Panel Data Regression Analysis in R

Akif

2024-09-04

Introduction

In this analysis, we investigate the effectiveness of two treatments (A and B) on lowering diastolic blood pressure (DBP) using panel data regression models. We use both fixed and random effects models to compare the treatments over time.

#Import data to R

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
```

```
## Warning: package 'tidyr' was built under R version 4.3.3
```

```
## Warning: package 'purrr' was built under R version 4.3.3
```

```
## Warning: package 'lubridate' was built under R version 4.3.3
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be
come errors
```

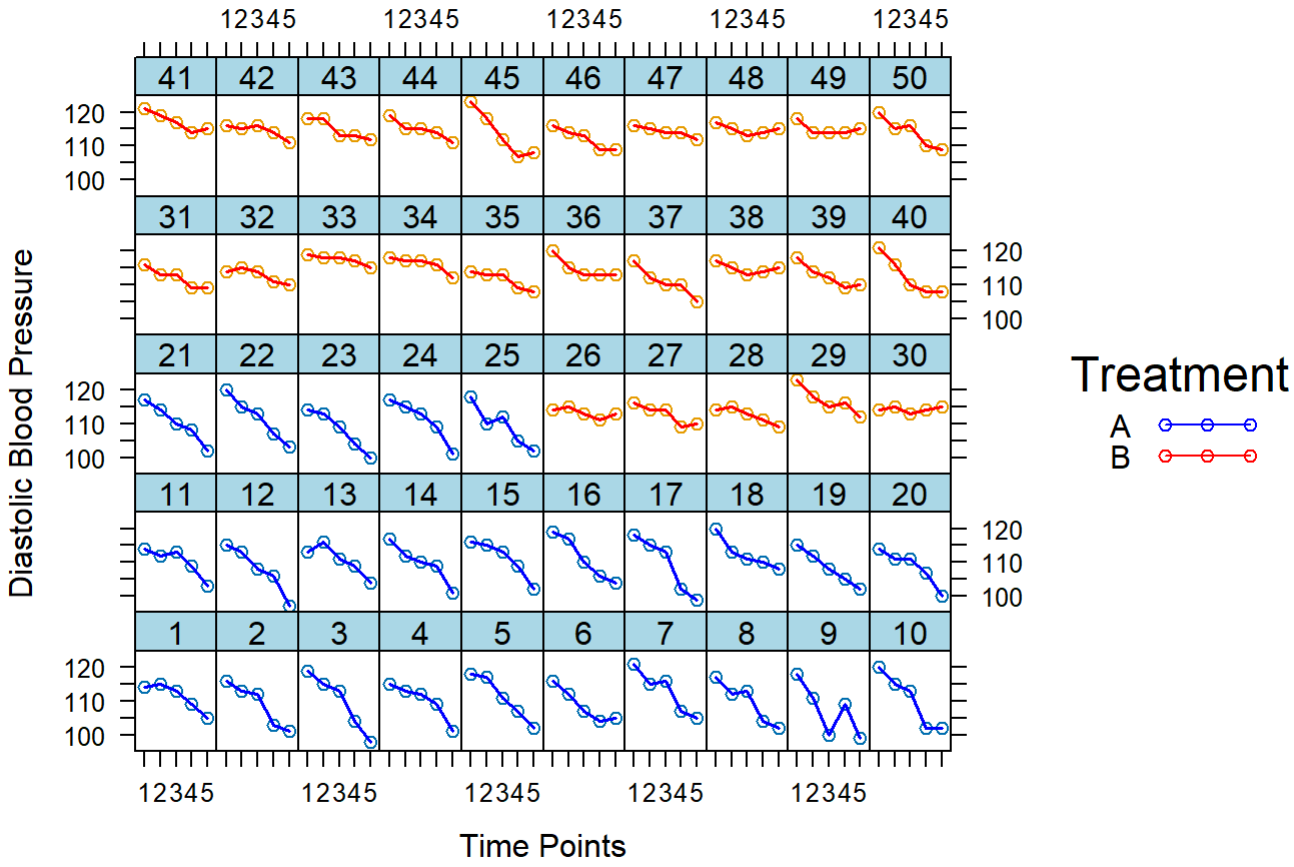
```
library(readxl)

dbp_data <- read_excel("C:/Users/akifi/OneDrive/Desktop/Medium/Panel Regression/dbp_data.xls
x", col_names = T)
head(dbp_data)
```

```
## # A tibble: 6 × 9
##      id treatment dbp_1 dbp_2 dbp_3 dbp_4 dbp_5   age sex
##   <dbl> <chr>     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr>
## 1     1 A           114   115   113   109   105    43 F
## 2     2 A           116   113   112   103   101    51 M
## 3     3 A           119   115   113   104    98    48 F
## 4     4 A           115   113   112   109   101    42 F
## 5     5 A           118   117   111   107   102    46 F
## 6     6 A           116   112   107   104   105    49 M
```

