---

title: "panel\_growth"

output: pdf\_document

editor\_options:

chunk\_output\_type: inline

---

by Aytaç Aydın

# Initial Clean-Up

```{r}

if(!is.null(dev.list())) dev.off()

cat("\014")

rm(list=ls(all=TRUE))

```

# Creating ID Codes

```{r}

library(readxl)

e7[[2]] <- rep(1:7, each = 19)

```

# Descriptive Statistics

```{r}

library(psych)

describe(e7)

write.xlsx(describe(e7), "betimsel\_e7.xlsx")

write.xlsx(describe(g7), "betimsel\_g7.xlsx")

```

# Converting Data to Panel Data Frame and Logarithm

```{r}

library(readxl)

id <- rep(1:7, each = (2022 - 2004 + 1))

e7[[2]] <- id

g7[[2]] <- id

names(e7)[1:2] <- c("id", "year")

names(g7)[1:2] <- c("id", "year")

names(g7)[9] <- "gdp"

names(e7)[9] <- "gdp"

g7$fdi <- min(g7$fdi) \* -1 + g7$fdi + 1000

g7$trade\_deficit <- min(g7$trade\_deficit) \* -1 + g7$trade\_deficit + 1000

min(e7$fdi)

log(e7$automation)

library(plm)

e7p <- pdata.frame(e7, index = c("id", "year"))

g7p <- pdata.frame(g7, index = c("id", "year"))

g7p[[5]] <- g7p[[5]] + (min(g7p[[5]]) \* -1) + 100

g7p[[7]] <- g7p[[7]] + (min(g7p[[7]]) \* -1) + 100

e7p[[5]] <- e7p[[5]] + (min(e7p[[5]]) \* -1) + 100

e7p[[7]] <- e7p[[7]] + (min(e7p[[7]]) \* -1) + 100

e7p$fdi <- log(e7p$fdi)

e7p$trade\_deficit <- log(e7p$trade\_deficit)

e7p$gdp <- log(e7p$gdp)

e7p$automation <- log(e7p$automation)

e7p$wage <- log(e7$wage)

e7p$robot <- log(e7$robot)

g7p$fdi <- log(g7p$fdi)

g7p$trade\_deficit <- log(g7p$trade\_deficit)

g7p$gdp <- log(g7p$gdp)

g7p$automation <- log(g7p$automation)

g7p$wage <- log(g7$wage)

```

# CD TEST FOR G7

```{r}

library(plm)

l\_plm\_models <- list()

cd\_test\_g7 <- matrix(0, nrow = length((3:(length(g7[1,])))), ncol = 2)

sutun\_isimleri <- c("test\_ist", "p\_degeri")

satir\_isimleri <- names(g7)[3:length(g7[1,])]

row.names(cd\_test\_g7) <- satir\_isimleri

colnames(cd\_test\_g7) <- sutun\_isimleri

for (i in 1 : (ncol(g7)-2)) {

l\_plm\_models[[i]] <- plm(formula = as.formula(paste0(colnames(g7)[i+2], "~1", sep = "")),

data = g7,

model = "pooling")

cd\_test\_g7[c(i:i), c(1, 2)] <-c((pcdtest(l\_plm\_models[[i]])[[1]]),

(pcdtest(l\_plm\_models[[i]])[[5]]))

}

round(cd\_test\_g7, 4)

write.xlsx(round(cd\_test\_g7, 4), "cd\_test\_g7.xlsx")

```

# CD TEST FOR E7

```{r}

library(plm)

l\_plm\_models <- list()

cd\_test\_e7 <- matrix(0, nrow = 10, ncol = 2)

sutun\_isimleri <- c("test\_ist", "p\_degeri")

satir\_isimleri <- names(e7)[3:length(g7[1,])]

row.names(cd\_test\_e7) <- satir\_isimleri

colnames(cd\_test\_e7) <- sutun\_isimleri

for (i in 1 : (ncol(e7)-2)) {

l\_plm\_models[[i]] <- plm(formula = as.formula(paste0(colnames(e7)[i+2], "~1", sep = "")),

data = e7,

model = "pooling")

cd\_test\_e7[c(i:i), c(1, 2)] <-c((pcdtest(l\_plm\_models[[i]])[[1]]),

(pcdtest(l\_plm\_models[[i]])[[5]]))

}

cd\_test\_e7

write.xlsx(round(cd\_test\_e7, 4), "cd\_e7.xlsx")

