```r
############################################################
# Title:        ps3.r
# Author:       Jason Parker
# Date:         2017-10-16
# Description:  Turn-in product for problem set 3
############################################################

### File Labels
## Question 1
# variable name   type     format      label      variable label
# price           float    %9.0g                  house price, $1000s
# assess          float    %9.0g                  assessed value, $1000s
# bdrms           byte     %9.0g                  number of bdrms
# lotsize         float    %9.0g                  size of lot in square feet
# sqrft           int      %9.0g                  size of house in square feet
# colonial        byte     %9.0g                  =1 if home is colonial style
## Question 2
# variable name   type     format      label      variable label
# month           int      %d                     dec 2000 through feb 2012
# urate           double   %10.0g                 unemployment rate, percent
# vrate           double   %10.0g                 vacancy rate, percent
# t               int      %9.0g                  linear time trend
## Question 3
# variable name   type     format      label      variable label
# year            int      %9.0g                  1987, 1988, or 1989
# fcode           float    %9.0g                  firm code number
# employ          int      %9.0g                  # employees at plant
# sales           float    %9.0g                  annual sales, $
# avgsal          float    %9.0g                  average employee salary
# scrap           float    %9.0g                  scrap rate (per 100 items)
# rework          float    %9.0g                  rework rate (per 100 items)
# tothrs          int      %9.0g                  total hours training
# union           byte     %9.0g                  =1 if unionized
# grant           byte     %9.0g                  = 1 if received grant
# totrain         int      %9.0g                  total employees trained

## Setup Workplace
rm(list=ls(all=TRUE))
library(data.table)
library(lmtest)
library(sandwich)
library(tseries)
library(plm)

repkpss <- function(x,test=c("Level","Trend"),dmax=5,level=0.05) {
  diff <- 0
  while (diff<=dmax) {
    suppressWarnings ( results <- kpss.test(x,null="Level") )
    if  (results$p.value>level)
      return(c(diff,"Level",round(results$statistic,digits=3),round(results$p.value,digits=3)))
    if (test == "Trend") {
      suppressWarnings( results <- kpss.test(x,null="Trend") )
      if (results$p.value>level)
        return(c(diff,"Trend",round(results$statistic,digits=3),round(results$p.value,digits=3)))
    }
    diff <- diff + 1
    x    <- diff(x)
  }
  return(c(NaN,NaN,NaN,NaN))
}

context1 <- fread('hprice1.csv')
context2 <- fread('beveridge.csv')
context3 <- fread('JTRAIN.csv')

## Summary statistics
# head(context1)
# head(context2)
# head(context3)
# summary(context1)
# summary(context2)
# summary(context3)

## Generate CTE dummies
context3$d88 <- as.numeric(context3$year==1988)
context3$d89 <- as.numeric(context3$year==1989)

## Another way
context3$d88 <- ifelse(context3$year==1988,1,0)
context3$d89 <- ifelse(context3$year==1989,1,0)

## Generate lag(grant)
context3$grant_lag <- rep(0,471)
context3 <- context3[order(fcode,year)]
for(j in 1:471)
```

```
  if(context3$year[j]!=1987)
    context3$grant_lag[j] <- context3$grant[j-1]


## Different way
context3$grant_lag <- rep(0,471)
for(i in 2:471)
  context3$grant_lag[i] <- context3$grant[i-1]
context3[year==1987]$grant_lag <- rep(0,471/3)


## Easier to code
context3$grant_lag <- rep(0,471)
context3[year==1988]$grant_lag <- context3[year==1987]$grant
context3[year==1989]$grant_lag <- context3[year==1988]$grant


## Declare panel
context3 <- plm.data(context3,indexes=c('fcode','year'))


## Generate models
model1 <- lm(price~bdrms+lotsize+sqrft,data=context1)
model2 <- lm(log(price)~bdrms+log(lotsize)+log(sqrft),data=context1)
model3 <- lm(urate~vrate,data=context2)
model4 <- lm(diff(urate)~diff(vrate),data=context2)
model5 <- plm(log(scrap)~d88+d89+grant+grant_lag,model="pooling",data=context3)
model6 <- plm(log(scrap)~d88+d89+grant+grant_lag,model="within",data=context3)


## Summarize models
# summary(model1)
# summary(model2)
# summary(model3)
# summary(model4)
# summary(model5)
# summary(model6)


## OLS Coefficient Tests
coeftest(model1)
coeftest(model2)
coeftest(model3)
coeftest(model4)
coeftest(model5)
coeftest(model6)


## Corrected significance tests
coeftest(model1,vcov.=vcovHC)
coeftest(model2,vcov.=vcovHC)
coeftest(model3,vcov=NeweyWest(model3,lag=5))
coeftest(model4,vcov=NeweyWest(model4,lag=5))
coeftest(model5,vcov=vcovHC(model5,method="arellano"))
coeftest(model6,vcov=vcovHC(model6,method="arellano"))


## KPSS Tests urate
kpss.test(context2$urate,null="Level")
kpss.test(context2$urate,null="Trend")
kpss.test(diff(context2$urate),null="Level")
# kpss.test(diff(context2$urate),null="Trend")
# kpss.test(diff(diff(context2$urate)),null="Level")
# kpss.test(diff(diff(context2$urate)),null="Trend")


## KPSS Tests vrate
kpss.test(context2$vrate,null="Level")
kpss.test(context2$vrate,null="Trend")
kpss.test(diff(context2$vrate),null="Level")
# kpss.test(diff(context2$vrate),null="Trend")
# kpss.test(diff(diff(context2$vrate)),null="Level")
# kpss.test(diff(diff(context2$vrate)),null="Trend")

repkpss(context2$urate,test="Trend")
repkpss(context2$vrate,test="Trend")
```