Lab 3: Panel Models

US Traffic Fatalities: 1980 - 2004

Contents

```r
install.packages('ggrepel')
```

```
## Installing package into '/usr/local/lib/R/site-library'
## (as 'lib' is unspecified)
```

1 U.S. traffic fatalities: 1980-2004

In this lab, we are asking you to answer the following **causal** question:

> **"Do changes in traffic laws affect traffic fatalities?"**

To answer this question, please complete the tasks specified below using the data provided in `data/driving.Rdata`. This data includes 25 years of data that cover changes in various state drunk driving, seat belt, and speed limit laws.

Specifically, this data set contains data for the 48 continental U.S. states from 1980 through 2004. Various driving laws are indicated in the data set, such as the alcohol level at which drivers are considered legally intoxicated. There are also indicators for "per se" laws—where licenses can be revoked without a trial—and seat belt laws. A few economics and demographic variables are also included. The description of the each of the variables in the dataset is also provided in the dataset.

```r
load(file="driving.RData")

## please comment these calls in your work
glimpse(data)
```

```
## Rows: 1,200
## Columns: 56
## $ year         <int> 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 198~
## $ state        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ sl55         <dbl> 1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 0.542, 0~
## $ sl65         <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.458, 1~
## $ sl70         <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0~
## $ sl75         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ slnone       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ seatbelt     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 2, ~
## $ minage       <dbl> 18, 18, 18, 18, 18, 20, 21, 21, 21, 21, 21, 21, 21, 21, 2~
## $ zerotol      <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0~
## $ gdl          <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.0~
## $ bac10        <dbl> 1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1.000, 1~
## $ bac08        <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0~
## $ perse        <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0~
## $ totfat       <int> 940, 933, 839, 930, 932, 882, 1080, 1111, 1024, 1029, 112~
## $ nghtfat      <int> 422, 434, 376, 397, 421, 358, 500, 499, 423, 418, 466, 47~
## $ wkndfat      <int> 236, 248, 224, 223, 237, 224, 279, 300, 226, 247, 271, 27~
## $ totfatpvm    <dbl> 3.200, 3.350, 2.810, 3.000, 2.830, 2.510, 3.177, 2.970, 2~
## $ nghtfatpvm   <dbl> 1.437, 1.558, 1.259, 1.281, 1.278, 1.019, 1.471, 1.334, 1~
## $ wkndfatpvm   <dbl> 0.803, 0.890, 0.750, 0.719, 0.720, 0.637, 0.821, 0.802, 0~
## $ statepop     <int> 3893888, 3918520, 3925218, 3934109, 3951834, 3972527, 399~
## $ totfatrte    <dbl> 24.14, 24.07, 21.37, 23.64, 23.58, 22.20, 27.08, 27.67, 2~
## $ nghtfatrte   <dbl> 10.84, 11.08, 9.58, 10.09, 10.65, 9.01, 12.53, 12.43, 10.~
## $ wkndfatrte   <dbl> 6.060000, 6.330000, 5.710000, 5.670000, 6.000000, 5.64000~
## $ vehicmiles   <dbl> 29.37500, 27.85200, 29.85765, 31.00000, 32.93286, 35.1394~
## $ unem         <dbl> 8.8, 10.7, 14.4, 13.7, 11.1, 8.9, 9.8, 7.8, 7.2, 7.0, 6.9~
## $ perc14_24    <dbl> 18.9, 18.7, 18.4, 18.0, 17.6, 17.3, 17.0, 16.6, 16.2, 15.~
## $ sl70plus     <dbl> 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0~
## $ sbprim       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ sbsecon      <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, ~
## $ d80          <int> 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d81          <int> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d82          <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d83          <int> 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d84          <int> 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d85          <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d86          <int> 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d87          <int> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d88          <int> 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d89          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d90          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, ~
## $ d91          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, ~
## $ d92          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, ~
## $ d93          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ d94          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, ~
## $ d95          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, ~
## $ d96          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, ~
## $ d97          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, ~
## $ d98          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, ~
## $ d99          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d00          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d01          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d02          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d03          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ d04          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ vehicmilespc <dbl> 7543.874, 7107.785, 7606.622, 7879.802, 8333.562, 8845.61~
```

```
desc
```

```
##       variable                                          label
## 1         year                              1980 through 2004
## 2        state            48 continental states, alphabetical
## 3         sl55                              speed limit == 55
## 4         sl65                              speed limit == 65
## 5         sl70                              speed limit == 70
## 6         sl75                              speed limit == 75
## 7       slnone                                 no speed limit
## 8     seatbelt     =0 if none, =1 if primary, =2 if secondary
## 9       minage                           minimum drinking age
## 10     zerotol                             zero tolerance law
## 11         gdl                  graduated drivers license law
## 12       bac10                        blood alcohol limit .10
## 13       bac08                        blood alcohol limit .08
## 14       perse administrative license revocation (per se law)
## 15      totfat                       total traffic fatalities
## 16     nghtfat                     total nighttime fatalities
## 17     wkndfat                       total weekend fatalities
## 18   totfatpvm         total fatalities per 100 million miles
## 19  nghtfatpvm     nighttime fatalities per 100 million miles
## 20  wkndfatpvm       weekend fatalities per 100 million miles
## 21    statepop                               state population
## 22   totfatrte        total fatalities per 100,000 population
## 23  nghtfatrte    nighttime fatalities per 100,000 population
## 24  wkndfatrte       weekend accidents per 100,000 population
## 25  vehicmiles               vehicle miles traveled, billions
## 26        unem                     unemployment rate, percent
## 27   perc14_24          percent population aged 14 through 24
## 28    sl70plus                           sl70 + sl75 + slnone
## 29      sbprim                     =1 if primary seatbelt law
## 30     sbsecon                   =1 if secondary seatbelt law
## 31         d80                             =1 if year == 1980
## 32         d81
## 33         d82
## 34         d83
## 35         d84
## 36         d85
## 37         d86
## 38         d87
## 39         d88
## 40         d89
## 41         d90
## 42         d91
## 43         d92
## 44         d93
## 45         d94
## 46         d95
## 47         d96
## 48         d97
## 49         d98
## 50         d99
## 51         d00
## 52          d01
## 53          d02
## 54          d03
## 55          d04                             =1 if year == 2004
## 56 vehicmilespc
```

2 (30 points, total) Build and Describe the Data

1. (5 points) Load the data and produce useful features. Specifically:
   - Produce a new variable, called `speed_limit` that re-encodes the data that is in `sl55`, `sl65`, `sl70`, `sl75`, and `slnone`;
   - Produce a new variable, called `year_of_observation` that re-encodes the data that is in `d80`, `d81`, . . . , `d04`.
   - Produce a new variable for each of the other variables that are one-hot encoded (i.e. `bac*` variable series).
   - Rename these variables to sensible names that are legible to a reader of your analysis. For example, the dependent variable as provided is called, `totfatrte`. Pick something more sensible, like, `total_fatalities_rate`. There are few enough of these variables to change, that you should change them for all the variables in the data. (You will thank yourself later.)
2. (5 points) Provide a description of the basic structure of the dataset. What is this data? How, where, and when is it collected? Is the data generated through a survey or some other method? Is the data that is presented a sample from the population, or is it a *census* that represents the entire population? Minimally, this should include:
   - How is the our dependent variable of interest `total_fatalities_rate` defined?
3. (20 points) Conduct a very thorough EDA, which should include both graphical and tabular techniques, on the dataset, including both the dependent variable `total_fatalities_rate` and the potential explanatory variables. Minimally, this should include:
   - How is the our dependent variable of interest `total_fatalities_rate` defined?
   - What is the average of `total_fatalities_rate` in each of the years in the time period covered in this dataset?

As with every EDA this semester, the goal of this EDA is not to document your own process of discovery – save that for an exploration notebook – but instead it is to bring a reader that is new to the data to a full understanding of the important features of your data as quickly as possible. In order to do this, your EDA should include a detailed, orderly narrative description of what you want your reader to know. Do not include any output – tables, plots, or statistics – that you do not intend to write about.

##EDA ###EDA

```r
pdata <- pdata.frame(data, index = c("state", "year"))
pdata_speed_vars<- pdata %>%
  select(c('sl55','sl65','sl70','sl75'))
pdata <- within(pdata,{
              speed_limit <- 0
              speed_limit[sl55 >= 0.5]        <-0
              speed_limit[sl65 >= 0.5]        <-0
              speed_limit[sl70 >= 0.5]        <-1
              speed_limit[sl75 >= 0.5]        <-1
              speed_limit[sl70plus >= 0.5]    <-1
})

pdata <- pdata %>%
  mutate(speed_limit = is.ordered(speed_limit))
table(is.na(pdata$speed_limit))
```

```
##
## FALSE
##  1200
```

```
NA_sl<- pdata %>%
  filter(is.na(speed_limit) == TRUE)%>%
  select(speed_limit, sl55, sl65, sl70, sl75, sl70plus, year, state)
#number of nas

NA_sl
```

```
## [1] speed_limit sl55        sl65        sl70        sl75        sl70plus
## [7] year        state
## <0 rows> (or 0-length row.names)
```

```
head(pdata['speed_limit'])
```

```
##        speed_limit
## 1-1980       FALSE
## 1-1981       FALSE
## 1-1982       FALSE
## 1-1983       FALSE
## 1-1984       FALSE
## 1-1985       FALSE
```

Answer The data comprises 1200 observations, each observation represents yearly data for a single contiguous US state from 1980 to 2004, so for each of the 48 states, there are 25 observations in the dataset and is therefore in panel data structure. There are 56 variables in total, relating to driving fatality statistics, driving laws, and some demographic factors. The fatality data is from the Federal Analysis Reporting System (FARS) by the National Highway and Traffic Safety Administration which records data on all crashes that are fatal, the reporting is standard across states. Every fatal car crash is recorded so this is census data that represents all traffic fatalities in the contiguous US. The variable of interest, ‘total_fatalities_rate’, is defined as the number of fatalities due to car crashes for a given state and year per 100,000 people, based on the state population for that year.

```
pdata <- within(pdata,{
              state_no <- state})
#pdata$state_no
#pdata$stat_no <- pdata$state
pdata$state <- recode(pdata$state,
              '1' =  'AL',
              '3' = 'AZ',
              '4' = 'AR',
              '5' = 'CA',
              '6' = 'CO',
              '7' = 'CT',
              '8' = 'DE',
              '9' = 'DC',
              '10' = 'FL',
              '11' =  'GA',
              '12' = 'HI',
              '13' = 'ID',
              '14' = 'IL',
              '15' = 'IN',
              '16' = 'IA',
              '17' = 'KS',
              '18' = 'KY',
              '19' = 'LA',
              '20' =  'ME',
              '21' = 'MT',
              '22' = 'NE',
              '23' = 'NV',
              '24' = 'NH',
              '25' = 'NJ',
              '26' = 'NM',
              '27' = 'NY',
              '28' = 'NC',
              '29' = 'ND',
              '30' = 'OH',
              '31' = 'OK',
              '32' = 'OR',
              '33' = 'MD',
              '34' = 'MA',
              '35' = 'MI',
              '36' = 'MN',
              '37' = 'MS',
              '38' = 'MO',
              '39' = 'PA',
              '40' = 'RI',
              '41' = 'SC',
              '42' = 'SD',
              '43' = 'TN',
              '44' = 'TX',
              '45' = 'UT',
              '46' = 'VT',
              '47' = 'VA',
              '48' = 'WA',
              '49' = 'WV',
              '50' = 'WI',
              '51' = 'WY'
)

#Creating a year variable
pdata <- within(pdata,{
              year_of_observation <- 0
              year_of_observation[d80 == 1]        <-1980
              year_of_observation[d81 == 1]        <-1981
              year_of_observation[d82 == 1]        <-1982
              year_of_observation[d83 == 1]        <-1983
              year_of_observation[d84 == 1]        <-1984
              year_of_observation[d85 == 1]        <-1985
              year_of_observation[d86 == 1]        <-1986
              year_of_observation[d87 == 1]        <-1987
              year_of_observation[d88 == 1]        <-1988
              year_of_observation[d89 == 1]        <-1989
              year_of_observation[d90 == 1]        <-1990
              year_of_observation[d91 == 1]        <-1991
              year_of_observation[d92 == 1]        <-1992
              year_of_observation[d93 == 1]        <-1993
              year_of_observation[d94 == 1]        <-1994
              year_of_observation[d95 == 1]        <-1995
              year_of_observation[d96 == 1]        <-1996
              year_of_observation[d97 == 1]        <-1997
              year_of_observation[d98 == 1]        <-1998
              year_of_observation[d99 == 1]        <-1999
              year_of_observation[d00 == 1]        <-2000
              year_of_observation[d01 == 1]        <-2001
              year_of_observation[d02 == 1]        <-2002
              year_of_observation[d03 == 1]        <-2003
              year_of_observation[d04 == 1]        <-2004
})

table(is.na(pdata$year_of_observation))
```

```
##
## FALSE
##  1200
```

```
head(pdata['year_of_observation'])
```

```
##        year_of_observation
## 1-1980                1980
## 1-1981                1981
## 1-1982                1982
## 1-1983                1983
## 1-1984                1984
## 1-1985                1985
```

```
#converting the blood alcohol variable to 0, 8, 10


pdata <- within(pdata,{
              bac <- 0
              bac[bac08 >= 0.5]                   <- 0.08
              bac[bac10 >= 0.5]                   <- 0.10
              bac[!is.finite(bac08)]              <- 0
              bac[!is.finite(bac10)]              <- 0
              bac[bac08 ==0 & bac10 == 0]         <- 0
              bac[bac10 == 0.5 & bac08 >= 0.5]    <- 0.08
              bac[bac10 > 0 & bac08 == 0]         <- 0.10
              bac[bac10 == 0 & bac08 > 0]         <- 0.08
})

pdata<- within(pdata, {
            minimum_age <- 0
            minimum_age[minage<18.5]                  <- 18
            minimum_age[minage>=18.5 & minage<19.5]   <- 19
            minimum_age[minage>= 19.5 & minage<20.5]  <- 20
            minimum_age[minage>= 20.5 & minage<21.5]  <- 21

})

table(is.na(pdata$bac))
```

```
##
## FALSE
##  1200
```

```
NA_obs<- pdata %>%
  filter(is.na(bac) == TRUE)%>%
  select(bac, bac08, bac10, year, state)

NA_obs
```

```
## [1] bac   bac08 bac10 year  state
## <0 rows> (or 0-length row.names)
```

```
pdata <- pdata %>%
  mutate(bac <- is.ordered(bac)) %>%
  mutate(minimum_age <- is.ordered(minimum_age))
#Order minimum drinking age
```

```
#Clean per se variable or the one indicating DUIs can be issued for increased BAC above level for drivi
pdata <- within(pdata,{
              per_se <- 0
              per_se[perse > 0]        <- 1
})

pdata <- within(pdata, {
  primary_seatbelt <-0
  primary_seatbelt[seatbelt == 1]     <-1
})

pdata <- within(pdata, {
  secondary_seatbelt <- 0
  secondary_seatbelt[seatbelt == 2] <-1
})

pdata <- pdata %>%
  #select(!c(d80:d04)) %>%
  select(!c(sl55:sl75)) %>%
  select(!c(bac08:bac10)) %>%
  select(!perse) %>%
  #select(!seatbelt) %>%
  select(!sbprim) %>% select(!sbsecon)
#Renaming variables
change <- c(total_fatalities_rate = 'totfatrte', unemployment = 'unem',
            total_fatalities = 'totfat', night_fatalities = 'nghtfat' , weekend_fatalities = 'wkndfat',
            total_fatalities_one_hundred_million_miles = 'totfatpvm',  night_fatalities_one_hundred_mil
            night_fatalities_rate = 'nghtfatrte',
            zero_tolerance = 'zerotol',  weekend_fatalities_rate = 'wkndfatrte',
            vehicle_miles = 'vehicmiles', blood_alcohol = 'bac', percent_14_to_24 = 'perc14_24')
pdata <- pdata %>%
  rename(all_of(change))
```

##Data Transformations

```
# speed limit column
data$speed_limit <- ifelse(data$sl55 >0, '55', NA)
data$speed_limit <- ifelse(data$sl65 > 0, '65', data$speed_limit)
```

```r
data$speed_limit <- ifelse(data$sl70 > 0, '70', data$speed_limit)
data$speed_limit <- ifelse(data$sl75 > 0, '75', data$speed_limit)
data$speed_limit <- ifelse(data$slnone > 0, 'None', data$speed_limit)
data$perse_binary <- ifelse(data$perse > 0.5, 1, 0)
data$gdl_binary <- ifelse(data$gdl > 0.5, 1, 0)
data$sbprim_binary <- ifelse(data$sbprim > 0.5, 1, 0)
data$sbsec_binary <- ifelse(data$sbsecon > 0.5, 1, 0)
data$sl70plus <- ifelse(data$sl70plus > 0.5, 1, 0)

col_num <- 31:55
years <- 1980:2004

year_of_observation <- rep(NA, 1200)

data$year_test <- year_of_observation



for (i in 1:25){
 data$year_test <- ifelse(data[, col_num[i]]== 1, years[i], data$year_test)
}


data$bac <- ifelse(data$bac08 >0.5, "0.08", "None")
data$bac <- ifelse(data$bac10 > 0.5, "0.1", data$bac)



data$total_fatalities_rate <- data$totfatrte
data$night_fatalities_rate <- data$nghtfatrte
data$weekend_fatalities_rate <- data$wkndfatrte
data <- data %>%
  select(-totfatrte, -nghtfatrte, -wkndfatrte)
head(data)
```

```
##   year state sl55 sl65 sl70 sl75 slnone seatbelt minage zerotol gdl bac10 bac08
## 1 1980     1    1    0    0    0      0        0     18       0   0     1     0
## 2 1981     1    1    0    0    0      0        0     18       0   0     1     0
## 3 1982     1    1    0    0    0      0        0     18       0   0     1     0
## 4 1983     1    1    0    0    0      0        0     18       0   0     1     0
## 5 1984     1    1    0    0    0      0        0     18       0   0     1     0
## 6 1985     1    1    0    0    0      0        0     20       0   0     1     0
##   perse totfat nghtfat wkndfat totfatpvm nghtfatpvm wkndfatpvm statepop
## 1     0    940     422     236      3.20      1.437      0.803  3893888
## 2     0    933     434     248      3.35      1.558      0.890  3918520
## 3     0    839     376     224      2.81      1.259      0.750  3925218
## 4     0    930     397     223      3.00      1.281      0.719  3934109
## 5     0    932     421     237      2.83      1.278      0.720  3951834
## 6     0    882     358     224      2.51      1.019      0.637  3972527
##   vehicmiles unem perc14_24 sl70plus sbprim sbsecon d80 d81 d82 d83 d84 d85 d86
## 1   29.37500  8.8      18.9        0      0       0   1   0   0   0   0   0   0
## 2   27.85200 10.7      18.7        0      0       0   0   1   0   0   0   0   0
## 3   29.85765 14.4      18.4        0      0       0   0   0   1   0   0   0   0
## 4   31.00000 13.7      18.0        0      0       0   0   0   0   1   0   0   0
## 5   32.93286 11.1      17.6        0      0       0   0   0   0   0   1   0   0
## 6   35.13944  8.9      17.3        0      0       0   0   0   0   0   0   1   0
##   d87 d88 d89 d90 d91 d92 d93 d94 d95 d96 d97 d98 d99 d00 d01 d02 d03 d04
## 1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 3   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 4   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 5   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##   vehicmilespc speed_limit perse_binary gdl_binary sbprim_binary sbsec_binary
## 1     7543.874          55            0          0             0            0
## 2     7107.785          55            0          0             0            0
## 3     7606.622          55            0          0             0            0
## 4     7879.802          55            0          0             0            0
## 5     8333.562          55            0          0             0            0
## 6     8845.614          55            0          0             0            0
##   year_test bac total_fatalities_rate night_fatalities_rate
## 1      1980 0.1                 24.14                 10.84
## 2      1981 0.1                 24.07                 11.08
## 3      1982 0.1                 21.37                  9.58
## 4      1983 0.1                 23.64                 10.09
## 5      1984 0.1                 23.58                 10.65
## 6      1985 0.1                 22.20                  9.01
##   weekend_fatalities_rate
## 1                    6.06
## 2                    6.33
## 3                    5.71
## 4                    5.67
## 5                    6.00
## 6                    5.64
```

2.0.1 Total Fatalities Rate

```
avg_fatalities <- mean(pdata$total_fatalities, na.rm=TRUE)
avg_fatalities
```

```
## [1] 900.7267
```

```
avg_population <- mean((pdata$state_population/100000), na.rm = TRUE)
avg_population
```

```
## [1] 53.29896
```

```
mean_tot_fat_rate <- avg_fatalities/avg_population
mean_tot_fat_rate
```

```
## [1] 16.89952
```

```
mean_total_fatalities_rate_variable = mean(pdata$total_fatalities_rate, na.rm=TRUE)
mean_total_fatalities_rate_variable
```

```
## [1] 18.91856
```

```
- What is the average of `total_fatalities_rate` in each of the years in the time period covered in thi
```

```
annual_mean_total_fatalities_rate <- pdata%>%
  group_by(year_of_observation)%>%
  summarize(avg_total_fatalities_rate = mean(total_fatalities_rate))
```

Answer: The main variable of interest is the total fatalities rate (TFR). The mean TFR across all time and observations is 18.92. This is slightly higher than the median TFR, 18.435, illustrated as the red line in the histogram. The fatalities distribution is skewed to the right. Outliers would lie either about the 95% quartile of a TFR greater or equal to 29.89 or below the 5th percentile (TFR$<=$9.58).In the following ACF diagram of annual average TFR across states also illustrates the autocorrelation across lagged TFR values.