```
dbp_data <- pivot_longer(dbp_data, cols = starts_with("dbp"),
                         names_to = "time",
                         values_to = "dbp",
                         names_prefix = "dbp_")
dbp_data$time = as.numeric(dbp_data$time)
dbp_data <- as.data.frame(dbp_data) #save data as data frame
head(dbp_data)
```

```
##   id treatment age sex time dbp
## 1  1         A  43   F    1 114
## 2  1         A  43   F    2 115
## 3  1         A  43   F    3 113
## 4  1         A  43   F    4 109
## 5  1         A  43   F    5 105
## 6  2         A  51   M    1 116
```

```
#plotting the DBP of each individual across the five time points.
# first install lattice package if not already installed
#call the package
library(lattice)
xyplot(dbp ~ as.factor(time) | factor(id),
       data = dbp_data,
       type = "o",
       lwd = 1.5,
       layout = c(10, 5),
       xlab = "Time Points",
       ylab = "Diastolic Blood Pressure",
       groups = dbp_data$treatment, # Color lines by treatment group
       auto.key = list(space = "right", title = "Treatment", lines = TRUE, points = FALSE),
       par.settings = list(superpose.line = list(col = c("blue", "red"))), # Define colors fo
r the groups
       strip = strip.custom(bg = "lightblue")
)
```

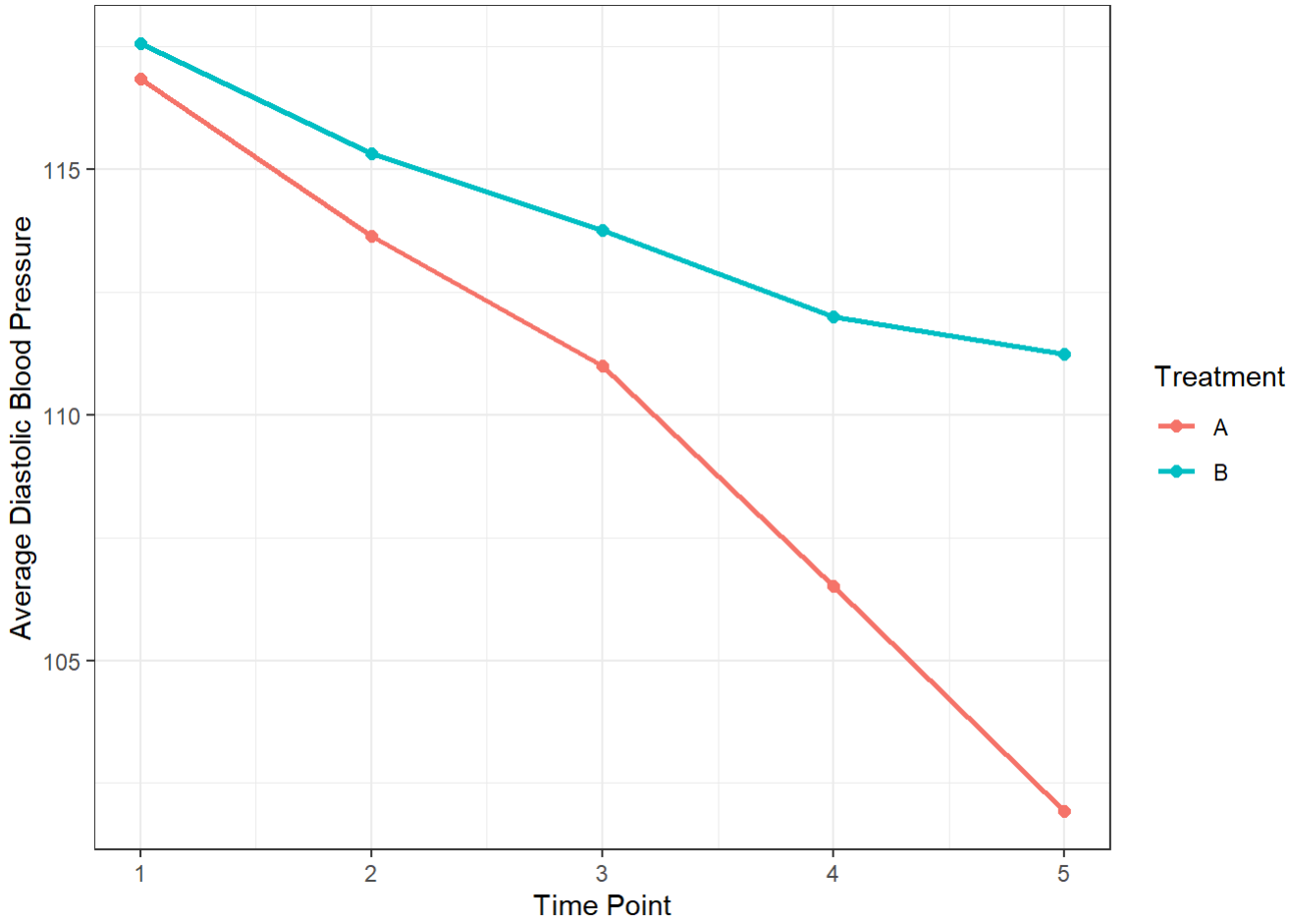



