Unit 11 Live Session (Solutions)

Analysis of Panel Data: An Introduction

Figure 1: South Hall

Class Announcements

- Congratulations on finishing the second part of the course!
- HW 11 is out this week
- Lab-3 is due in 3 weeks

Roadmap

Rearview Mirror

- Univariate Time Series Models
- Multivariate Time Series Models

Today

- Introduction to panel data
- Exploratory panel data analysis
- Pooled OLS models
- First-Difference models

Looking Ahead

- Fixed effect and random effect models
- Linear mixed-effect model

Start-up Code

```
# Insert the function to *tidy up* the code when they are printed out
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)

# Load libraries

## Load a set of packages inclusing: broom, cli, crayon, dbplyr , dplyr, dtplyr, forcats,
#googledrive, googlesheets4, ggplot2, haven, hms, httr, jsonlite, lubridate , magrittr,
#modelr, pillar, purrr, readr, readxl, reprex, rlang, rstudioapi, rvest, stringr, tibble,
#tidyr, xml2
library(tidyverse)

# Provide a set of estimators for models and (robust) covariance matrices and tests for panel data econometrics,
library(plm)

## Functions, data sets, examples, demos, and vignettes for the book Christian Kleiber and Achim Zeileis (2008),
#Applied Econometrics with R
library(AER)

## provides geoms for ggplot2 to repel overlapping text labels.
library(ggrepel)
library(stargazer)
library(gridExtra)
```

Introduction to Panel Data

Panel data combines cross-sectional and time series data: the same individuals (persons, firms, cities, etc.) are observed at several points in time (days, years, before and after treatment, etc.).

Panel data allows us to control for individual characteristics that we cannot observe or measure, like:

- Cultural (like country or region-specific) factors;
- Difference in business practices across companies;

or variables that change over time but not across individuals:

- National policies;
- Federal regulations;
- International agreements;

Panel structure

Panel data includes N individuals observed at T regular periods. There are three general types of panel data:

- Short panel: many individuals (large N) over a few periods (small T) (we use this case in class)
- Long panel: many periods (large T) and few individuals (small N)
- Both: many periods and many individuals (large N and large T)

Panel data can be balanced when all individuals are observed in all periods ($T_i = T$ for all i) or unbalanced when individuals are not observed in all periods ($T_i \neq T$).

Analyzing unbalanced panel data typically raises a few additional issues compared with analysis of balanced data. For example, if the panel is unbalanced for reasons that are not entirely random (e.g., because firms with relatively low levels of productivity have relatively high exit rates), then we need to consider this when estimating the model.

Repeated cross-sections are not the same as panel data. Repeated cross-sections are obtained by sampling from the same population at different points in time. The identity of the individuals (or firms, households, etc.) is not recorded, and there is no attempt to follow the same individuals over time. If each cross-section is drawn independently, combining the resulting random sample gives us an independent cross-section that can be modeled using OLS.

Framework for Panel Data

- Consider the multiple linear regression model for individual $i = 1, ..., N$ who is observed at several time periods $t = 1, ..., T$:

$$y_{it} = \beta_0 + \beta_1 x_{it} + \gamma_i + u_{it}$$

where:

- y_{it}: a dependent variable
- x_{it}: an explanatory variable
- γ_i: an unobserved individual-specific effect(time-invariant)
- u_{it}: - an idiosyncratic error term (observation-specific zero-mean random-error term, analogous to the random-error term of cross-sectional regression analysis).

Estimation Methods

Panel data models describe the individual behavior both across time and across individuals. We can consider three panel data models depending on the time-invariant unobserved effect and its relation to the other independent variables. These are:

1- Pooled Cross Sections

- When there is no time-invariant unobserved effect

2- Fixed Effects ("Within") Model:

- When there is a time-invariant unobserved effect

- The time-invariant unobserved effect is correlated with the explanatory variables

3- Random Effects Model

- When there is a time-invariant unobserved effect

- The time-invariant unobserved effect is uncorrelated with the explanatory variables

The Pooled Cross Sections

The pooled model simply applies an OLS estimate to the pooled data set where each individual i's data is ordered from $t = 1, \ldots, T$, and then vertically stacked.

For pooled OLS to be the appropriate estimator, we need to assume:

1- **Linearity**: the model is linear in parameters

2- **i.i.d.** : The observations are independent across individuals but not necessarily across time. This is guaranteed by random sampling of individuals.

3- **Indentifiability**: the regressors, including a constant, are not perfectly collinear, and all regressors (but the constant) have non-zero variance and not too many extreme values.

4- $x_i t$ is uncorrelated with idiosyncratic error term u_{it} and individual-specific effect γ_i

a)

$$E(u_{it} x_{it}) = 0$$

b)

$$E(x_{it}, \gamma_i) = 0$$

The pooled OLS estimator is consistent under assumptions 1-4.

We need to assume homoskedasticity and no serial correlation in the data to do inference based on the conventional OLS estimator of the covariance matrix. Both of these assumptions can be restrictive. Therefore, it is a good idea to be conservative and obtain an estimate of the covariance matrix that is robust to heteroskedasticity and autocorrelation.

a)- What are the main issues of pooled OLS?

- The main problem is that we do not observe γ_i, which is constant over time for each individual (hence no t subscript) but varies across individuals. Hence if we estimate the model in levels using OLS then γ_i will go into the error term: $\epsilon_{it} = \gamma_i + u_{it}$.
- If $gamma_i$ is correlated with x_{it}, then putting γ_i in the error term can cause serious problems. This, of course, is **an omitted variable problem**. For the single regressor model:

$$plim\widehat{\beta_{ols}} = \beta + \frac{cov(x_{it}, \gamma_i)}{\sigma_x^2}$$

- which shows that the OLS estimator is inconsistent unless $cov(x_{it}, \gamma_i) = 0$. If $x_i t$ is positively correlated with the unobserved effect, then there is an upward bias. If the correlation is negative, we get a negative bias.
- **In this case, the fixed effect model is preferred because it is consistent in the case of these unobserved individual characteristics.**
- But if γ_i is uncorrelated with x_{it}, then γ_i is just another unobserved factor making up the residual. However, OLS will not be efficient (smallest variance) because the error term ϵ_{it} is serially correlated:

$$corr(\epsilon_{it}, \epsilon_{it-s}) = \frac{\sigma_\gamma}{\sigma_\gamma^2 + \sigma_u^2}$$

- In this case, OLS is still consistent. However, the standard formula for calculating the standard errors is wrong.
- **, In this case, the random effects model is more efficient.**

Fixed Effects Model

- The fixed effects (FE) model is commonly applied to remove omitted variable bias in the case of unobserved individual characteristics. By estimating changes within a specific group (over time), all time-invariant differences between entities (individuals, firms, ...) are controlled for.
- The fixed effect model(FD) could be estimated using three estimation methods:

1) Least Squares Dummy Variable Estimation (LSDV)
2) First-difference Estimator (FD)
3) Fixed Effect or Within-groups Estimator (FE) (Next week)

All fixed effect estimation methods are consistent under the following assumptions:

1- **Linearity**: the model is linear in parameters

2- **i.i.d.** : The observations are independent across individuals but not necessarily across time. This is guaranteed by random sampling of individuals.

3- **Indentifiability**: the regressors, including a constant, are not perfectly collinear, and all regressors (but the constant) have non-zero variance and not too many extreme values.

4- **Zero conditional means (strict exogeneity)**

$E(x_{it}, u_{is}) = 0$ for $s = 1, 2, 3, ..., T$

- Under the above assumptions, we can use the Fixed Effects (FE) estimators to obtain consistent estimates of β, allowing unobserved individual-specific γ_i to be freely correlated with x_{it}.
- Note that strict exogeneity rules out feedback from past u_{is} shocks to current x_{it}. One implication of this is that estimators will not yield consistent estimates if x_{it} depends on lagged dependent variables $(y_{it-1}, y_{it-2}, ...)$ as in the case of a VAR model.

Least Squares Dummy Variable Estimator (LSDV)

- Least Squares Dummy Variable Estimator assumes different intercepts for each individual by including one dummy variable.
- If our N is large so that we have a large number of dummy variables, this may not be a very practical approach, and the Within-groups Estimator(FE) is a better option with the same results.

$$y_{it} = \beta_0 + \beta_1 x_{it} + \sum_{i=2}^{N} \beta_i I_i + u_{it}$$

- Where I_i is an indicator variable.

The First Differencing Estimator (FD)

A simple approach to rid the model of the individual-specific effects γ_i is first differencing. Subtracting the lagged value y_{it-1} from the initial model:

$$y_{it} - y_{it-1} = (\beta_0 - \beta_0) + \beta_1(x_{it} - x_{it-1}) + (\gamma_i - \gamma_i) + (u_i t - u_{it-1})$$

$$\Delta y_{it} = \beta_1 \Delta x_{it} + \Delta u_{it}$$

Note that the individual-specific effect γ_i , the intercept β_0, and the parameters β_0 are not estimated by the FD estimator because they are time-invariant and therefore cancel out when differencing.

The Within Estimator

In the within estimator, we first demean the variables to remove group averages and then run our regression, which eliminates the fixed effect coefficients γ_i:

$$(y_{it} - \bar{y}_i) = \beta_1(x_{1it} - \bar{x}_{1i}) + ... + \beta_p(x_{pit} - \bar{x}_{pi}) + (\epsilon_{it} - \bar{\epsilon}_i)$$

This will produce equivalent results to running a regression with the fixed effects per group included as dummy variables (LSDV), but it can be faster to run things this way when there are many groups.

Using R for panel data

In this course, we use the plm package to estimate various specifications and to conduct various specification tests of panel data models. You can find more information about this package and its functionality in the following link:

https://cran.r-project.org/web/packages/plm/vignettes/A_plmPackage.html

Estimating panel data models with the plm package requires that the data sets are in "long format." If a data set is in "wide format," it can be converted to "long format" using `pivot_longer()` from the `tidyr` package or `melt` from the `reshape2` package.

- the long-form has a column for each variable and a row for each individual-period (in our example below: state-year).
- The wide form has a column for each variable-period and a row for each individual (in our example below: state).

Then we need to create a panel structure for the dataset using the function `pdata.frame` in the `plm` package. The `pdata.frame()` function adds information about the panel data structure in the following ways:

1) Convert a data frame class to "pdata.frame"
2) Convert the class of each individual variable to "pseries", based on the original class of the variable
3) Convert the row names so that they indicate the individual identifier and the time identifier.
4) Converts the variables that identify the individuals and the periods to categorical class.

Using the `pdata.frame()` is not necessary if the first variable of the data set is the individual identifier and the second variable is the time identifier. The package can then infer the structure automatically.

"Analysis of Panel Data Using R" by A. Henningsen and G. Henningsen has more details if interested. The electronic version is available in the UC Berkeley library.

We estimate different panel data models using the `plm()` function from the `plm` package.

