Example 8-7 and Figures 8-2 & 8-3

September 11, 2020

[ ]:
```
# install the following packages and libraries
install.packages("pder")
install.packages("plm")
install.packages("texreg")
install.packages("msm")

library("plm")
library("texreg")
library("msm")
library("ggplot2")

# import the data
data("RDSpillovers", package = "pder")

# create the following formula
fm.rds <- lny ~ lnl + lnk + lnrd
```

[4]:
```
##---------------------------------Block 1---------------------------------

#### Example 8-7 ####

## -----------------------------------------------------------------------

# common correlated effects pooled model with 3 alternative estimates for the␣
↪standard errors
ccep.rds <- pcce(fm.rds, RDSpillovers, model="p")
library(lmtest)
ccep.tab <- cbind(coeftest(ccep.rds)[, 1:2],
                  coeftest(ccep.rds, vcov = vcovNW)[, 2],
                  coeftest(ccep.rds, vcov = vcovHC)[, 2])
dimnames(ccep.tab)[[2]][2:4] <- c("Nonparam.", "vcovNW", "vcovHC")
round(ccep.tab, 3)
```

	Estimate	Nonparam.	vcovNW	vcovHC
lnl	0.562	0.088	0.031	0.045
lnk	0.289	0.161	0.045	0.077
lnrd	0.084	0.068	0.020	0.033

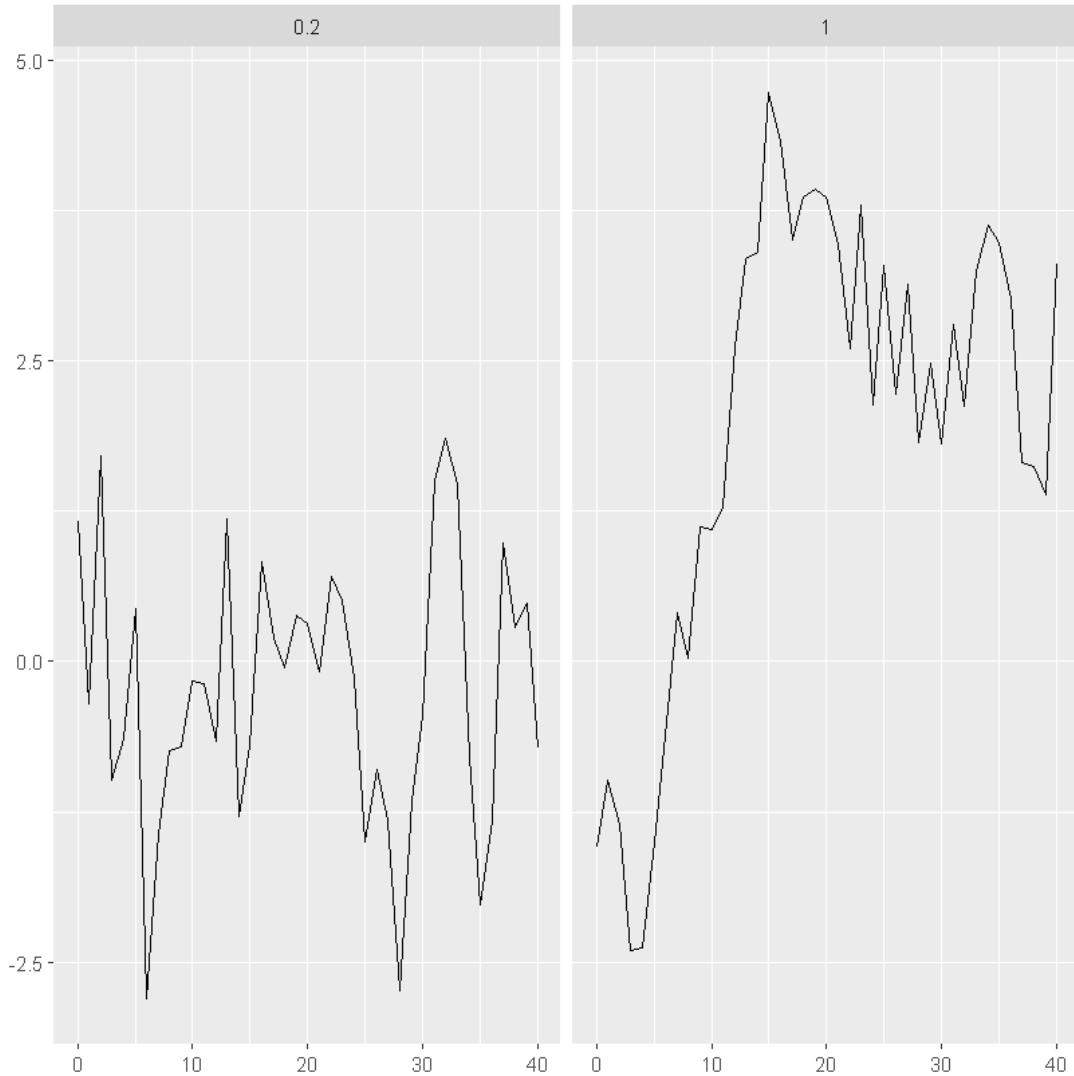
[5]:

```r
##----------------------------------Block 2----------------------------------

#### Figure 8-2 ####

autoreg <- function(rho = 0.1, T = 100){
  e <- rnorm(T+1)
  for (t in 2:(T+1)) e[t] <- e[t]+rho*e[t-1]
  e
}
set.seed(20)

f <- data.frame(time = rep(0:40, 2),
                rho = rep(c(0.2, 1), each = 41),
                y = c(autoreg(rho = 0.2, T = 40),
                      autoreg(rho = 1, T = 40)))
ggplot(f, aes(time, y)) + geom_line() + facet_wrap(~ rho) + xlab("") + ylab("")
```



