Appendix-Codes

Section One

```
#Downloading the libraries, setting the working directory and importing
 the data set
library(tidyverse)
library(stargazer)
library(dagitty)
library(gridExtra)
library(tinytex)
library(ggplot2)
library(tidyr)
library(dplyr)
library(plyr)
library(reshape2)
library(sandwich)

dir <- "C:/Users/Administrator/Desktop/NewStart/Courses/AdvancedStatist
icsandProgramming/assignment2/github/BAM_ASP_A2"
dirProg <- paste0(dir, "/programs/")
dirData <- paste0(dir, "/Data/")

dfDiD <- read.csv(file=paste0(dirData, "DiD_dataset.csv"))
```

Preparing and analyzing the dataset

```
# no need to transform the dataset, already in the long format
str(dfDiD) # all variables are numeric or integer, no need to tranform


dfDiD$dPeriod = ifelse(dfDiD$year >= 1993, 1, 0) # dummy variable for p
eriod
dfDiD$cChildren = ifelse(dfDiD$children >= 1, 1, 0) # dummy for differe
nt groups

dfDiD.sub <- subset(dfDiD, work=="1") #creating a subset of employed wo
men
```

1 Plotting the dependent variables

```
#Earn
#6 years for both groups, total of 12 averages (average by year and chi
ldren (0/1)
earn.agg = aggregate(dfDiD.sub$earn, list(dfDiD.sub$year, dfDiD.sub$cCh
ildren == 1),
                     FUN = mean, na.rm = TRUE)
```

```r
names(earn.agg) = c("Year","Children","Earn") #rename variables
#new variable with group name
earn.agg$Group[1:6] = "Women without children"
earn.agg$Group[7:12] = "Women with children"

Earn.plot <- qplot(Year, Earn, data=earn.agg, geom=c("point","line"),
  colour = Group,
  xlab="Year", ylab="Annual earnings") +
  geom_vline(xintercept = 1993) +
  theme_bw()
ggsave(file="Earn.pdf", width=7, height=4)

#Finc
finc.agg = aggregate(dfDiD.sub$finc, list(dfDiD.sub$year, dfDiD.sub$cCh
ildren == 1),
                     FUN = mean, na.rm = TRUE)
names(finc.agg) = c("Year","Children","Finc")
finc.agg$Group[1:6] = "Women without children"
finc.agg$Group[7:12] = "Women with children"

Finc.plot <- qplot(Year, Finc, data=finc.agg, geom=c("point","line"),
  colour = Group,
  xlab="Year", ylab="Annual Family Income") +
  geom_vline(xintercept = 1993) +
  theme_bw()
ggsave(file="Finc.pdf", width=7, height=4)

#Work
work.agg = aggregate(dfDiD$work, list(dfDiD$year, dfDiD$cChildren == 1),

                     FUN = mean, na.rm = TRUE)
names(work.agg) = c("Year","Children","Work")

work.agg$Group[1:6] = "Women without children"
work.agg$Group[7:12] = "Women with children"

Work.plot <- qplot(Year, Work, data=work.agg, geom=c("point","line"),
  colour = Group,
  xlab="Year", ylab="Work")+
  geom_vline(xintercept = 1993) +
  theme_bw()
ggsave(file="Work.pdf", width=7, height=4)
```

2 Summary statistics of the dataset

```r
stargazer(dfDiD, type = "text")
stargazer(dfDiD[, c("children", "finc", "earn", "age", "work", "unearn
")], type = "text")
```

