Econometrics_Problem_Set_7.R

Bernardo

Tue Oct 29 09:41:26 2019

```
library(readstata13)
```

```
## Warning: package 'readstata13' was built under R version 3.4.4
```

```
library(MASS)


kt <- read.dta13("kt_data.dta")

year10 <- subset(kt, year == 10)


firstreg <- lm(data = year10, wage ~ educ + I(exper^2) + abil + Fath_ed)
summary(firstreg)
```

```
##
## Call:
## lm(formula = wage ~ educ + I(exper^2) + abil + Fath_ed, data = year10)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.35884 -0.27775  0.04841  0.31240  1.35068
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.1304059  0.1271802   8.888  < 2e-16 ***
## educ        0.0743388  0.0083968   8.853  < 2e-16 ***
## I(exper^2)  0.0018780  0.0002808   6.689 3.15e-11 ***
## abil        0.0907044  0.0169678   5.346 1.04e-07 ***
## Fath_ed     0.0082859  0.0036598   2.264   0.0237 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4815 on 1515 degrees of freedom
## Multiple R-squared:  0.1323, Adjusted R-squared:   0.13
## F-statistic: 57.73 on 4 and 1515 DF,  p-value: < 2.2e-16
```

```
step1 <- lm(data = year10, wage ~ I(exper^2) + abil + Fath_ed)
step2 <- lm(data = year10, educ ~ I(exper^2) + abil + Fath_ed)

resid1 <- resid(step1)
resid2 <- resid(step2)

residuals <- as.data.frame(cbind(resid1, resid2))

secondreg <- lm(data = residuals, resid1 ~ resid2)
summary(secondreg)
```

```
##
```

```
## Call:
## lm(formula = resid1 ~ resid2, data = residuals)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.35884 -0.27775  0.04841  0.31240  1.35068
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.196e-17  1.234e-02   0.000        1
## resid2       7.434e-02  8.389e-03   8.862   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4811 on 1518 degrees of freedom
## Multiple R-squared:  0.04919,    Adjusted R-squared:  0.04856
## F-statistic: 78.53 on 1 and 1518 DF,  p-value: < 2.2e-16
```

```
thirdreg <- lm(data = year10, wage ~ I(exper^2) + abil + Fath_ed)
resid3 <- resid(thirdreg)

year10withresid <- cbind(year10, resid3)
fourthreg <- lm(data = year10, resid3 ~ educ)
summary(fourthreg)
```

```
##
## Call:
## lm(formula = resid3 ~ educ, data = year10)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.31302 -0.27680  0.04248  0.31269  1.49453
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.49961    0.08018  -6.231 5.99e-10 ***
## educ         0.03860    0.00612   6.308 3.70e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.487 on 1518 degrees of freedom
## Multiple R-squared:  0.02554,    Adjusted R-squared:  0.0249
## F-statistic: 39.79 on 1 and 1518 DF,  p-value: 3.696e-10
```

```
ttests <- vector()


## k = 1

alphaestimate <- vector()
betaestimate <- vector()
for (i in 1:1000) {

x <- rnorm(n = 100, mean = 1)
e <- rnorm(n = 100)
```

```r
f <- function(x) {
  1/sqrt(2*pi)*exp((-1/2)*x^2)
}

f2 <- function(x, k) {
  (x^k)*f(x)
}

f3 <- function(k) {
 integrate(f2, -Inf, Inf, k)

}

n <- vector()

for (i in 1:100) {
  expect <- f3(1)
  expect2 <- f3(2)

  expect <- as.numeric(unlist(expect)[1])
  expect2 <- as.numeric(unlist(expect2)[1])

  n[i] <- (e[i]^1 - expect)/(expect2 - expect^2)^(1/2)

}
y <- x + n