[6]:
```
##----------------------------------Block 3----------------------------------

# the following code is for the purpose of illustrating spurios regression.
# a simulation of data is preformed by drawing two AR series independently
# and regress one on the another. the t-stat is then calculated

autoreg <- function(rho = 0.1, T = 100){
  e <- rnorm(T)
  for (t in 2:(T)) e[t] <- e[t] + rho *e[t-1]
  e
}
tstat <- function(rho = 0.1, T = 100){
  y <- autoreg(rho, T)
```

```
  x <- autoreg(rho, T)
  z <- lm(y ~ x)
  coef(z)[2] / sqrt(diag(vcov(z))[2])
}
result <- c()
R <- 1000
for (i in 1:R) result <- c(result, tstat(rho = 0.2, T = 40))
quantile(result, c(0.025, 0.975))
prop.table(table(abs(result) > 2))
```

2.5\% -2.11402406977969 **97.5\%** 1.99003122343967

```
FALSE  TRUE
0.943 0.057
```

[7]:

```
##--------------------------------Block 4--------------------------------

# now includes a unit root
result <- c()
R <- 1000
for (i in 1:R) result <- c(result, tstat(rho = 1, T = 40))
quantile(result, c(0.025, 0.975))
prop.table(table(abs(result) > 2))
```

2.5\% -9.15844814463279 **97.5\%** 8.22705902068572

```
FALSE  TRUE
0.379 0.621
```

[8]:

```
##--------------------------------Block 5--------------------------------

# unit root test
R <- 1000
T <- 100
result <- c()
for (i in 1:R){
  y <- autoreg(rho=1, T=100)
  Dy <- y[2:T] - y[1:(T-1)]
  Ly <- y[1:(T-1)]
  z <- lm(Dy ~ Ly)
  result <- c(result, coef(z)[2] / sqrt(diag(vcov(z))[2]))
}

prop.table(table(result < -1.64))
```

```
FALSE  TRUE
0.542 0.458
```

[9]:
```
##--------------------------------Block 6--------------------------------

#### Figure 8.3 ####

ggplot(data.frame(x = result), aes(x = x)) +
    geom_histogram(fill = "white", col = "black",
                   bins = 20, aes(y = ..density..)) +
    stat_function(fun = dnorm) + xlab("") + ylab("")
```