```

```{r}

birim\_kök\_g7 <- matrix(0, 12, 2)

say <- 0

for (i in 3:14){

say <- say + 1

kök <- purtest(rbind(g7)[i],

lags = 1,

type = "levinlin")

birim\_kök\_g7[say, 1] <- kök[[1]][[1]]

birim\_kök\_g7[say, 2] <- kök[[1]][[6]]

}

birim\_kök\_e7 <- matrix(0, 12, 2)

say <- 0

for (i in 3:14){

say <- say + 1

kök <- purtest(rbind(e7)[i],

lags = 1,

type = "levinlin")

birim\_kök\_e7[say, 1] <- kök[[1]][[1]]

birim\_kök\_e7[say, 2] <- kök[[1]][[6]]

}

round(birim\_kök\_g7, 4)

round(birim\_kök\_e7, 4)

rownames(birim\_kök\_e7) <- colnames(e7)[3:14]

rownames(birim\_kök\_g7) <- colnames(g7)[3:14]

colnames(birim\_kök\_e7) <- colnames(cd\_test\_g7)

colnames(birim\_kök\_g7) <- colnames(cd\_test\_g7)

write.xlsx(round(birim\_kök\_g7, 4), "birim\_g7.xlsx")

write.xlsx(round(birim\_kök\_e7, 4), "birim\_e7.xlsx")

```

# pooled estimation without variable names

```{r}

wage\_pooled <- plm(e7p[[13]] ~ e7p[[3]] + e7p[[4]] + e7p[[5]] + e7p[[6]] + e7p[[7]] + e7p[[8]] + e7p[[9]] + e7p[[10]] + e7p[[11]] + e7p[[12]] + e7p[[14]], data = e7p, model = "pooling")

wage\_fe <- plm(e7p[[13]] ~ e7p[[3]] + e7p[[4]] + e7p[[5]] + e7p[[6]] + e7p[[7]] + e7p[[8]] + e7p[[9]] + e7p[[10]] + e7p[[11]] + e7p[[12]] + e7p[[14]], data = e7p, model = "within")

summary(wage\_pooled)

summary(wage\_fe)

names(e7p)

```

# pooled estimation with variable names

```{r}

wage\_pooled <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "pooling")

summary(wage\_pooled)

wage\_fe <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "within")

wage\_re <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "random")

summary(wage\_pooled)

summary(wage\_fe)

summary(wage\_re)

```

# wage estimation with variable names

```{r}

wage\_pooled\_e7 <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "pooling")

wage\_fe\_e7 <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "within")

wage\_re\_e7 <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "random")

summary(wage\_pooled\_e7)

# write.xlsx(round(summary(wage\_pooled\_e7)[[1]], 4), "ücret\_havuz\_e7.xlsx")

# write.xlsx(round(summary(wage\_fe\_e7)[[1]], 4), "ücret\_sabit\_e7.xlsx")

# write.xlsx(round(summary(wage\_re\_e7)[[1]], 4), "ücret\_rassal\_e7.xlsx")

phtest(wage\_fe\_e7, wage\_re\_e7)

plmtest(wage\_pooled\_e7)

library(lmtest)

bgtest(plm(wage ~ inflation + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = g7p, model = "pooling"))

```

# wage estimaton of alternative models

```{r}

wage\_pooled <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = g7p, model = "pooling")

wage\_re <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = g7p, model = "random")

summary(wage\_re)

wage\_fe <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = g7p, model = "within")

summary(wage\_pooled)

summary(wage\_fe)

summary(wage\_re)

pFtest(wage\_fe, wage\_pooled)

plmtest(wage\_re)

names(g7p)[9] <- "gdp"

wage\_pooled <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = g7p, model = "pooling")

phtest(wage\_fe, wage\_re)

plmtest(wage\_pooled)

```

# employment - estimation and tests

```{r}

emp\_pooled\_g7 <- plm(unemployment ~ inflation + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "pooling")

emp\_fe\_g7 <- plm(unemployment ~ inflation + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "within")

emp\_re\_g7 <- plm(unemployment ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "random")

summary(wage\_re)

emp\_fe <- plm(unemployment ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "within")

summary(emp\_fe)

summary(emp\_re)

pFtest(emp\_fe, emp\_pooled)

plmtest(emp\_re)

summary(emp\_pooled\_g7)

summary(emp\_fe\_g7)

summary(emp\_re\_g7)

phtest(plm(unemployment ~ inflation + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "random"), plm(unemployment ~ inflation + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "within"))

plmtest(plm(unemployment ~ inflation + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = g7p, model = "random"))

```

# pooled estimation of different models

```{r}

emp\_pooled <- plm(unemployment ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = e7p, model = "pooling")

emp\_re <- plm(unemployment ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = e7p, model = "random")

summary(wage\_re)

emp\_fe <- plm(unemployment ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + wage + automation + robot + population, data = e7p, model = "within")

write.xlsx(round(summary(emp\_pooled)[[1]], 4), "istihdam\_e7.xlsx")

summary(emp\_fe)

summary(emp\_re)

pFtest(emp\_fe, emp\_pooled)

plmtest(emp\_re)