For an unbalanced panel, we could use `make.pbalanced(data, balance.type)` to covert an unbalanced panel to a balanced panel. There are different ways to do this, and the argument `balance.type` must be supplied with one of three options:

1) Using "fill" creates a new row with NAs for each missing time point.
2) Using "shared.times" keeps all available individuals in the dataset but drops all time periods where at least one individual has no data.
3) By using "shared.individuals," all available time periods are kept but only for those individuals with information for all time periods.

The function `is.pconsecutive()` can test whether the entities in the data have any gaps in their time series.

Case study: Traffic Deaths and Alcohol Taxes

About 40,000 traffic fatalities occur each year in the U.S., and approximately 25% of fatal crashes involve a driver who drank alcohol.

Government officials are looking for a possible policy to reduce traffic fatalities. One potential policy is to increase the tax on alcoholic beverages to reduce their consumption and then drive down alcohol-related traffic fatalities.

In this case study, we'll use the `Fatalities` data set from the `AER` package. This data set contains traffic fatality rate and tax on beer for 48 U.S. states from 1982-1988.

Our primary research question is: **Is the tax on beer related to the traffic fatality rate (the number of fatalities per 10,000 inhabitants)**.

Data Description

In this section, we explore the employment data set by answering the following questions:

- What are the names of the variables in the data set?
- What is the number of observations?
- What is the number of states in the data set?
- Which years are included in the data set?
- Are there any duplicate state-year combinations in the data set?
- Is the data in "long format' or"wide format"?
- Is the data a balanced or unbalanced panel?
- Are there individual and time identifier variables in the data?
- Do we need to convert the data frame to a `pdata.frame()` to set up our model?
- Given this data set, how would you study whether the tax rate tax on beer impacts the traffic fatality rate?

```
# load dataset
data(Fatalities)

## Variable names

names(Fatalities)
```

```
##  [1] "state"        "year"         "spirits"      "unemp"        "income"
##  [6] "emppop"       "beertax"      "baptist"      "mormon"       "drinkage"
## [11] "dry"          "youngdrivers" "miles"        "breath"       "jail"
## [16] "service"      "fatal"        "nfatal"       "sfatal"       "fatal1517"
## [21] "nfatal1517"   "fatal1820"    "nfatal1820"   "fatal2124"    "nfatal2124"
## [26] "afatal"       "pop"          "pop1517"      "pop1820"      "pop2124"
## [31] "milestot"     "unempus"      "emppopus"     "gsp"
```

```
# check the  dimension and structure of the data set

dim(Fatalities)
```

```
## [1] 336  34
```

```
#str(Fatalities)
head(Fatalities)
```

```
##   state year spirits unemp   income   emppop  beertax baptist  mormon drinkage
## 1    al 1982    1.37  14.4 10544.15 50.69204 1.539379 30.3557 0.32829    19.00
## 2    al 1983    1.36  13.7 10732.80 52.14703 1.788991 30.3336 0.34341    19.00
## 3    al 1984    1.32  11.1 11108.79 54.16809 1.714286 30.3115 0.35924    19.00
## 4    al 1985    1.28   8.9 11332.63 55.27114 1.652542 30.2895 0.37579    19.67
## 5    al 1986    1.23   9.8 11661.51 56.51450 1.609907 30.2674 0.39311    21.00
## 6    al 1987    1.18   7.8 11944.00 57.50988 1.560000 30.2453 0.41123    21.00
##       dry youngdrivers    miles breath jail service fatal nfatal sfatal
## 1 25.0063     0.211572 7233.887     no   no      no   839    146     99
## 2 22.9942     0.210768 7836.348     no   no      no   930    154     98
## 3 24.0426     0.211484 8262.990     no   no      no   932    165     94
## 4 23.6339     0.211140 8726.917     no   no      no   882    146     98
## 5 23.4647     0.213400 8952.854     no   no      no  1081    172    119
## 6 23.7924     0.215527 9166.302     no   no      no  1110    181    114
##   fatal1517 nfatal1517 fatal1820 nfatal1820 fatal2124 nfatal2124  afatal
## 1        53          9        99         34       120         32 309.438
## 2        71          8       108         26       124         35 341.834
## 3        49          7       103         25       118         34 304.872
## 4        66          9       100         23       114         45 276.742
## 5        82         10       120         23       119         29 360.716
## 6        94         11       127         31       138         30 368.421
##       pop  pop1517  pop1820  pop2124 milestot unempus emppopus         gsp
## 1 3942002 208999.6 221553.4 290000.1    28516     9.7     57.8 -0.02212476
## 2 3960008 202000.1 219125.5 290000.2    31032     9.6     57.9  0.04655825
## 3 3988992 197000.0 216724.1 288000.2    32961     7.5     59.5  0.06279784
## 4 4021008 194999.7 214349.0 284000.3    35091     7.2     60.1  0.02748997
## 5 4049994 203999.9 212000.0 263000.3    36259     7.0     60.7  0.03214295
## 6 4082999 204999.8 208998.5 258999.8    37426     6.2     61.5  0.04897637
```

```
# check if it's balanced
Fatalities %>%
  dplyr::select(year, state) %>%
  table()
```

```
##       state
## year   al az ar ca co ct de fl ga id il in ia ks ky la me md ma mi mn ms mo mt
##   1982  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1983  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
##   1984  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1985  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1986  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1987  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1988  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##       state
## year   ne nv nh nj nm ny nc nd oh ok or pa ri sc sd tn tx ut vt va wa wv wi wy
##   1982  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1983  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1984  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1985  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1986  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1987  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   1988  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
# Check for gaps in the time series of each state
Fatalities%>%
  is.pconsecutive()
```

```
##   al   az   ar   ca   co   ct   de   fl   ga   id   il   in   ia   ks   ky   la
## TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
##   me   md   ma   mi   mn   ms   mo   mt   ne   nv   nh   nj   nm   ny   nc   nd
## TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
##   oh   ok   or   pa   ri   sc   sd   tn   tx   ut   vt   va   wa   wv   wi   wy
## TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
## convert data frame to pdata.frame
pfatalities <- pdata.frame(Fatalities, index=c("state", "year")) %>%
  mutate(fatal_rate = (fatal/pop)*10000)

## Check the structure of panel data
pdim(pfatalities)
```

```
## Balanced Panel: n = 48, T = 7, N = 336
```

The dataset consists of 336 observations on 34 variables.

The variable state is the individual identifier and a factor variable with 48 levels (one for each of the 48 contiguous federal states of the U.S.).

The variable year is a time identifier and factor variable with seven levels identifying the period when the observation was made.

It's in long format, and we don't need to reshape it since there is a column for each variable and a row for each state-year).

The panel is balanced since all variables are observed for all entities and over all periods.

Using is.pconsecutive() shows that no gaps in the periods are present for any state.

Since the first variable of the data set is the individual identifier and the second variable is the time identifier, it's not necessary to use `pdata.frame()`. However, we convert the data frame to "pdata.frame" for practice.

We use beertax to operationalize the tax on beer. It's a numeric class with a min of 0.04, a max of 2.72, and a mean of 0.51

To operationalize the fatality rate, we use the fatal variable, the number of vehicle fatalities. It's a numeric class with min 79, max 5504, and a mean of 928.7.

To create a fatality rate per 10000, we divide the number of fatalities by population and multiply it by 10000.

Descriptive Statistics

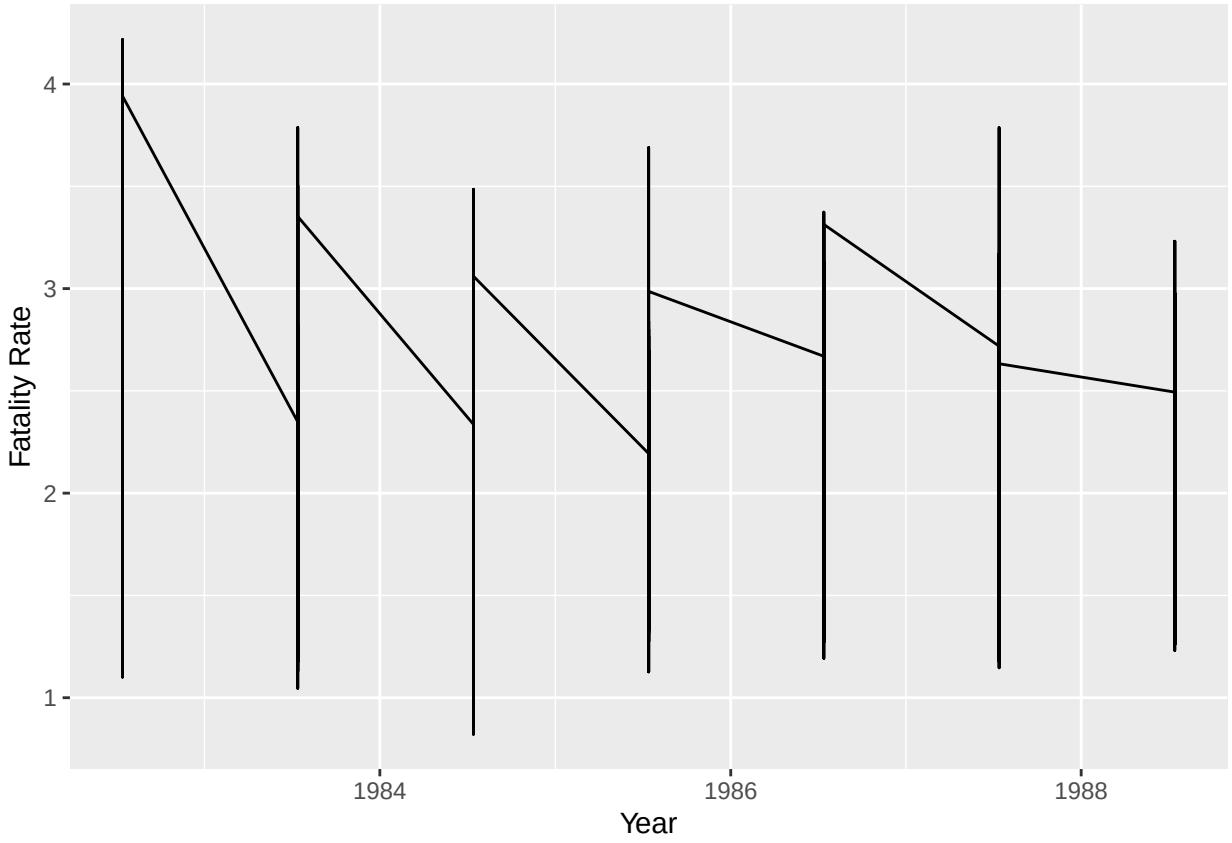
In this section, we use exploratory analysis to answer the following questions:

1- How has the traffic fatality rate changed over time?

2- How has the tax on beer changed over time?