dataset <- as.data.frame(cbind(y, x, n))

model <- lm(data = dataset, y ~ x + n)

alphaestimate <- append(alphaestimate, as.numeric(unlist(model[1])[1]))
betaestimate <- append(betaestimate, as.numeric(unlist(model[1])[2]))

}
## ttests <- append(ttests, t.test(betaestimate, mu=1))
summary(betaestimate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
```

```r
## k = 2



alphaestimate <- vector()
betaestimate <- vector()
for (i in 1:1000) {

  x <- rnorm(n = 100, mean = 1)
  e <- rnorm(n = 100)

  f <- function(x) {
    1/sqrt(2*pi)*exp((-1/2)*x^2)
  }
```

```
  f2 <- function(x, k) {
    (x^k)*f(x)
  }

  f3 <- function(k) {
    integrate(f2, -Inf, Inf, k)

  }

  n <- vector()

  for (i in 1:100) {
    expect <- f3(2)
    expect2 <- f3(4)

    expect <- as.numeric(unlist(expect)[1])
    expect2 <- as.numeric(unlist(expect2)[1])

    n[i] <- (e[i]^2 - expect)/(expect2 - expect^2)^(1/2)

  }
  y <- x + n
  dataset <- as.data.frame(cbind(y, x, n))

  model <- lm(data = dataset, y ~ x + n)

  alphaestimate <- append(alphaestimate, as.numeric(unlist(model[1])[1]))
  betaestimate <- append(betaestimate, as.numeric(unlist(model[1])[2]))
}
##ttests <- append(ttests, t.test(betaestimate, mu=1))

summary(betaestimate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
```

```
## k = 4
```

```
alphaestimate <- vector()
betaestimate <- vector()
for (i in 1:1000) {

  x <- rnorm(n = 100, mean = 1)
  e <- rnorm(n = 100)

  f <- function(x) {
    1/sqrt(2*pi)*exp((-1/2)*x^2)
  }

  f2 <- function(x, k) {
    (x^k)*f(x)
```

```
  }

  f3 <- function(k) {
    integrate(f2, -Inf, Inf, k)

  }

  n <- vector()

  for (i in 1:100) {
    expect <- f3(4)
    expect2 <- f3(8)

    expect <- as.numeric(unlist(expect)[1])
    expect2 <- as.numeric(unlist(expect2)[1])

    n[i] <- (e[i]^4 - expect)/(expect2 - expect^2)^(1/2)

  }
  y <- x + n
  dataset <- as.data.frame(cbind(y, x, n))

  model <- lm(data = dataset, y ~ x + n)

  alphaestimate <- append(alphaestimate, as.numeric(unlist(model[1])[1]))
  betaestimate <- append(betaestimate, as.numeric(unlist(model[1])[2]))
}
##ttests <- append(ttests, t.test(betaestimate, mu=1))

summary(betaestimate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
```

```
## k = 8


alphaestimate <- vector()
betaestimate <- vector()
for (i in 1:1000) {

  x <- rnorm(n = 100, mean = 1)
  e <- rnorm(n = 100)

  f <- function(x) {
    1/sqrt(2*pi)*exp((-1/2)*x^2)
  }

  f2 <- function(x, k) {
    (x^k)*f(x)
  }

  f3 <- function(k) {
    integrate(f2, -Inf, Inf, k)
```

```
  }

  n <- vector()

  for (i in 1:100) {
    expect <- f3(8)
    expect2 <- f3(16)

    expect <- as.numeric(unlist(expect)[1])
    expect2 <- as.numeric(unlist(expect2)[1])

    n[i] <- (e[i]^8 - expect)/(expect2 - expect^2)^(1/2)

  }
  y <- x + n
  dataset <- as.data.frame(cbind(y, x, n))

  model <- lm(data = dataset, y ~ x + n)

  alphaestimate <- append(alphaestimate, as.numeric(unlist(model[1])[1]))
  betaestimate <- append(betaestimate, as.numeric(unlist(model[1])[2]))
}
##ttests <- append(ttests, t.test(betaestimate, mu=1))

summary(betaestimate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
```