = "Sample Size(+10)",ylab = "the P-value of Homoscedasticit
#plot(P.value_W_chisq_container/(loop*m), xlab = "Sample Size(+10)",ylab = "the P-value of Wald statist
#plot(P.value_LM_chisq_container/(loop*m),xlab = "Sample Size(+10)",ylab = "the P-value of LM statistcs
par(mfrow=c(2,1))
plot(W_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of Wald statistcs following the
points(lowess(W_count/(loop*m)),type="l",col="red")
plot(LM_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of LM statistcs following the c
points(lowess(LM_count/(loop*m)),type="l",col="red")
```



```
#plot(LR_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of LR statistcs following the
```

From these plots, after corrected heteroscedacity issue, I can find that the Wald and LM statistics are valid, all the p-value of ks.test of each test statistics are over 5%, which means the these test statistics follow the chi-squared distribution.that is consistent with the definition of their own.

1.4 Question 4

Compare the size of different test statistics (frequencies of making Type 1 error) from Monte Carlo using 5% level of significance for different sample sizes. Explain the results.

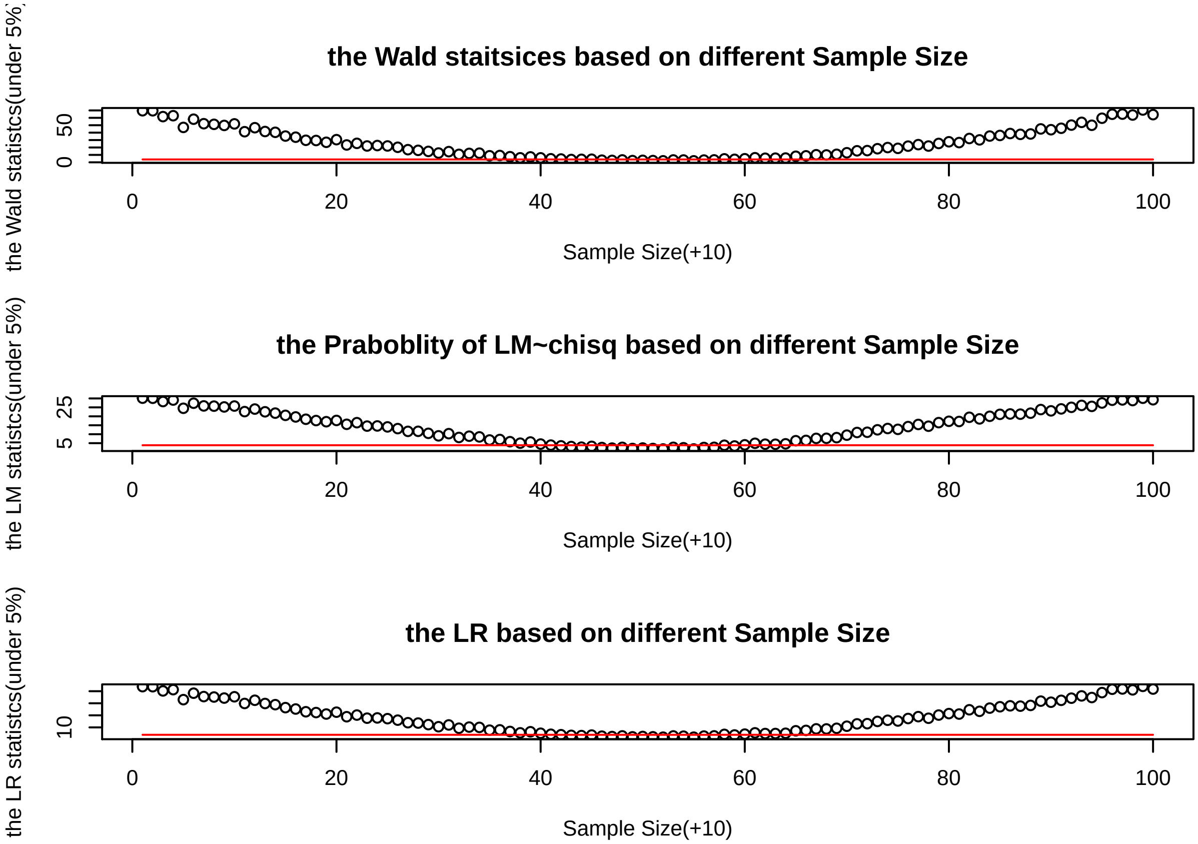
```
require(lmtest)
require(MASS)