3 Difference-in-Difference

```
# creating averages per group per period
avgEarn <- ddply (dfDiD.sub, .(dPeriod, cChildren), summarise,
                  avgEarn = mean(earn, na.rm=TRUE))

avgFinc <- ddply (dfDiD.sub, .(dPeriod, cChildren), summarise,
                  avgFinc = mean(finc, na.rm=TRUE))

avgWork <- ddply (dfDiD, .(dPeriod, cChildren), summarise,
                  avgWork = mean(work, na.rm=TRUE))

#Remodel the avg table from long to wide, add  row for the difference i
n averages
avgtable.Earn <- dcast (avgEarn, dPeriod ~ cChildren, value.var = "avgE
arn")
avgtable.Earn <- rbind(avgtable.Earn, avgtable.Earn[2,]-avgtable.Earn
[1,])
rownames(avgtable.Earn) <- c("Before", "After", "Difference") # renamin
g the rows
colnames(avgtable.Earn) <- c("dPeriod", "Women without children (0)",
                             "Women with children (1)") # renaming the
columns
avgtable.Earn[3, "dPeriod"] <- NA

avgtable.Finc <- dcast (avgFinc, dPeriod ~ cChildren, value.var = "avgF
inc")
avgtable.Finc <- rbind(avgtable.Finc, avgtable.Finc[2,]-avgtable.Finc
[1,])
rownames(avgtable.Finc) <- c("Before", "After", "Difference")
colnames(avgtable.Finc) <- c("dPeriod", "Women without children (0)",
                             "Women with children (1)")
avgtable.Finc[3, "dPeriod"] <- NA

avgtable.Work <- dcast (avgWork, dPeriod ~ cChildren, value.var = "avgW
ork")
avgtable.Work <- rbind(avgtable.Work, avgtable.Work[2,]-avgtable.Work
[1,])
rownames(avgtable.Work) <- c("Before", "After", "Difference")
colnames(avgtable.Work) <- c("dPeriod", "Women without children (0)",
                             "Women with children (1)")
avgtable.Work[3, "dPeriod"] <- NA

stargazer(avgtable.Earn, summary=FALSE, align = TRUE, type="text",
          title = "Average Annual Earnings")
stargazer(avgtable.Finc, summary=FALSE, align = TRUE, type="text",
          title = "Average Indicator Annual Family Income")
stargazer(avgtable.Work, summary=FALSE, align = TRUE, type="text",
          title = "Average Indicator Work Status")
```

4 Regression analysis

```
mdlEarn <- earn ~ cChildren + dPeriod + cChildren:dPeriod
rsltOLSEarn <- lm(mdlEarn, data=dfDiD.sub)

mdlFinc <- finc ~ cChildren + dPeriod + cChildren:dPeriod
rsltOLSFinc <- lm(mdlFinc, data=dfDiD.sub)

mdlWork <- work ~ cChildren + dPeriod + cChildren:dPeriod
rsltOLSWork <- lm(mdlWork, data=dfDiD)

stargazer(rsltOLSEarn, rsltOLSFinc, rsltOLSWork,
          intercept.bottom = FALSE, align = TRUE, no.space=TRUE,
          type="text")
```

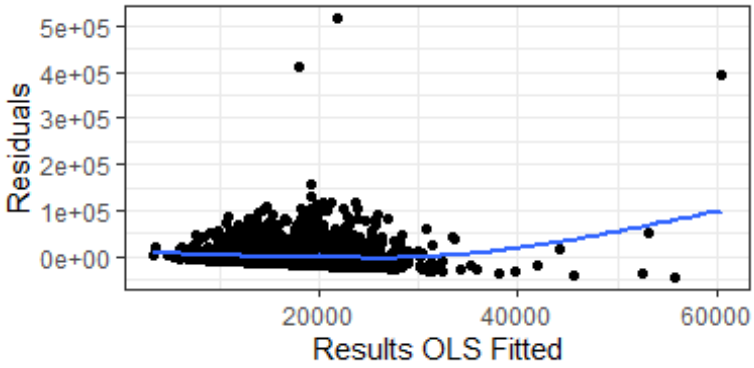
Control variables

```
# adding urate, unearn and children as control variables
# Earn
mdl.control.earn <- earn ~ cChildren + dPeriod + cChildren:dPeriod +
  urate + unearn + children
rsltOLS.control.earn <- lm(mdl.control.earn, data=dfDiD.sub)

# Finc
mdl.control.finc <- finc ~ cChildren + dPeriod + cChildren:dPeriod +
  urate + unearn + children
rsltOLS.control.finc <- lm(mdl.control.finc, data=dfDiD.sub)

# Work
mdl.control.work <- work ~ cChildren + dPeriod + cChildren:dPeriod +
  urate + unearn + children
rsltOLS.control.work <- lm(mdl.control.work, data=dfDiD)

stargazer(rsltOLS.control.earn, rsltOLS.control.finc,
          rsltOLS.control.work,
          intercept.bottom = FALSE,
          align = TRUE,
          no.space=TRUE, type="text")
```

