Panel data Analysis

A Fixed Effects Approach

Litta Jose Thottam,135546

Contents

Introduction to Fixed Effects for Panel Data

In econometrics and statistical analysis, fixed effects models are a powerful tool for analyzing panel data. Panel data, also known as longitudinal data, consists of observations on multiple entities (such as individuals, firms, or countries) over multiple time periods. This structure provides a rich dataset that allows researchers to account for both cross-sectional and temporal variations.

Fixed effects models are designed to control for unobserved heterogeneity when this heterogeneity is constant over time and correlated with the independent variables. The essential idea is to isolate the impact of the explanatory variables by controlling for all time-invariant characteristics of the entities.

Essential Meaning

The core principle of fixed effects is to focus on within-entity variations. By doing so, the model effectively removes the influence of all entity-specific attributes that do not change over time, such as cultural factors, innate abilities, or other intrinsic properties. This is achieved through a transformation that demeans the data, subtracting the entity-specific mean of each variable.

Benefits

Control for Unobserved Heterogeneity: Fixed effects models account for unobservable factors that differ between entities but remain constant over time. This helps in reducing bias in the estimated coefficients.

Reduction of Omitted Variable Bias: By controlling for entity-specific effects, fixed effects models mitigate the risk of omitted variable bias, which occurs when a model leaves out an important variable that is correlated with both the dependent and independent variables.

Improved Causal Inference: Fixed effects models enhance the reliability of causal inference by controlling for time-invariant characteristics, allowing researchers to more confidently attribute changes in the dependent variable to changes in the independent variables.

Flexibility in Application: These models are widely applicable across various fields such as economics, sociology, political science, and public health, making them a versatile tool for panel data analysis.

Cross-sectional Data (2 pt)

Suppose you want to learn the **effect of price on the demand** for back massages. Read in the following data from four Midwest locations (call it `crosssection`). please create the plot

```
# Crosssection
crosssection <- data.frame(Location = c("Chicago", "Peoria", "Milwaukee", "Madison"),
                           Year = rep(2003, 4),
                           Price = c(75, 50, 60, 55),
                           Quantity = c(2.0, 1.0, 1.5, 0.8))
```

Create the table with the `gt` package. Use `tab_header()` to set a header and `cols_label()` to label columns.