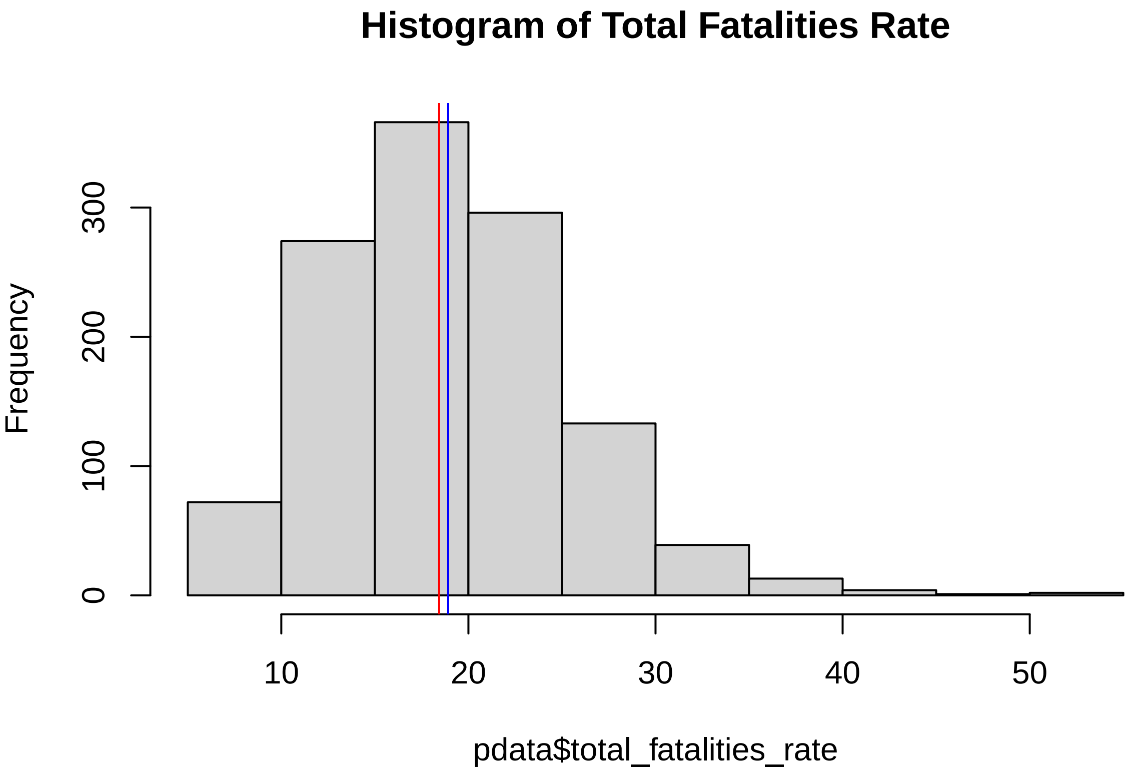
```
#Total_Fatalities
mean.tfr<-mean(pdata$total_fatalities_rate)
mean.tfr
```

```
## [1] 18.91856
```

```
median.tfr<-median(pdata$total_fatalities_rate)
median.tfr
```

```
## [1] 18.435
```

```
hist.tfr<- hist(pdata$total_fatalities_rate, main = "Histogram of Total Fatalities Rate")
abline(v = median(pdata$total_fatalities_rate),col = "red")
abline(v = mean(pdata$total_fatalities_rate),col = "blue")
```



```
labs("Histogram of Total Fatalities Rate")
```

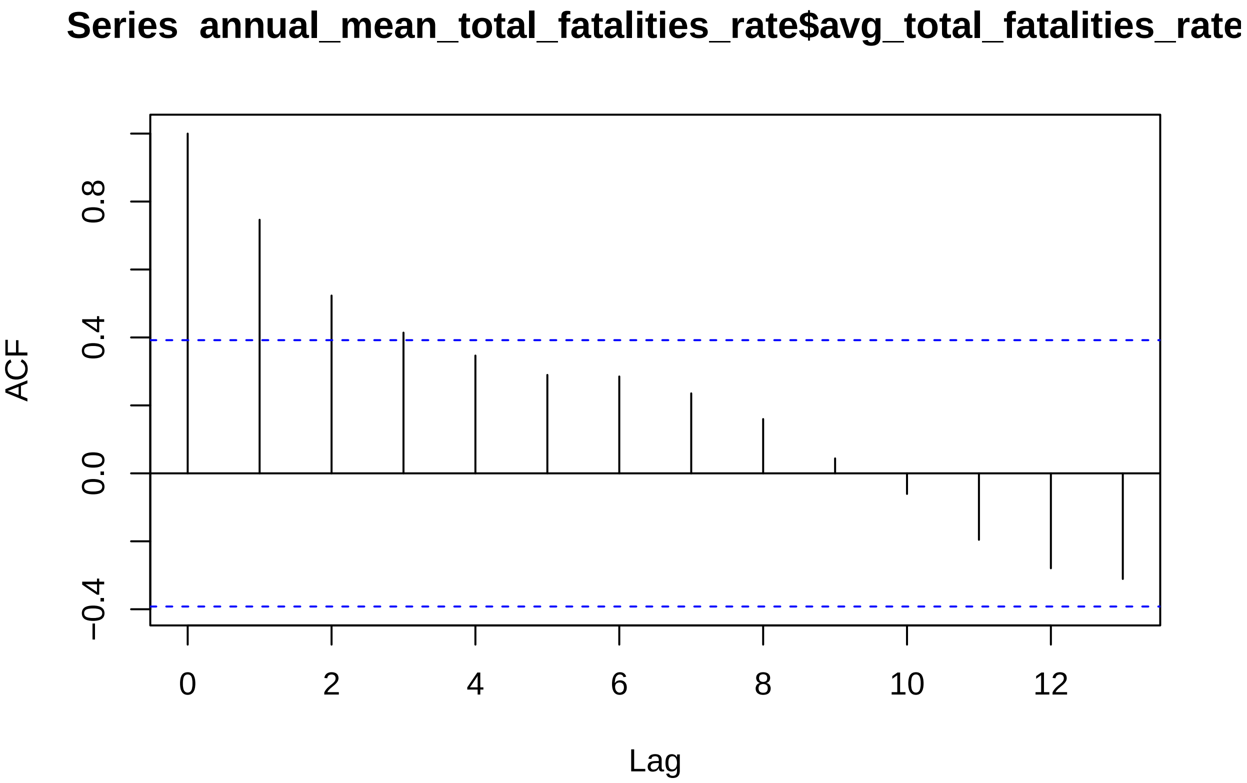
```
## [[1]]
## [1] "Histogram of Total Fatalities Rate"
##
## attr(,"class")
## [1] "labels"
```

```
tf_quantiles <- quantile(pdata[['total_fatalities_rate']], p = c(0.05, 0.25, 0.50, 0.75, 0.95))
```

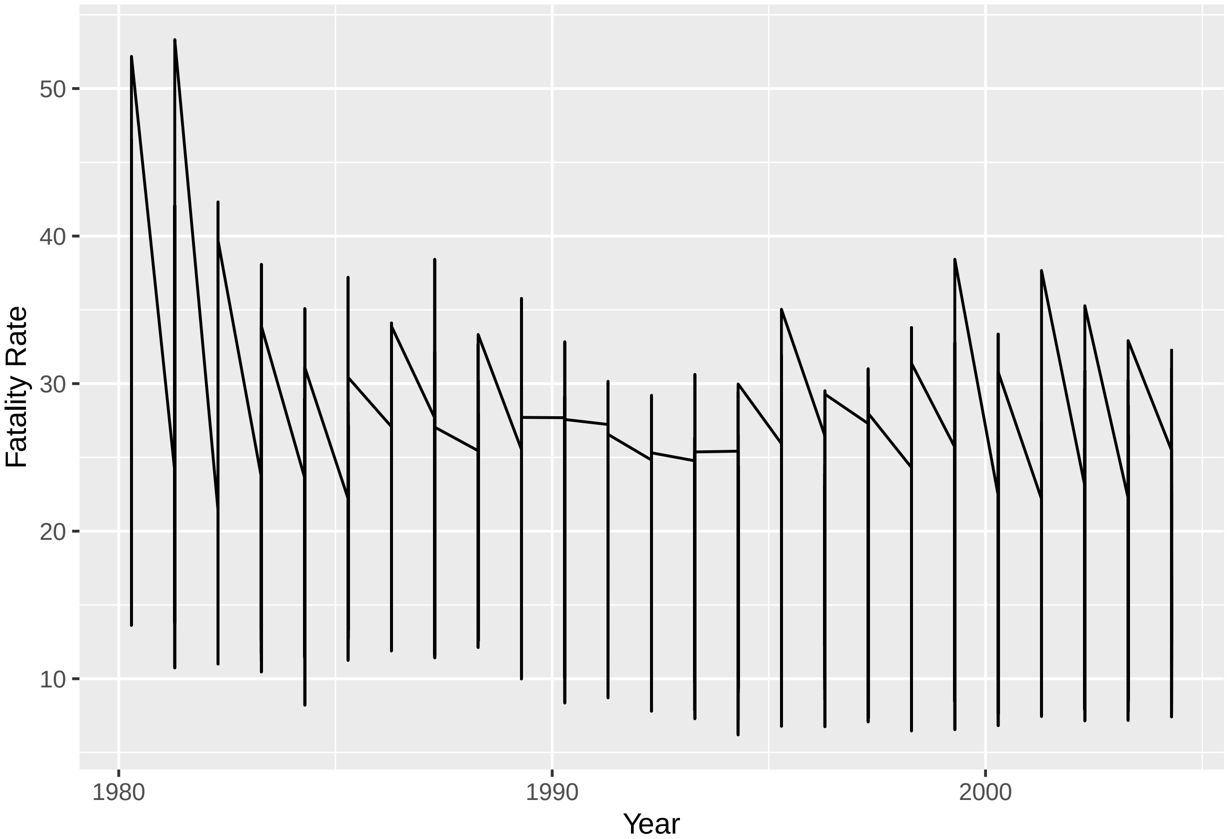
```
tf_quantiles
```

```
##      5%     25%     50%     75%     95% 
##  9.5780 14.3775 18.4350 22.7725 29.8945
```

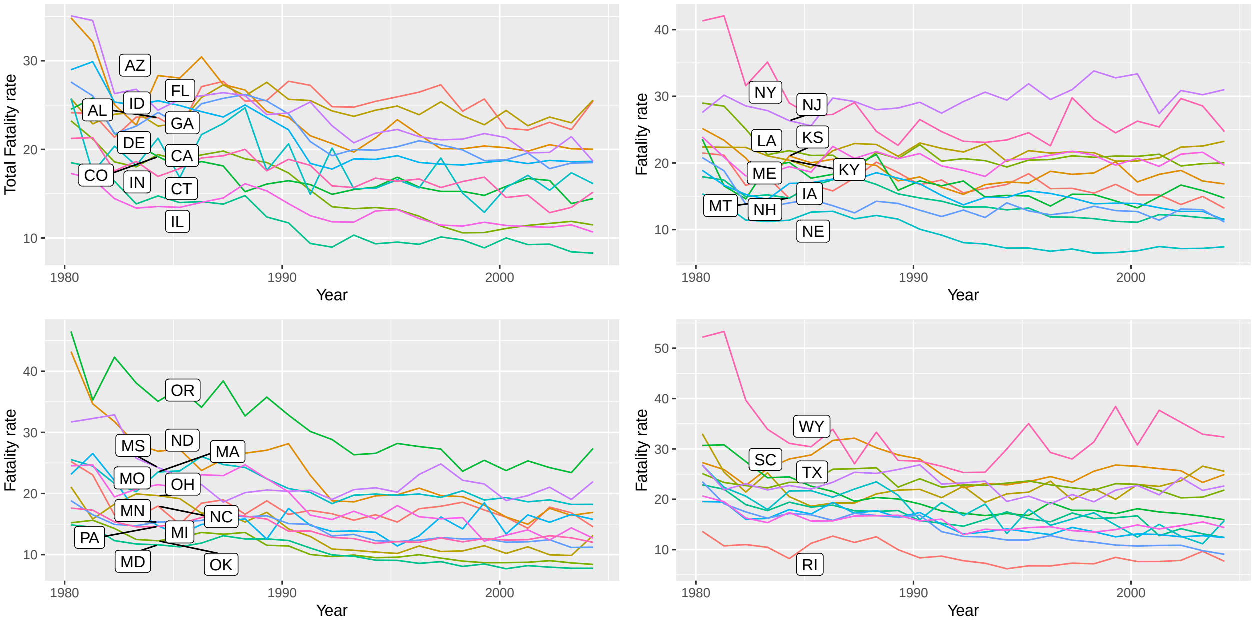
```
acf(annual_mean_total_fatalities_rate$avg_total_fatalities_rate)
```



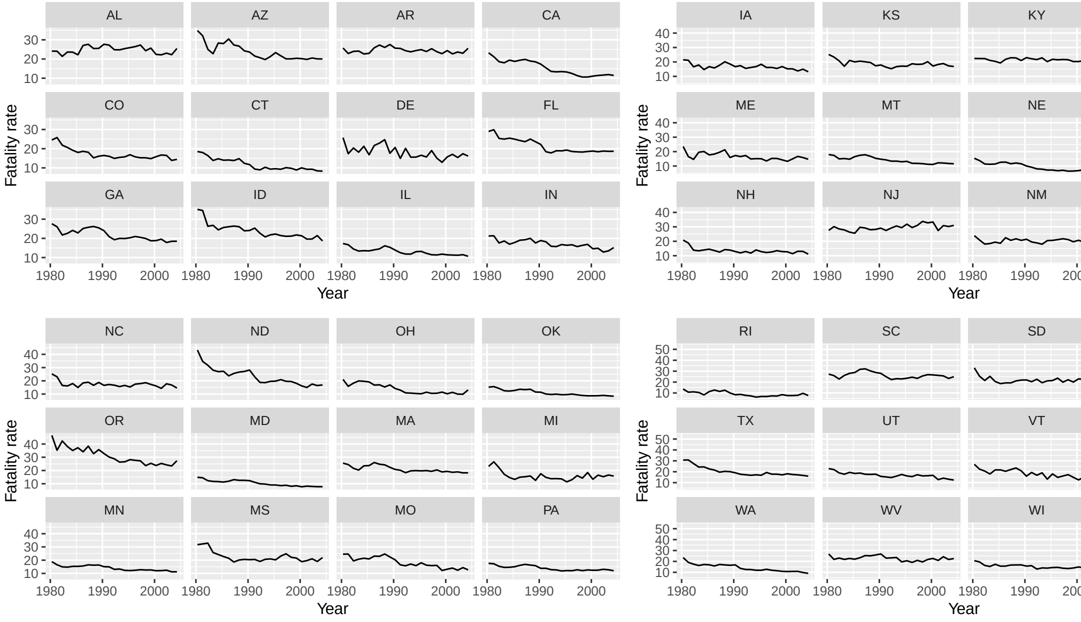
The total fatality rate is dynamic over time, as illustrated in the line plot of TFR across time.



Further visualization of TFR is presented by state in the following diagrams. The four panels include different groupings of the states, by alphabetical order. In the upper left panel, the state with the highest initial TFR is Arizona and the lowest is Illinois. As time progresses, TFR drops across all states, especially Deleware and Colorado. For the states represented in the upper right-hand panel, New York has a TFR about 40, one of the highest of all time. Only Nebraska appears to maintain a low TFR amoung members of this group. The next group of states, represented in the lower left-hand panel, appear to be more likely to have TFRs below 20 per 100,000 persons. The next group, in the lower right-hand panel, has the highest TFR per 100,000 persons across all groups. Wyoming starts out with well over 40 TFR per 100,000 persons in the 1980s. Rhode Island, also in the group, is among the states with the lowest TFR per 100,000.