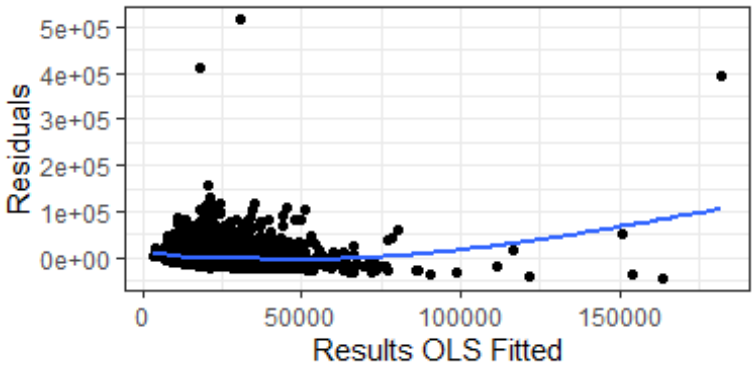
Robust standard errors

```
#Test for heteroskedasticity
rsltOLS.control.earn2 <- lm(mdl.control.earn, data=dfDiD.sub)
rsltOLS.control.finc2 <- lm(mdl.control.finc, data=dfDiD.sub)
rsltOLS.control.work2 <- lm(mdl.control.work, data=dfDiD)

# EARN
ggplot(data = data.frame(fit = fitted(rsltOLS.control.earn2),
  rsid = residuals(rsltOLS.control.earn2)),
  aes(fit, rsid)) +
  geom_point() +
```

```
  stat_smooth(se = F) +
  theme_bw() +
  labs(x = "Results OLS Fitted") +
  labs(y = "Residuals")
```

