Homework 4--R computing for Business Data Analytics

好棒棒:葉早彬、陳威宇、劉瑞祥

Q1. Import the library AER in R, and attach the data set CPS1988.

(a) Run the linear regression model below (using lm( )) and save the model as object “CPS_lm”.Note that the ethnicity is a categorical/dummy variable.

```
 library("AER")
 data("CPS1988")
 attach(CPS1988)
experience2=experience*experience
CPS_lm=lm(log(wage)~experience+experience2+education+as.factor(ethnicity))
```

(b) Explain the results in detail. What is the statistical significance of each independent variable? What are the implications of the findings? Particularly, what is the association between wage and experience? Is the identified association linear? If not, what is the shape of the association?

What is the statistical significance of each independent variable?

experience,$expreience^{2$},education,ethnicity的p-value都小於0.001,因此這幾個獨立變數都是統計顯著。

What are the implications of the findings? Particularly, what is the association between wage and experience?

wage與experience,$expreience^2$
$, education, ethnicity$都是指數成長的關係, 其中$experience, education$對$wage$都是正面的影響, expreience2$則是負面影響,

ethnicity為afam的話,wage成長的幅度會比較小。
experience越大,wage越大,兩者呈指數成長。

Is the identified association linear?

no,log(wage)與獨立變數之間是線性關係,所以wage與獨立變數之間不是線性

If not, what is the shape of the association?

指數關係

(c) Based on the estimated coefficients, write out the two equations of predictive models for Africa-American and Caucasian respectively.

$\beta_0 = 4.321395, \beta_1 = 0.077473, \beta_2 = -0.001316, \beta_3 = 0.085673, \beta_4 = -0.243364$

$if ethnicity = Africa American :$
$$wage = e^{4.078031+0.077473*experience-0.001316*experience^2+0.085673*education}$$

$if ethnicity = Caucasian :$
$$wage = e^{4.321395+0.077473*experience-0.001316*experience^2+0.085673*education}$$

Q2. Monte-Carlo simulation experiments of linear regression (OLS)

(a) Multicollinearity

answer:

```
library(mvtnorm)
set.seed(121402)
# Create two correlated independent variables
n=1000
b0=0.2
b1=0.5
b2=0.75
mclvls= seq(0, 0.95, 0.05)

simulate=function(n){

        b1.sds=c()
        for(i in 1:length(mclvls)){
                mclvl=mclvls[i]
                b1.estimate=c()
                for(j in 1:1000)
                {
                        x.corr=matrix(c(1, mclvl, mclvl, 1), ncol=2)
                        x=rmvnorm(n, mean=c(0, 0), sigma=x.corr) # n is the sample size
                        x1=x[ , 1]
                        x2=x[, 2]
                        y=b0+b1*x1+b2*x2+rnorm(n, 0, 1)
                        lm(y~x1+x2)
                        b1.estimate[j]=coef(lm(y~x1+x2))[2]
                }
                b1.sds[i]=sd(b1.estimate)
        }
        b1.sds
}

b1.1000=simulate(1000)
b1.5000=simulate(5000)


plot(mclvls,seq(0,0.1,0.1/19),type='n',ylab='the corresponding standard deviation of b1')
lines(mclvls,b1.1000,col='green')
```

```
lines(mclvls,b1.5000,col='red')
```

,b1的標準差越大,也就是說兩個變數的covariance的絕對值越接近1,兩者共線性的程度越明顯,regression的估計效果會越差。

(b) Omitted variable

Following the procedure in (a), for each mclvl in c(0, 0.5, 1), set n=1000 and simulate x1& x2. Then simulate the dependent variable using

```
> y=b0+b1*x1+b2*x2+rnorm(n, 0, 1) #set b0=0.2; b1=0.5; b2=0.75
```

b1 from lm(y_{x1}) – we intentionally omit x2 – and repeat the estimation for 1000 times (for a given mclvl). Save all of the estimated b1 and plot the three distributions of estimated b1 in each mclvl. Compare the distributions to the true b1=0.5. What is the impact of omitting x2? Discuss what you observe.