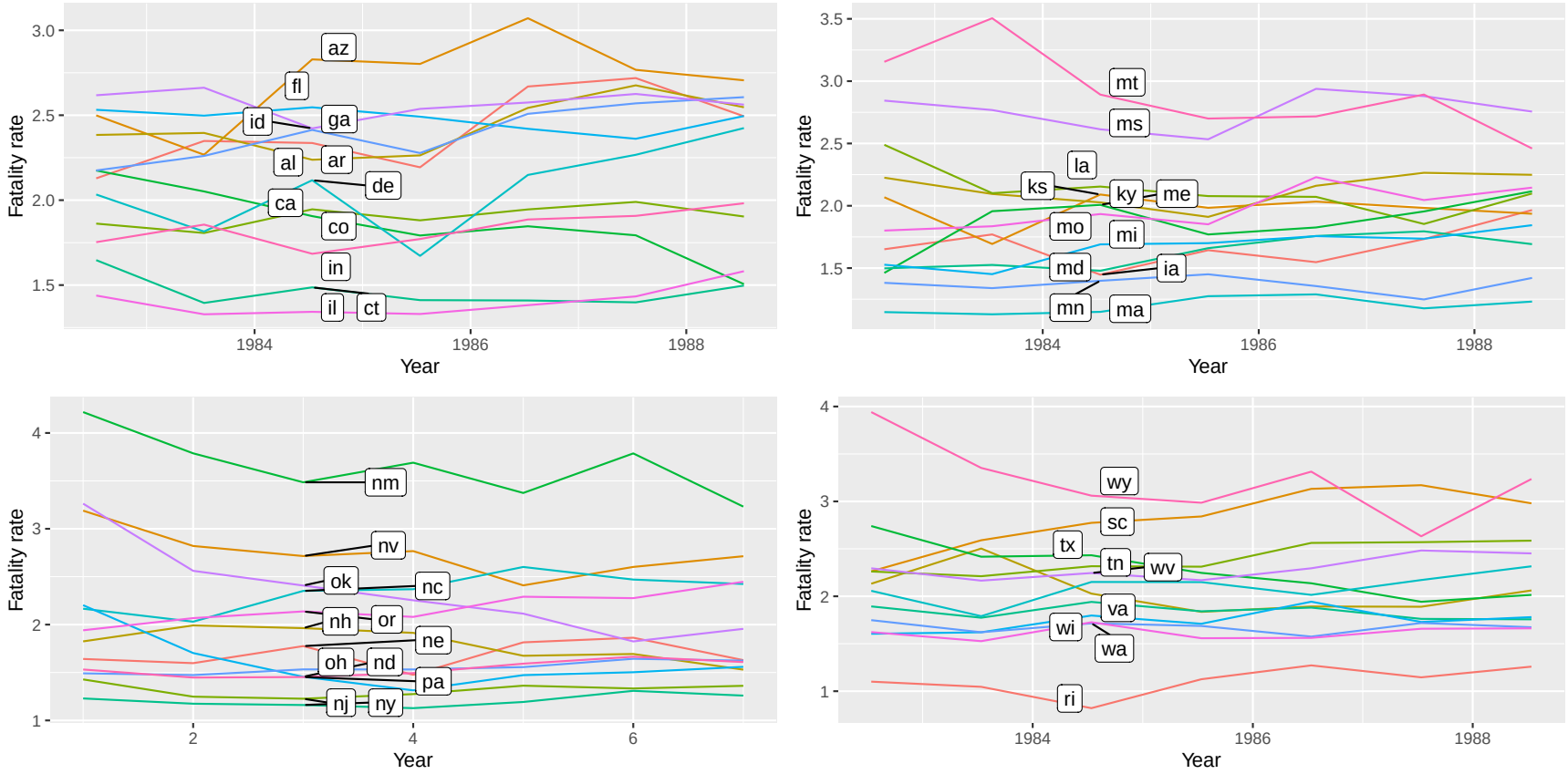
3- Is the tax on beer related to the traffic fatality rate?

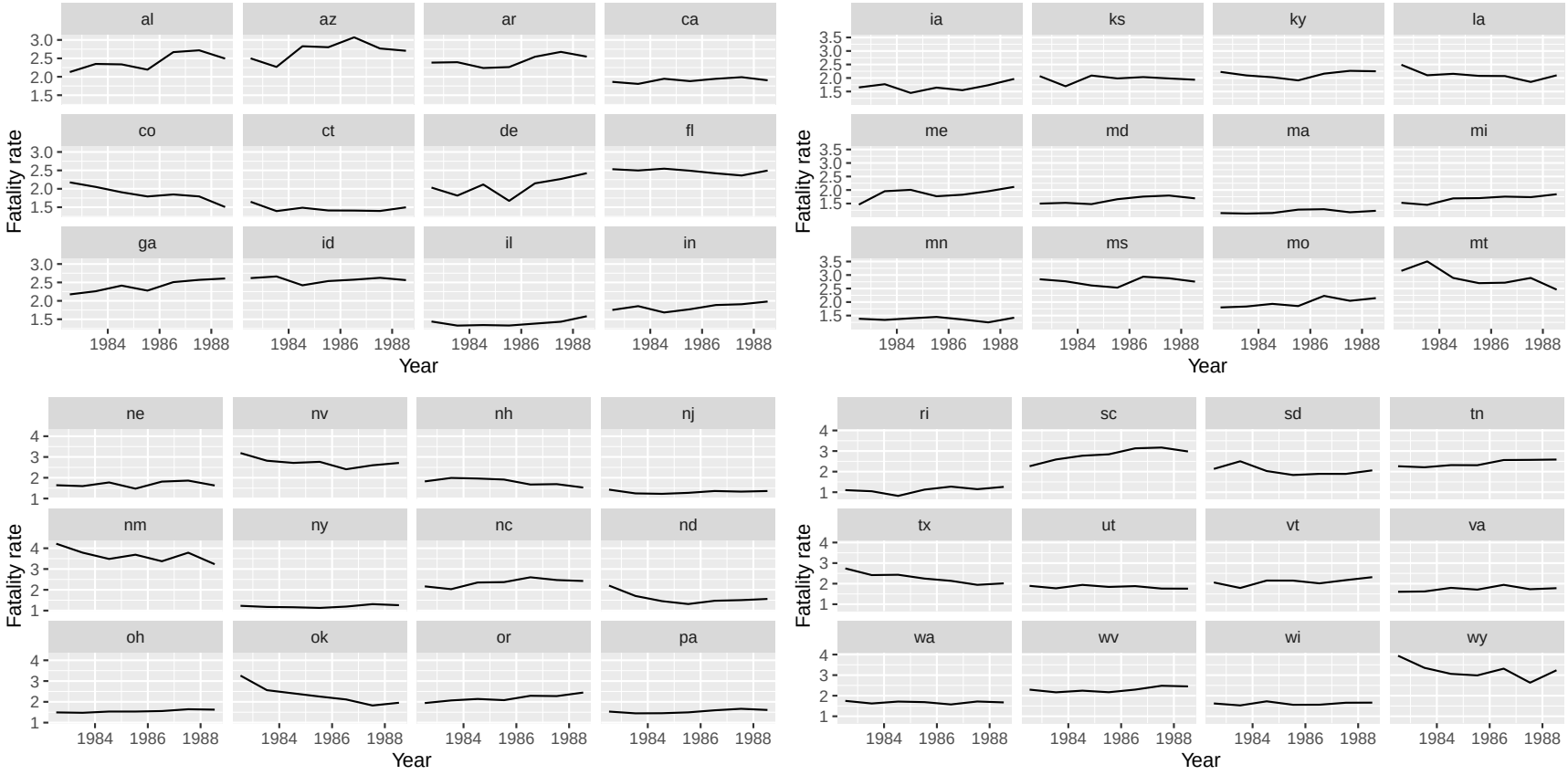
a)- Let's start with a simple line plot of traffic fatality rate over time (year) without considering the state variations. What did you notice?