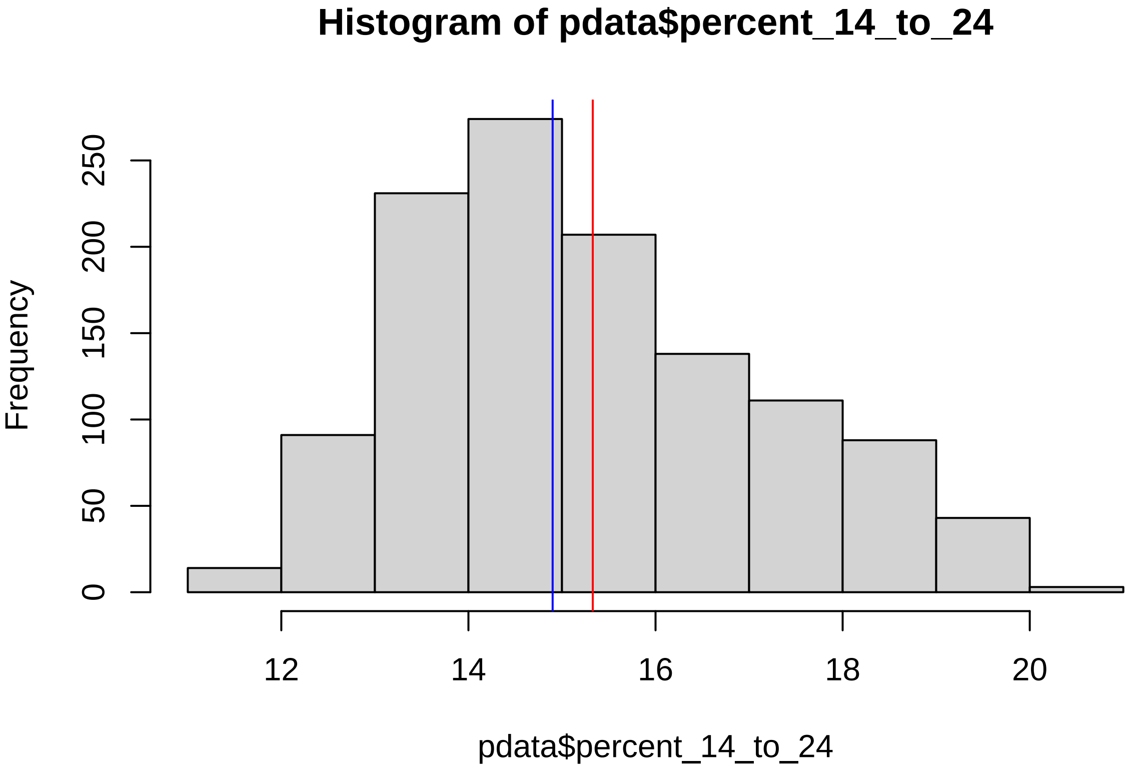
The next set of diagrms seperates all TFRs across all states. This additional view of TFRs allows us to see the diferences in TFR variation with more clarity. While states such as Deleware had relatively low TFR by 2004, the diagram indicates higher rates of variation than statues such as Kansas, Kentucky, Nebraska, Minnesota, Pennsylvania, and Washington.

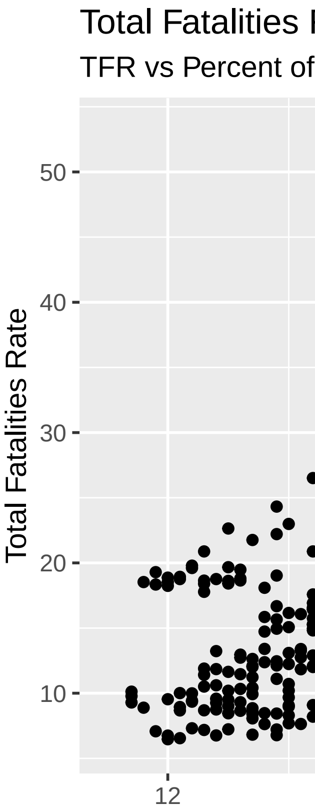


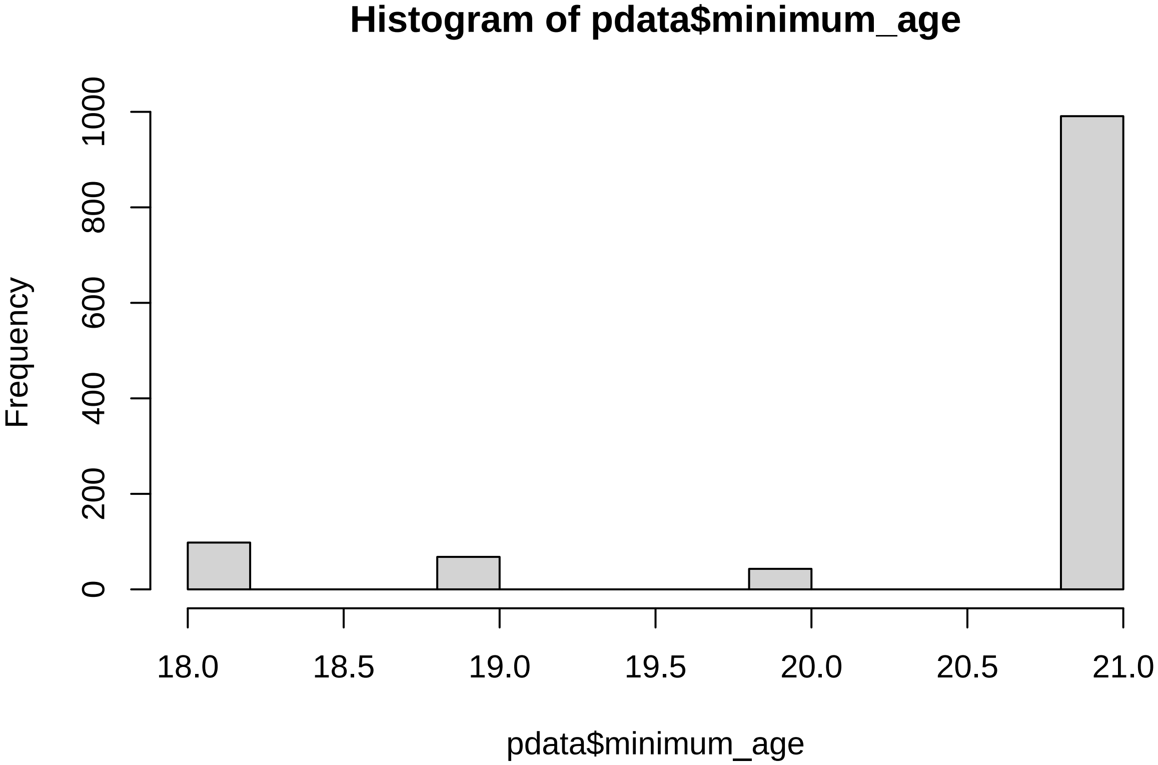
We now turn our attention to other variables of interest with relationship to TFR. These variables include measures of young people in the population, unemployment, miles driven per capita, seatbelt laws, and drunk driving regulations. Across the nation, the percent of the population between ages 14 and 24 years is 15.33 percent. This is almost half a percent higher than the median, 14.29 percent. The distribution is skewed to the right. The legal age to drink alcohol is primary 21 or in 991 of the cases minimum age is 21. It is 18 in 98 cases, 19 in 68 observations, and 20 in 42 observations. We may want to consider a dummy variable for minimum age is at or above 21 is one and zero otherwise.

2.0.2 Youth

```
## [1] 15.32908
## [1] 14.9
```

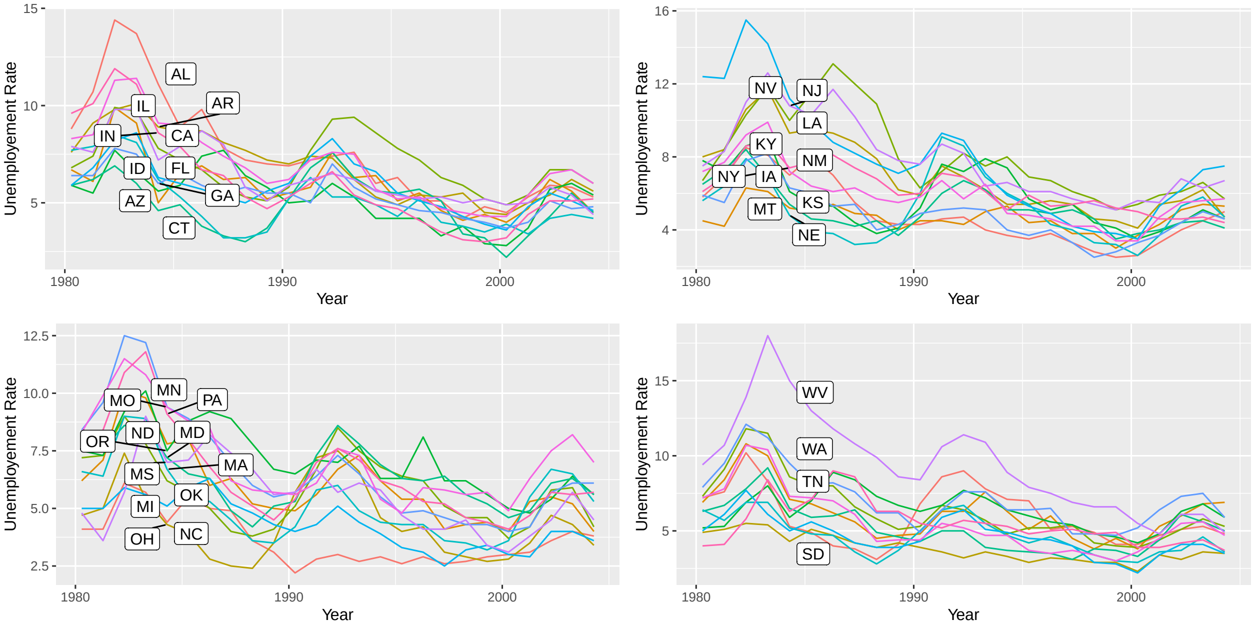






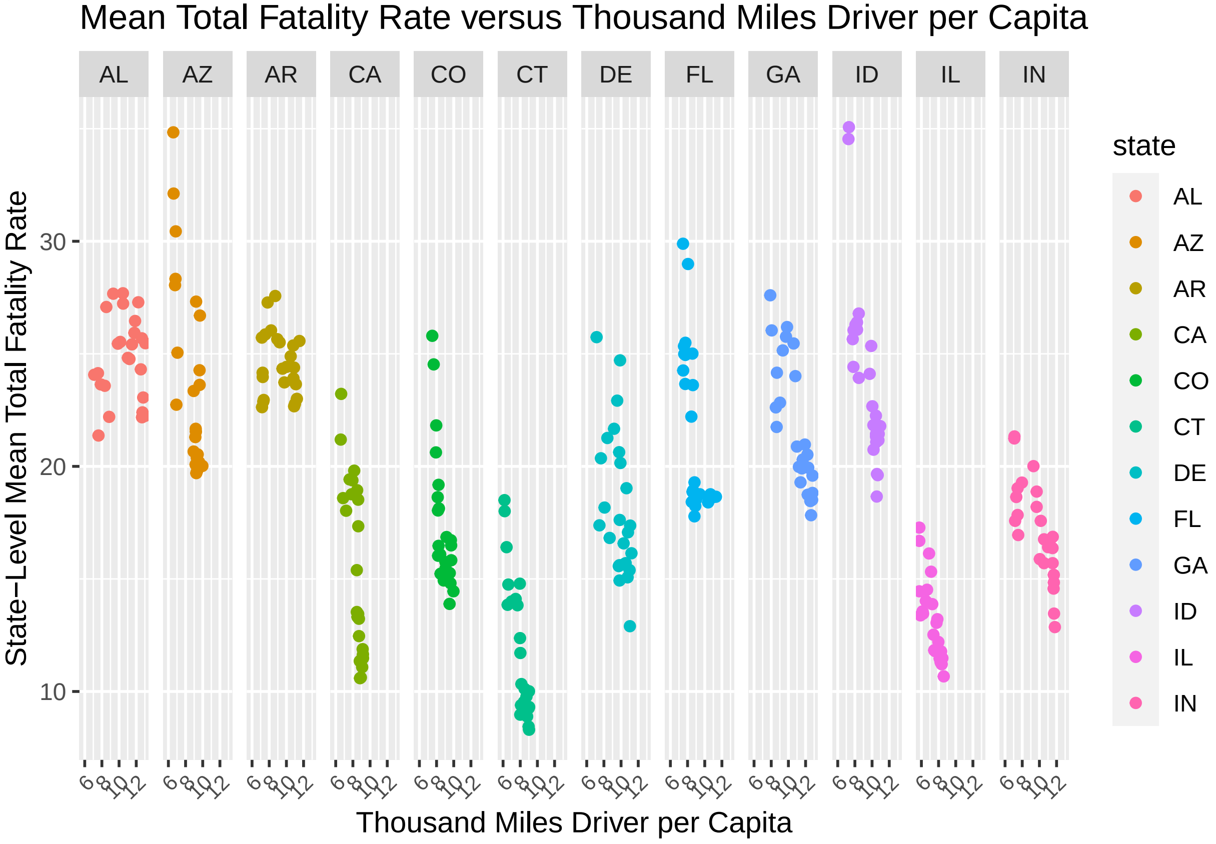
```
##  [1]   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0  98  68
## [20]  43 991
```