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

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

```
##boost up-end for continues
#set seed
set.seed(2112)
#assumption part

loop=100
```

```r
m=1#I have a multiplication factor:1, which means when you set loop=N,
#It will generate N different (increased) sample size, and for each sample will do N*10 times Monte Car
#be careful your settings, your computer may explode.
#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

#initial container for Wald, LM, LR
W=rep(0,loop)
LM=rep(0,loop)
LR=rep(0,loop)
W_count=rep(0,loop)
LM_count=rep(0,loop)
LR_count=rep(0,loop)
P.value_homo_container=rep(0,loop)
P.value_W_chisq_container=rep(0,loop)
P.value_LM_chisq_container=rep(0,loop)
theta=rep(seq(0.25,1.75,len=loop))
crv=rep(qchisq(0.95,1),loop)





# for loop start:Monte Carlo
#for(j in 1:loop){#first for-loop for generating multi-sample



#generation part:data


#U_q=matrix(0,3,3)
#U_tilde=matrix(0,3,3)
#R=cbind(0,1,1)
#r=cbind(1,0,0)
#X=cbind(rep(1,N),x1,x2)

for(j in 1:loop){
N=original_N+j
x1=rchisq(N, 2)
x2=runif(N,0,10)

for (i in 1:loop*m){# second for-loop: the main Monte Carlo code
e=rnorm(N,0,1)
y=beta_0+beta_1*x1+(1-beta_1)*x2+sqrt(x1)*e
#generation part:regression
```

```
#true_equ2<-lm(I(y-x2)~I(x1-x2),weights=1/sqrt(x1))
equ1<-lm(y~x1+x2,weights=1/sqrt(x1))
equ2<-lm(I(y-theta[i]*x2)~I(x1-x2),weights=1/sqrt(x1))

SSRu=sum(residuals(equ1)^2)
SSRr=sum(residuals(equ2)^2)
#true_SSRr=sum(residuals(true_equ2)^2)
W[i]=W[i]+N*((SSRr-SSRu)/(SSRu))
LM[i]=LM[i]+N*((SSRr-SSRu)/(SSRr))
LR[i]=LR[i]+N*(log(SSRr/SSRu))

#LR[i]=N*(log(SSRr/SSRu))
W_count[j]=W_count[j]+(mean(W)>qchisq(0.95,1))
LM_count[j]=LM_count[j]+(mean(LM)>qchisq(0.95,1))
LR_count[j]=LR_count[j]+(mean(LR)>qchisq(0.95,1))
}

}
#W_count=W_count+(ks.test(W,'pchisq',1)$p.value>signlevel)
#LM_count=LM_count+(ks.test(LM,'pchisq',1)$p.value>signlevel)
par(mfrow=c(3,1))
plot(W/(loop*m),xlab = "Sample Size(+10)",ylab = "the Wald statistcs(under 5%)",main="the Wald staitsic
points(crv,type="l",col="red")

plot(LM/(loop*m),xlab = "Sample Size(+10)",ylab = "the LM statistcs(under 5%)",main="the Praboblity of I
#points(lowess(LM/(loop*m)),type="l",col="red")
points(crv,type="l",col="red")
plot(LR/(loop*m),xlab = "Sample Size(+10)",ylab = "the LR statistcs(under 5%)",main="the LR based on di
points(crv,type="l",col="red")
```



```
#plot(W_count/(loop*m),xlab = "Sample Size(+10)",ylab = "the Wald statistcs following the chisq distrib
#points(crv,type="l",col="red")

#plot(LM/(loop*m),xlab = "Sample Size(+10)",ylab = "the LM statistcs following the chisq distribution(u
#points(lowess(LM/(loop*m)),type="l",col="red")
#points(crv,type="l",col="red")
#plot(LR/(loop*m),xlab = "Sample Size(+10)",ylab = "the LR statistcs following the chisq distribution(u
#points(crv,type="l",col="red")
```

from these test statistics plots, I can find that after sample size=50, the wald, LM and LR statistcs will increase by the sample size. And before 50, will decrease by the sample size.