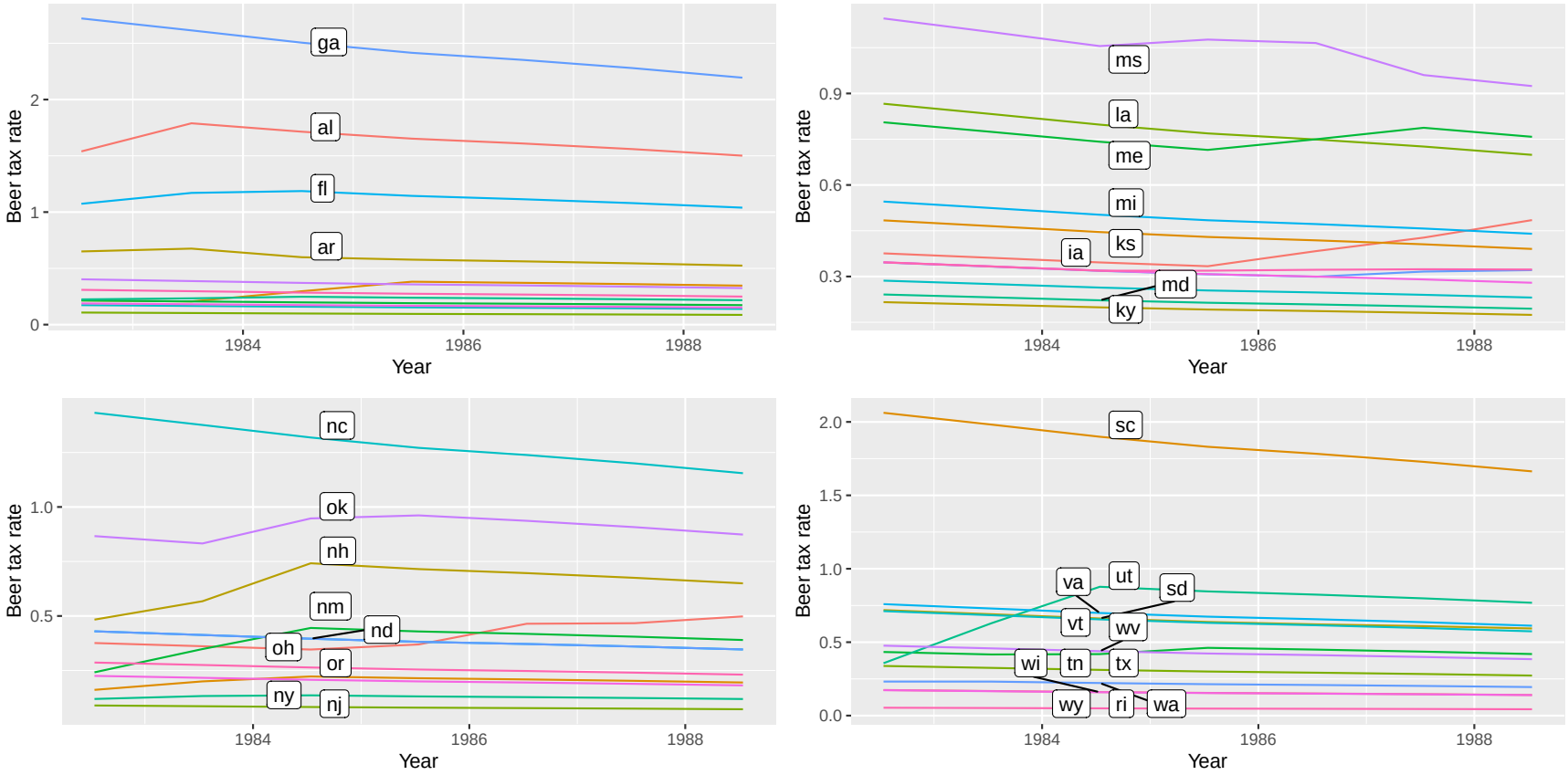


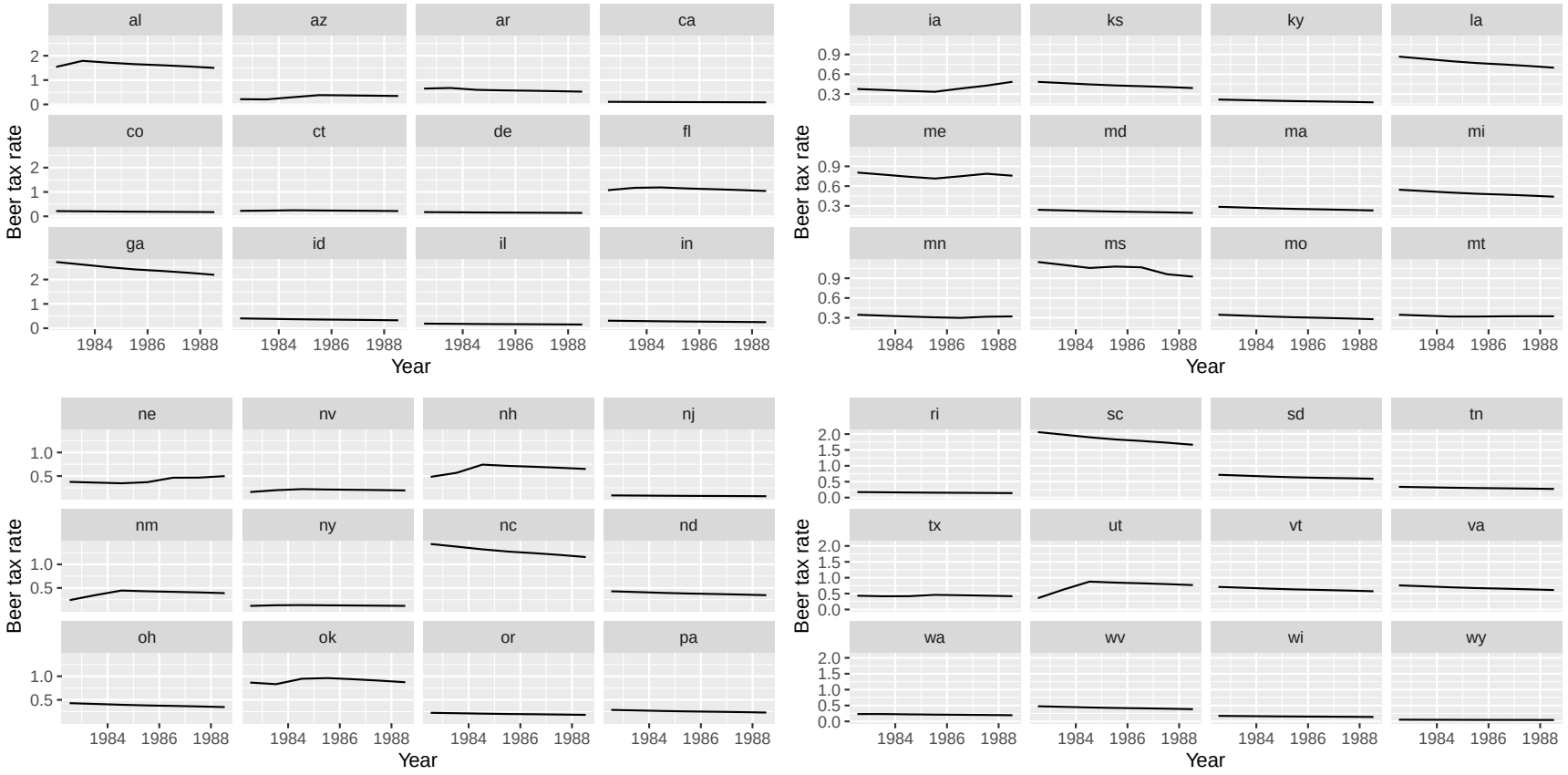
As you can see, it does not give meaningful results! When doing graphical analysis with panel data, the individual and time dimensions must be considered to get interpretable results.

b)- Create a line plot for each state separated by color or using `facet_wrap`.



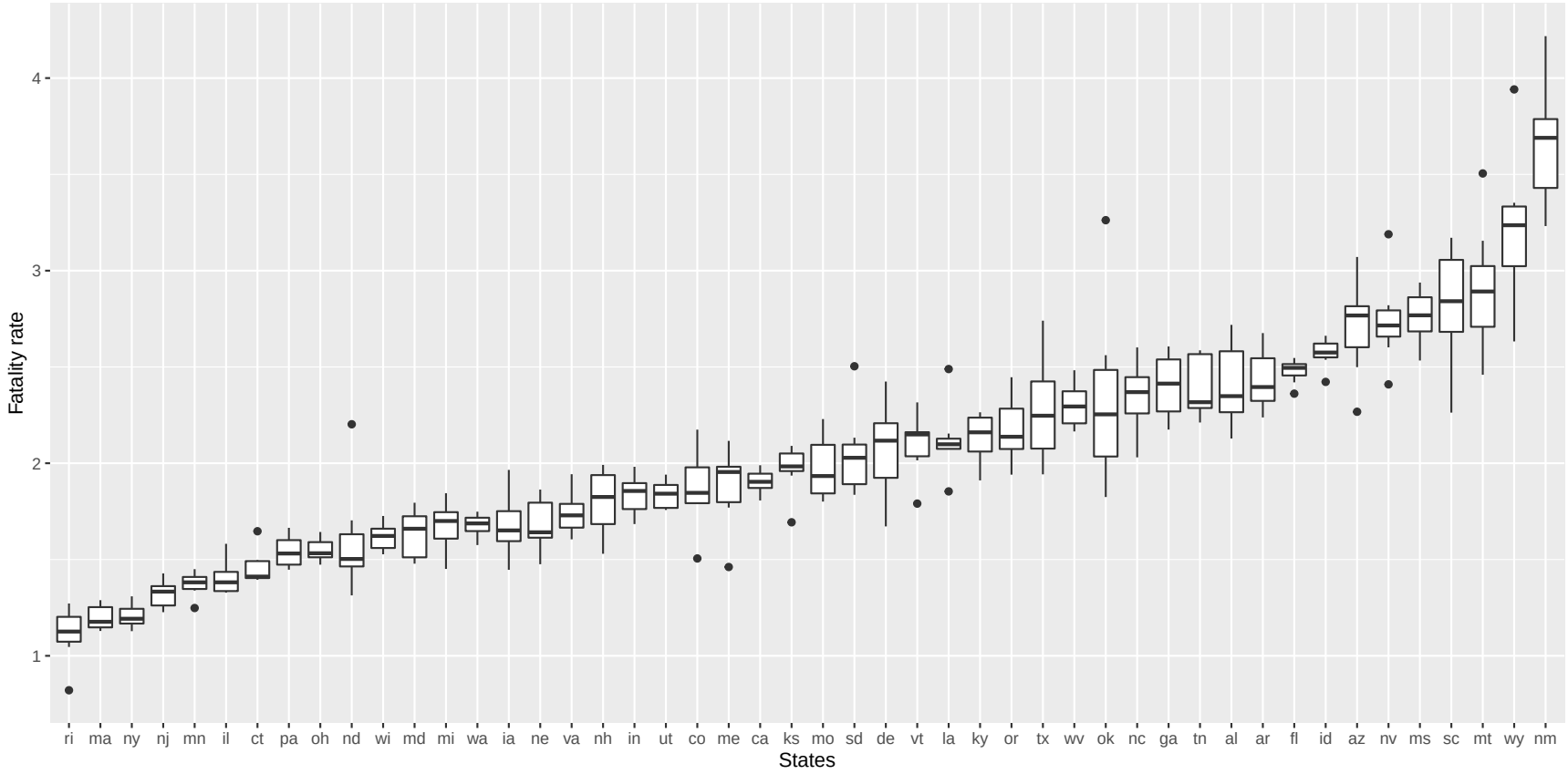

al
az
ar
ca
co
ct
de
fl
ga
id
il
in
ia
ks
ky
la
me
md
ma
mi
mn
ms
mo
mt
ne
nv
nh
nj
nm
ny
nc
nd
oh
ok
or
pa
ri
sc
sd
tn
tx
ut
vt
va
wa
wv
wi
wy
Fatality rate
Year
1984
1986
1988