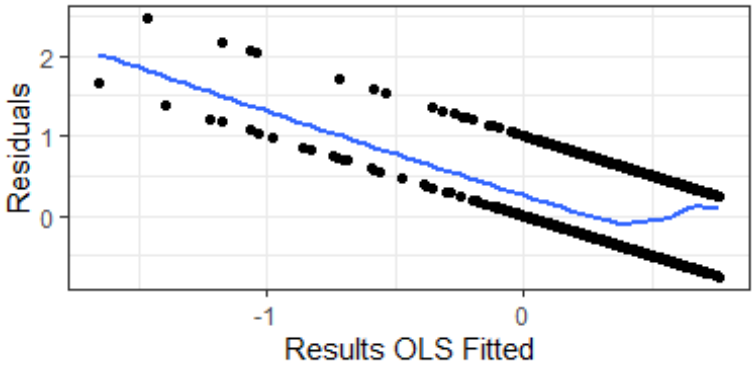



```
lmtest::bptest(rsltOLS.control.earn2)
# p < 0.01, heteroskedastiscity is detected.

#FINC
ggplot(data = data.frame(fit = fitted(rsltOLS.control.finc2),
  rsid = residuals(rsltOLS.control.finc2)),
  aes(fit, rsid)) +
  geom_point() +
  stat_smooth(se = F) +
  theme_bw() +
  labs(x = "Results OLS Fitted") +
  labs(y = "Residuals")
```




```
lmtest::bptest(rsltOLS.control.finc2)
# p < 0.01, heteroskedastiscity is detected.

#WORK
ggplot(data = data.frame(fit = fitted(rsltOLS.control.work2),
  rsid = residuals(rsltOLS.control.work2)),
  aes(fit, rsid)) +
  geom_point() +
```

```
  stat_smooth(se = F) +
  theme_bw() +
  labs(x = "Results OLS Fitted") +
  labs(y = "Residuals")
```




```
lmtest::bptest(rsltOLS.control.work2)
# p < 0.01, heteroskedastiscity is detected

#Standard  errors
seBasicEarn  <- sqrt(diag(vcov(rsltOLS.control.earn2)))
seWhiteEarn  <- sqrt(diag(vcovHC(rsltOLS.control.earn2, type="HC0")))
seClusterEarn <- sqrt(diag(vcov(rsltOLS.control.earn2, cluster="state
")))
stargazer(rsltOLS.control.earn2, rsltOLS.control.earn2, rsltOLS.control.
earn2,
          se=list(seBasicEarn, seWhiteEarn, seClusterEarn), type="text")

#No impact on the significance of the DiD effect, still insignificant
#Standard error of seWhite seems smaller than basic and clustered

seBasicFinc  <- sqrt(diag(vcov(rsltOLS.control.finc2)))
seWhiteFinc  <- sqrt(diag(vcovHC(rsltOLS.control.finc2, type="HC0")))
seClusterFinc <- sqrt(diag(vcov(rsltOLS.control.finc2, cluster="state
")))
stargazer(rsltOLS.control.finc2, rsltOLS.control.finc2, rsltOLS.control.
finc2,
          se=list(seBasicFinc, seWhiteFinc, seClusterFinc), type="text")
#No impact on the significance of the DiD effect, still insignificant
#Standard error of seWhite seems smaller than basic and clustered

seBasicWork  <- sqrt(diag(vcov(rsltOLS.control.work2)))
seWhiteWork  <- sqrt(diag(vcovHC(rsltOLS.control.work2, type="HC0")))
seClusterWork <- sqrt(diag(vcov(rsltOLS.control.work2, cluster="state
")))
stargazer(rsltOLS.control.work2, rsltOLS.control.work2, rsltOLS.control.
work2,
          se=list(seBasicWork, seWhiteWork, seClusterWork), type="text")
#No impact on the significance of the DiD effect, all three significant
```

```
 (p<0.05).
#Standard error for all three remains the same
```

section2_IVA

Instrumental Variable Analysis: Effect of Compulsory Schooling on Wages

Downloading the libraries

```
# Load libraries
library(tidyverse)
library(stargazer)
library(dagitty)
library(gridExtra)
library(tinytex)
library(stargazer)
library(AER)
library(ivpack)

# Set working director
setwd("C:/Users/Administrator/Desktop/NewStart/Courses/AdvancedStatisti
csandProgramming/assignment2/github/BAM_ASP_A2/data")

# Load csv and generate subset containing only variables for interest
da.IV <- read.csv("IV_dataset.csv", header = TRUE)
da.IV <- subset(da.IV, select = c("age", "educ", "lnwage", "married", "
qob",
                                  "SMSA", "yob"))

## Subset the data set so that we could focus on the variables above ac
cording to the order
da.IV <- read.csv("IV_dataset.csv", header = TRUE)
da.IV <- subset(da.IV,select = c("age","educ","lnwage","married","qob",
"SMSA","yob"))
## Subset the dataset so that we could focus on the variables above acc
ording to the order

stargazer(da.IV,type = "text")
summary(as.factor(da.IV$married))

# Convert to factor variables
da.IV$married <- as.factor(da.IV$married)
da.IV$qob <- as.factor(da.IV$qob)
da.IV$SMSA <- as.factor(da.IV$SMSA)
```

```r
da.IV$yob <- as.factor(da.IV$yob)

# To change those variables which should be factor variables into facto
r variables
g1.1 <- ggplot(data = da.IV, aes(qob, educ)) +
  geom_point(size = 0.5) +
  geom_smooth(method = "lm", color = "blue", alpha = 0.2)  +
  theme_bw() +
  labs(caption = "Figure 2.1") +
  geom_boxplot() +
  theme(plot.caption =  element_text(hjust = 0.5, size = 12, face = "bo
ld")) +
  labs(x = "Quarter of Birth", y = "Education(in years)")
g1.1



rsltIV <- ivreg(lnwage ~ educ|qob,data = da.IV)
summary(rsltIV, diagnostics = TRUE)

library(ivreg)
rslt2SLS.A <- ivreg(lnwage ~ educ | qob, data=da.IV)
summary(rslt2SLS.A)
stargazer(rslt2SLS.A, type= "text")

rslt2SLS.B <- ivreg(lnwage ~ educ + married + SMSA | married + SMSA + q
ob,
                    data=da.IV)
summary(rslt2SLS.A)
stargazer(rslt2SLS.A, rslt2SLS.B)

#Robust standard errors
modelIV <- ivreg(lnwage ~ educ + married + SMSA | married + SMSA + qob ,

                 data=da.IV)
summary(modelIV)

#Standard  errors (superfluous  in the  case of  seBasic)
seBasic  <- sqrt(diag(vcov(modelIV)))
seWhite  <- sqrt(diag(vcovHC(modelIV , type="HC0")))
library(vcov)
# Make  table  with  stargazer
stargazer(modelIV , modelIV ,align=TRUE , no.space=TRUE ,intercept.bott
om = FALSE ,se = list(seBasic , seWhite), type= "text")

da.IV_sub <- subset(da.IV,select = c("age", "educ", "lnwage", "married",
 "qob",
                                     "SMSA", "yob"))

# Convert to factor variables
da.IV_sub$married <- as.factor(da.IV_sub$married)
```