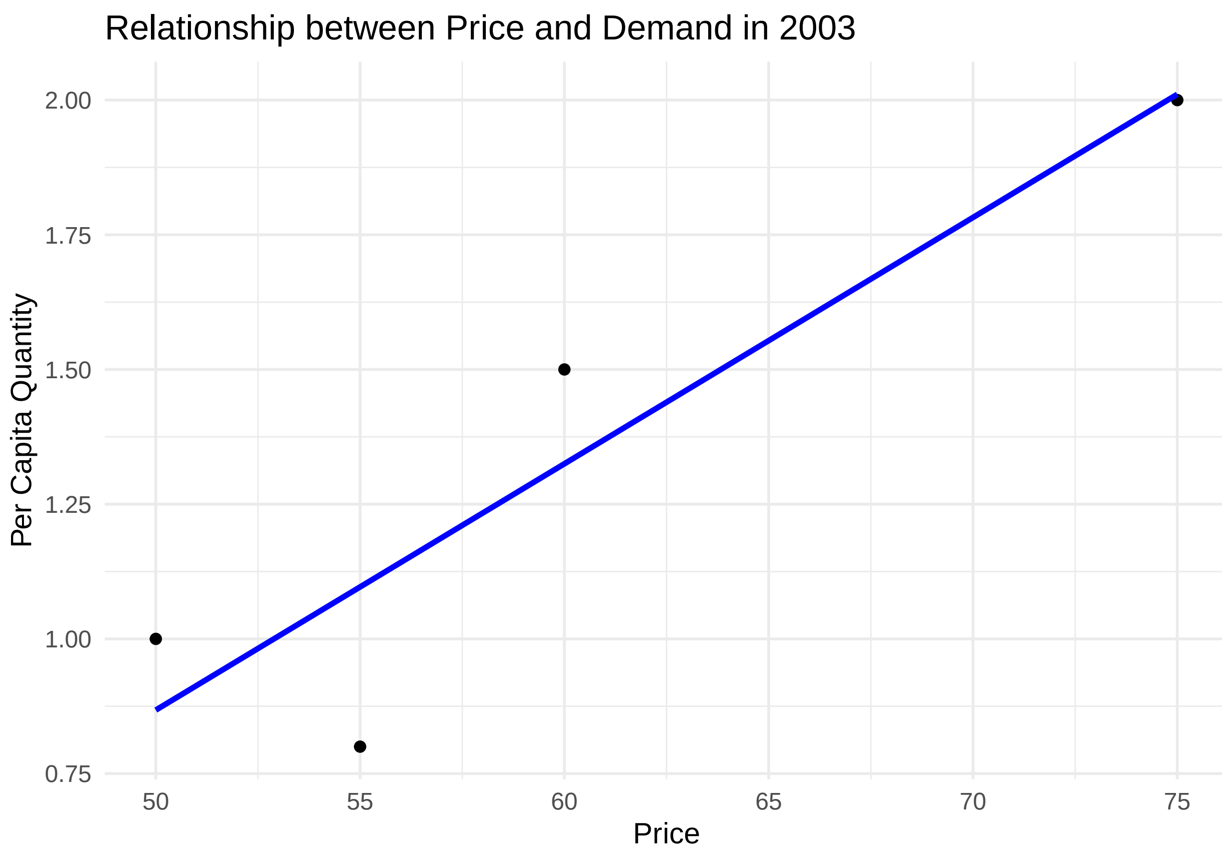
Table 1: Cross-sectional Data

Location	Year	Price	Per CapitaQuantity
Chicago	2003	75	2.0
Peoria	2003	50	1.0
Milwaukee	2003	60	1.5
Madison	2003	55	0.8

```
# Load necessary libraries
# Load necessary libraries
library(gt)
library(ggplot2)
## Warning: package 'ggplot2' was built under R version 4.2.3


# Create the data frame
crosssection <- data.frame(
  Location = c("Chicago", "Peoria", "Milwaukee", "Madison"),
  Year = rep(2003, 4),
  Price = c(75, 50, 60, 55),
  Quantity = c(2.0, 1.0, 1.5, 0.8)
)
```

```r
# Create the plot using ggplot2
ggplot(crosssection, aes(x = Price, y = Quantity)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(
    title = "Relationship between Price and Demand in 2003",
    x = "Price",
    y = "Per Capita Quantity"
  ) +
  theme_minimal()
## `geom_smooth()` using formula = 'y ~ x'
```



From this plot a clear positive relationship is visible.

Panel Data (2 pt)

Read in additional data from Table 2 (call it `paneldata`).

```r
library(ggplot2)
library(tidyverse)

# Panel Data
```

```r
paneldata <- data.frame(Location = c("Chicago", "Chicago", "Peoria", "Peoria",
                                     "Milwaukee", "Milwaukee", "Madison", "Madison"),
                        Year = rep(2003:2004, 4),
                        Price = c(75, 85, 50, 48, 60, 65, 55, 60),
                        Quantity = c(2.0, 1.8, 1.0, 1.1, 1.5, 1.4, 0.8, 0.7))
```

create the table

```r
# Load necessary libraries
library(gt)

# Create the data frame
paneldata <- data.frame(
  Location = c("Chicago", "Chicago", "Peoria", "Peoria",
               "Milwaukee", "Milwaukee", "Madison", "Madison"),
  Year = rep(2003:2004, 4),
  Price = c(75, 85, 50, 48, 60, 65, 55, 60),
  Quantity = c(2.0, 1.8, 1.0, 1.1, 1.5, 1.4, 0.8, 0.7)
)

# Create the table using gt
gt_table <- gt(paneldata) %>%
  tab_header(
    title = "Table 2: Panel Data"
  ) %>%
  cols_label(
      Location = md("**Location**"),
    Year = md("**Year**"),
    Price = md("**Price**"),
    Quantity = md("**Per Capita**<br>**Quantity**")
  ) %>%
  cols_align(
    align = "center",
    columns = vars(Quantity)
  )

# Display the table
gt_table
```

Table 2: Panel Data

Location	**Year**	**Price**	**Per CapitaQuantity**
Chicago	2003	75	2.0
Chicago	2004	85	1.8
Peoria	2003	50	1.0
Peoria	2004	48	1.1
Milwaukee	2003	60	1.5
Milwaukee	2004	65	1.4
Madison	2003	55	0.8