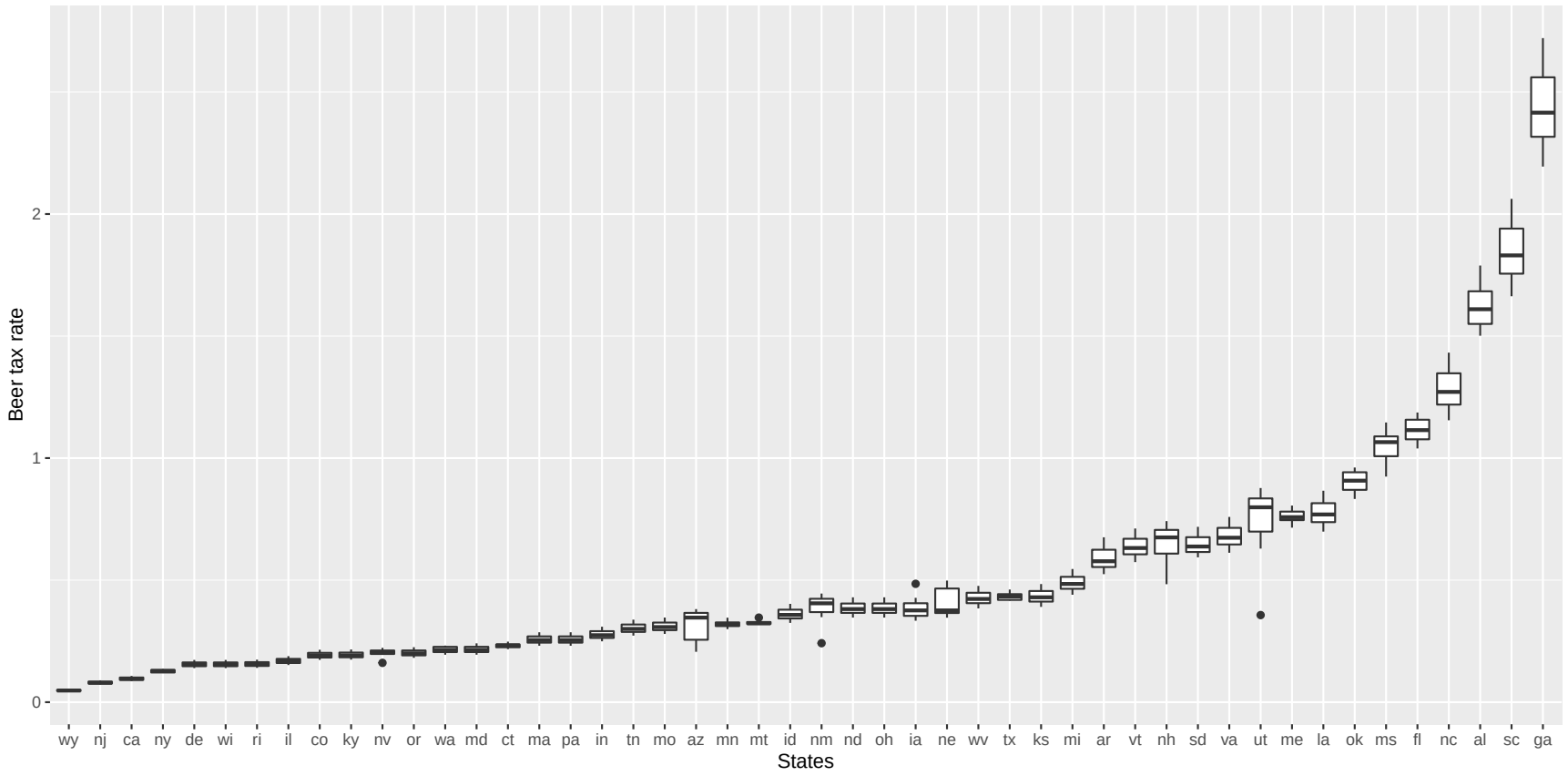

Beer tax rate
Year
1984
1986
1988
0
1
2
ga
al
fl
ar
0.3
0.6
0.9
ms
la
me
mi
ks
ia
md
ky
0.5
1.0
nc
ok
nh
nm
nd
oh
or
ny
nj
0.0
0.5
1.0
1.5
2.0
sc
va
ut
sd
vt
wv
wi
tn
tx
wy
ri
wa



The fatality rate increases in some states and decrease in others, but the tax on beer have a small variation over time in many states.

c)- Use a boxplot to show the heterogeneity of the fatality rate and beer tax across the states.

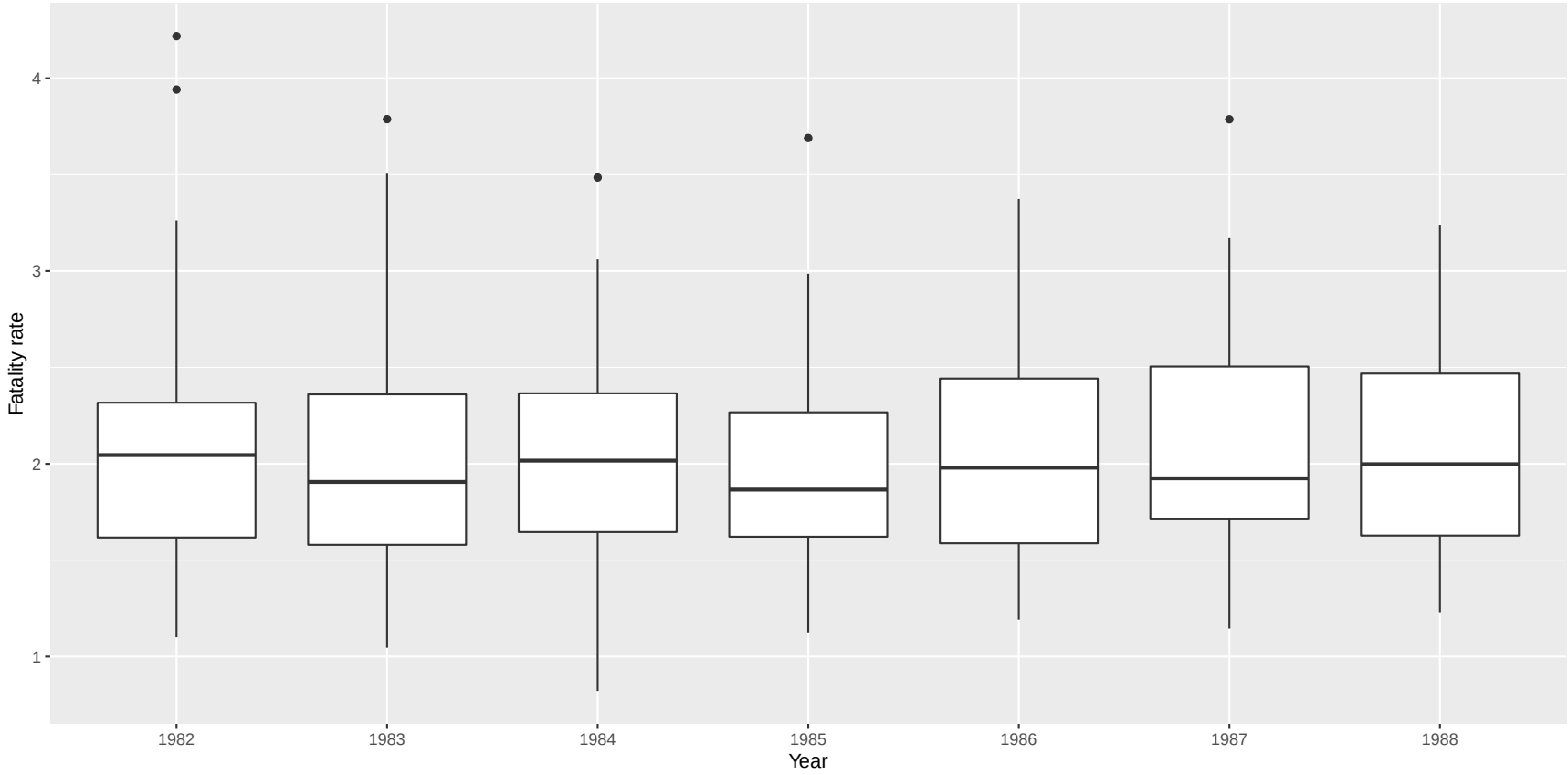


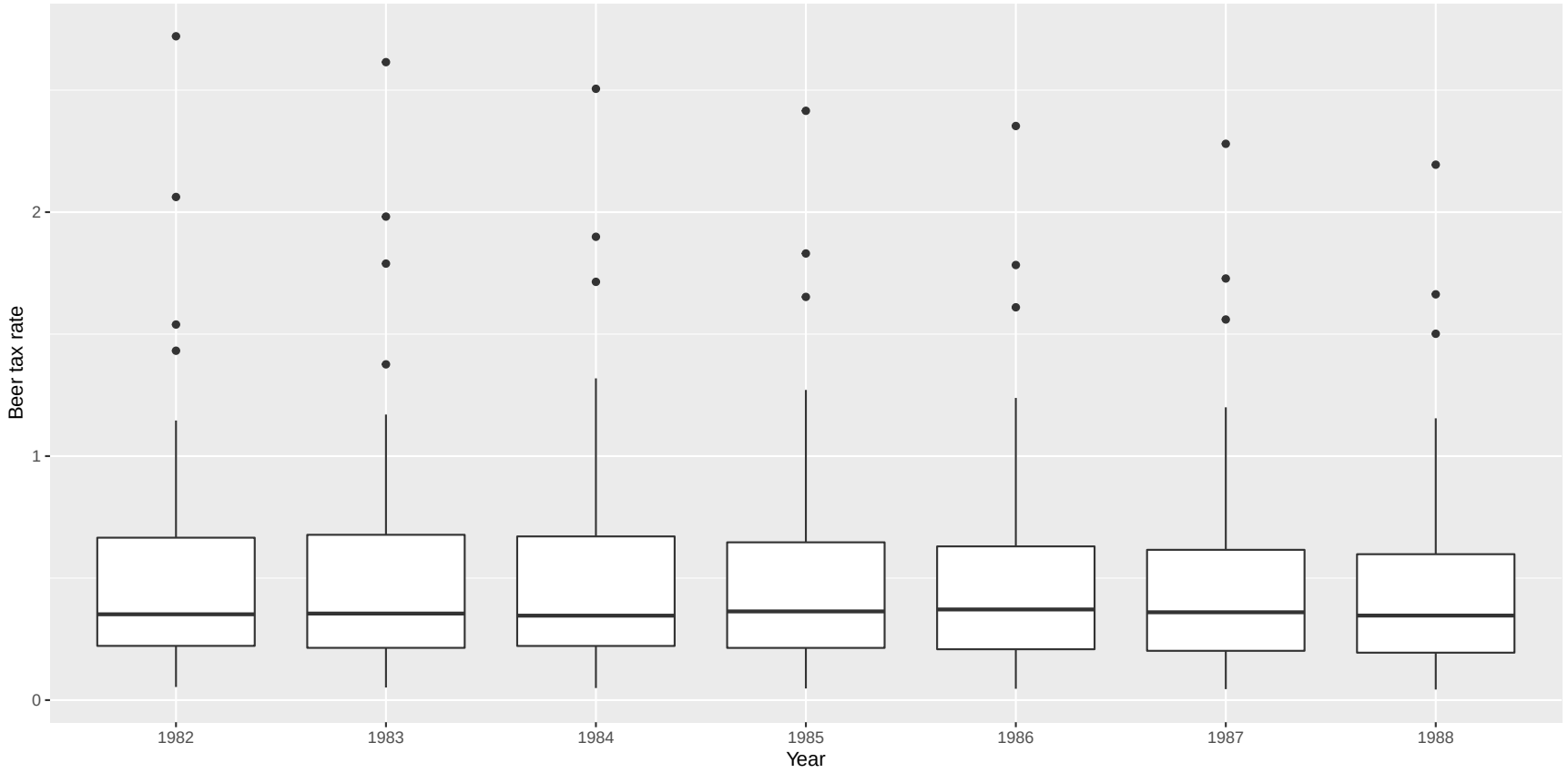


In the first plot, the states have different averages of fatality rates, and states with higher fatality rates also have high variation or variance. RI (Rhode Island) and MA(Massachusetts) have the lowest fatality rate on average, and WY(Wyoming) and NM (New Mexico) have the highest fatality rate on average.