```
#Let us plot the average DBP at the 5 time points separately for treatment A and B and plot i
t against time.
mean_dbp <- dbp_data %>%
  group_by(time, treatment) %>%
  summarise(meandbp = mean(dbp)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'time'. You can override using the
## `.groups` argument.
```

```
ggplot(mean_dbp, aes(x = time, y = meandbp, color = treatment, group = treatment)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(x = "Time Point",
       y = "Average Diastolic Blood Pressure",
       color = "Treatment") +                 # Label the plot
  theme_bw()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```
#Runnig Fixed Effect Regression
library(plm)
```

```
## Warning: package 'plm' was built under R version 4.3.3
```

```
##
## Attaching package: 'plm'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```
m1.fe <- plm(dbp ~ time, data = dbp_data, index = c("id", "time"),
             model = "within")
summary(m1.fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = dbp ~ time, data = dbp_data, model = "within",
##     index = c("id", "time"))
##
## Balanced Panel: n = 50, T = 5, N = 250
##
## Residuals:
##    Min. 1st Qu.  Median 3rd Qu.    Max.
##  -7.800  -1.815   0.180   1.870   6.200
##
## Coefficients:
##        Estimate Std. Error  t-value  Pr(>|t|)
## time2  -2.72000    0.54443  -4.9961 1.292e-06 ***
## time3  -4.82000    0.54443  -8.8534 5.061e-16 ***
## time4  -7.94000    0.54443 -14.5842 < 2.2e-16 ***
## time5 -10.62000    0.54443 -19.5068 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    4963.2
## Residual Sum of Squares: 1452.4
## R-Squared:      0.70737
## Adj. R-Squared: 0.62825
## F-statistic: 118.449 on 4 and 196 DF, p-value: < 2.22e-16
```

```
#Running fixed effect regression with interaction between time and treatmetn
#making treatment "B" as the reference treatment for better interpretation
dbp_data$treatment <- relevel(factor(dbp_data$treatment), ref = "B")

#running the model with interaction term
m2.fe <- plm(dbp ~ time*treatment, data = dbp_data, index = c("id", "time"),
             model = "within")
summary(m2.fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = dbp ~ time * treatment, data = dbp_data, model = "within",
##     index = c("id", "time"))
##
## Balanced Panel: n = 50, T = 5, N = 250
##
## Residuals:
##    Min. 1st Qu.  Median 3rd Qu.    Max.
##  -8.416  -1.144   0.056   1.264   5.816
##
## Coefficients:
##                  Estimate Std. Error  t-value  Pr(>|t|)
## time2            -2.24000    0.59520  -3.7635 0.0002226 ***
## time3            -3.80000    0.59520  -6.3844 1.264e-09 ***
## time4            -5.56000    0.59520  -9.3414 < 2.2e-16 ***
## time5            -6.32000    0.59520 -10.6183 < 2.2e-16 ***
## time2:treatmentA -0.96000    0.84174  -1.1405 0.2554988
## time3:treatmentA -2.04000    0.84174  -2.4236 0.0162966 *
## time4:treatmentA -4.76000    0.84174  -5.6550 5.584e-08 ***
## time5:treatmentA -8.60000    0.84174 -10.2170 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    4963.2
## Residual Sum of Squares: 850.22
## R-Squared:      0.82869
## Adj. R-Squared: 0.77784
## F-statistic: 116.1 on 8 and 192 DF, p-value: < 2.22e-16
```