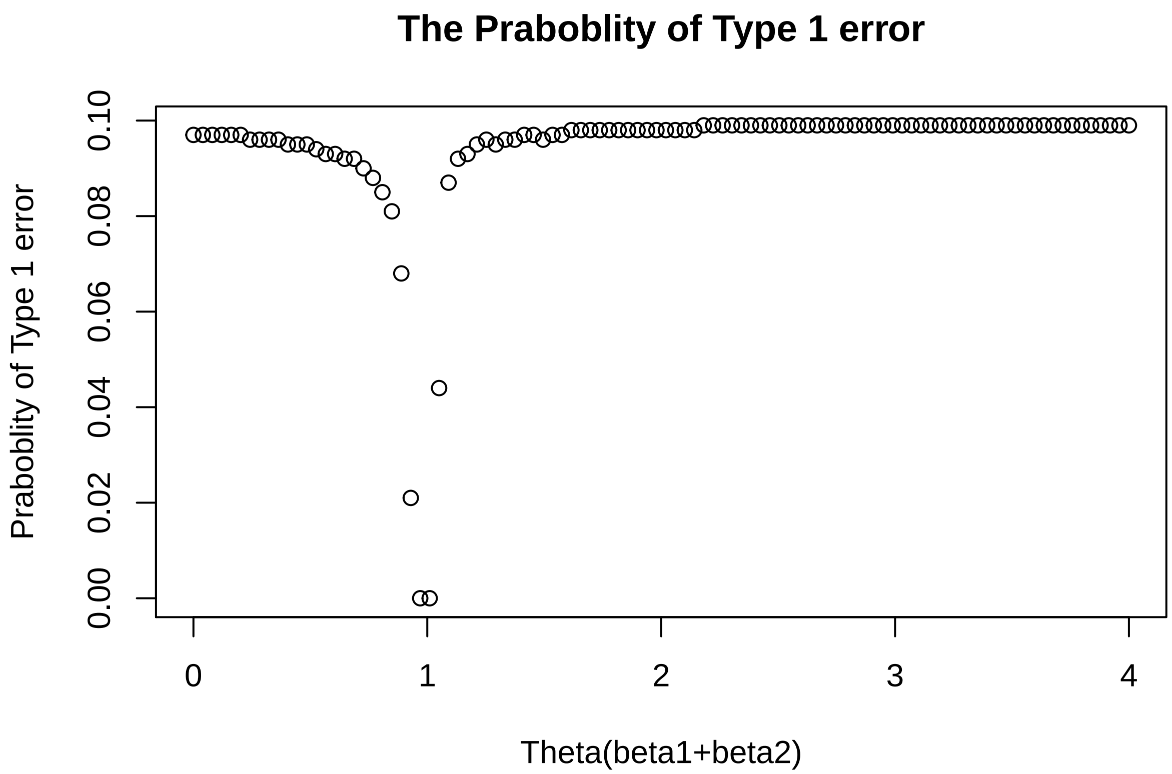
I aslo do the t test for $\beta_1 + \beta_2 = 1$ to show the type I error.

```
require(lmtest)
require(MASS)

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

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

```
##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
#initial valueset
```

```
original_N=100
signlevel=0.05
theta_count=rep(0,loop)
# for loop start:Monte Carlo
for(j in 1:loop){#first for-loop for generating multi-sample
theta=0
N=original_N+j
#generation part:data
s=sample(1:length(pbp$y),N,replace = 1)
question.pre.y<-pbp$y[s]
question.pre.x1<-pbp$x1[s]
question.pre.x2<-pbp$x2[s]
pre_equ<-lm(I(question.pre.y-question.pre.x2)~I(question.pre.x1-question.pre.x2))
theta_test=rep(seq(0,4,len=loop))
for (i in 1:loop*10){
question5.residuals=sample(residuals(pre_equ),N,replace=1)
question5.y=pre_equ$coefficients[1]+pre_equ$coefficients[2]*question.pre.x1+(1-pre_equ$coefficients[2])
#generation part:regression
question5.equ1<-lm(question5.y~question.pre.x1+question.pre.x2)
#residuals.question5.equ1=resid(question5.equ1)
#question5.equ1<-lm((question5.y-question5.x2)~(question5.x1-question5.x2))
#calculation
##pre-cal:heteroscedasticity
#if(bptest(residuals.question5.equ1^2~question5.x1*question5.x2+question5.x1^2+question5.x2^2)$p.value<
#  question5.equ1<-lm(question5.y~question5.x1+question5.x2,weights=(1/question5.x1^0.5))
  #question5.equ1<-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(length(theta)>2){if(t.test(theta,mu=1)$p.value>signlevel){theta_count[j]=theta_count[j]+1}}
theta_count[j]=theta_count[j]+(t.test(theta,mu=theta_test[j])$p.value<0.05)
}
}