In the second, the states have different tax rates on beer. WY(Wyoming) and NJ(New Jersey) have the lowest tax on beer on average, and SC(South Carolina) and GA (Georgia) have the highest tax on beer. Also, the variation in tax on beer is low in some states, and this could create a problem when we use the first differencing estimator.

d)- Use a boxplot to show heterogeneity of fatality rate and beer tax across time.

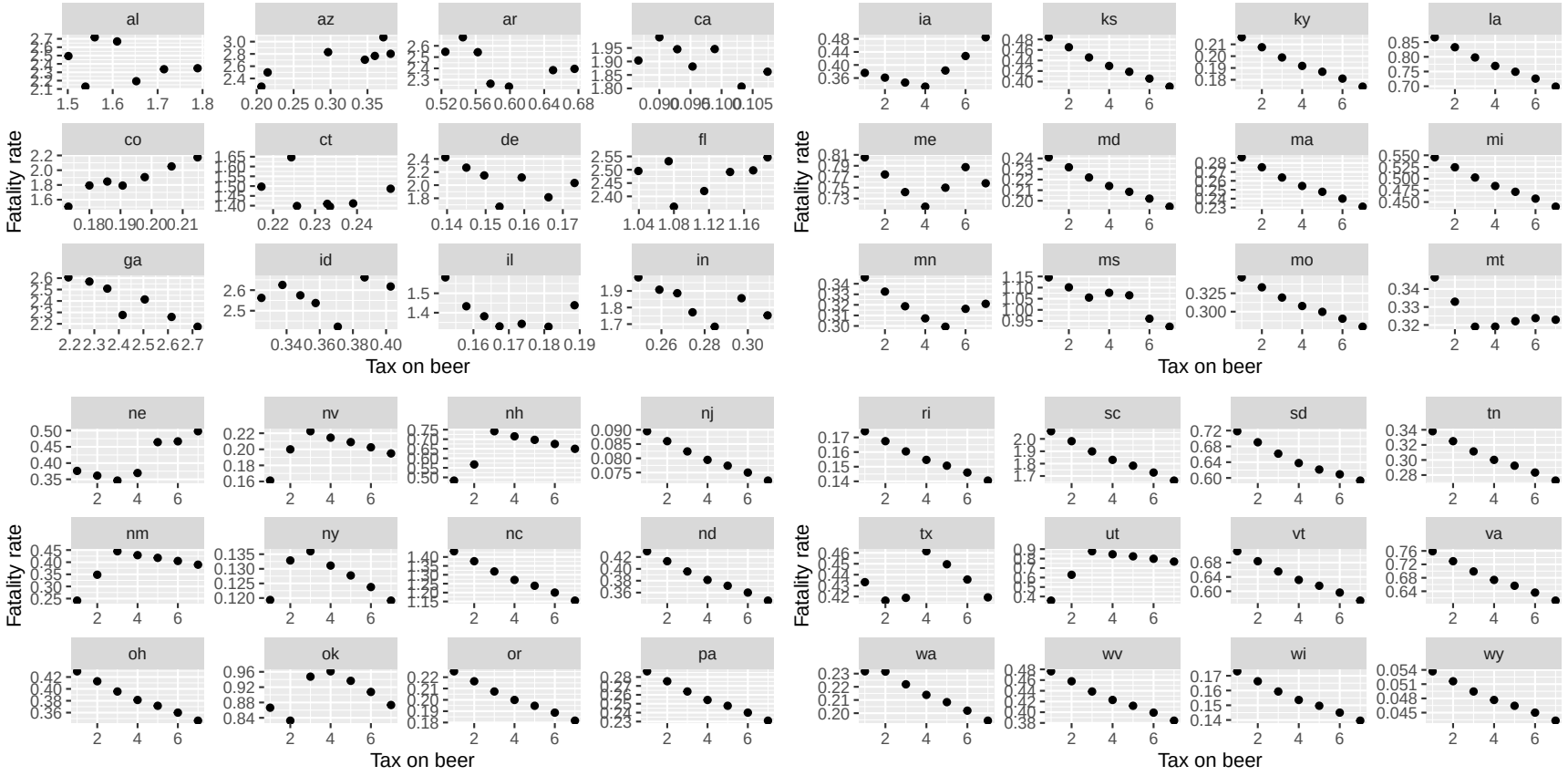




In the first plot above, the average fatality rate calculated across all 48 states differs each year, indicating that the average fatality rate varies with time.

In the second plot, the average tax rate calculated across all 48 states is almost the same each year, with the same variation.

e) Use a scatter plot to check how the fatality rate correlates with the tax rate for each state and each rate?



Based on these scatter plots, there is a negative correlation between tax on beer and fatality rate in most states, but in some states, there seems to be a positive correlation between these two variables, like Colorado. Based on these results, we expect that there is a negative association between fatality rate and tax on beer if we correctly specify and estimate the model.

Model Development

The Pooled OLS Estimator

- Estimate the Pooled OLS using either lm() or plm(). What do you notice? Is the association between fatality rate and tax on beer as we expected?

```
names(pfatalities)
```

```
##  [1] "state"        "year"         "spirits"      "unemp"        "income"
##  [6] "emppop"       "beertax"      "baptist"      "mormon"       "drinkage"
## [11] "dry"          "youngdrivers" "miles"        "breath"       "jail"
## [16] "service"      "fatal"        "nfatal"       "sfatal"       "fatal1517"
## [21] "nfatal1517"   "fatal1820"    "nfatal1820"   "fatal2124"    "nfatal2124"
## [26] "afatal"       "pop"          "pop1517"      "pop1820"      "pop2124"
## [31] "milestot"     "unempus"      "emppopus"     "gsp"          "fatal_rate"
```

```
pooled_ols <- plm(fatal_rate ~ beertax, data = pfatalities,
                      index = c("state", "year"),
                      effect = "individual", model = "pooling")
summary(pooled_ols)
```

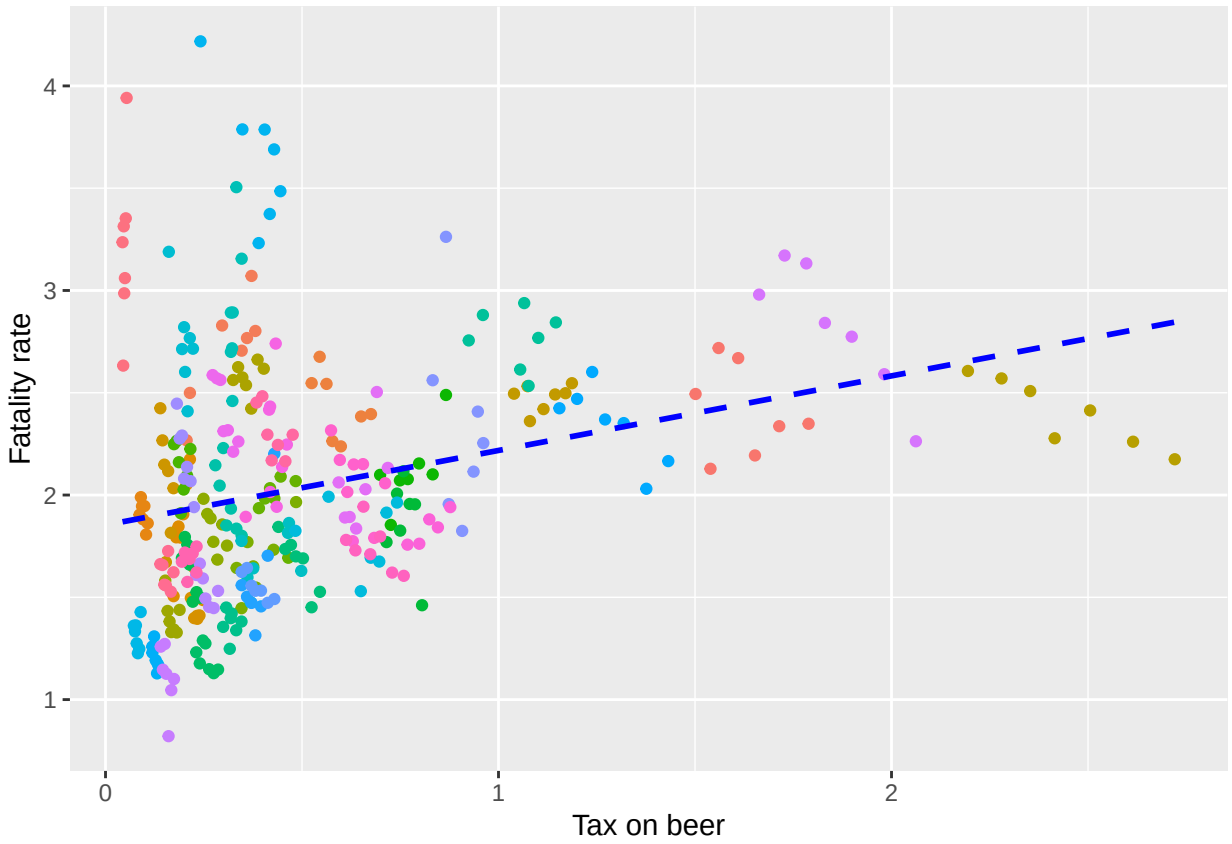
```
## Pooling Model
##
## Call:
## plm(formula = fatal_rate ~ beertax, data = pfatalities, effect = "individual",
##     model = "pooling", index = c("state", "year"))
##
## Balanced Panel: n = 48, T = 7, N = 336
##
## Residuals:
##     Min.  1st Qu.   Median  3rd Qu.     Max.
## -1.09060 -0.37768 -0.09436  0.28548  2.27643
##
## Coefficients:
##             Estimate Std. Error t-value  Pr(>|t|)
## (Intercept) 1.853308   0.043567 42.5391 < 2.2e-16 ***
## beertax     0.364605   0.062170  5.8647 1.082e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Total Sum of Squares:    108.92
## Residual Sum of Squares: 98.747
## R-Squared:      0.093363
## Adj. R-Squared: 0.090648
## F-statistic: 34.3943 on 1 and 334 DF, p-value: 1.0822e-08
```

```
#for comparison; note it is the same, so pooled OLS is doing a simple ols
lm(pfatalities$fatal_rate ~ pfatalities$beertax)
```

```
##
## Call:
## lm(formula = pfatalities$fatal_rate ~ pfatalities$beertax)
##
## Coefficients:
##         (Intercept)  pfatalities$beertax
##              1.8533               0.3646
```

```
pfatalities%>%
  ggplot(aes(x = beertax, y = fatal_rate)) +
  geom_point(aes(color = state))+
  geom_line(data=broom::augment(pooled_ols),
            aes(x = beertax, y =.fitted),
            color = "blue", lty="dashed", size = 1)+
  theme(legend.position = "none") +labs(x = "Tax on beer",
       y = "Fatality rate")
```



We use both lm() or plm() to estimate the pooled OLS model.

The coefficient of tax on beer is significant and positive. So, according to this model, taxing beer would increase the number of deaths from road accidents.