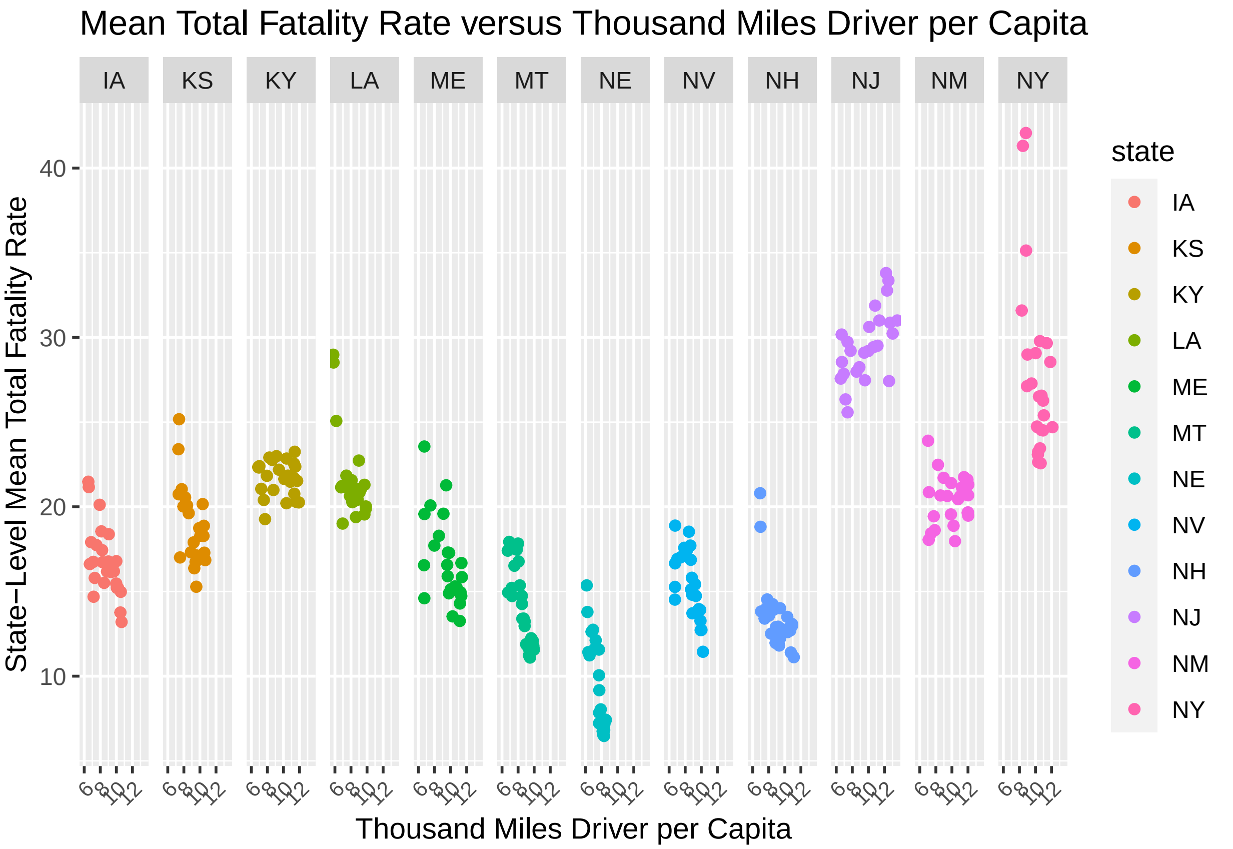
2.0.3 Unemployment

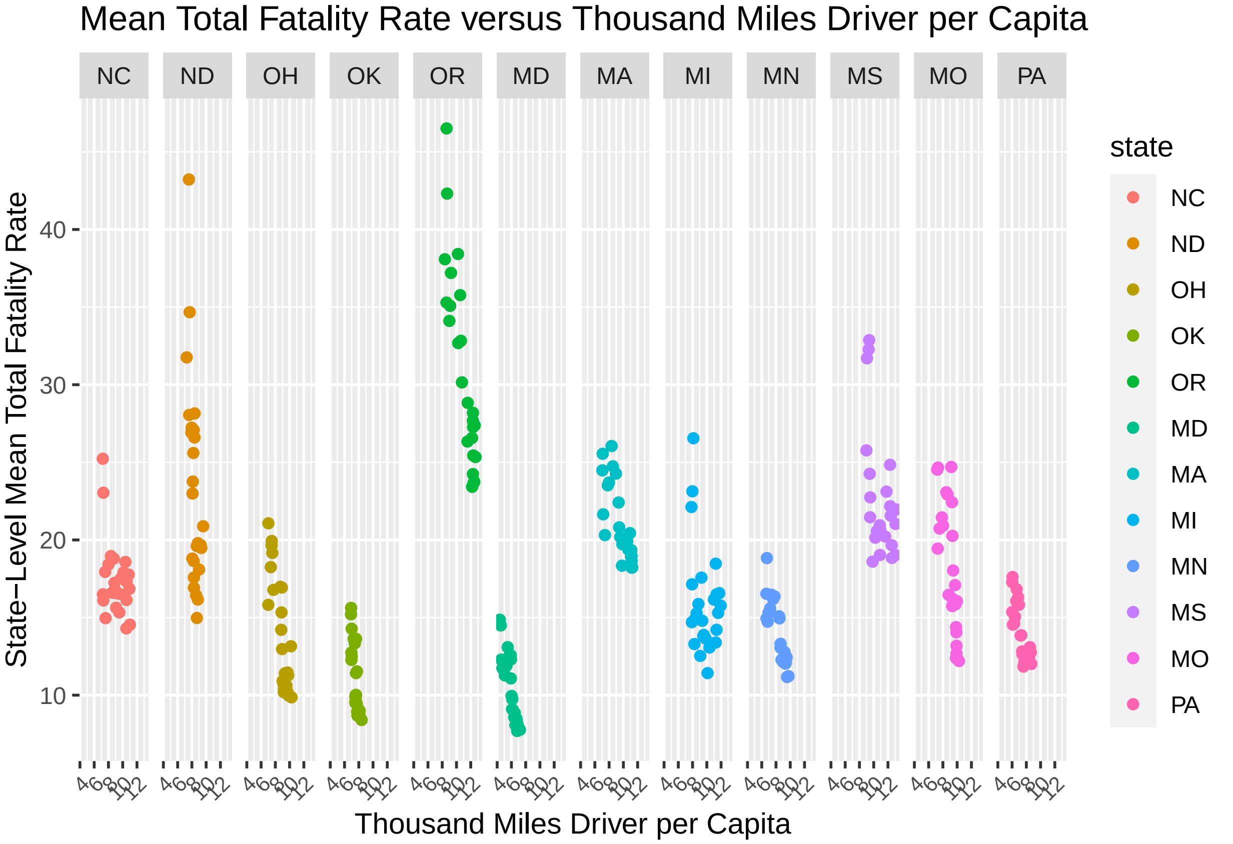


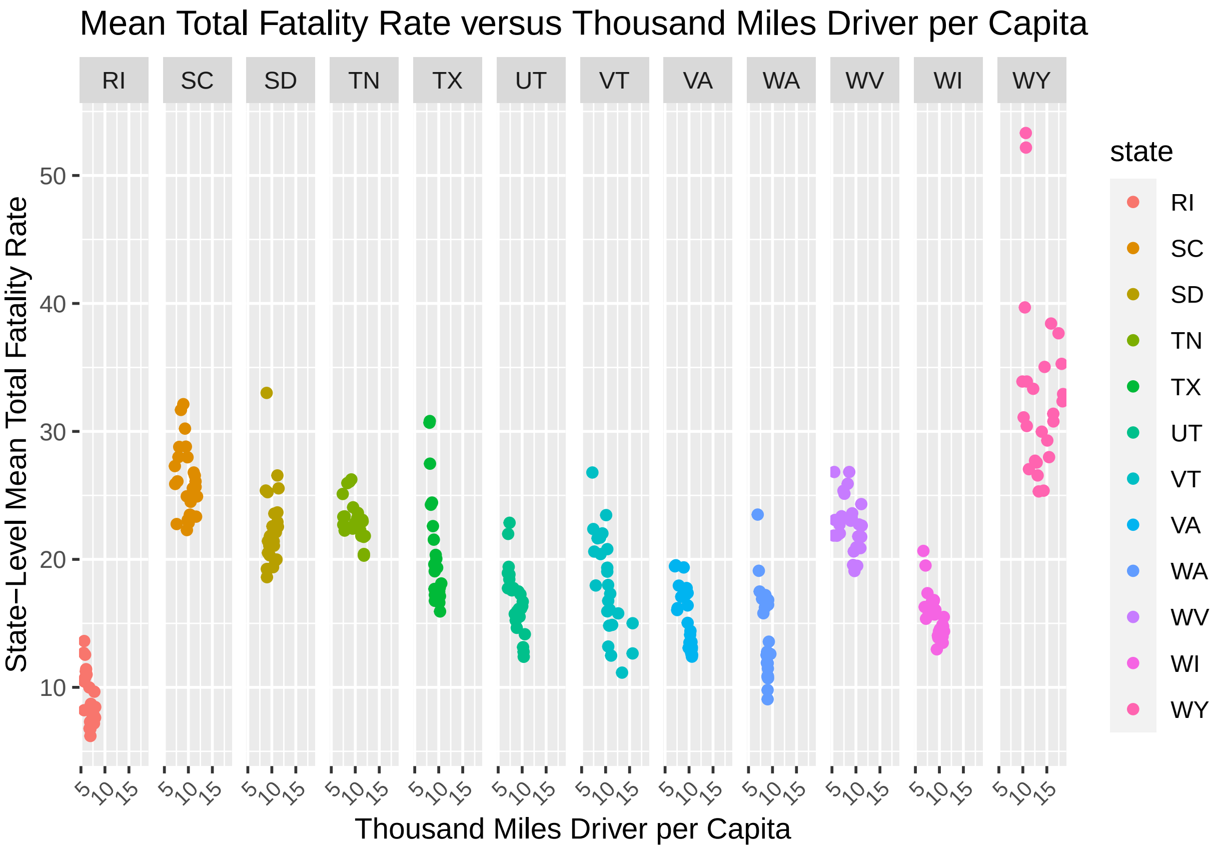
### Miles Per Capita

We converted the miles driven per state variable from billion miles per state to thousands of miles per capita for visualiation. The four panel visualization illustrates a mixed relationship between miles driven per capita and total fatalities rate. There is an indication of a positive relationship between miles driven per capita and total fatalities rate when looking at extreme examples of high fatality states (i.e., Wyoming) and low fatality states (i.e., Delaware and Rhode Island). The correlation between the TFR and total vehicle miles driven is -0.26.



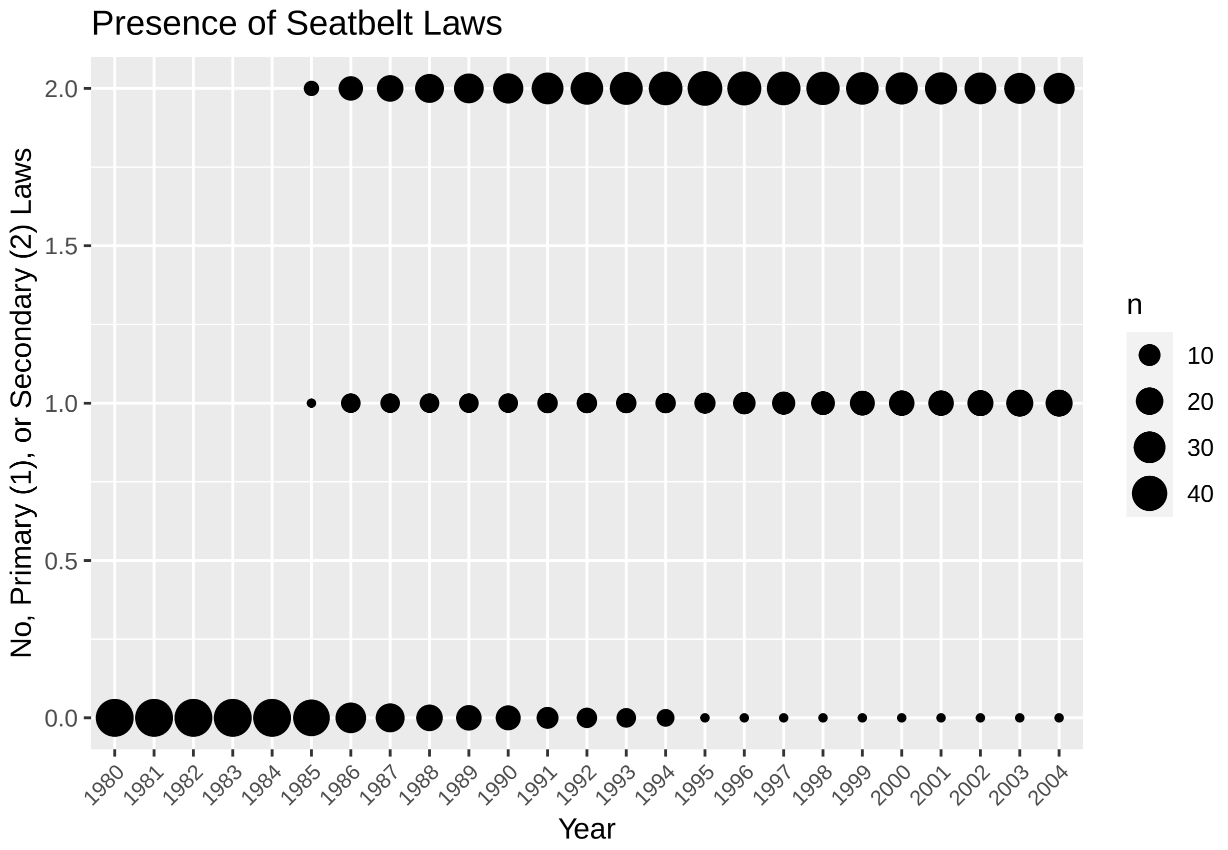

Mean Total Fatality Rate versus Thousand Miles Driver per Capita
IA
KS
KY
LA
ME
MT
NE
NV
NH
NJ
NM
NY
State−Level Mean Total Fatality Rate
40
30
20
10
6
8
10
12
Thousand Miles Driver per Capita
state
● IA
● KS
● KY
● LA
● ME
● MT
● NE
● NV
● NH
● NJ
● NM
● NY


Mean Total Fatality Rate versus Thousand Miles Driver per Capita
NC
ND
OH
OK
OR
MD
MA
MI
MN
MS
MO
PA
State−Level Mean Total Fatality Rate
40
30
20
10
4
6
8
10
12
Thousand Miles Driver per Capita
state
● NC
● ND
● OH
● OK
● OR
● MD
● MA
● MI
● MN
● MS
● MO
● PA



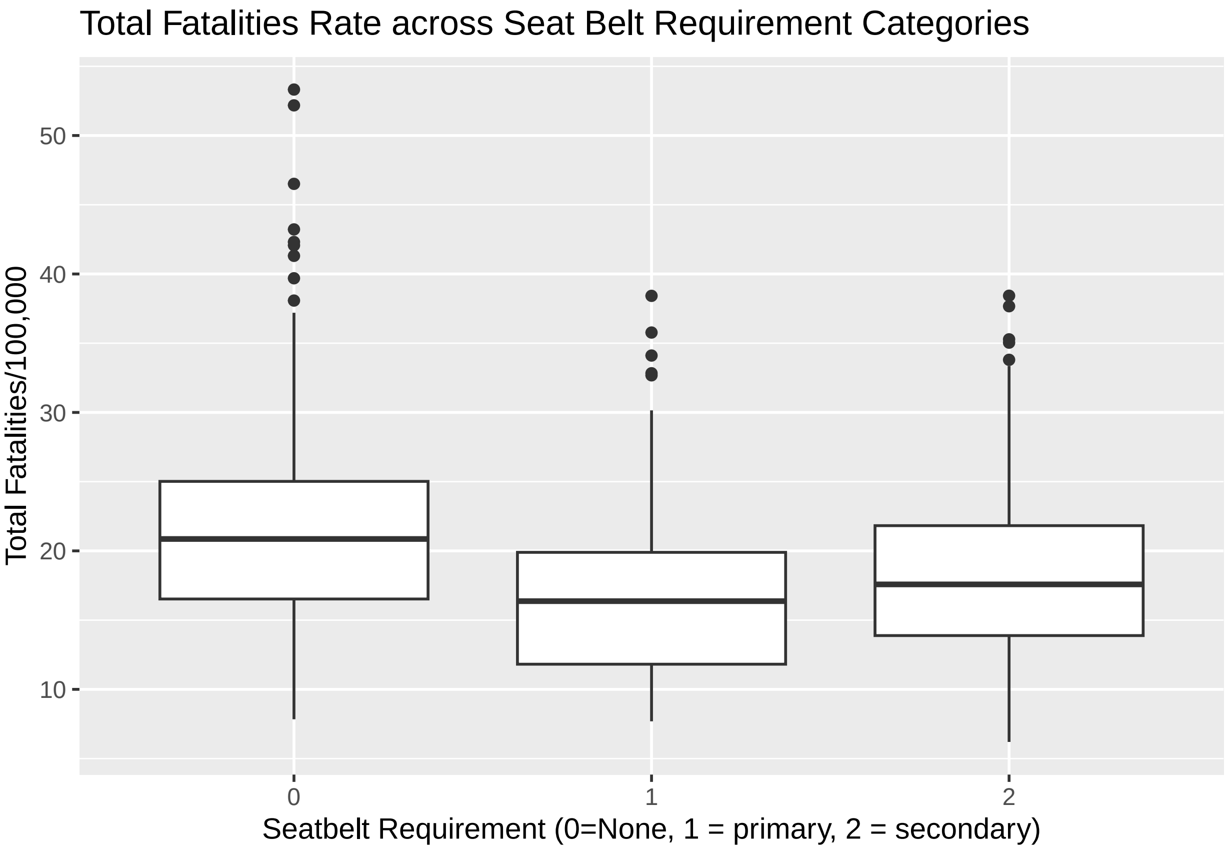
### Seatbelt Regulation

There are a number of NAs in the Seatbelt regulation variable (n > 900). Still, over 99 percent of all panel observations included either primary or secondary seatbelt use. There is little variation in this variable. Essentially, it appears seatbelts were mandatory across locations by 1985.



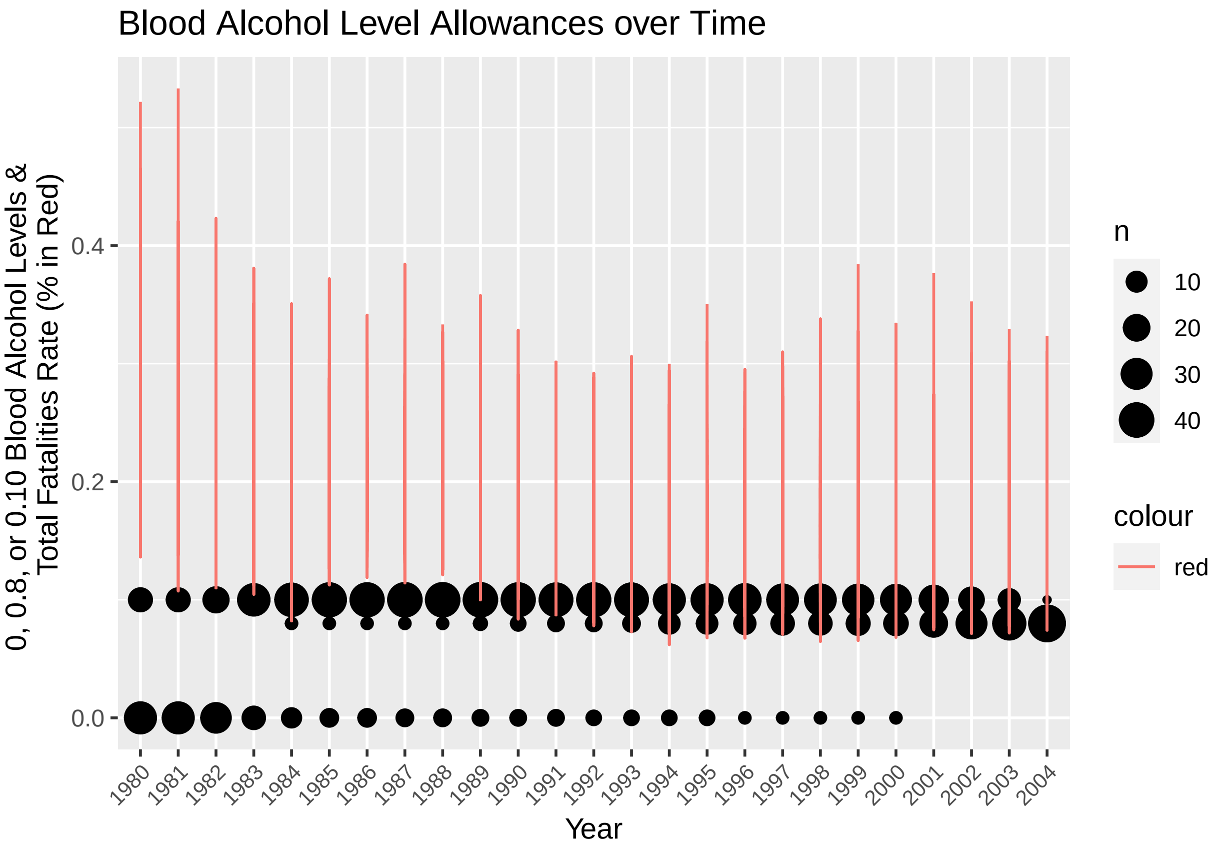
```
fatalities_seatbelt
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA

## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
```

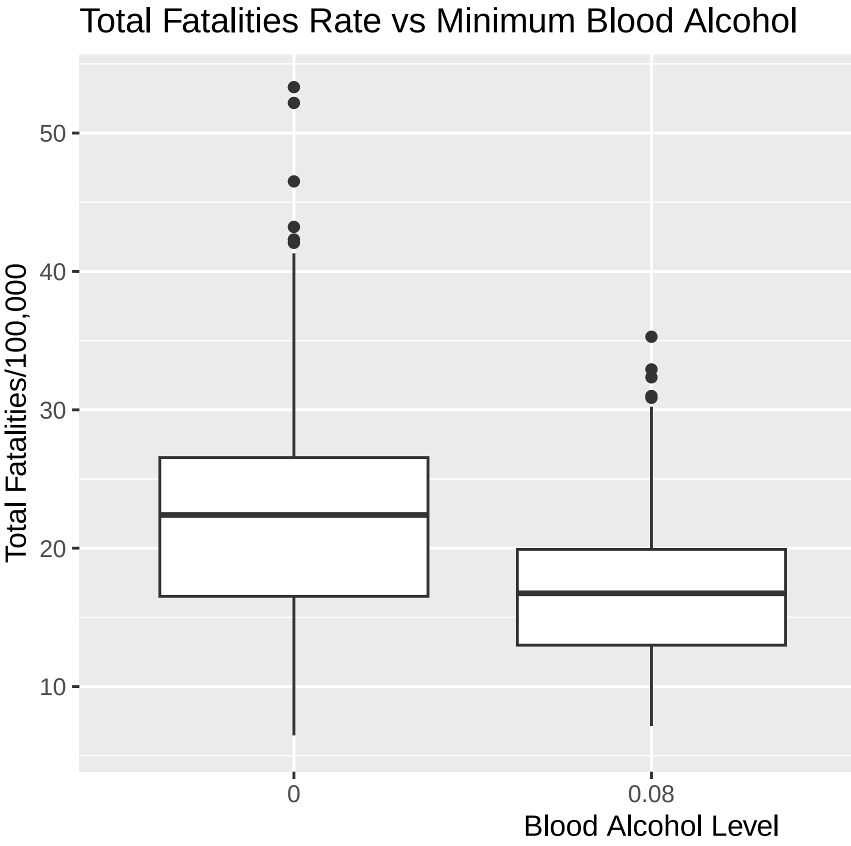


2.0.4 Drunk Driving

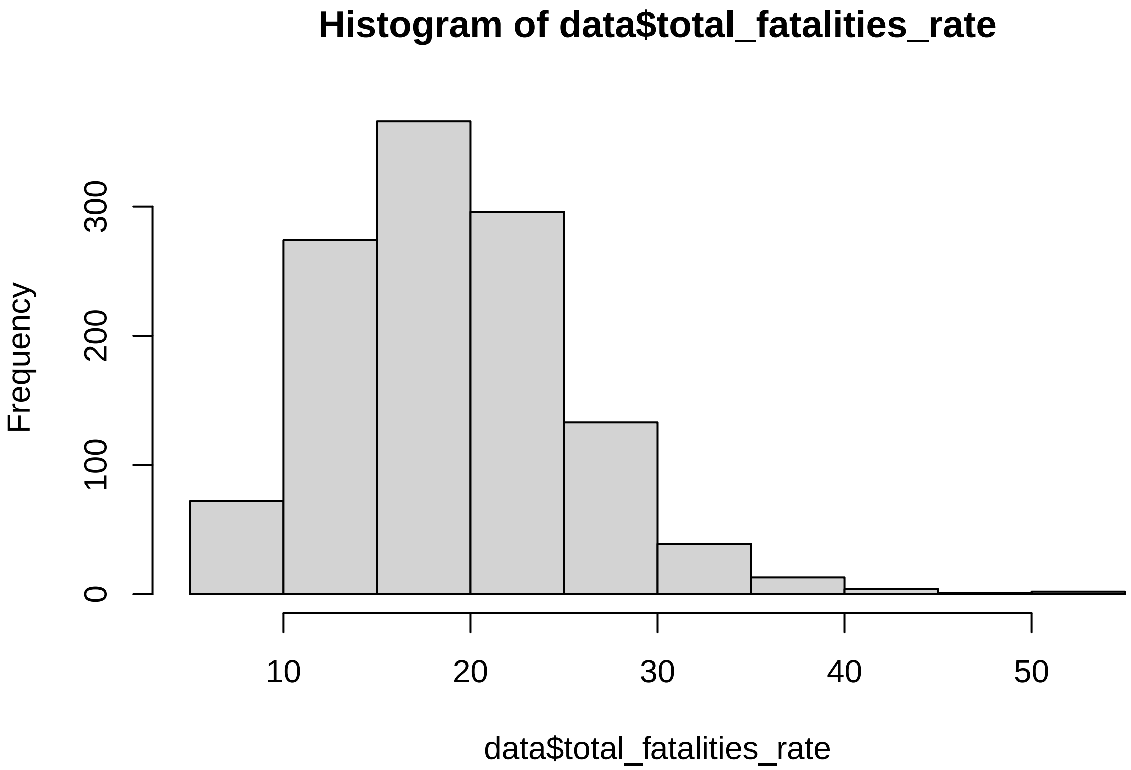
The blood alcohol variable indicates the level of constraint on drivers regarding the amount of alcohol they may consume and not be considered driving drunk. As the following figure illustrates, most states started to cap blood alcohol at 0.10 in prior to 1980. Then, starting in 1984, states increasingly capped the blood alcohol level at 0.85 for drunk driving. The red vertical bars represent the spread of TFR rates across states each year. The trend indicates a decrease in TFR as more states lower allowable blood alcohol levels.