plot(theta_test,theta_count/(loop*10),xlab="Theta(beta1+beta2)",ylab="Praboblity of Type 1 error",main=
```



form graph, I can find that the Theta which equals to $\beta_1 + \beta_2 = 1$, when the type I error will dramatically increase when theta has a value different from true 1.

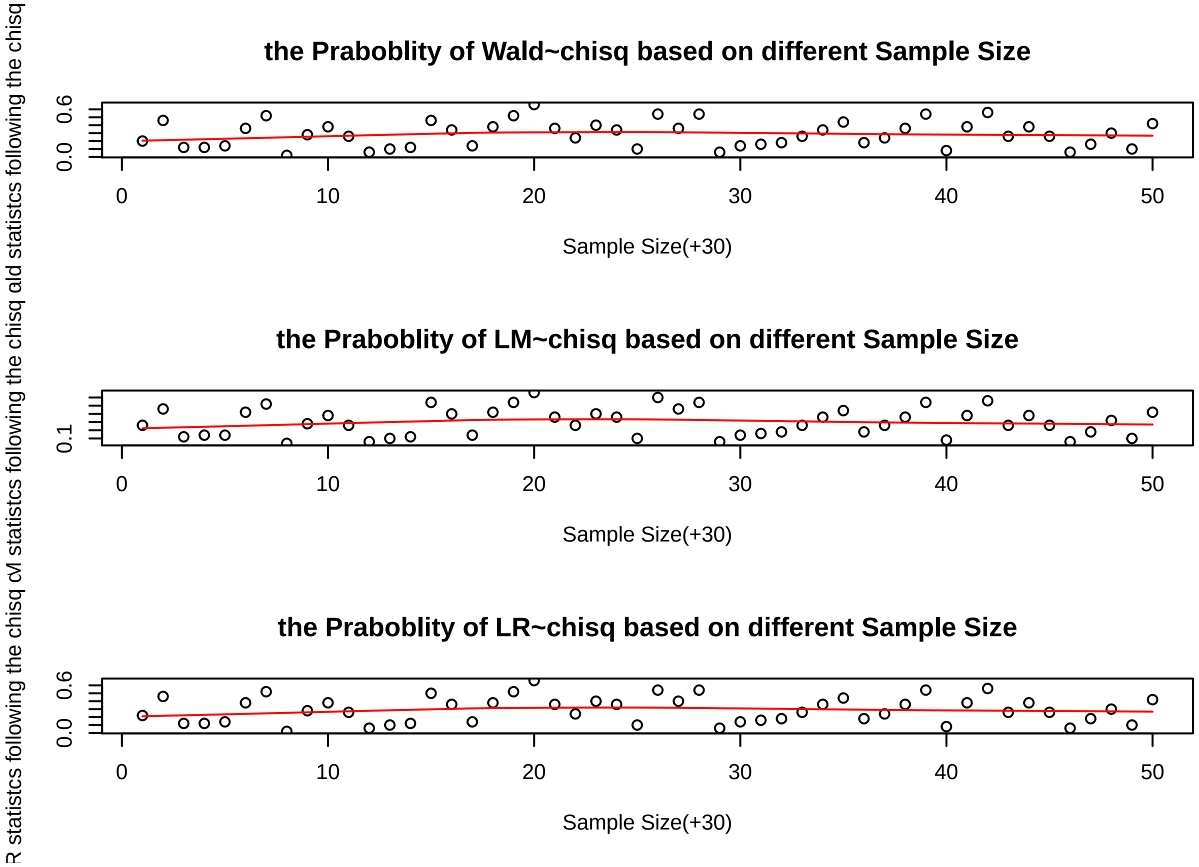
1.5 Question 5

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.