```

# pooled estimation of different models

```{r}

wage\_pooled <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "pooling")

wage\_fe <- plm(wage ~ inflation + investment\_ratio + fdi + skilled\_unskilled + trade\_deficit + education + gdp + unemployment + automation + robot + population, data = e7p, model = "within")

summary(emp\_fe)

write.xlsx(round(summary(emp\_fe)[[1]], 4), "istihdam\_sabit\_e7.xlsx")

summary(emp\_re)

write.xlsx(round(summary(emp\_re)[[1]], 4), "istihdam\_rassal\_e7.xlsx")

phtest(emp\_fe, emp\_re)

plmtest(wage\_re, type = "bp")

pchisq(7.9256, 1, lower.tail = FALSE)

e7p

```

# HAUSMAN TESTS

```{r}

phtest(tahmin\_fe\_ücret\_e7[[1]], tahmin\_re\_ücret\_e7[[1]])

hausman\_testleri\_e7 <- matrix(NA, 126, 3)

say <- 0

for (i in sıra\_e7) {

say <- say + 1

hausman\_testleri\_e7[say,1] <- phtest(tahmin\_fe\_ücret\_e7[[i]], tahmin\_re\_ücret\_e7[[i]])[[1]]

hausman\_testleri\_e7[say,2] <- phtest(tahmin\_fe\_ücret\_e7[[i]], tahmin\_re\_ücret\_e7[[i]])[[2]]

if (hausman\_testleri\_e7[say,2] < 0.05) {

hausman\_testleri\_e7[say, 3] <- 1

} else {

hausman\_testleri\_e7[say, 3] <- 2

}

}

colnames(hausman\_testleri\_e7) <- c("Ki-Kare Test İstatistiği", "p-değeri", "Geçerli Model")

rownames(hausman\_testleri\_e7) <- modeller

hausman\_testleri

write.xlsx(round(hausman\_testleri\_e7,4), "hausman\_ücret\_e7.xlsx")

```

# SERIAL CORRELATION TESTS

```{r}

sıra

library(lmtest)

otokorelasyon <- matrix(NA, 126, 2)

colnames(otokorelasyon) <- c("Test İstatistiği", "p-değeri")

rownames(otokorelasyon) <- paste0("model-", sıra)

say <- 0

for (i in sıra) {

say <- say + 1

otokorelasyon[say,1] <- bgtest(tahmin\_re\_ücret[[i]])[[1]]

otokorelasyon[say,2] <- bgtest(tahmin\_re\_ücret[[i]])[[4]]

}

write.xlsx(round(otokorelasyon, 4), "otokorelasyon\_g7.xlsx")

any(otokorelasyon[,2] < 0.5)

```

# SERIAL CORRELATION TESTS

```{r}

sıra

library(lmtest)

otokorelasyon\_e7 <- matrix(NA, 126, 2)

colnames(otokorelasyon\_e7) <- c("Test İstatistiği", "p-değeri")

rownames(otokorelasyon\_e7) <- paste0("model-", sıra\_e7)

say <- 0

for (i in sıra\_e7) {

say <- say + 1

otokorelasyon\_e7[say,1] <- bgtest(tahmin\_re\_ücret\_e7[[i]])[[1]]

otokorelasyon\_e7[say,2] <- bgtest(tahmin\_re\_ücret\_e7[[i]])[[4]]

}

write.xlsx(round(otokorelasyon, 4), "otokorelasyon\_e7.xlsx")

any(otokorelasyon[,2] < 0.5)

```

# HETEROSCEDASTICITY TESTS

```{r}

bptest(tahmin\_re\_ücret[[i]])

sıra

library(lmtest)

değişen\_varyans\_g7 <- matrix(NA, 126, 2)

colnames(değişen\_varyans\_g7) <- c("Test İstatistiği", "p-değeri")

rownames(değişen\_varyans\_g7) <- paste0("model-", sıra)

say <- 0

for (i in sıra) {

say <- say + 1

değişen\_varyans\_g7[say,1] <- bptest(tahmin\_re\_ücret[[i]])[[1]]

değişen\_varyans\_g7[say,2] <- bptest(tahmin\_re\_ücret[[i]])[[4]]

}

write.xlsx(round(değişen\_varyans\_g7, 4), "değişen\_varyans\_g7.xlsx")

any(değişen\_varyans\_g7[,2] < 0.05)

```

# HETEROSCEDASTICITY TESTS

```{r}

sıra

library(lmtest)

değişen\_varyans\_e7 <- matrix(NA, 126, 2)

colnames(değişen\_varyans\_e7) <- c("Test İstatistiği", "p-değeri")

rownames(değişen\_varyans\_e7) <- paste0("model-", sıra\_e7)

say <- 0

for (i in sıra\_e7) {

say <- say + 1

değişen\_varyans\_e7[say,1] <- bptest(tahmin\_re\_ücret\_e7[[i]])[[1]]

değişen\_varyans\_e7[say,2] <- bptest(tahmin\_re\_ücret\_e7[[i]])[[4]]

}

write.xlsx(round(değişen\_varyans\_e7, 4), "değişen\_varyans\_e7.xlsx")

any(değişen\_varyans\_e7[,2] < 0.05)

which((değişen\_varyans\_e7[,2] < 0.05))

```