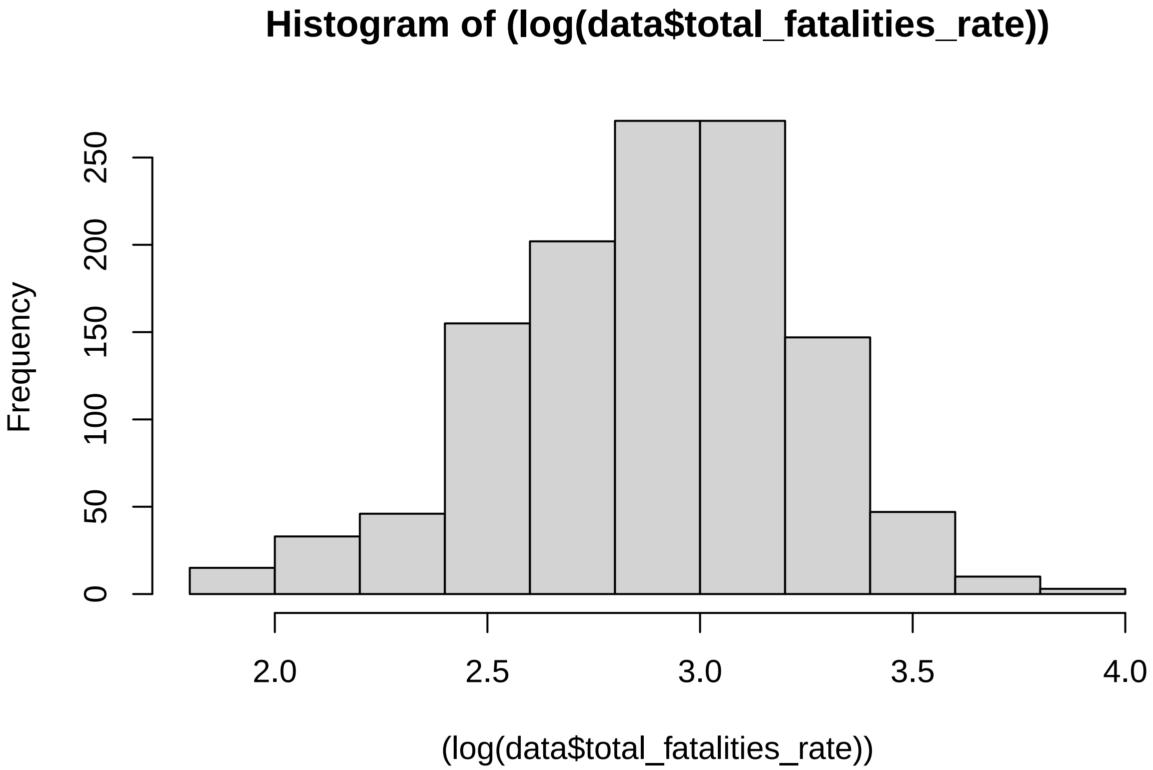
The following box plot shows the relationship between blood alcohol restrictions and fatalities. Mean TFR is lowest when the blood alcohol limit is 0.85 in a state.



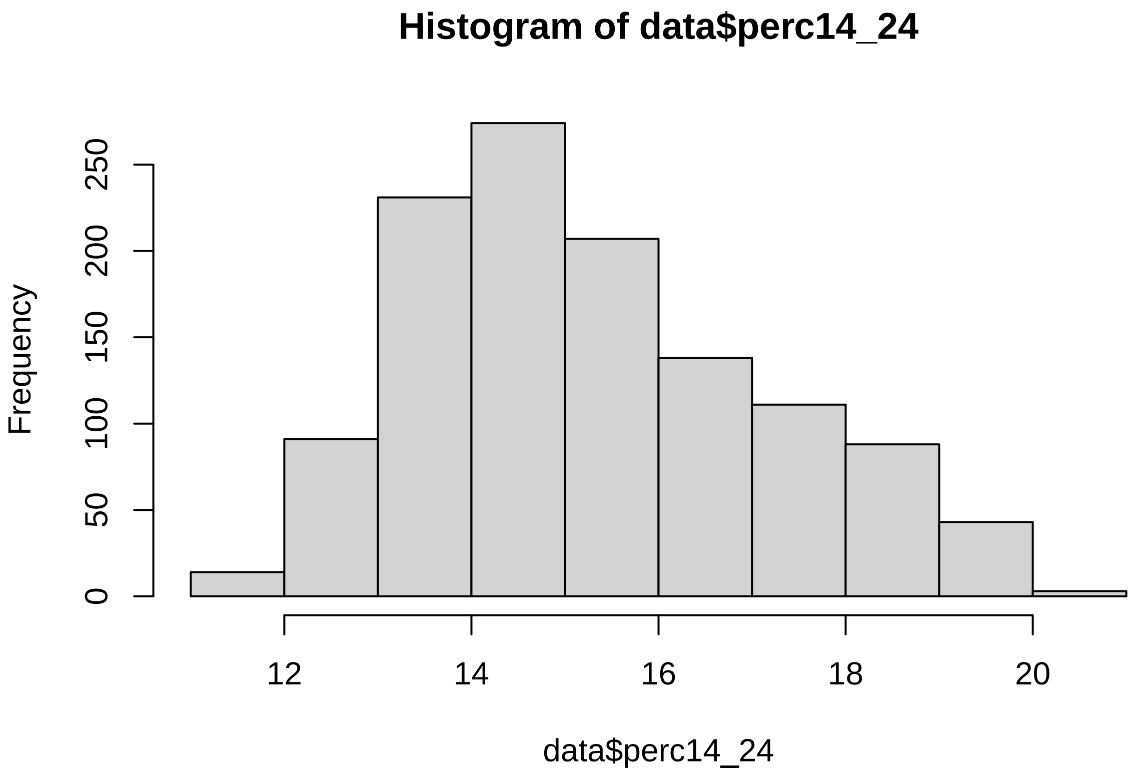
```r
hist(data$total_fatalities_rate)
```



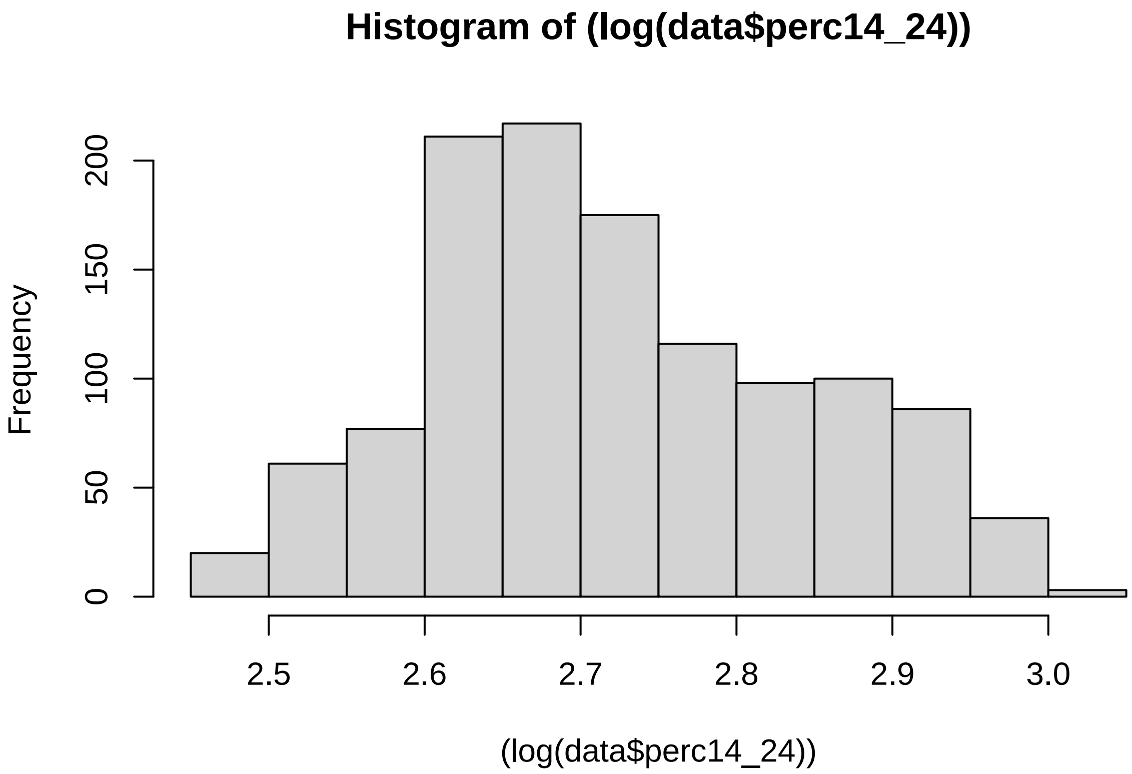
```r
hist((log(data$total_fatalities_rate)))
```



