Example 3-1

September 12, 2020

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

[15]:
```
##---------------------------------Block 1---------------------------------

#### Example 3-1 ####

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

#import the data and check to see if the panel is balanced
data("Tileries", package = "pder")
head(Tileries, 3)
pdim(Tileries)
```

id	week	area	output	labor	machine
2	1	fayoum	5.650487	4.532599	4.663439
2	2	fayoum	6.522328	5.347108	4.234107
2	3	fayoum	6.302619	4.969813	4.234107

```
Unbalanced Panel: n = 25, T = 12-22, N = 483
```

[16]:
```
##---------------------------------Block 2---------------------------------

Tileries <- pdata.frame(Tileries)

# within regression
plm.within <- plm(log(output) ~ log(labor) + log(machine), Tileries)

# create variables to estimate the within regression with transformed
# variables in OLS
y <- log(Tileries$output)
x1 <- log(Tileries$labor)
x2 <- log(Tileries$machine)
lm.within <- lm(I(y - Between(y)) ~ I(x1 - Between(x1)) + I(x2 - Between(x2)) -␣
↪1)
```

```
# ldsv version of the within regression
lm.lsdv <- lm(log(output) ~ log(labor) + log(machine) + factor(id), Tileries)
coef(lm.lsdv)[2:3]
coef(lm.within)
coef(plm.within)
```

log(labor) 0.870617084483074 **log(machine)** 0.0243774520324716

I(x1 - Between(x1)) 0.870617084483074 **I(x2 - Between(x2))** 0.0243774520324716

log(labor) 0.870617084483074 **log(machine)** 0.0243774520324716

[10]:
```
##---------------------------------Block 3---------------------------------

# one-way random effect model
tile.r <- plm(log(output) ~ log(labor) + log(machine), Tileries, model =␣
 ↪"random")
summary(tile.r)
```

```
Oneway (individual) effect Random Effect Model
   (Swamy-Arora's transformation)

Call:
plm(formula = log(output) ~ log(labor) + log(machine), data = Tileries,
    model = "random")

Unbalanced Panel: n = 25, T = 12-22, N = 483

Effects:
                    var   std.dev share
idiosyncratic 0.0026396 0.0513772 0.809
individual    0.0006232 0.0249641 0.191
theta:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.4892  0.5730  0.5820  0.5775  0.5903  0.5982

Residuals:
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-0.186639 -0.027250  0.003087  0.000006  0.033398  0.226778

Coefficients:
             Estimate Std. Error z-value  Pr(>|z|)
(Intercept)  0.277926   0.060769  4.5735 4.796e-06 ***
log(labor)   0.908793   0.030045 30.2477 < 2.2e-16 ***
log(machine) 0.023958   0.027049  0.8857    0.3758
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Total Sum of Squares:    4.845
```

```
Residual Sum of Squares: 1.2973
R-Squared:      0.73224
Adj. R-Squared: 0.73112
Chisq: 995.917 on 2 DF, p-value: < 2.22e-16
```

[11]:
```
##---------------------------------Block 4---------------------------------

# two-ways random effect model
plm.within <- plm(log(output) ~ log(labor) + log(machine),
                  Tileries, effect = "twoways")
lm.lsdv <- lm(log(output) ~ log(labor) + log(machine) +
                  factor(id) + factor(week), Tileries)
y <- log(Tileries$output)
x1 <- log(Tileries$labor)
x2 <- log(Tileries$machine)

# removing the individual and time means
y <- y - Between(y, "individual") - Between(y, "time") + mean(y)
x1 <- x1 - Between(x1, "individual") - Between(x1, "time") + mean(x1)
x2 <- x2 - Between(x2, "individual") - Between(x2, "time") + mean(x2)

# OLS model
lm.within <- lm(y ~ x1 + x2 - 1)

coef(plm.within)
coef(lm.within)
coef(lm.lsdv)[2:3]
```

log(labor) 0.869511305231281 **log(machine)** 0.0353861361840868

x1 0.880851667777117 **x2** 0.0355406590129144

log(labor) 0.86951130523128 **log(machine)** 0.035386136184087

[12]:
```
##---------------------------------Block 5---------------------------------

# 3 different individual random effects models
wh <- plm(log(output) ~ log(labor) + log(machine), Tileries,
          model = "random", random.method = "walhus",
          effect = "twoways")
am <- update(wh, random.method = "amemiya")
sa <- update(wh, random.method = "swar")

# variance of the error component for the Swamy and Aurora model
ercomp(sa)
```

```
                    var   std.dev share
idiosyncratic 0.0025892 0.0508844 0.768
```

```
individual    0.0006250 0.0250006 0.185
time          0.0001575 0.0125515 0.047
theta:
           Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
id    0.4934203 0.5769144 0.5857693 0.5813370 0.5940905 0.6019296
time  0.1961577 0.3460803 0.3544246 0.3487106 0.3624573 0.3701975
total 0.1665414 0.3023471 0.3096829 0.3058107 0.3186426 0.3294559
```

[13]:
```
##---------------------------------Block 6---------------------------------

# standard deviations of the variance of the error components for all models
re.models <- list(walhus = wh, amemiya = am, swar = sa)
sapply(re.models, function(x) sqrt(ercomp(x)$sigma2))
```

	walhus	amemiya	swar
idios	0.05167175	0.05088437	0.05088437
id	0.02778348	0.03191929	0.02500061
time	0.01177146	0.01267110	0.01255146

[14]:
```
##---------------------------------Block 7---------------------------------

# extract the coefficients of the three models
sapply(re.models, coef)
```

	walhus	amemiya	swar
(Intercept)	0.27420240	0.28560346	0.26527995
log(labor)	0.90777910	0.90061898	0.91278677
log(machine)	0.02696203	0.02774253	0.02692352