```
#Running Random Effect Regression Model
#with time and treatment as covariate

m1.re <- plm(dbp ~ time + treatment, data = dbp_data, index = c("id", "time"),
             model = "random")
summary(m1.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ time + treatment, data = dbp_data, model = "random",
##     index = c("id", "time"))
##
## Balanced Panel: n = 50, T = 5, N = 250
##
## Effects:
##                 var std.dev share
## idiosyncratic 7.410   2.722 0.881
## individual    1.000   1.000 0.119
## theta: 0.2273
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -9.796687 -1.772785 -0.094166  1.981222  6.373087
##
## Coefficients:
##              Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept) 119.19600    0.46674 255.3820 < 2.2e-16 ***
## time2        -2.72000    0.54443  -4.9961 5.851e-07 ***
## time3        -4.82000    0.54443  -8.8534 < 2.2e-16 ***
## time4        -7.94000    0.54443 -14.5842 < 2.2e-16 ***
## time5       -10.62000    0.54443 -19.5068 < 2.2e-16 ***
## treatmentA   -3.99200    0.44561  -8.9586 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    5913.6
## Residual Sum of Squares: 1808
## R-Squared:      0.69426
## Adj. R-Squared: 0.68799
## Chisq: 554.054 on 5 DF, p-value: < 2.22e-16
```

```
#with interaction between time and treatment
m2.re <- plm(dbp ~ time*treatment, data = dbp_data, index = c("id", "time"),
             model = "random")
summary(m2.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ time * treatment, data = dbp_data, model = "random",
##     index = c("id", "time"))
##
## Balanced Panel: n = 50, T = 5, N = 250
##
## Effects:
##                 var std.dev share
## idiosyncratic 4.428   2.104 0.735
## individual    1.596   1.263 0.265
## theta: 0.4027
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -9.959536 -1.238310 -0.071241  1.590768  5.591399
##
## Coefficients:
##                   Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept)      117.56000    0.49090 239.4766 < 2.2e-16 ***
## time2             -2.24000    0.59520  -3.7635 0.0001676 ***
## time3             -3.80000    0.59520  -6.3844 1.720e-10 ***
## time4             -5.56000    0.59520  -9.3414 < 2.2e-16 ***
## time5             -6.32000    0.59520 -10.6183 < 2.2e-16 ***
## treatmentA        -0.72000    0.69424  -1.0371 0.2996889
## time2:treatmentA  -0.96000    0.84174  -1.1405 0.2540782
## time3:treatmentA  -2.04000    0.84174  -2.4236 0.0153691 *
## time4:treatmentA  -4.76000    0.84174  -5.6550 1.559e-08 ***
## time5:treatmentA  -8.60000    0.84174 -10.2170 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    5531.1
## Residual Sum of Squares: 1062.8
## R-Squared:      0.80786
## Adj. R-Squared: 0.80065
## Chisq: 1009.06 on 9 DF, p-value: < 2.22e-16
```

```
#with sex and age as covariates
m3.re <- plm(dbp ~ time*treatment + age + sex, data = dbp_data, index = c("id", "time"),
             model = "random")
summary(m3.re)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = dbp ~ time * treatment + age + sex, data = dbp_data,
##     model = "random", index = c("id", "time"))
##
## Balanced Panel: n = 50, T = 5, N = 250
##
## Effects:
##                 var std.dev share
## idiosyncratic 4.428   2.104 0.791
## individual    1.168   1.081 0.209
## theta: 0.3433
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -10.353890  -1.251649  -0.098874   1.454012   6.188886
##
## Coefficients:
##                     Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept)      112.869393   1.638285  68.8948 < 2.2e-16 ***
## time2             -2.240000   0.595197  -3.7635 0.0001676 ***
## time3             -3.800000   0.595197  -6.3844 1.720e-10 ***
## time4             -5.560000   0.595197  -9.3414 < 2.2e-16 ***
## time5             -6.320000   0.595197 -10.6183 < 2.2e-16 ***
## treatmentA        -0.894178   0.673411  -1.3278 0.1842328
## age                0.104637   0.032849   3.1853 0.0014458 **
## sexM              -0.334059   0.412869  -0.8091 0.4184476
## time2:treatmentA  -0.960000   0.841736  -1.1405 0.2540782
## time3:treatmentA  -2.040000   0.841736  -2.4236 0.0153691 *
## time4:treatmentA  -4.760000   0.841736  -5.6550 1.559e-08 ***
## time5:treatmentA  -8.600000   0.841736 -10.2170 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    5649.6
## Residual Sum of Squares: 1053.9
## R-Squared:      0.81345
## Adj. R-Squared: 0.80483
## Chisq: 1037.81 on 11 DF, p-value: < 2.22e-16
```

```
#Hausman Test Example
# Example data
data("Grunfeld", package = "plm")


# Fixed effects model
fe_model <- plm(inv ~ value + capital, data = Grunfeld, model = "within")

# Random effects model
re_model <- plm(inv ~ value + capital, data = Grunfeld, model = "random")

# Hausman test
hausman_test <- phtest(fe_model, re_model)
print(hausman_test)
```

```
##
##  Hausman Test
##
## data:  inv ~ value + capital
## chisq = 2.3304, df = 2, p-value = 0.3119
## alternative hypothesis: one model is inconsistent
```