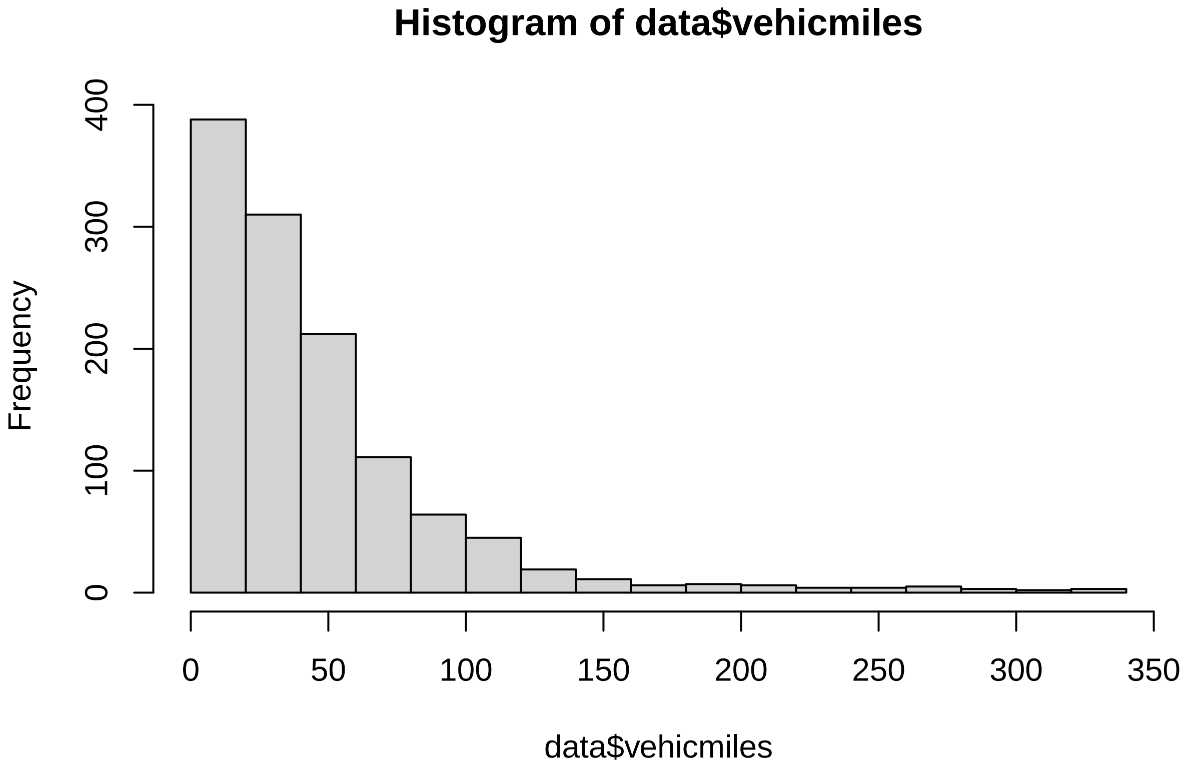
```
hist(data$perc14_24)
```



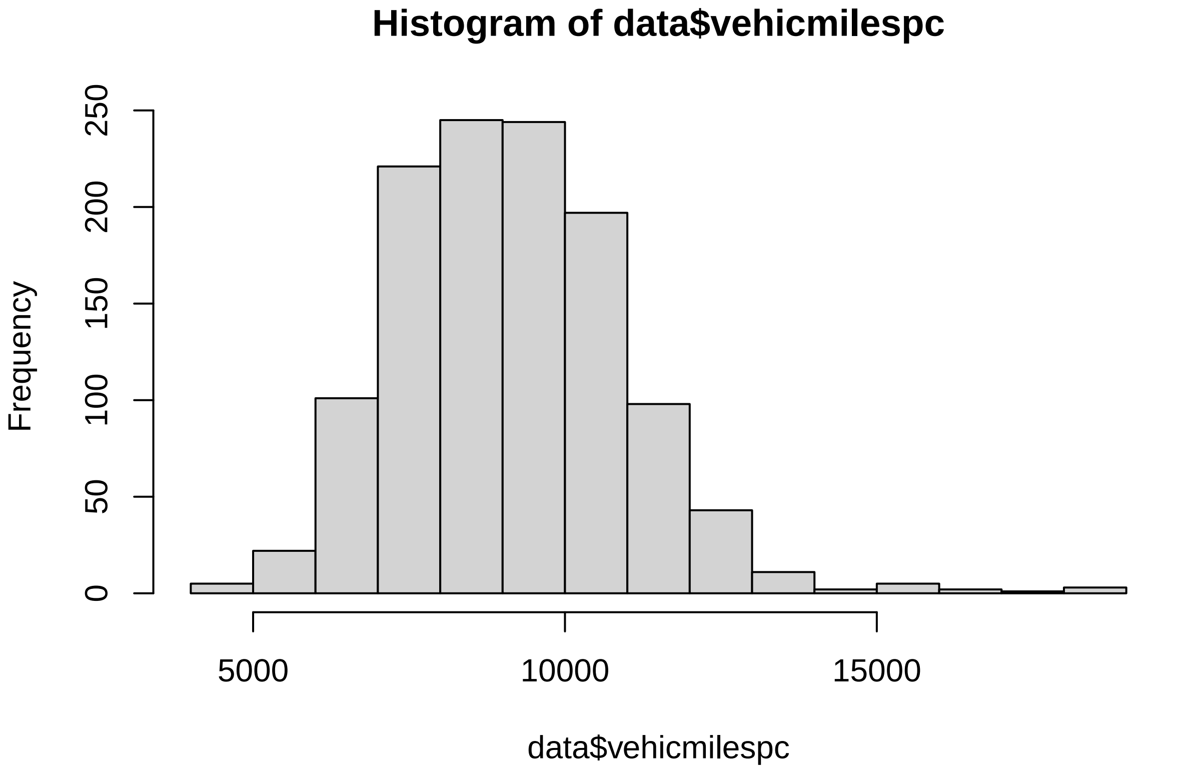
```
hist((log(data$perc14_24)))
```



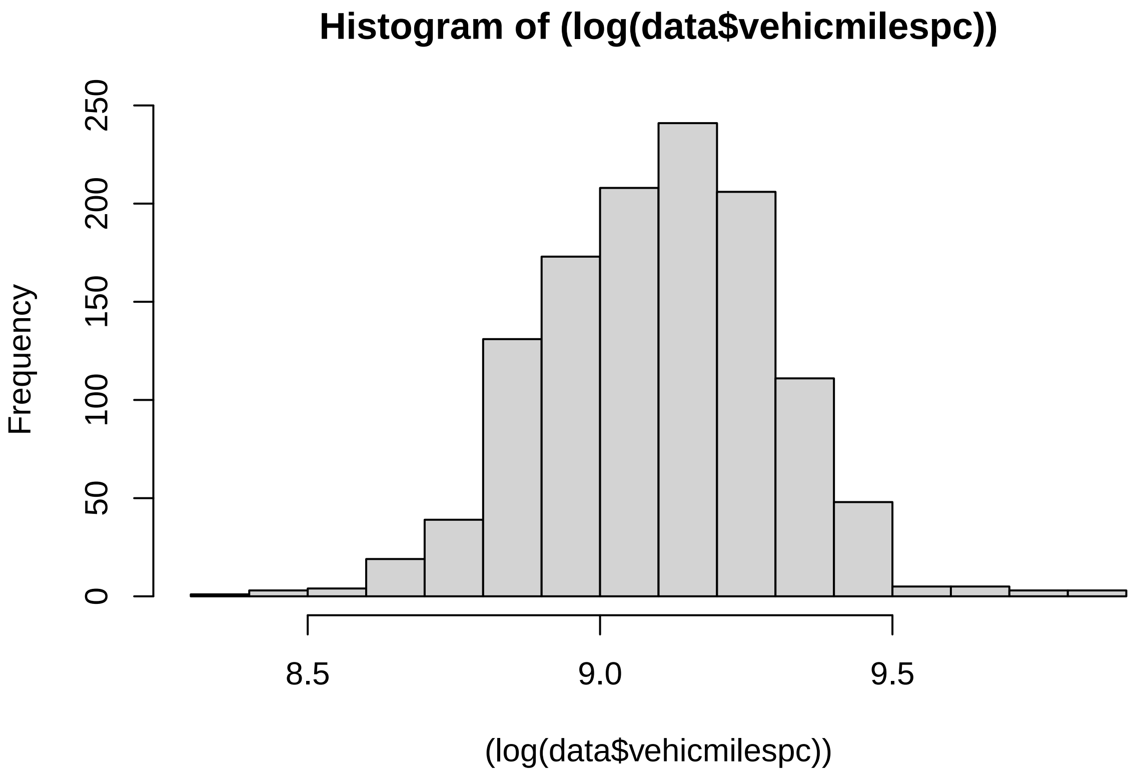
```
hist(data$vehicmiles)
```



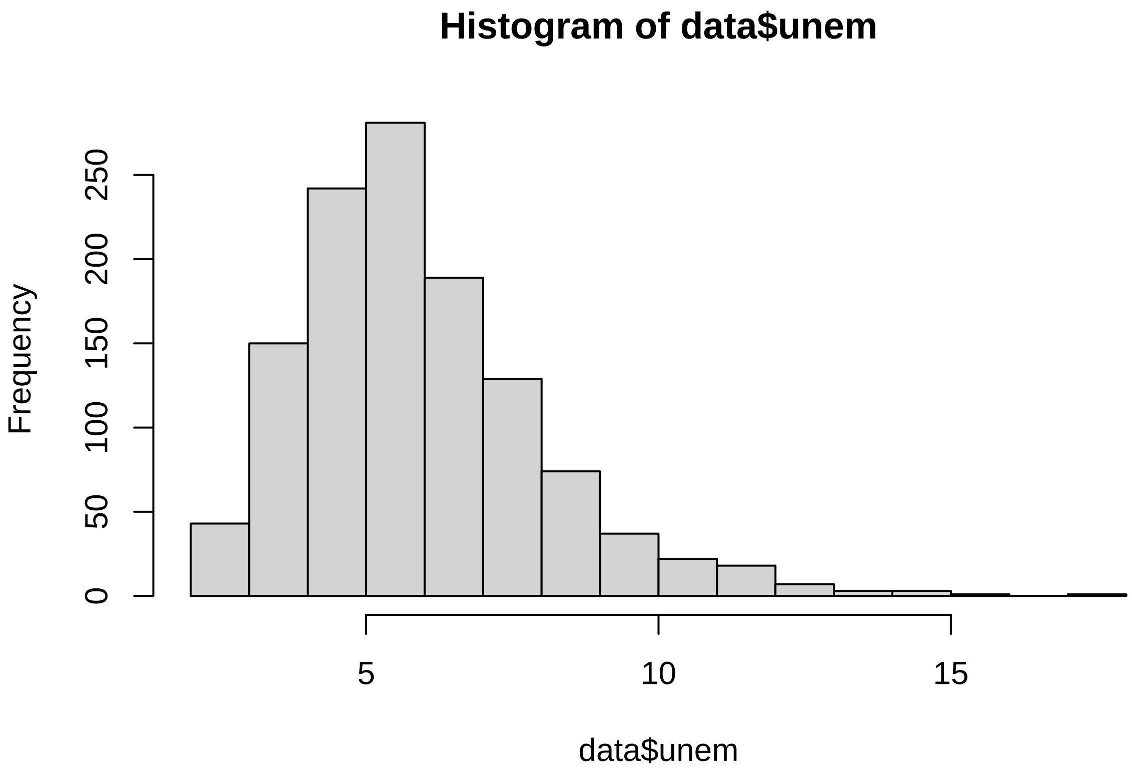
```
hist(data$vehicmilespc)
```



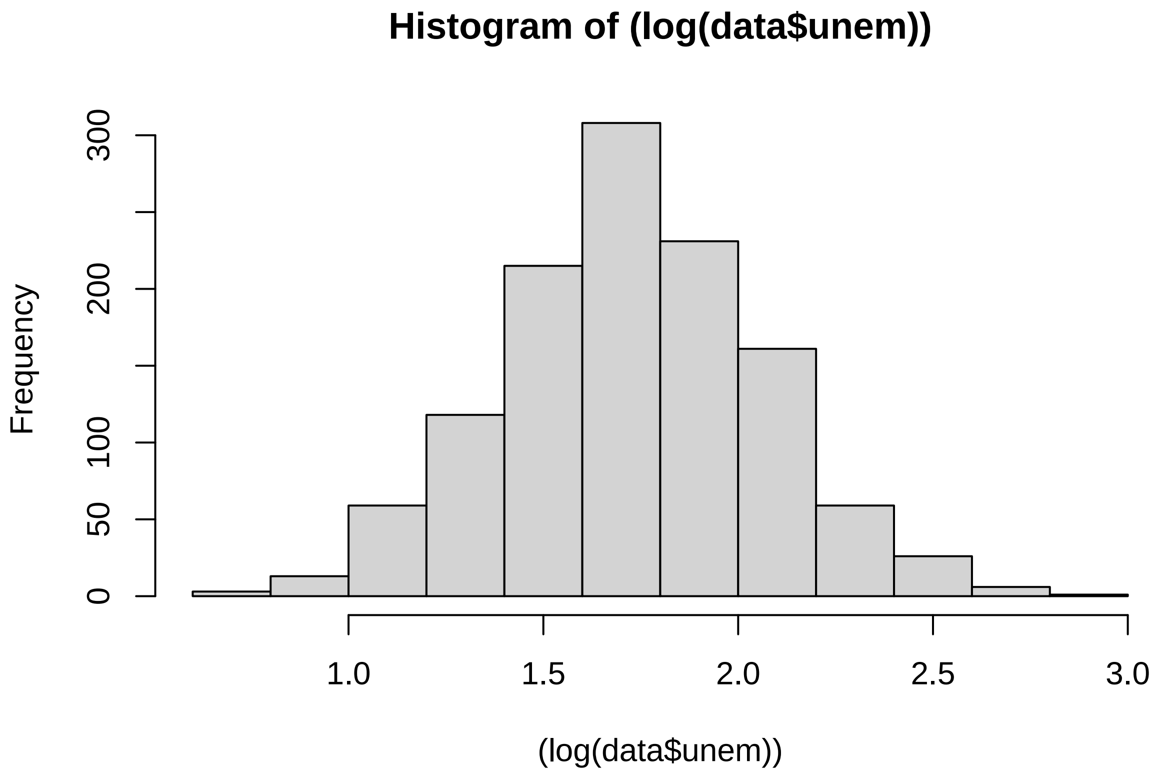
```
hist((log(data$vehicmilespc)))
```



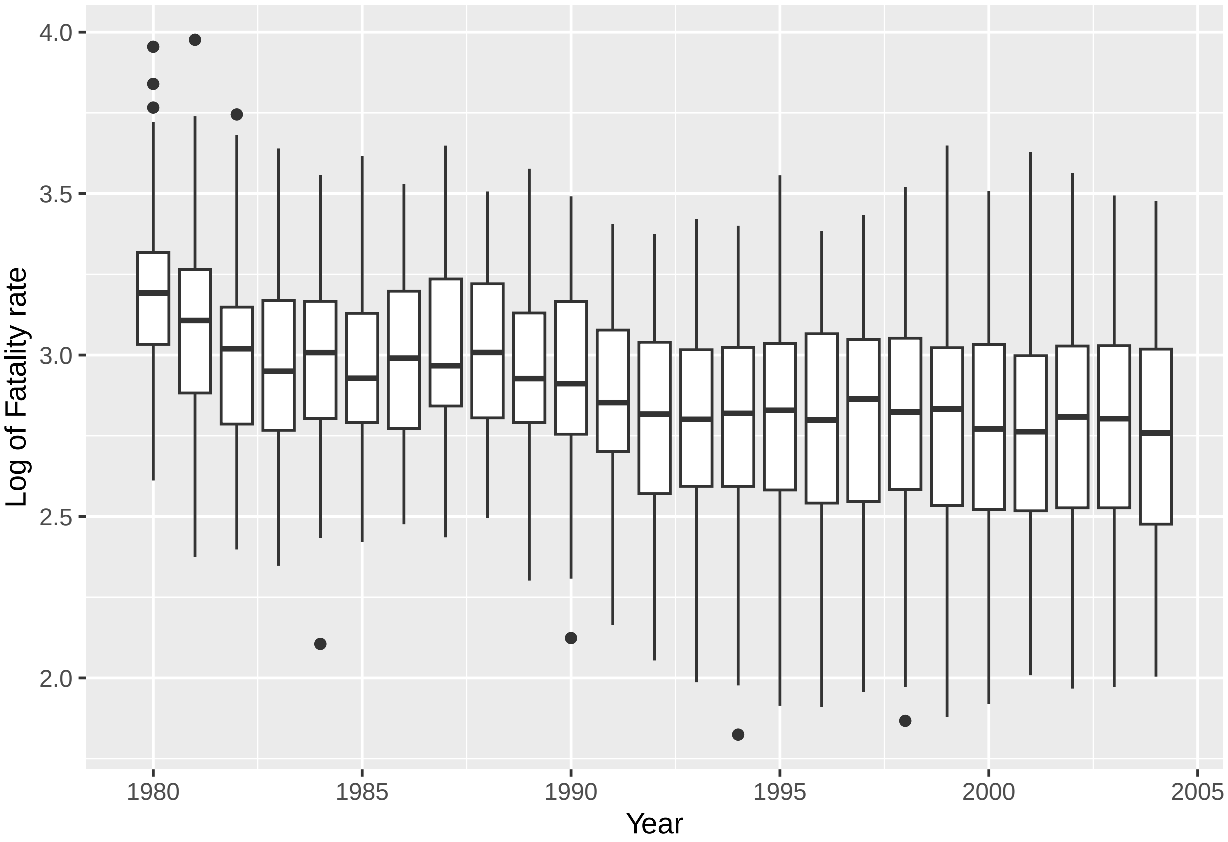
```
hist(data$unem)
```



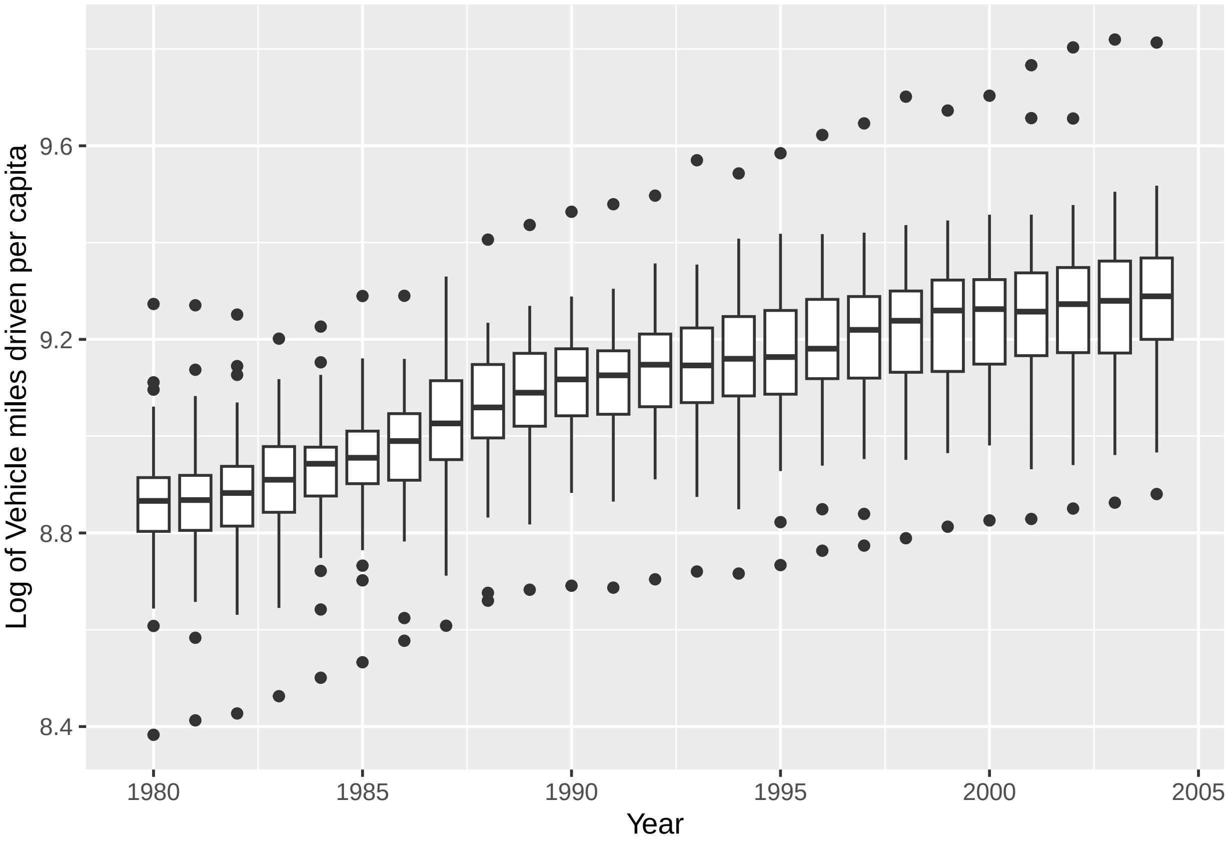
```r
hist((log(data$unem)))
```



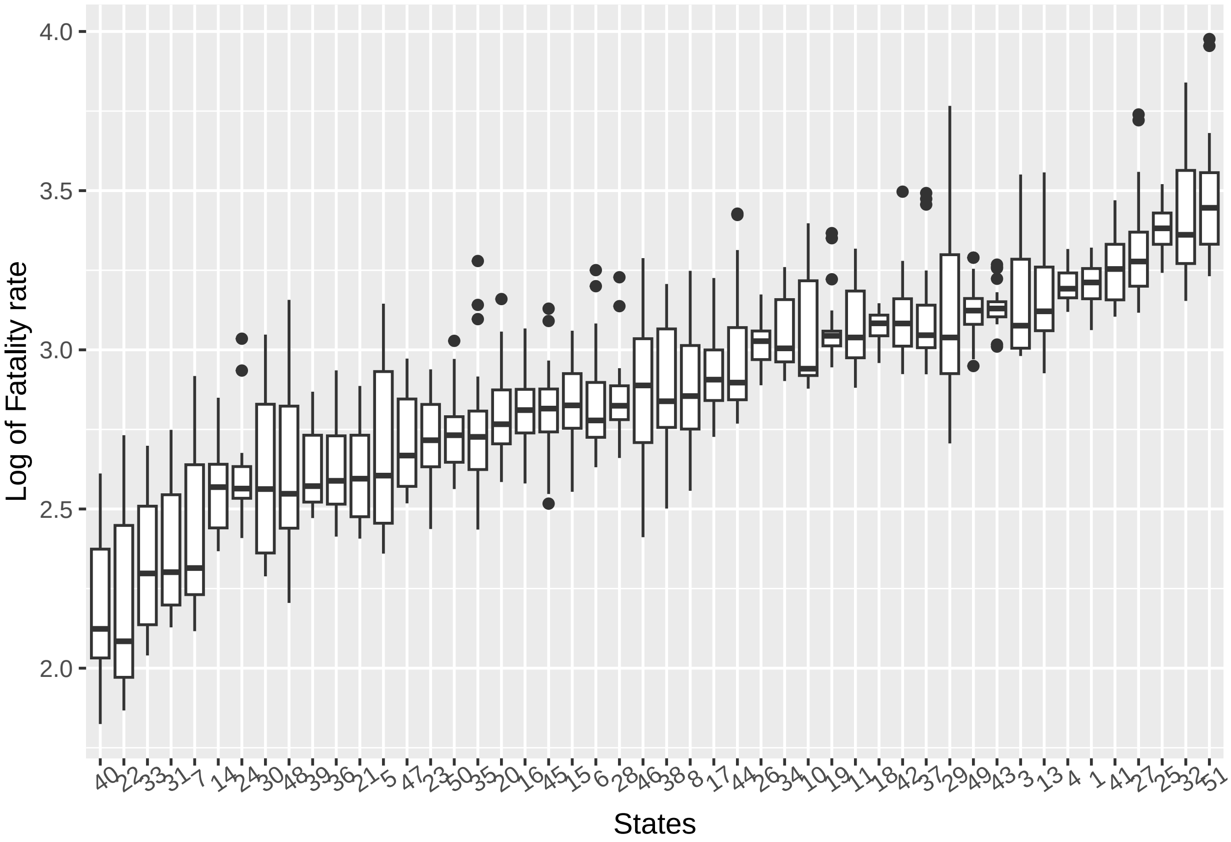
```r
data %>%
  group_by(year) %>%
  ggplot(aes(x = year, y = log(total_fatalities_rate), group = year)) +
  geom_boxplot() +
  labs(x = "Year",  y = "Log of Fatality rate")
```



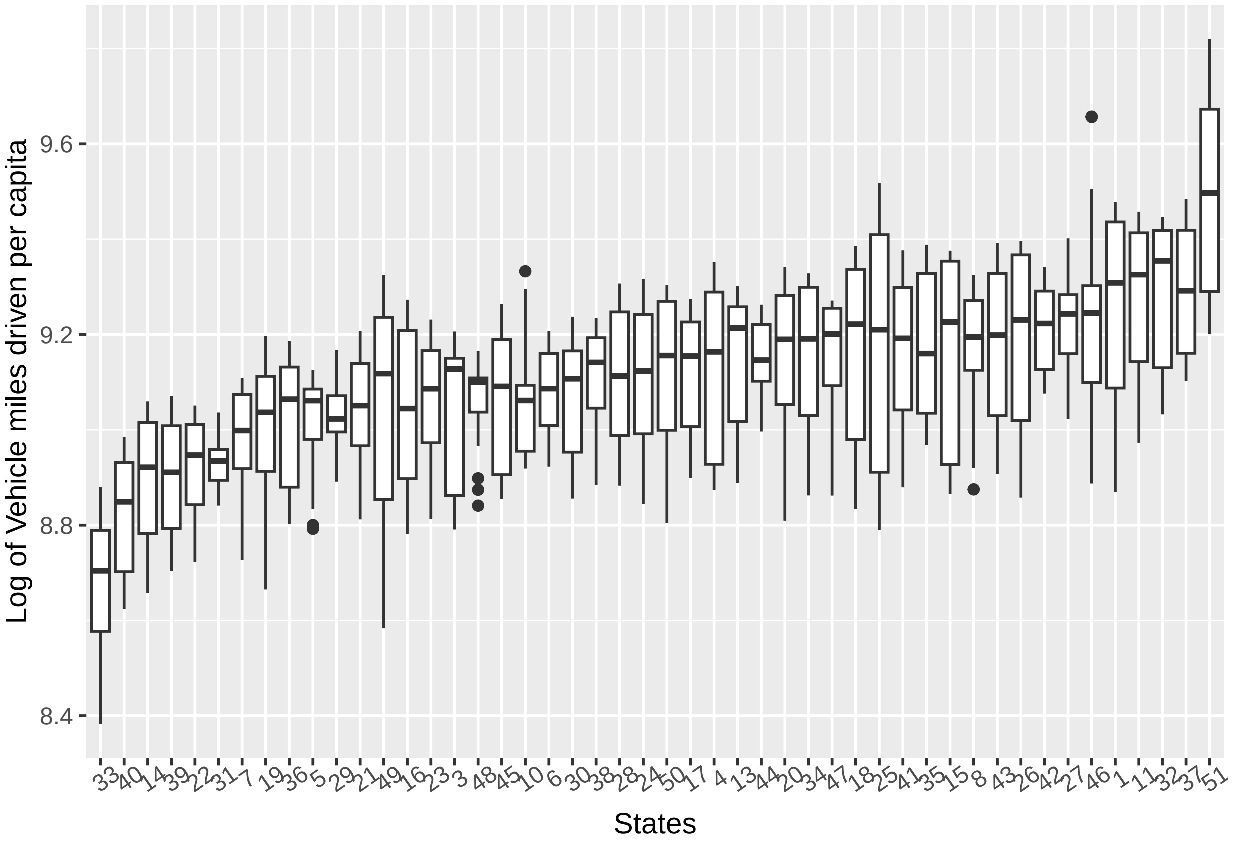
```
data %>%
  group_by(year) %>%
  ggplot(aes(x = year, y = log(data$vehicmilespc), group = year)) +
  geom_boxplot() +
  labs(x = "Year",  y = "Log of Vehicle miles driven per capita")
```



```
data %>%
  data.frame() %>%
  ggplot(aes(x = reorder(state,log(total_fatalities_rate)), y = log(total_fatalities_rate), group=state
  geom_boxplot() +
  labs(x = "States",  y = "Log of Fatality rate") + theme(axis.text.x = element_text(angle=35))+ theme(
```



```r
data %>%
  group_by(state) %>%
  ggplot(aes(x = reorder(state,log(vehicmilespc)), y = log(vehicmilespc))) +
  geom_boxplot() +
  labs(x = "States",  y = "Log of Vehicle miles driven per capita") + theme(axis.text.x = element_text(
```



3 (15 points) Preliminary Model

Estimate a linear regression model of *totfatrte* on a set of dummy variables for the years 1981 through 2004 and interpret what you observe. In this section, you should address the following tasks:

- Why is fitting a linear model a sensible starting place?
- What does this model explain, and what do you find in this model?