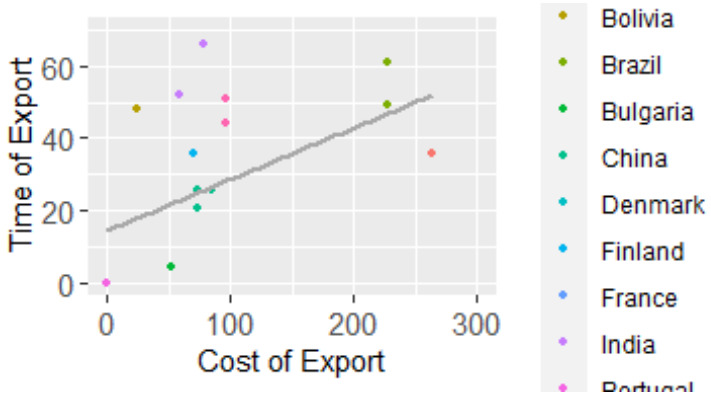
```r
da.IV_sub$qob <- as.factor(da.IV_sub$qob)
da.IV_sub$SMSA <- as.factor(da.IV_sub$SMSA)
da.IV_sub$yob <- as.factor(da.IV_sub$yob)

# Define OLS models
rsltOLS.A <- lm(lnwage ~ educ, data=da.IV_sub)
rsltOLS.B <- lm(lnwage ~ educ + married + SMSA, data=da.IV_sub)

# Define IV model
rsltSLS.A <- ivreg(lnwage ~ educ | qob, data=da.IV_sub)
rsltSLS.B <- ivreg(lnwage ~ educ + married + SMSA | married + SMSA + qo
b,
                   data=da.IV_sub)
rsltSLS.C <- ivreg(lnwage ~ educ + married + SMSA | married + SMSA + ag
e + qob,
                   data=da.IV_sub)

# Generate table containing both models
stargazer(rsltOLS.A, rsltOLS.B, rsltSLS.A, rsltSLS.B, rsltSLS.C, type="
text")

# Test for violation over-identification
summary(rsltSLS.A, diagnostics = TRUE)
summary(rsltSLS.B, diagnostics = TRUE)
summary(rsltSLS.C, diagnostics = TRUE)
```