```
pbp=read.csv("/Users/sn0wfree/Dropbox/PhD_1st_study/BST169_Econometrics/Crousework_Project/pbp.csv")
loop=50
m=1#I have a multiplication factor:1, which means when you set loop=N,
#It will generate N different (increased) sample size, and for each sample will do N*10 times Monte Car
#be careful your settings, your computer may explode.
#Warning: the loop time cannot be larger any more;please forgive me, this all my Macbook fault. And the


#initial valueset
original_N=30
signlevel=0.05

#initial container for Wald, LM, LR
W_count=rep(0,loop)
LM_count=rep(0,loop)
LR_count=rep(0,loop)
P.value_homo_container=rep(0,loop)
P.value_W_chisq_container=rep(0,loop)
P.value_LM_chisq_container=rep(0,loop)
```

```
# for loop start:Monte Carlo

for(j in 1:loop){#first for-loop for generating multi-sample
W=0
LM=0
LR=0
N=original_N+j



for (i in 1:loop*m){# second for-loop: the main Monte Carlo code
s=sample(1:length(pbp$y),N,replace = 1)
question5.y<-pbp$y[s]
question5.x1<-pbp$x1[s]
question5.x2<-pbp$x2[s]
#generation part:data
question5_equ2<-lm(I(question5.y-question5.x2)~I(question5.x1-question5.x2),weights=1/sqrt(question5.x1
#generation part:regression
question5_equ1<-lm(question5.y~question5.x1+question5.x2,weights=1/(question5.x1^.5))

#calculate beta and residual
#beta=matrix(equ1$coefficients)
#residual=matrix(resid(equ1))
#calc SSR and Wald,LM, and LR

SSRu=sum(residuals(question5_equ1)^2)
SSRr=sum(residuals(question5_equ2)^2)

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


#if (bptest(equ1,studentize = 0)$p.value<signlevel){P.value_homo_container[j]=P.value_homo_container[j]
P.value_homo_container[j]=P.value_homo_container[j]+bptest(equ1,studentize = 0)$p.value
P.value_W_chisq_container[j]=P.value_W_chisq_container[j]+ks.test(W,'pchisq',1)$p.value
P.value_LM_chisq_container[j]=P.value_LM_chisq_container[j]+ks.test(LM,'pchisq',1)$p.value
if(ks.test(W,'pchisq',1)$p.value>signlevel){W_count[j]=W_count[j]+1}
if(ks.test(LM,'pchisq',1)$p.value>signlevel){LM_count[j]=LM_count[j]+1}
if(ks.test(LR,'pchisq',1)$p.value>signlevel){LR_count[j]=LR_count[j]+1}
gc()
}

}

#plot(P.value_homo_container/(loop*m), xlab = "Sample Size(+10)",ylab = "the P-value of Homoscedasticit
#plot(P.value_W_chisq_container/(loop*m), xlab = "Sample Size(+10)",ylab = "the P-value of Wald statist
#plot(P.value_LM_chisq_container/(loop*m),xlab = "Sample Size(+10)",ylab = "the P-value of LM statistcs
par(mfrow=c(3,1))
plot(W_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of Wald statistcs following the
points(lowess(W_count/(loop*m)),type="l",col="red")
```

```
plot(LM_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of LM statistcs following the cl
points(lowess(LM_count/(loop*m)),type="l",col="red")
plot(LR_count/(loop*m),xlab = "Sample Size(+30)",ylab = "the Praboblity of LR statistcs following the cl
points(lowess(LR_count/(loop*m)),type="l",col="red")
```



I use the WLS to estmate the model, which driectly avoid the potential heteroscedacity. From the plots, I can find the each of these three test statistics (in red line) show a convex function, which means when the sample size increase the statistics will increase, which suggest me that the statistics will follow chi-squred disrtribution. that is consistent with the definition.