answer:

```
library(mvtnorm)
set.seed(121402)
# Create two correlated independent variables
mclvls= c(0,0.5,1)

omitted.plot=function(n){

        par.est=matrix(NA, nrow=n, ncol=3)

        for(j in 1:n)
        {

                for(i in 1:length(mclvls)){
                        mclvl=mclvls[i]
                        x.corr=matrix(c(1, mclvl, mclvl, 1), ncol=2)
                        x=rmvnorm(n, mean=c(0, 0), sigma=x.corr) # n is the sample size
                        x1=x[ , 1]
                        x2=x[, 2]
                        y=b0+b1*x1+b2*x2+rnorm(n, 0, 1)
                        model=lm(y~x1)
                        par.est[j, i]=model$coef[2]

                }
        }
        par.est


        plot(c(min(par.est),max(par.est)),c(0,15),main="",lwd=2,xlab='b1',ylab='density',ty
        lines(density(par.est[,3]),col= 'green', lwd=3, lty=3)
        lines(density(par.est[,2]),col= 'red', lwd=2, lty=2)
        lines(density(par.est[,1]),col= 'black', lwd=1, lty=1)
        abline(v=b1,col='blue')

        legend(0.9,15,c("mclvl=0","mclvl=0.5","mclvl=1"), lty=c(1,2,3),lwd=c(1,2,3),bty="n
}
omitted.plot(1000)
```

mclvl小,意味著x1,x2的共線性的程度小,線性迴歸得到的$\hat{b}1$會接近真實的b1。

mclvl大時,x1,x2的共線性程度大,線性迴歸得到的$\hat{b}1$與真實的b1的誤差變大,而且$\hat{b}1$越接近b1+b2。

(c)Measurement error

Run the following codes in R

```
> set.seed(385062) > n=1000
> x=runif(n, -1, 1)
```

each errlvl in c(0, 0.5, 1), generate x with measurement error > xp=x+rnorm(n, 0, errlvl)
Then repeat the following process for 1000 times. First simulate the dependent variable using

```
> y=b0+b1*x+rnorm(n, 0, 1) #set b0=0.2; b1=0.5
```

estimate b1 from OLS regression (lm( )) > lm(y_{xp})
Save the estimated b1 in each of the 1000 replications (for this given errlvl).
Plot the three distributions of estimated b1s for errlvl in c(0, 0.5, 1). Compare the distributions to the true b1=0.5. What's the impact of measurement errors? Discuss what you observe.

answer:

```
measurement.plot=function(){
        errlvls=c(0,0.5,1)
        set.seed(385062)
        n=1000
        b0=0.2
        b1=0.5
        x=runif(n, -1, 1)
        par.est=matrix(NA, nrow=n, ncol=3)


        for(i in 1:length(errlvls)){
                errlvl=errlvls[i]
                xp=x+rnorm(n, 0, errlvl)
                for(j in 1:n){
                        y=b0+b1*x+rnorm(n, 0, 1)
                        model=lm(y~xp)
                        par.est[j, i]=model$coef[2]
                }
        }
        par.est
        plot(c(min(par.est),max(par.est)),c(0,15),main="",lwd=2,xlab='b1',ylab='density',ty
        lines(density(par.est[,3]),col= 'green', lwd=3, lty=3)
        lines(density(par.est[,2]),col= 'red', lwd=2, lty=2)
        lines(density(par.est[,1]),col= 'black', lwd=1, lty=1)
        abline(v=b1,col='blue')

        legend(0.3,15,c("errlvl=0","errlvl=0.5","errlvl=1"), lty=c(1,2,3),lwd=c(1,2,3),bty=

}
measurement.plot()
```

,線性迴歸得到的$\hat{b1}$會接近真實的b1。

errlvl越大,線性迴歸得到的$\hat{b1}$與真實的b1的誤差越大。

所以跑線性迴歸前,要確認收集到的資料是乾淨的 :D