section3_PDM

```r
# Load libraries
library(tidyverse)
library(stargazer)
library(wbstats)
library(ggplot2)
library(plyr)
library(plm)

# Load world bank data
dfExport <- wb_data(indicator=c("IC.EXP.TMBC",       # Time to export
                                "NY.GDP.PCAP.CD",    # GDP per capita
                                "TG.VAL.TOTL.GD.ZS", # Merchandise trad
e % GDP
                                "NE.EXP.GNFS.ZS",    # Exports of goods
 and services (% of GDP)
                                "IC.EXP.CSDC.CD"),   # Cost to export
                                country = "countries_only",
                                start_date = 2014,
                                end_date = 2019)
```

```r
# Rename column names
colnames(dfExport)[colnames(dfExport) == "date"]              <- "Year"
colnames(dfExport)[colnames(dfExport) == "country"]           <- "Count
ry"
colnames(dfExport)[colnames(dfExport) == "date"]              <- "Year"
colnames(dfExport)[colnames(dfExport) == "IC.EXP.TMBC"]       <- "TimeE
xport"
colnames(dfExport)[colnames(dfExport) == "NY.GDP.PCAP.CD"]    <- "GDPPe
rCap"
colnames(dfExport)[colnames(dfExport) == "TG.VAL.TOTL.GD.ZS"] <- "Merch
andiseGDP"
colnames(dfExport)[colnames(dfExport) == "NE.EXP.GNFS.ZS"]    <- "Expor
tGoodsServices"
colnames(dfExport)[colnames(dfExport) == "IC.EXP.CSDC.CD"]    <- "CostE
xport"

# Subset complete observations, and implement an admittedly arbitrary
# observation period
dfExport.sub <- dfExport[complete.cases(dfExport),]

# Generate list with all countries with complete observations
complete <- dfExport.sub %>%
  dplyr::count(Country) %>%
  filter(n == 6)
completeCountry <- as.vector(complete$Country)

# Generate data frame only containing countries with complete observati
ons
dfExport.sub.cmplt <- dfExport.sub %>%
  filter(Country %in% completeCountry)

# Convert to data frame
dfExport.sub.cmplt <- as.data.frame(dfExport.sub.cmplt)

# Generate table with summary statistics
stargazer(dfExport.sub.cmplt)

# Plot Cost Export

subCountries <- c("Australia", "Bolivia", "Brazil", "Portugal", "Thaila
nd",
                  "Zimbabwe", "Bangladesh", "Bulgaria", "China", "Denma
rk",
                  "France", "Finland", "India")

dfExport.sub.cmplt <-
  dfExport.sub.cmplt[dfExport.sub.cmplt$Country %in% subCountries,]

ggplot(dfExport.sub.cmplt, aes(x=CostExport, y=TimeExport))+
```

```
  #add the annual outcomes coloured by Country
  geom_point(aes(color=Country), size=1)+
  #add regression lines for the countries
  geom_smooth(method="lm", se=FALSE, colour="dark grey")+
  #label the axis
  xlim(0, 300) + ylim(0, 70)+
  xlab("Cost of Export")+
  ylab("Time of Export")+
  theme(axis.title= element_text(size=rel(1)),
        axis.text= element_text(size=rel(1)))+
  guides(colour = guide_legend(override.aes = list(size=1)))
```



Preparing data for regression

```
# Determine country averages of the included variables, as well as the
number of
# non missing observations during the selected observation period
dfExport.sub.cmplt.avg <-
  ddply(dfExport.sub.cmplt, .(Country), summarise,
        avg.TimeExport    = mean(TimeExport, na.rm=TRUE),
        avg.GDPPerCap     = mean(GDPPerCap, na.rm=TRUE),
        avg.CostExport    = mean(CostExport, na.rm=TRUE),
        avg.ExportGoodsServices     = mean(ExportGoodsServices, na.rm=T
RUE),
        avg.MerchandiseGDP    = mean(MerchandiseGDP, na.rm=TRUE),
        numValid          = length(Country))

# Merge averages in dfWorld.avg with dfWorld.sub (this can be done with

# 'mutate', but then the concise data frame with country average will n
ot be
# made available
dfExport.sub.cmplt <- merge(dfExport.sub.cmplt, dfExport.sub.cmplt.avg,

                            by="Country")

attach(dfExport.sub.cmplt)
dfExport.sub.cmplt$diff.TimeExport    <- TimeExport    - avg.TimeExport
```

```
dfExport.sub.cmplt$diff.GDPPerCap    <- GDPPerCap    - avg.GDPPerCap
dfExport.sub.cmplt$diff.CostExport   <- CostExport   - avg.CostExport
dfExport.sub.cmplt$diff.ExportGoodsServices <- ExportGoodsServices  -
  avg.ExportGoodsServices
dfExport.sub.cmplt$diff.MerchandiseGDP   <- MerchandiseGDP   -
  avg.MerchandiseGDP
detach(dfExport.sub.cmplt)
```