Answer A linear model makes sense as a starting place because it is an easy model to implement and interpret.

The preliminary model is structured such that every year is represented by a dummy variable with 1980 as the baseline. A negative coefficient in this case represents a reduction in the mean rate of traffic fatalities for that year compared to the mean traffic fatalities rate in 1980. All of the coefficients are negative, meaning that there were less fatalities per 100,000 on average for that year compared to 1980.

In 1981, the reduction in traffic fatalities rate was 1.8 per 100,000 on average. From 1982 to 1990, the mean difference per year stayed between approximately 4.5 and 6, decreasing by more in the late 80s. In 1991 the mean difference in traffic fatalities rate since 1980 was -7.4 per 100,000. The difference stays mostly level, dropping a little throughout the rest of the data until 2004, when the mean difference was -8.8 per 100,000.

- Did driving become safer over this period? Please provide a detailed explanation. *Answer* The increasingly negative coefficients for each of the dummy variables shows that over time the trend of the total traffic fatalities is decreasing. This indicates that driving is becoming safer over the period
- What, if any, are the limitation of this model. In answering this, please consider **at least**:
  - Are the parameter estimates reliable, unbiased estimates of the truth? Or, are they biased due to the way that the data is structured?

  - Are the uncertainty estimate reliable, unbiased estimates of sampling based variability? Or, are they biased due to the way that the data is structured? *Answer* The linear model makes a couple of assumptions of the data. The first assumption is that all of the observations are independent from one another. This cannot be the case for this data there are repeated measures of individual states, which creates correlation. OLS regression does not account for this and therefore the parameter estimates will not be reliable. Another common feature of panel data is heteroskedasticity in the errors, which violates the second OLS assumption that the errors are independent and normally distributed.

```r
data$year_test_fac <- factor(data$year_test)
linear_mod <- lm(log(total_fatalities_rate)~year_test_fac, data = data)
summary(linear_mod)
```

```
##
## Call:
## lm(formula = log(total_fatalities_rate) ~ year_test_fac, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.96324 -0.22134  0.01005  0.23221  0.86830
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.19577    0.04697  68.035  < 2e-16 ***
## year_test_fac1981 -0.07878    0.06643  -1.186 0.235904
## year_test_fac1982 -0.19957    0.06643  -3.004 0.002719 **
## year_test_fac1983 -0.23523    0.06643  -3.541 0.000414 ***
## year_test_fac1984 -0.22585    0.06643  -3.400 0.000697 ***
## year_test_fac1985 -0.24301    0.06643  -3.658 0.000265 ***
## year_test_fac1986 -0.19681    0.06643  -2.963 0.003111 **
## year_test_fac1987 -0.19871    0.06643  -2.991 0.002836 **
## year_test_fac1988 -0.18885    0.06643  -2.843 0.004547 **
## year_test_fac1989 -0.24815    0.06643  -3.735 0.000196 ***
## year_test_fac1990 -0.26785    0.06643  -4.032 5.89e-05 ***
## year_test_fac1991 -0.34372    0.06643  -5.174 2.69e-07 ***
## year_test_fac1992 -0.40229    0.06643  -6.056 1.88e-09 ***
## year_test_fac1993 -0.40257    0.06643  -6.060 1.83e-09 ***
## year_test_fac1994 -0.40798    0.06643  -6.142 1.12e-09 ***
## year_test_fac1995 -0.38492    0.06643  -5.794 8.79e-09 ***
## year_test_fac1996 -0.39949    0.06643  -6.014 2.42e-09 ***
## year_test_fac1997 -0.38596    0.06643  -5.810 8.03e-09 ***
## year_test_fac1998 -0.40954    0.06643  -6.165 9.67e-10 ***
## year_test_fac1999 -0.41450    0.06643  -6.240 6.11e-10 ***
## year_test_fac2000 -0.43694    0.06643  -6.578 7.18e-11 ***
## year_test_fac2001 -0.43521    0.06643  -6.552 8.50e-11 ***
## year_test_fac2002 -0.42672    0.06643  -6.424 1.93e-10 ***
## year_test_fac2003 -0.43978    0.06643  -6.620 5.44e-11 ***
## year_test_fac2004 -0.44853    0.06643  -6.752 2.29e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3254 on 1175 degrees of freedom
## Multiple R-squared:  0.126,  Adjusted R-squared:  0.1081
## F-statistic: 7.057 on 24 and 1175 DF,  p-value: < 2.2e-16
```

4 (15 points) Expanded Model

Expand the **Preliminary Model** by adding variables related to the following concepts:

- Blood alcohol levels
- Per se laws
- Primary seat belt laws (Note that if a law was enacted sometime within a year the fraction of the year is recorded in place of the zero-one indicator.)
- Secondary seat belt laws
- Speed limits faster than 70
- Graduated drivers licenses
- Percent of the population between 14 and 24 years old
- Unemployment rate
- Vehicle miles driven per capita.

If it is appropriate, include transformations of these variables. Please carefully explain carefully your rationale, which should be based on your EDA, behind any transformation you made. If no transformation is made, explain why transformation is not needed.

Answer We performed log transformations on the fatalities rate, unemployment, and vehicle miles driven per capita variables to normalize them after seeing that the distributions of the variables were skewed.

- How are the blood alcohol variables defined? Interpret the coefficients that you estimate for this concept.

Answer The blood alcohol variable is a categorical variable defined with 3 levels. The most restrictive BAC law is set at 0.08, the next level describes states who prohibit driving with a BAC of 0.10, the next level describes states with no BAC restrictions.

The coefficient of $BAC_{0.10}$ is 0.04, this means that mean total fatality rate increases by 0.04 when a state has a law prohibiting driving with a BAC level of 0.10 compared to 0.08.

The coefficient of $BAC_{0.08}$ is 0.06, which means that the total fatality rate increases by 0.06 per 100,000 on average, when a state has no BAC restrictions compared to a restriction of 0.08.

- Do *per se laws* have a negative effect on the fatality rate?

Answer The coefficient representing if a state has per se laws has a coefficient of -0.02. The model indicates that having per se laws does have a negative effect on the fatality rate, however, this coefficient is not significant, so we can not conclude that the effect on mean fatality rate related to having per se laws is significantly different than 0.

- Does having a primary seat belt law?

Answer The dummy variable representing a state having a primary seatbelt law has a coefficient of 0.0009. The model indicates that having per se laws has a positive effect on the fatality rate, however, this coefficient is not significant, so we can not conclude that the effect on mean fatality rate related to having per se laws is significantly different than 0 from this model.

```
expanded_mod <- lm(log(total_fatalities_rate)~year_test_fac + factor(bac) + factor(sl70plus) + factor(p
summary(expanded_mod)
```

```
##
## Call:
## lm(formula = log(total_fatalities_rate) ~ year_test_fac + factor(bac) +
##     factor(sl70plus) + factor(perse_binary) + factor(sbprim_binary) +
##     factor(sbsec_binary) + factor(gdl_binary) + perc14_24 + log(unem) +
##     log(vehicmilespc), data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.58513 -0.12787 -0.00044  0.14047  0.62367
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -1.135e+01  4.017e-01 -28.267  < 2e-16 ***
## year_test_fac1981      -9.192e-02  4.121e-02  -2.231  0.02589 *
## year_test_fac1982      -2.956e-01  4.206e-02  -7.028 3.57e-12 ***
## year_test_fac1983      -3.520e-01  4.269e-02  -8.247 4.35e-16 ***
## year_test_fac1984      -2.997e-01  4.346e-02  -6.896 8.77e-12 ***
## year_test_fac1985      -3.371e-01  4.443e-02  -7.586 6.75e-14 ***
## year_test_fac1986      -3.133e-01  4.640e-02  -6.752 2.30e-11 ***
## year_test_fac1987      -3.492e-01  4.839e-02  -7.216 9.60e-13 ***
## year_test_fac1988      -3.602e-01  5.094e-02  -7.071 2.64e-12 ***
## year_test_fac1989      -4.451e-01  5.291e-02  -8.412  < 2e-16 ***
## year_test_fac1990      -5.053e-01  5.407e-02  -9.345  < 2e-16 ***
## year_test_fac1991      -6.207e-01  5.517e-02 -11.252  < 2e-16 ***
## year_test_fac1992      -7.266e-01  5.627e-02 -12.914  < 2e-16 ***
## year_test_fac1993      -7.185e-01  5.694e-02 -12.617  < 2e-16 ***
## year_test_fac1994      -7.055e-01  5.802e-02 -12.161  < 2e-16 ***
## year_test_fac1995      -6.818e-01  5.950e-02 -11.459  < 2e-16 ***
## year_test_fac1996      -8.080e-01  6.161e-02 -13.114  < 2e-16 ***
## year_test_fac1997      -8.170e-01  6.266e-02 -13.038  < 2e-16 ***
## year_test_fac1998      -8.654e-01  6.381e-02 -13.562  < 2e-16 ***
## year_test_fac1999      -8.671e-01  6.459e-02 -13.424  < 2e-16 ***
## year_test_fac2000      -8.794e-01  6.568e-02 -13.388  < 2e-16 ***
## year_test_fac2001      -9.349e-01  6.610e-02 -14.144  < 2e-16 ***
## year_test_fac2002      -9.794e-01  6.653e-02 -14.721  < 2e-16 ***
## year_test_fac2003      -1.003e+00  6.679e-02 -15.012  < 2e-16 ***
## year_test_fac2004      -9.839e-01  6.853e-02 -14.357  < 2e-16 ***
## factor(bac)0.1          4.494e-02  1.846e-02   2.435  0.01504 *
## factor(bac)None         6.190e-02  2.433e-02   2.545  0.01107 *
## factor(sl70plus)1       2.219e-01  2.162e-02  10.262  < 2e-16 ***
## factor(perse_binary)1  -1.882e-02  1.464e-02  -1.286  0.19869
## factor(sbprim_binary)1  9.419e-04  2.456e-02   0.038  0.96942
## factor(sbsec_binary)1   2.043e-02  2.144e-02   0.953  0.34084
## factor(gdl_binary)1    -2.129e-02  2.529e-02  -0.842  0.40000
## perc14_24               1.779e-02  6.111e-03   2.911  0.00367 **
## log(unem)               2.673e-01  2.414e-02  11.071  < 2e-16 ***
## log(vehicmilespc)       1.541e+00  4.432e-02  34.765  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2015 on 1165 degrees of freedom
## Multiple R-squared:  0.6678, Adjusted R-squared:  0.6581
## F-statistic: 68.87 on 34 and 1165 DF,  p-value: < 2.2e-16
```

5 (15 points) State-Level Fixed Effects

Re-estimate the **Expanded Model** using fixed effects at the state level.

- What do you estimate for coefficients on the blood alcohol variables? How do the coefficients on the blood alcohol variables change, if at all?
- What do you estimate for coefficients on per se laws? How do the coefficients on per se laws change, if at all?

- What do you estimate for coefficients on primary seat-belt laws? How do the coefficients on primary seatbelt laws change, if at all?

Answer The coefficient for bac = 0.1 variable in this fixed effects model is now 0.0042323,which is a smaller coefficient compared to the expanded model which has a positive coefficient for bac laws (4.494e-02). What is also notable is that this bac law was a significant explanatory variable in the expanded model, and in the fixed effects model it is not significant to explaining fatality rates.

In the fixed effects model, the coefficient for perse laws is -0.0534343,compared to -1.882e-02 in the expanded model. The direction is still negative, implying that the addition of per se laws decrease fatality rates. In the expanded model this coefficient wasn't significant, but in the fixed effects model it is highly significant.

The coefficient for the primary seatbelt law in this model is -0.0404917, and for the secondary seat belt law it is 0.0056980. The coefficient for the primary seatbelt law is small and positive for the expanded model , but negative in the fixed effect model. The coefficients for the secondary seat belt law are both small and positive in the fixed effects model and the expanded model. These coefficients are not significant in the expanded model, but in the fixed effects model the primary seatbelt law coefficient is significant. This implies that the primary seatbelt law has a negative effect on fatality rates.

The fixed effects model is likely more reliable since it is able to avoid omitted variable bias in the model. We can see from the difference in some coefficients that the expanded model was underestimating the effects of some of the variables. One clear example is how only the fixed effects model recognized the magnitude and significance of the per se laws coefficient.

We also ran a pFtest that rejected the null hypothesis (p-value = 2.2e-16) which proves that there are individual effects that are better handled with the fixed effects model. We further tested and made sure to use the within model instead of first differencing to estimate our fixed effects model.

Which set of estimates do you think is more reliable? Why do you think this?

- What assumptions are needed in each of these models?

- Are these assumptions reasonable in the current context?

Answer

The assumptions for the expanded model would be the OLS regression assumptions in order to get statistically correct estimates. Because of unobserved effects, repeated observations, and time-constant explanatory variables, this model is likely statistically inaccurate.

The assumptions for the fixed effect model are linearity (model is linear in parameters), strict exogeneity (zero conditional means), observations are IID across entities, and no perfect collinearity. The second assumption is necessary to avoid omitted variable bias. Using the within model for the fixed effects means we are removing the unobserved individual heterogeneity by using the time-demeaned model. This allows the equation to meet the assumptions for the fixed effect model since we are averaging by state, making the observations independent and removing the fixed effects coefficients.

```
within_model<- plm(log(total_fatalities_rate)~year_test_fac + factor(bac) + factor(sl70plus) + factor(pe
                   data = data,
                   index = c("state", "year_test_fac"),
                   effect = "individual", model = "within")

summary(within_model)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = log(total_fatalities_rate) ~ year_test_fac + factor(bac) +
##     factor(sl70plus) + factor(perse_binary) + factor(sbprim_binary) +
##     factor(sbsec_binary) + factor(gdl_binary) + perc14_24 + log(unem) +
##     log(vehicmilespc), data = data, effect = "individual", model = "within",
##     index = c("state", "year_test_fac"))
##
## Balanced Panel: n = 48, T = 25, N = 1200
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.3809194 -0.0517516  0.0041331  0.0535470  0.2912838
##
## Coefficients:
##                          Estimate Std. Error  t-value  Pr(>|t|)
## year_test_fac1981      -0.0629920  0.0180829  -3.4835  0.000514 ***
## year_test_fac1982      -0.1368192  0.0190106  -7.1970 1.126e-12 ***
## year_test_fac1983      -0.1721909  0.0196020  -8.7844 < 2.2e-16 ***
## year_test_fac1984      -0.2071493  0.0203281 -10.1903 < 2.2e-16 ***
## year_test_fac1985      -0.2307212  0.0213486 -10.8073 < 2.2e-16 ***
## year_test_fac1986      -0.1923439  0.0229726  -8.3727 < 2.2e-16 ***
## year_test_fac1987      -0.2380879  0.0249768  -9.5323 < 2.2e-16 ***
## year_test_fac1988      -0.2696855  0.0273541  -9.8591 < 2.2e-16 ***
## year_test_fac1989      -0.3426135  0.0292016 -11.7327 < 2.2e-16 ***
## year_test_fac1990      -0.3535455  0.0303472 -11.6500 < 2.2e-16 ***
## year_test_fac1991      -0.3895487  0.0310366 -12.5513 < 2.2e-16 ***
## year_test_fac1992      -0.4505470  0.0321027 -14.0345 < 2.2e-16 ***
## year_test_fac1993      -0.4677952  0.0327017 -14.3049 < 2.2e-16 ***
## year_test_fac1994      -0.5033180  0.0335512 -15.0015 < 2.2e-16 ***
## year_test_fac1995      -0.5000318  0.0346860 -14.4159 < 2.2e-16 ***
## year_test_fac1996      -0.5531910  0.0366731 -15.0844 < 2.2e-16 ***
## year_test_fac1997      -0.5754564  0.0376269 -15.2938 < 2.2e-16 ***
## year_test_fac1998      -0.6300575  0.0385917 -16.3262 < 2.2e-16 ***
## year_test_fac1999      -0.6497906  0.0390355 -16.6461 < 2.2e-16 ***
## year_test_fac2000      -0.6827377  0.0396002 -17.2408 < 2.2e-16 ***
## year_test_fac2001      -0.6538599  0.0396768 -16.4797 < 2.2e-16 ***
## year_test_fac2002      -0.6158438  0.0399246 -15.4252 < 2.2e-16 ***
## year_test_fac2003      -0.6188054  0.0401019 -15.4308 < 2.2e-16 ***
## year_test_fac2004      -0.6553754  0.0412336 -15.8942 < 2.2e-16 ***
## factor(bac)0.1          0.0042323  0.0106432   0.3977  0.690962
## factor(bac)None         0.0204314  0.0144299   1.4159  0.157082
## factor(sl70plus)1       0.0720100  0.0113558   6.3412 3.302e-10 ***
## factor(perse_binary)1  -0.0534343  0.0097673  -5.4707 5.529e-08 ***
## factor(sbprim_binary)1 -0.0404917  0.0150058  -2.6984  0.007072 **
## factor(sbsec_binary)1   0.0056980  0.0110141   0.5173  0.605019
## factor(gdl_binary)1    -0.0154092  0.0122313  -1.2598  0.207998
## perc14_24               0.0202582  0.0041644   4.8646 1.311e-06 ***
## log(unem)              -0.1926117  0.0171865 -11.2072 < 2.2e-16 ***
## log(vehicmilespc)       0.6784007  0.0507510  13.3672 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    31.924
## Residual Sum of Squares: 8.6994
## R-Squared:      0.7275
## Adj. R-Squared: 0.70776
## F-statistic: 87.7863 on 34 and 1118 DF, p-value: < 2.22e-16
```

```r
pFtest(within_model, expanded_mod)
```

```
##
##  F test for individual effects
##
## data:  log(total_fatalities_rate) ~ year_test_fac + factor(bac) + factor(sl70plus) +  ...
## F = 105.56, df1 = 47, df2 = 1118, p-value < 2.2e-16
## alternative hypothesis: significant effects
```

```r
fd_model<- plm(log(total_fatalities_rate)~year_test_fac + factor(bac) + factor(sl70plus) + factor(perse_
                  data = data,
                  index = c("state", "year_test_fac"),
                  effect = "individual", model = "fd")

summary(fd_model)
```

```
## Oneway (individual) effect First-Difference Model
##
## Call:
## plm(formula = log(total_fatalities_rate) ~ year_test_fac + factor(bac) +
##     factor(sl70plus) + factor(perse_binary) + factor(sbprim_binary) +
##     factor(sbsec_binary) + factor(gdl_binary) + perc14_24 + log(unem) +
##     log(vehicmilespc), data = data, effect = "individual", model = "fd",
##     index = c("state", "year_test_fac"))
##
## Balanced Panel: n = 48, T = 25, N = 1200
## Observations used in estimation: 1152
##
## Residuals:
##        Min.     1st Qu.      Median     3rd Qu.        Max.
## -0.35620582 -0.04808329 -0.00089455  0.05049240  0.35957740
##
## Coefficients: (1 dropped because of singularities)
##                          Estimate Std. Error t-value  Pr(>|t|)
## (Intercept)            -0.0132866  0.0044907 -2.9587 0.0031544 **
## year_test_fac1981      -0.0467548  0.0129269 -3.6169 0.0003115 ***
## year_test_fac1982      -0.1132115  0.0199652 -5.6704 1.812e-08 ***
## year_test_fac1983      -0.1116046  0.0246051 -4.5358 6.357e-06 ***
## year_test_fac1984      -0.0853257  0.0280132 -3.0459 0.0023741 **
## year_test_fac1985      -0.0738606  0.0315085 -2.3442 0.0192448 *
## year_test_fac1986       0.0016734  0.0346728  0.0483 0.9615160
## year_test_fac1987       0.0174542  0.0385674  0.4526 0.6509503
## year_test_fac1988       0.0446360  0.0428004  1.0429 0.2972258
## year_test_fac1989       0.0123142  0.0462716  0.2661 0.7901888
## year_test_fac1990       0.0230577  0.0478566  0.4818 0.6300372
## year_test_fac1991      -0.0131418  0.0484646 -0.2712 0.7863154
## year_test_fac1992      -0.0485953  0.0482967 -1.0062 0.3145446
## year_test_fac1993      -0.0399791  0.0471415 -0.8481 0.3965832
## year_test_fac1994      -0.0420582  0.0456660 -0.9210 0.3572506
## year_test_fac1995      -0.0101937  0.0442762 -0.2302 0.8179556
## year_test_fac1996      -0.0170723  0.0426792 -0.4000 0.6892217
## year_test_fac1997      -0.0071239  0.0399174 -0.1785 0.8583900
## year_test_fac1998      -0.0311151  0.0366644 -0.8486 0.3962604
## year_test_fac1999      -0.0346024  0.0327970 -1.0550 0.2916315
## year_test_fac2000      -0.0509967  0.0295083 -1.7282 0.0842258 .
## year_test_fac2001      -0.0258596  0.0246084 -1.0508 0.2935573
## year_test_fac2002       0.0064812  0.0194245  0.3337 0.7386970
## year_test_fac2003       0.0083404  0.0137526  0.6065 0.5443337
## factor(bac)0.1         -0.0083167  0.0152431 -0.5456 0.5854482
## factor(bac)None         0.0337329  0.0158818  2.1240 0.0338893 *
## factor(sl70plus)1       0.0199198  0.0201110  0.9905 0.3221480
## factor(perse_binary)1  -0.0123776  0.0153211 -0.8079 0.4193328
## factor(sbprim_binary)1 -0.0132541  0.0232290 -0.5706 0.5683966
## factor(sbsec_binary)1  -0.0118338  0.0142402 -0.8310 0.4061431
## factor(gdl_binary)1     0.0129765  0.0152284  0.8521 0.3943279
## perc14_24               0.0402364  0.0149674  2.6883 0.0072890 **
## log(unem)              -0.0911956  0.0233147 -3.9115 9.725e-05 ***
## log(vehicmilespc)       0.1143020  0.0942753  1.2124 0.2256048
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    10.684
## Residual Sum of Squares: 8.7927
## R-Squared:      0.17704
## Adj. R-Squared: 0.15275
## F-statistic: 7.28826 on 33 and 1118 DF, p-value: < 2.22e-16
```

```r
pwfdtest(fd_model, data = data, h0="fe")
```

```
##
##  Wooldridge's first-difference test for serial correlation in panels
##
## data:  fd_model
## F = 6.4253, df1 = 1, df2 = 1102, p-value = 0.01139
## alternative hypothesis: serial correlation in original errors
```

```r
pwfdtest(fd_model, data = data, h0="fd")
```

```
##
##  Wooldridge's first-difference test for serial correlation in panels
##
## data:  fd_model
## F = 133.32, df1 = 1, df2 = 1102, p-value < 2.2e-16
## alternative hypothesis: serial correlation in differenced errors
```