Madison	2004	60	0.7

please create the plot

```r
# Load necessary libraries
library(ggplot2)
library(tidyverse)

# Panel Data
paneldata <- data.frame(
  Location = c("Chicago", "Chicago", "Peoria", "Peoria",
               "Milwaukee", "Milwaukee", "Madison", "Madison"),
  Year = rep(2003:2004, 4),
  Price = c(75, 85, 50, 48, 60, 65, 55, 60),
  Quantity = c(2.0, 1.8, 1.0, 1.1, 1.5, 1.4, 0.8, 0.7)
)

# Create the plot using ggplot2
ggplot(paneldata, aes(x = Price, y = Quantity, color = Location)) +
  geom_point(size = 3) +
  geom_line(aes(group = Location), size = 1) +
  geom_smooth(method = "lm", se = FALSE, color = "blue", size = 1.5) +  # Add the overall trend line
  labs(
    title = "Relationship between Price and Quantity",
    x = "Price",
    y = "Quantity"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.title = element_blank(),
    legend.position = "right"
  )
```

The plot demonstrates the relationship between price and quantity for four locations (Chicago, Madison, Milwaukee, and Peoria) over the years 2003 and 2004. Here are the key takeaways:

1.**Overall Positive Trend**: The blue line indicates a general positive relationship between price and quantity when considering all locations together. This suggests that, on average, higher prices are associated with higher quantities.

2.**Location-Specific Negative Trends**: The colored lines connecting the points for each location show that within each location, the quantity decreases as the price increases from 2003 to 2004. This suggests a negative relationship between price and quantity at the individual location level.

Fixed Effects Regression

Manual FD (1 pt)

Please add two columns, i.e. the change in price "(Δ)" and the change in quantity "(Δ)" to your dataframe. Create the following table:

```
## Data for Difference Equation Estimation

library(dplyr)
library(knitr)
library(kableExtra)
## Warning: package 'kableExtra' was built under R version 4.2.3
##
```

```
## Attaching package: 'kableExtra'
## The following object is masked from 'package:dplyr':
##
##     group_rows

# Create the dataframe
data <- data.frame(
  Location = c("Chicago", "Chicago", "Peoria", "Peoria", "Milwaukee", "Milwaukee", "Madison", "Madison")
  Year = c(2003, 2004, 2003, 2004, 2003, 2004, 2003, 2004),
  Price = c(75, 85, 50, 48, 60, 65, 55, 60),
  Quantity = c(2.0, 1.8, 1.0, 1.1, 1.5, 1.4, 0.8, 0.7)
)

# Calculate changes in Price (ΔP) and Quantity (ΔQ)
data <- data %>%
  group_by(Location) %>%
  mutate(
    ΔP = c(NA, diff(Price)),
    ΔQ = c(NA, diff(Quantity))
  )

# Replace 0 values with NA
data <- data %>%
  mutate(
    ΔP = ifelse(ΔP == 0, NA, ΔP),
    ΔQ = ifelse(ΔQ == 0, NA, ΔQ)
  )

# Create the table with a centered title, caption, and footnote
kable(data, caption = "Table 3: Data for Difference Equation Estimation") %>%
  kable_styling(full_width = FALSE) %>%
  add_header_above(c(" " = 2, "Data for Difference Equation Estimation" = 4), bold = TRUE, align = "c")
  footnote(general = "Note: ΔP and ΔQ represent the change in Price and Quantity, respectively.")
```

Table 3: Table 3: Data for Difference Equation Estimation

		Data for Difference Equation Estimation			
Location	Year	Price	Quantity	ΔP	ΔQ
Chicago	2003	75	2.0	NA	NA
Chicago	2004	85	1.8	10	-0.2
Peoria	2003	50	1.0	NA	NA
Peoria	2004	48	1.1	-2	0.1
Milwaukee	2003	60	1.5	NA	NA
Milwaukee	2004	65	1.4	5	-0.1
Madison	2003	55	0.8	NA	NA
Madison	2004	60	0.7	5	-0.1

Note:

Note: ΔP and ΔQ represent the change in Price and Quantity, respectively.

The modelsummary package creates nice tables from a `dataframe` with `datasummary_df()`.

First Difference (1 pt)