Pooled Regression

```
#Formulate the model (very ad hoc)
mdlA <- TimeExport ~ GDPPerCap + CostExport + ExportGoodsServices +
  MerchandiseGDP

#Make between and within group data frames

#For convenience two datasets are made that contain the model
#variables for the within group differences and the between
#group difference

# find the variable of interest
mdlvars <- all.vars(mdlA)
mdlvars.avg <- paste0("avg.", mdlvars)
mdlvars.diff <- paste0("diff.", mdlvars)

# Select variables from the data frames
dfExport.between <- dfExport.sub.cmplt.avg[mdlvars.avg]
dfExport.within <-dfExport.sub.cmplt[mdlvars.diff]

# Rename column names in order to make use  of the same model specifica
tion
# mdlA, and to conveniently merge the regression objects in stargazer

colnames(dfExport.within) <-
  gsub("diff\\.", "", colnames(dfExport.within))
colnames(dfExport.between) <-
  gsub("avg\\.", "", colnames(dfExport.between))

## Estimation of the pooled model
rsltPool <- lm(mdlA, data= dfExport.sub.cmplt)
summary(rsltPool)
stargazer::stargazer(rsltPool, align=TRUE, no.space=TRUE,
                     intercept.bottom=FALSE, type="text")
```

Between regression

```
rsltwithin <- lm(mdlA, data= dfExport.within)
summary(rsltwithin)
rsltBetween <- lm (mdlA, data= dfExport.between)
summary(rsltBetween)
```

```
stargazer::stargazer(rsltPool, rsltBetween, aling=TRUE, no.space=TRUE,
                     intercept.bottom= FALSE, type= "text")
```

Fixed Effect Regression

```
rsltFE.Country <- plm(mdlA, data= dfExport.sub.cmplt,
                      index= c("Country", "Year"), model="within")
#Tabulate the results
summary(rsltFE.Country)
stargazer::stargazer(rsltPool, rsltFE.Country, align=TRUE, no.space=TRU
E,
                     intercept.bottom=FALSE, type="text")
#Explore the estimated intercepts
summary(fixef(rsltFE.Country, type="dmean"))
```

Random Effect Regression

```
#Estimate random effect model ('random')
rsltRE.Country <- plm(mdlA, data=dfExport.sub.cmplt,
                      index=c("Country", "Year"), model= "random")

#Tabulate the results
summary(rsltRE.Country)
stargazer::stargazer(rsltPool, rsltFE.Country, rsltRE.Country,
                     align=TRUE, no.space=TRUE, intercept.bottom=FALSE,

                     type="text")

# Evaluate the fixed effects model versus the pooled regression model
# Last minute of tutorial #4 Panel Data
# An insignificant tests tells that all models are consistent
# A significant tests rejects the hypothesis in favor of the fix effect
s model
pFtest(rsltFE.Country, rsltPool)

# How do we now when to use fixed and when to use random?
# Hausman test: compare random and fixed effects models
# Under H0, no correlation between disturbance and explanatory variable
s,
# both RE and FE are consistent (though FE is not efficient), under H1,

# correlation between disturbance, only FE consistent
# Last two minutes of tutorial #5 Panel Data
phtest(rsltFE.Country, rsltRE.Country)
```