In the scatter plot of tax on beer and fatality rate, each point represents observations of beer tax and the fatality rate for a given state and year. It is easy to see that, although the variable state could distinguish the state, OLS estimation treats all observations as if they come from the same state and fits the regression line accordingly.

As a result, The regression results indicate a positive relationship between the beer tax and the fatality rate. This is contrary to our expectations: alcohol taxes are supposed to lower the rate of traffic fatalities.

This counter-intuitive result is possibly due to omitted variable bias since the model does not include any other explanatory

variable such as economic conditions. This could be corrected by adding more explanatory variables. However, this cannot account for unobserved omitted factors that differ from state to state but can be assumed to be constant over time, e.g., the population's attitude towards drunk driving.

Panel data analysis will provide a solution to this puzzle by controlling for the effect of unobserved state heterogeneity.

We also observe signs of heteroskedasticity in the response variable fatality rate. Specifically, the variance in a fatality is not constant for different values of tax on beer.

The Least-Squares Dummy Variables Estimator(LSDV) Use Least-Squares Dummy Variables Estimator (LSDV) to control time-invariant state heterogeneity. What do you notice?

- Use model = pooling in plm but explicitly add different intercepts for each state.

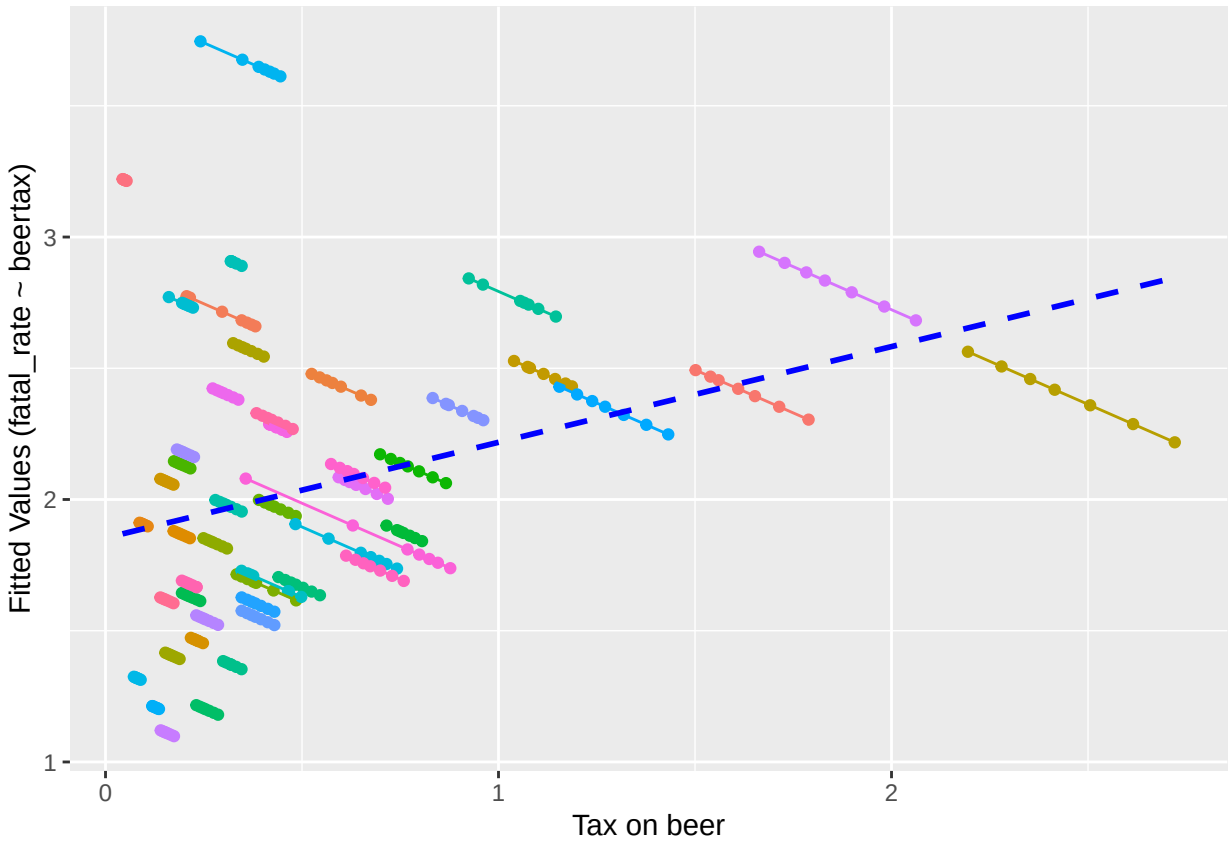
```
lsdv_model <- plm(fatal_rate ~ beertax + state-1, data = pfatalities,
                      index = c("state", "year"),
                      effect = "individual", model = "pooling")
summary(lsdv_model)
```

```
## Pooling Model
##
## Call:
## plm(formula = fatal_rate ~ beertax + state - 1, data = pfatalities,
##     effect = "individual", model = "pooling", index = c("state",
##         "year"))
##
## Balanced Panel: n = 48, T = 7, N = 336
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.5869619 -0.0828376 -0.0012701  0.0795454  0.8977960
##
## Coefficients:
##          Estimate Std. Error t-value  Pr(>|t|)
## beertax -0.655874   0.187850 -3.4915  0.000556 ***
## stateal  3.477630   0.313357 11.0980 < 2.2e-16 ***
## stateaz  2.909903   0.092539 31.4452 < 2.2e-16 ***
## statear  2.822679   0.132125 21.3636 < 2.2e-16 ***
## stateca  1.968161   0.074007 26.5943 < 2.2e-16 ***
## stateco  1.993350   0.080371 24.8019 < 2.2e-16 ***
## statect  1.615373   0.083913 19.2506 < 2.2e-16 ***
```

```
## statede  2.170028   0.077457 28.0159 < 2.2e-16 ***
## statefl  3.209500   0.221513 14.4890 < 2.2e-16 ***
## stategа  4.002233   0.464031  8.6249 4.435e-16 ***
## stateid  2.808608   0.098767 28.4368 < 2.2e-16 ***
## stateil  1.516008   0.078478 19.3176 < 2.2e-16 ***
## statein  2.016088   0.088672 22.7364 < 2.2e-16 ***
## stateia  1.933698   0.102217 18.9176 < 2.2e-16 ***
## stateks  2.254414   0.108632 20.7528 < 2.2e-16 ***
## stateky  2.260113   0.080462 28.0893 < 2.2e-16 ***
## statela  2.630514   0.162664 16.1714 < 2.2e-16 ***
## stateme  2.369683   0.160065 14.8045 < 2.2e-16 ***
## statemd  1.771190   0.082458 21.4800 < 2.2e-16 ***
## statema  1.367884   0.086477 15.8178 < 2.2e-16 ***
## statemi  1.993103   0.116632 17.0888 < 2.2e-16 ***
## statemn  1.580417   0.093628 16.8797 < 2.2e-16 ***
## statems  3.448550   0.209363 16.4717 < 2.2e-16 ***
## statemo  2.181368   0.092523 23.5764 < 2.2e-16 ***
## statemt  3.117239   0.094413 33.0170 < 2.2e-16 ***
## statene  1.955452   0.105505 18.5342 < 2.2e-16 ***
## statenv  2.876855   0.081056 35.4922 < 2.2e-16 ***
## statenh  2.223176   0.141143 15.7512 < 2.2e-16 ***
## statenj  1.371881   0.073328 18.7089 < 2.2e-16 ***
## statenm  3.904005   0.101537 38.4492 < 2.2e-16 ***
## stateny  1.290960   0.075629 17.0696 < 2.2e-16 ***
## statenc  3.187165   0.251734 12.6609 < 2.2e-16 ***
## statend  1.854191   0.101928 18.1912 < 2.2e-16 ***
## stateoh  1.803211   0.101928 17.6910 < 2.2e-16 ***
## stateok  2.932569   0.184285 15.9133 < 2.2e-16 ***
## stateor  2.309630   0.081175 28.4526 < 2.2e-16 ***
## statepa  1.710164   0.086477 19.7759 < 2.2e-16 ***
## stateri  1.212576   0.077533 15.6395 < 2.2e-16 ***
## statesc  4.034804   0.354789 11.3724 < 2.2e-16 ***
## statesd  2.473909   0.141209 17.5195 < 2.2e-16 ***
## statetn  2.601971   0.091624 28.3983 < 2.2e-16 ***
## statetx  2.560157   0.108532 23.5889 < 2.2e-16 ***
## stateut  2.313680   0.154532 14.9721 < 2.2e-16 ***
## statevt  2.511586   0.139726 17.9751 < 2.2e-16 ***
## stateva  2.187447   0.146641 14.9170 < 2.2e-16 ***
## statewa  1.818106   0.082328 22.0836 < 2.2e-16 ***
```

```
## statewv  2.580876   0.107668 23.9707 < 2.2e-16 ***
## statewi  1.718364   0.077457 22.1848 < 2.2e-16 ***
## statewy  3.249126   0.072328 44.9219 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    108.92
## Residual Sum of Squares: 10.345
## R-Squared:      0.90501
## Adj. R-Squared: 0.88913
## F-statistic: 847.812 on 49 and 287 DF, p-value: < 2.22e-16
```

```
ggplot(data = broom::augment(lsdv_model), aes(x = beertax, y = .fitted)) +
  geom_point(aes(color =state)) +
  geom_line(aes(color = state)) +
  geom_line(data=broom::augment(pooled_ols),
            aes(x = beertax, y =.fitted),
            color = "blue", lty="dashed", size = 1) +
  labs(x = "Tax on beer", y = "Fitted Values (fatal_rate ~ beertax)",
       color = "state") +
  theme(legend.position = "none")
```



We can control for the unobserved factors by including dummy variables for all states (or years). This is the so-called least squares dummy variable (LSDV) approach.

We can drop the intercept by adding -1 to the formula so that no coefficient (level) of the state is excluded.

Note that in this model, the coefficient estimate of tax is now different compared to the pooled OLS approach, and it is negative as we expect.

The estimated coefficient on beer tax is now negative and significantly different from zero at 5%. Its interpretation is that raising the beer tax by 1% causes the traffic fatalities rate to decrease by 0.65 per 10,000 people. This is rather large as the average fatality rate is 2 approximately per 10,000 people

```r
# compute mean fatality rate over all states for all time periods
mean(pfatalities$fatal_rate)
```

```
## [1] 2.040444
```

All coefficients for the 48 dummy variables representing the state-specific effects are statistically significant.

In terms of the goodness-of-fit, the FE model seems to have improved upon the Pooled OLS model by a large amount of 9% to 92% based on adjusted R-squared

We see that the F-test's statistic of 447.8 is significant at a $p < .001$, thereby implying that the model's goodness-of-fit is better than the mean model.

From the scatter plot, we can see that due to the introduction of state dummy variables, each state has its own intercept with the y axis! For comparison, I plotted the fitted values from the pooled OLS model (blue dashed line).