Run a first difference estimation by regressing change in quantity on change in price (1 pt). Please exclude the intercept in the estimation.

```
library(dplyr)
library(knitr)

# Create the dataframe
data <- data.frame(
  Location = c("Chicago", "Chicago", "Peoria", "Peoria", "Milwaukee", "Milwaukee", "Madison", "Madison")
  Year = c(2003, 2004, 2003, 2004, 2003, 2004, 2003, 2004),
  Price = c(75, 85, 50, 48, 60, 65, 55, 60),
  Quantity = c(2.0, 1.8, 1.0, 1.1, 1.5, 1.4, 0.8, 0.7)
)

# Calculate changes in Price (ΔP) and Quantity (ΔQ)
data <- data %>%
  group_by(Location) %>%
  mutate(
    ΔP = c(NA, diff(Price)),
    ΔQ = c(NA, diff(Quantity))
  )

# Replace 0 values with NA
data <- data %>%
  mutate(
    ΔP = ifelse(ΔP == 0, NA, ΔP),
    ΔQ = ifelse(ΔQ == 0, NA, ΔQ)
  )

# Remove rows with NA in ΔP or ΔQ
data_clean <- data %>%
  filter(!is.na(ΔP) & !is.na(ΔQ))

# Perform the regression excluding the intercept
model_fd <- lm(ΔQ ~ ΔP - 1, data = data_clean)

# Display the summary of the regression
summary(model_fd)
##
## Call:
## lm(formula = ΔQ ~ ΔP - 1, data = data_clean)
##
## Residuals:
##        1        2        3        4
## 0.007792 0.058442 0.003896 0.003896
##
## Coefficients:
##     Estimate Std. Error t value Pr(>|t|)
## ΔP -0.020779   0.002755  -7.542  0.00483 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.03419 on 3 degrees of freedom
## Multiple R-squared:  0.9499, Adjusted R-squared:  0.9332
## F-statistic: 56.89 on 1 and 3 DF,  p-value: 0.004832
```

Manual Time Demeaning (1 pt)

Please prepare the data for a manual time demeaning. Thus replicate the following table.

Calculate the mean of Q and P per individual (group). Subtract the individual mean from each observation. Regress these time-demeaned observation (with standard lm() command). Please exclude the intercept in the estimation.

```r
# Calculate the mean of Q and P per individual (group)
means <- data %>%
  group_by(Location) %>%
  summarize(
    mean_P = mean(Price),
    mean_Q = mean(Quantity)
  )

# Merge the means with the original data
data_demean <- data %>%
  left_join(means, by = "Location")

# Subtract the individual mean from each observation
data_demean <- data_demean %>%
  mutate(
    demeaned_P = Price - mean_P,
    demeaned_Q = Quantity - mean_Q
  )

# Display the prepared data
kable(data_demean, caption = "Data for Manual Time Demeaning")
```

Table 4: Data for Manual Time Demeaning

Location	Year	Price	Quantity	ΔP	ΔQ	mean_P	mean_Q	demeaned_P	demeaned_Q
Chicago	2003	75	2.0	NA	NA	80.0	1.90	-5.0	0.10
Chicago	2004	85	1.8	10	-0.2	80.0	1.90	5.0	-0.10
Peoria	2003	50	1.0	NA	NA	49.0	1.05	1.0	-0.05
Peoria	2004	48	1.1	-2	0.1	49.0	1.05	-1.0	0.05
Milwaukee	2003	60	1.5	NA	NA	62.5	1.45	-2.5	0.05
Milwaukee	2004	65	1.4	5	-0.1	62.5	1.45	2.5	-0.05
Madison	2003	55	0.8	NA	NA	57.5	0.75	-2.5	0.05
Madison	2004	60	0.7	5	-0.1	57.5	0.75	2.5	-0.05

```r
# Perform the regression excluding the intercept
model_td <- lm(demeaned_Q ~ demeaned_P - 1, data = data_demean)

# Display the summary of the regression
```

```r
summary(model_td)
##
## Call:
## lm(formula = demeaned_Q ~ demeaned_P - 1, data = data_demean)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.029221 -0.002435  0.000000  0.002435  0.029221
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## demeaned_P -0.020779   0.001804  -11.52 8.35e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01583 on 7 degrees of freedom
## Multiple R-squared:  0.9499, Adjusted R-squared:  0.9428
## F-statistic: 132.7 on 1 and 7 DF,  p-value: 8.352e-06
```

Time Demeaning (1 pt)

Run a within fixed effects regression with the function plm() from plm package. Please exclude the intercept in the estimation.

```r
library(plm)

# Create a pdata.frame for plm
pdata <- pdata.frame(data, index = c("Location", "Year"))

# Perform the within fixed effects regression excluding the intercept
model_fe <- plm(Quantity ~ Price - 1, data = pdata, model = "within")

# Display the summary of the regression
summary(model_fe)
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = Quantity ~ Price - 1, data = pdata, model = "within")
##
## Balanced Panel: n = 4, T = 2, N = 8
##
## Residuals:
##   Chicago-2003   Chicago-2004   Madison-2003   Madison-2004 Milwaukee-2003
##     -0.0038961      0.0038961     -0.0019481      0.0019481     -0.0019481
## Milwaukee-2004    Peoria-2003    Peoria-2004
##      0.0019481     -0.0292208      0.0292208
##
## Coefficients:
##        Estimate Std. Error t-value Pr(>|t|)
## Price -0.020779   0.002755 -7.5425 0.004832 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Total Sum of Squares:    0.035
## Residual Sum of Squares: 0.0017532
## R-Squared:      0.94991
## Adj. R-Squared: 0.88312
## F-statistic: 56.8889 on 1 and 3 DF, p-value: 0.0048318
```