If there are many individuals, the LSDV method is expensive from a computational point of view.

The First Differencing Estimator (FD)

- Use the First-difference Estimator to control for state time-invariant state heterogeneity. You can do that by specifying model = "fd" in the plm() function.

Compare this to the LSDV model above.

```
first_diff_model<- plm(fatal_rate ~ beertax, data = pfatalities,
                       index = c("state", "year"),
                       effect = "individual", model = "fd")

summary(first_diff_model)
```

```
## Oneway (individual) effect First-Difference Model
##
## Call:
## plm(formula = fatal_rate ~ beertax, data = pfatalities, effect = "individual",
##     model = "fd", index = c("state", "year"))
##
## Balanced Panel: n = 48, T = 7, N = 336
## Observations used in estimation: 288
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -0.697391 -0.106403  0.010073  0.109465  0.606420
##
## Coefficients:
##               Estimate Std. Error t-value Pr(>|t|)
## (Intercept) -0.0031368  0.0119115 -0.2633   0.7925
## beertax      0.0136878  0.2852511  0.0480   0.9618
##
## Total Sum of Squares:    11.213
## Residual Sum of Squares: 11.213
## R-Squared:      8.0509e-06
## Adj. R-Squared: -0.0034884
## F-statistic: 0.00230256 on 1 and 286 DF, p-value: 0.96176
```

The estimated coefficient on beer tax is still positive but not statistically significant. This could be due to the low variation of tax on beer in many states

The coefficients and standard errors of the first-differenced model and LSDV are only identical when there are two time periods. For longer time series, the coefficients and the standard errors will differ.

The Within Estimator

- Use the Within Estimator to control for state time-invariant state heterogeneity. You can do that by specifying model = "within" in the plm() function.

Compare this to the LSDV model above.

```
within_model<- plm(fatal_rate ~ beertax, data = pfatalities,
                   index = c("state", "year"),
                   effect = "individual", model = "within")

summary(within_model)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = fatal_rate ~ beertax, data = pfatalities, effect = "individual",
##     model = "within", index = c("state", "year"))
##
## Balanced Panel: n = 48, T = 7, N = 336
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.5869619 -0.0828376 -0.0012701  0.0795454  0.8977960
##
## Coefficients:
##         Estimate Std. Error t-value Pr(>|t|)
## beertax -0.65587    0.18785 -3.4915 0.000556 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    10.785
## Residual Sum of Squares: 10.345
## R-Squared:      0.040745
## Adj. R-Squared: -0.11969
## F-statistic: 12.1904 on 1 and 287 DF, p-value: 0.00055597
```

The estimated coefficient on beer tax is exactly the same as in the LSDV model, showing it is doing the same thing but can more efficiently estimate the coefficient of interest.

Model Comparison

- Compare the four models you fit using stargazer

```
stargazer(pooled_ols, lsdv_model, first_diff_model, within_model, keep = "beertax", type = "text",
          omit.stat = c("ser","f","adj.rsq"), dep.var.labels = "",
          column.labels = c("Pooled OLS", "LSDV", "FD", "Within"))
```

```
##
## ================================================
##                        Dependent variable:
##              -----------------------------------
##
##              Pooled OLS   LSDV      FD     Within
##                 (1)        (2)      (3)      (4)
## ------------------------------------------------
## beertax       0.365***  -0.656***  0.014  -0.656***
##               (0.062)    (0.188)  (0.285)  (0.188)
##
## ------------------------------------------------
## Observations    336        336      288      336
## R2             0.093      0.905   0.00001   0.041
## ================================================
## Note:                   *p<0.1; **p<0.05; ***p<0.01
```

Time Fixed Effects

Let's extend our model to a case with both unobserved individual and unobserved time effects:

$$y_{it} = \beta_0 + \beta_1 x_{it} + \gamma_i + \eta_t + u_{it}$$

Where η_t is an unobserved time effect (invariant across individuals)

a) What is the consequence of excluding these unobserved time effects from the model?

b) What is an example of an unobserved time effect for fatality rate?

c) How do we control for this unobserved time heterogeneity?

Traffic Fatality Case Study

```
lsdv_model_time  <- plm(fatal_rate ~ beertax + state + year, data = pfatalities,
                        index = c("state", "year"),
                        model = "pooling")

summary(lsdv_model_time)
```

```
## Pooling Model
##
## Call:
## plm(formula = fatal_rate ~ beertax + state + year, data = pfatalities,
##     model = "pooling", index = c("state", "year"))
##
## Balanced Panel: n = 48, T = 7, N = 336
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.5955623 -0.0809619  0.0014301  0.0823356  0.8388350
##
## Coefficients:
##              Estimate Std. Error t-value  Pr(>|t|)
## (Intercept)  3.511375   0.332501 10.5605 < 2.2e-16 ***
## beertax     -0.639980   0.197377 -3.2424 0.0013280 **
## stateaz     -0.546862   0.277889 -1.9679 0.0500605 .
## statear     -0.638530   0.227320 -2.8089 0.0053191 **
```

```
## stateca     -1.485192   0.317771 -4.6738 4.594e-06 ***
## stateco     -1.461535   0.299792 -4.8752 1.821e-06 ***
## statect     -1.840129   0.292574 -6.2895 1.215e-09 ***
## statede     -1.284261   0.306769 -4.1864 3.795e-05 ***
## statefl     -0.260053   0.141938 -1.8322 0.0679855 .
## stategа      0.511622   0.189924  2.6938 0.0074887 **
## stateid     -0.648956   0.268670 -2.4154 0.0163546 *
## stateil     -1.938502   0.304175 -6.3730 7.568e-10 ***
## statein     -1.440141   0.284110 -5.0690 7.266e-07 ***
## stateia     -1.524283   0.263873 -5.7766 2.019e-08 ***
## stateks     -1.204308   0.255382 -4.7157 3.798e-06 ***
## stateky     -1.194788   0.299593 -3.9880 8.502e-05 ***
## statela     -0.833659   0.194970 -4.2758 2.612e-05 ***
## stateme     -1.094245   0.197590 -5.5380 7.029e-08 ***
## statemd     -1.684068   0.295416 -5.7007 3.015e-08 ***
## statema     -2.088021   0.287879 -7.2531 3.972e-12 ***
## statemi     -1.466498   0.245385 -5.9763 6.900e-09 ***
## statemn     -1.876493   0.276214 -6.7936 6.520e-11 ***
## statems     -0.019913   0.151810 -0.1312 0.8957364
## statemo     -1.275395   0.277913 -4.5892 6.714e-06 ***
## statemt     -0.339775   0.275025 -1.2354 0.2177016
## statene     -1.502914   0.259458 -5.7925 1.855e-08 ***
## statenv     -0.578156   0.298310 -1.9381 0.0536122 .
## statenh     -1.238930   0.217384 -5.6993 3.038e-08 ***
## statenj     -2.081217   0.320780 -6.4880 3.913e-10 ***
## statenm      0.446105   0.264804  1.6847 0.0931643 .
## stateny     -2.162882   0.312011 -6.9321 2.843e-11 ***
## statenc     -0.285072   0.120711 -2.3616 0.0188780 *
## statend     -1.603756   0.264267 -6.0687 4.161e-09 ***
## stateoh     -1.654736   0.264267 -6.2616 1.422e-09 ***
## stateok     -0.533614   0.174048 -3.0659 0.0023814 **
## stateor     -1.145402   0.298058 -3.8429 0.0001504 ***
## statepa     -1.745741   0.287879 -6.0641 4.267e-09 ***
## stateri     -2.241730   0.306569 -7.3123 2.746e-12 ***
## statesc      0.553584   0.109877  5.0382 8.422e-07 ***
## statesd     -0.988202   0.217313 -4.5474 8.084e-06 ***
## statetn     -0.854671   0.279319 -3.0598 0.0024286 **
## statetx     -0.898554   0.255510 -3.5167 0.0005093 ***
## stateut     -1.149722   0.203243 -5.6569 3.794e-08 ***
```

```
## statevt     -0.950379   0.218922 -4.3412 1.980e-05 ***
## stateva     -1.275195   0.211495 -6.0294 5.162e-09 ***
## statewa     -1.637130   0.295676 -5.5369 7.068e-08 ***
## statewv     -0.877737   0.256627 -3.4203 0.0007184 ***
## statewi     -1.735925   0.306769 -5.6587 3.757e-08 ***
## statewy     -0.203461   0.326805 -0.6226 0.5340678
## year1983    -0.079903   0.038354 -2.0833 0.0381264 *
## year1984    -0.072421   0.038352 -1.8883 0.0600119 .
## year1985    -0.123976   0.038442 -3.2250 0.0014083 **
## year1986    -0.037864   0.038588 -0.9813 0.3273123
## year1987    -0.050902   0.038974 -1.3061 0.1926000
## year1988    -0.051804   0.039623 -1.3074 0.1921450
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    108.92
## Residual Sum of Squares: 9.9193
## R-Squared:      0.90893
## Adj. R-Squared: 0.89143
## F-statistic: 51.9338 on 54 and 281 DF, p-value: < 2.22e-16
```

The Fixed Effects model may be enhanced by including appropriate dummy variables for time-fixed effects.

Typical example of time effects: macroeconomic conditions or federal policy common to all states but vary over time.

Model Diagnostics

Test the significance of fixed effects

We can do this test by running an F-test between two models, a restricted model, and an unrestricted model:

- The restricted model (the one with fewer variables) is the Pooled OLS model.
- The unrestricted model is the Fixed Effects model.

This is basically an ANOVA test if you remember back to the discrete choice model part of the course.

Suppose we have the fixed effects regression model:

$$y_{it} = \beta_0 + \beta_1 x_{1it} + ... + \beta_p x_{pit} + \gamma_i + \epsilon_{it}$$

We can use the F test with the following null hypothesis:

$$\gamma_1 = ... = \gamma_N = 0$$

In the plm package, the function `pFtest()` will run this with the null hypothesis that the pooled OLS model is better than the FE model, i.e., individual intercepts are zero. We can also run a test for time-fixed effects.

If we fail to reject the null hypothesis, then ignoring the panel data structure is probably ok though we would still want to check the residuals.

```
pFtest(within_model, pooled_ols)
```

```
##
##  F test for individual effects
##
## data:  fatal_rate ~ beertax
## F = 52.179, df1 = 47, df2 = 287, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```
pFtest(lsdv_model_time, lsdv_model)
```

```
##
##  F test for individual effects
##
## data:  fatal_rate ~ beertax + state + year
## F = 2.0117, df1 = 6, df2 = 281, p-value = 0.0642
## alternative hypothesis: significant effects
```

The null hypothesis is rejected in favor of individual fixed effects being significant, but time fixed effects are not.

Reminders

Before the next live session:

1. Complete HW-11
2. Complete all videos and reading for unit 12