LSDV (1 pt)

Run a Least Square Dummy Variable (LSDV) regression, i.e. include all locations as dummy variables in your standard lm regression. Please exclude the intercept in the estimation

```
# Create dummy variables for Location
data_lsdv <- data %>%
  mutate(
    Chicago = ifelse(Location == "Chicago", 1, 0),
    Peoria = ifelse(Location == "Peoria", 1, 0),
    Milwaukee = ifelse(Location == "Milwaukee", 1, 0),
    Madison = ifelse(Location == "Madison", 1, 0)
  )

# Perform the LSDV regression excluding the intercept
model_lsdv <- lm(Quantity ~ Price + Chicago + Peoria + Milwaukee + Madison - 1, data = data_lsdv)

# Display the summary of the regression
summary(model_lsdv)
##
## Call:
## lm(formula = Quantity ~ Price + Chicago + Peoria + Milwaukee +
##     Madison - 1, data = data_lsdv)
##
## Residuals:
##         1         2         3         4         5         6         7         8
## -0.003896  0.003896 -0.029221  0.029221 -0.001948  0.001948 -0.001948  0.001948
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## Price     -0.020779   0.002755  -7.542 0.004832 **
## Chicago    3.562338   0.221059  16.115 0.000520 ***
## Peoria     2.068182   0.136071  15.199 0.000618 ***
## Milwaukee  2.748701   0.173032  15.886 0.000542 ***
## Madison    1.944805   0.159330  12.206 0.001184 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02417 on 3 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9997
## F-statistic:  5061 on 5 and 3 DF,  p-value: 4.382e-06
```

Comparison (1 pt)

Create a model overview with modelsummary package. Show all 5 models. Drop the coefficients of location dummies in the LSDV model. Drop the goodness-of-fit statistics (except for n, R2, adjR2). Rename the coefficient in the FD model into "Price" such that all price coefficient are the same. Relabel the models according their names. Your result should look like this:

```
## Model Comparison

library(modelsummary)

# Create the Pooling model without intercept
pooling_model <- lm(Quantity ~ Price - 1, data = data)

# Summary of the pooling model to verify
summary(pooling_model)
##
## Call:
## lm(formula = Quantity ~ Price - 1, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55741 -0.12404  0.02823  0.13120  0.42823
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## Price 0.020957   0.001753   11.96 6.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.314 on 7 degrees of freedom
## Multiple R-squared:  0.9533, Adjusted R-squared:  0.9467
## F-statistic:   143 on 1 and 7 DF,  p-value: 6.509e-06

# Prepare models for comparison
models <- list(
  "Pooling" = pooling_model,
  "First Difference" = model_fd,
  "Manual Time Demeaning" = model_td,
  "Within Fixed Effects" = model_fe,
  "LSDV" = model_lsdv
)

# Define a custom gof_map to include R2 and adjusted R2 manually for Pooling model
custom_gof <- function(model, ...) {
  if (identical(model, pooling_model)) {
    return(c("R2" = 0.953, "adjR2" = 0.947))
  } else {
    return(c("R2" = summary(model)$r.squared, "adjR2" = summary(model)$adj.r.squared))
  }
}

# Drop location dummies from LSDV model
```

Model Overview

	Pooling	First Difference	Manual Time Demeaning	Within Fixed Effects	LSDV
Price	0.021***	−0.021**	−0.021***	−0.021**	−0.021**
	(0.002)	(0.003)	(0.002)	(0.003)	(0.003)
Num.Obs.	8	4	8	8	8
R2	0.953	0.950	0.950	0.950	1.000
R2 Adj.	0.947	0.933	0.943	0.883	1.000

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001
Note: The coefficients of location dummies in the LSDV model are omitted from the table.

```
coef_map <- c(
  "Price" = "Price",
  "ΔP" = "Price",
  "demeaned_P" = "Price"
)

modelsummary(models,
             coef_map = c("Price" = "Price", "demeaned_P" = "Price", "ΔP" = "Price"),
             coef_omit = "Chicago|Peoria|Milwaukee|Madison",

             gof_omit = "AIC|BIC|Log.Lik|F|sigma|RMSE",
             stars = TRUE,
             notes = "Note: The coefficients of location dummies in the LSDV model are omitted from the
             title = "Model Overview"
)
```