6 (10 points) Consider a Random Effects Model

Instead of estimating a fixed effects model, should you have estimated a random effects model?

- Please state the assumptions of a random effects model, and evaluate whether these assumptions are met in the data.

Answer

The random effects model requires a strong assumption of independence between random effects and other predictors. We use this model when we think the unobserved effect is uncorrelated with all the explanatory variables. For these assumptions to be met, there can't be a correlation between the state effects and the other explanatory variables. However, it is apparent that each state and their own fatality rates are correlated with the passing of laws related to driving safety. Therefore, the assumptions for a random effects model aren't met.

- If the assumptions are, in fact, met in the data, then estimate a random effects model and interpret the coefficients of this model. Comment on how, if at all, the estimates from this model have changed compared to the fixed effects model.

Answer

The assumptions are not met.

- If the assumptions are **not** met, then do not estimate the data. But, also comment on what the consequences would be if you were to *inappropriately* estimate a random effects model. Would your coefficient estimates be biased or not? Would your standard error estimates be biased or not? Or, would there be some other problem that might arise?

Answer

If we inappropriately estimate a random effects model, we risk biased coefficient estimates due to omitted variable bias. There would also be correlation in the errors. The random effects model is a more efficient estimator, meaning that if the assumptions were held true, that the standard errors of the betas would be less than the fixed effect models. However, the assumption is not true, so the standard error estimates would be biased.

```r
head(data)
```

```
##   year state sl55 sl65 sl70 sl75 slnone seatbelt minage zerotol gdl bac10 bac08
## 1 1980     1    1    0    0    0      0        0     18       0   0     1     0
## 2 1981     1    1    0    0    0      0        0     18       0   0     1     0
## 3 1982     1    1    0    0    0      0        0     18       0   0     1     0
## 4 1983     1    1    0    0    0      0        0     18       0   0     1     0
## 5 1984     1    1    0    0    0      0        0     18       0   0     1     0
## 6 1985     1    1    0    0    0      0        0     20       0   0     1     0
##   perse totfat nghtfat wkndfat totfatpvm nghtfatpvm wkndfatpvm statepop
## 1     0    940     422     236      3.20      1.437      0.803  3893888
## 2     0    933     434     248      3.35      1.558      0.890  3918520
## 3     0    839     376     224      2.81      1.259      0.750  3925218
## 4     0    930     397     223      3.00      1.281      0.719  3934109
## 5     0    932     421     237      2.83      1.278      0.720  3951834
## 6     0    882     358     224      2.51      1.019      0.637  3972527
##   vehicmiles unem perc14_24 sl70plus sbprim sbsecon d80 d81 d82 d83 d84 d85 d86
## 1   29.37500  8.8      18.9        0      0       0   1   0   0   0   0   0   0
## 2   27.85200 10.7      18.7        0      0       0   0   1   0   0   0   0   0
## 3   29.85765 14.4      18.4        0      0       0   0   0   1   0   0   0   0
## 4   31.00000 13.7      18.0        0      0       0   0   0   0   1   0   0   0
## 5   32.93286 11.1      17.6        0      0       0   0   0   0   0   1   0   0
## 6   35.13944  8.9      17.3        0      0       0   0   0   0   0   0   1   0
##   d87 d88 d89 d90 d91 d92 d93 d94 d95 d96 d97 d98 d99 d00 d01 d02 d03 d04
## 1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 3   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 4   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 5   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##   vehicmilespc speed_limit perse_binary gdl_binary sbprim_binary sbsec_binary
## 1     7543.874          55            0          0             0            0
## 2     7107.785          55            0          0             0            0
## 3     7606.622          55            0          0             0            0
## 4     7879.802          55            0          0             0            0
## 5     8333.562          55            0          0             0            0
## 6     8845.614          55            0          0             0            0
##   year_test bac total_fatalities_rate night_fatalities_rate
## 1      1980 0.1                 24.14                 10.84
## 2      1981 0.1                 24.07                 11.08
## 3      1982 0.1                 21.37                  9.58
## 4      1983 0.1                 23.64                 10.09
## 5      1984 0.1                 23.58                 10.65
## 6      1985 0.1                 22.20                  9.01
##   weekend_fatalities_rate year_test_fac
## 1                    6.06          1980
## 2                    6.33          1981
## 3                    5.71          1982
## 4                    5.67          1983
## 5                    6.00          1984
## 6                    5.64          1985
```

```
re.model <- plm(log(total_fatalities_rate)~year_test_fac + factor(bac) + factor(sl70plus) + factor(pers
                   data = data,
                index = c("state", "year_test_fac"), model = "random", random.method = "walhus")
summary(re.model)
```

```
## Oneway (individual) effect Random Effect Model
##    (Wallace-Hussain's transformation)
##
## Call:
## plm(formula = log(total_fatalities_rate) ~ year_test_fac + factor(bac) +
##     factor(sl70plus) + factor(perse_binary) + factor(sbprim_binary) +
##     factor(sbsec_binary) + factor(gdl_binary) + perc14_24 + log(unem) +
##     log(vehicmilespc), data = data, model = "random", random.method = "walhus",
##     index = c("state", "year_test_fac"))
##
## Balanced Panel: n = 48, T = 25, N = 1200
##
## Effects:
##                     var std.dev share
## idiosyncratic 0.01519 0.12324 0.385
## individual    0.02423 0.15566 0.615
## theta: 0.8436
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -0.4360358 -0.0573289  0.0015671  0.0599271  0.3014582
##
## Coefficients:
##                          Estimate Std. Error  z-value  Pr(>|z|)
## (Intercept)            -4.2799793  0.4560824  -9.3842 < 2.2e-16 ***
## year_test_fac1981      -0.0651733  0.0193817  -3.3626 0.0007721 ***
## year_test_fac1982      -0.1500847  0.0203324  -7.3815 1.565e-13 ***
## year_test_fac1983      -0.1883667  0.0209371  -8.9968 < 2.2e-16 ***
## year_test_fac1984      -0.2182580  0.0216850 -10.0649 < 2.2e-16 ***
## year_test_fac1985      -0.2443051  0.0227288 -10.7487 < 2.2e-16 ***
## year_test_fac1986      -0.2086666  0.0244166  -8.5461 < 2.2e-16 ***
## year_test_fac1987      -0.2570963  0.0264610  -9.7161 < 2.2e-16 ***
## year_test_fac1988      -0.2901498  0.0289033 -10.0386 < 2.2e-16 ***
## year_test_fac1989      -0.3659355  0.0308017 -11.8804 < 2.2e-16 ***
## year_test_fac1990      -0.3821921  0.0319611 -11.9580 < 2.2e-16 ***
## year_test_fac1991      -0.4241620  0.0326829 -12.9781 < 2.2e-16 ***
## year_test_fac1992      -0.4904816  0.0337546 -14.5308 < 2.2e-16 ***
## year_test_fac1993      -0.5065945  0.0343720 -14.7386 < 2.2e-16 ***
## year_test_fac1994      -0.5395915  0.0352566 -15.3047 < 2.2e-16 ***
## year_test_fac1995      -0.5361566  0.0364310 -14.7171 < 2.2e-16 ***
## year_test_fac1996      -0.5944130  0.0384841 -15.4457 < 2.2e-16 ***
## year_test_fac1997      -0.6168837  0.0394579 -15.6340 < 2.2e-16 ***
## year_test_fac1998      -0.6724879  0.0404411 -16.6288 < 2.2e-16 ***
## year_test_fac1999      -0.6920054  0.0408994 -16.9197 < 2.2e-16 ***
## year_test_fac2000      -0.7239697  0.0415005 -17.4449 < 2.2e-16 ***
## year_test_fac2001      -0.7029789  0.0415414 -16.9224 < 2.2e-16 ***
## year_test_fac2002      -0.6725819  0.0417593 -16.1062 < 2.2e-16 ***
## year_test_fac2003      -0.6775339  0.0419356 -16.1565 < 2.2e-16 ***
## year_test_fac2004      -0.7111372  0.0431337 -16.4868 < 2.2e-16 ***
## factor(bac)0.1          0.0058259  0.0113023   0.5155 0.6062286
## factor(bac)None         0.0240214  0.0153073   1.5693 0.1165840
## factor(sl70plus)1       0.0791120  0.0120967   6.5400 6.154e-11 ***
## factor(perse_binary)1  -0.0496071  0.0102842  -4.8236 1.410e-06 ***
## factor(sbprim_binary)1 -0.0367245  0.0158682  -2.3144 0.0206484 *
## factor(sbsec_binary)1   0.0067372  0.0117436   0.5737 0.5661765
## factor(gdl_binary)1    -0.0150714  0.0130729  -1.1529 0.2489613
## perc14_24               0.0209761  0.0043857   4.7829 1.728e-06 ***
## log(unem)              -0.1568234  0.0180193  -8.7031 < 2.2e-16 ***
## log(vehicmilespc)       0.8305639  0.0503909  16.4824 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    34.626
## Residual Sum of Squares: 10.417
## R-Squared:      0.69915
## Adj. R-Squared: 0.69037
## Chisq: 2707.37 on 34 DF, p-value: < 2.22e-16
```

```r
phtest(within_model, re.model)
```

```
##
##  Hausman Test
##
## data:  log(total_fatalities_rate) ~ year_test_fac + factor(bac) + factor(sl70plus) +  ...
## chisq = 392.57, df = 34, p-value < 2.2e-16
## alternative hypothesis: one model is inconsistent
```

```r
stargazer(linear_mod, expanded_mod, within_model, fd_model,
style="qje", type="text", omit.stat=c("adj.rsq","f"),
column.labels = c("Prelim Model","Expanded Model","Within","FD"))
```

```
##
## =============================================================================
##                                      log(total_fatalities_rate)
##                                        OLS                        panel
##                                                                  linear
##                          Prelim Model     Expanded Model    Within      FD
##                               (1)               (2)           (3)       (4)
## -----------------------------------------------------------------------------
## year_test_fac1981           -0.079           -0.092**      -0.063*** -0.047***
##                             (0.066)           (0.041)       (0.018)   (0.013)
##
## year_test_fac1982          -0.200***         -0.296***     -0.137*** -0.113***
##                             (0.066)           (0.042)       (0.019)   (0.020)
##
## year_test_fac1983          -0.235***         -0.352***     -0.172*** -0.112***
##                             (0.066)           (0.043)       (0.020)   (0.025)
##
## year_test_fac1984          -0.226***         -0.300***     -0.207*** -0.085***
##                             (0.066)           (0.043)       (0.020)   (0.028)
##
## year_test_fac1985          -0.243***         -0.337***     -0.231*** -0.074**
##                             (0.066)           (0.044)       (0.021)   (0.032)
##
## year_test_fac1986          -0.197***         -0.313***     -0.192***   0.002
##                             (0.066)           (0.046)       (0.023)   (0.035)
##
## year_test_fac1987          -0.199***         -0.349***     -0.238***   0.017
##                             (0.066)           (0.048)       (0.025)   (0.039)
##
## year_test_fac1988          -0.189***         -0.360***     -0.270***   0.045
##                             (0.066)           (0.051)       (0.027)   (0.043)
##
## year_test_fac1989          -0.248***         -0.445***     -0.343***   0.012
##                             (0.066)           (0.053)       (0.029)   (0.046)
##
## year_test_fac1990          -0.268***         -0.505***     -0.354***   0.023
##                             (0.066)           (0.054)       (0.030)   (0.048)
##
## year_test_fac1991          -0.344***         -0.621***     -0.390***  -0.013
##                             (0.066)           (0.055)       (0.031)   (0.048)
##
## year_test_fac1992          -0.402***         -0.727***     -0.451***  -0.049
##                             (0.066)           (0.056)       (0.032)   (0.048)
##
## year_test_fac1993          -0.403***         -0.718***     -0.468***  -0.040
##                             (0.066)           (0.057)       (0.033)   (0.047)
##
## year_test_fac1994          -0.408***         -0.706***     -0.503***  -0.042
##                             (0.066)           (0.058)       (0.034)   (0.046)
##
## year_test_fac1995          -0.385***         -0.682***     -0.500***  -0.010
##                             (0.066)           (0.059)       (0.035)   (0.044)
##
## year_test_fac1996          -0.399***         -0.808***     -0.553***  -0.017
##                             (0.066)           (0.062)       (0.037)   (0.043)
##
## year_test_fac1997          -0.386***         -0.817***     -0.575***  -0.007
##                             (0.066)           (0.063)       (0.038)   (0.040)
##
## year_test_fac1998          -0.410***         -0.865***     -0.630***  -0.031
##                             (0.066)           (0.064)       (0.039)   (0.037)
##
```

```
## year_test_fac1999          -0.414***         -0.867***     -0.650***  -0.035
##                             (0.066)           (0.065)       (0.039)   (0.033)
##
## year_test_fac2000          -0.437***         -0.879***     -0.683***  -0.051*
##                             (0.066)           (0.066)       (0.040)   (0.030)
##
## year_test_fac2001          -0.435***         -0.935***     -0.654***  -0.026
##                             (0.066)           (0.066)       (0.040)   (0.025)
##
## year_test_fac2002          -0.427***         -0.979***     -0.616***   0.006
##                             (0.066)           (0.067)       (0.040)   (0.019)
##
## year_test_fac2003          -0.440***         -1.003***     -0.619***   0.008
##                             (0.066)           (0.067)       (0.040)   (0.014)
##
## year_test_fac2004          -0.449***         -0.984***     -0.655***
##                             (0.066)           (0.069)       (0.041)
##
## factor(bac)0.1                                0.045**        0.004    -0.008
##                                               (0.018)       (0.011)   (0.015)
##
## factor(bac)None                               0.062**        0.020    0.034**
##                                               (0.024)       (0.014)   (0.016)
##
## factor(sl70plus)1                            0.222***      0.072***    0.020
##                                               (0.022)       (0.011)   (0.020)
##
## factor(perse_binary)1                         -0.019       -0.053***  -0.012
##                                               (0.015)       (0.010)   (0.015)
##
## factor(sbprim_binary)1                         0.001       -0.040***  -0.013
##                                               (0.025)       (0.015)   (0.023)
##
## factor(sbsec_binary)1                          0.020         0.006    -0.012
##                                               (0.021)       (0.011)   (0.014)
##
## factor(gdl_binary)1                           -0.021        -0.015     0.013
##                                               (0.025)       (0.012)   (0.015)
##
## perc14_24                                    0.018***      0.020***  0.040***
##                                               (0.006)       (0.004)   (0.015)
##
## log(unem)                                    0.267***      -0.193*** -0.091***
##                                               (0.024)       (0.017)   (0.023)
##
## log(vehicmilespc)                            1.541***      0.678***    0.114
##                                               (0.044)       (0.051)   (0.094)
##
## Constant                   3.196***         -11.355***               -0.013***
##                             (0.047)           (0.402)                 (0.004)
##
## N                             1,200             1,200         1,200     1,152
## R2                            0.126             0.668         0.727     0.177
## Residual Std. Error    0.325 (df = 1175) 0.202 (df = 1165)
## ===============================================================================
## Notes:                                  ***Significant at the 1 percent level.
##                                          **Significant at the 5 percent level.
##                                          *Significant at the 10 percent level.
```

7 (10 points) Model Forecasts

The COVID-19 pandemic dramatically changed patterns of driving. Find data (and include this data in your analysis, here) that includes some measure of vehicle miles driven in the US. Your data should at least cover the period from January 2018 to as current as possible. With this data, produce the following statements:

- Comparing monthly miles driven in 2018 to the same months during the pandemic:
  - What month demonstrated the largest decrease in driving? How much, in percentage terms, lower was this driving?
  - What month demonstrated the largest increase in driving? How much, in percentage terms, higher was this driving?

```r
miles_data <- read.csv('miles_driven.csv')
head(miles_data)
```

```
##         DATE M12MTVUSM227NFWA
## 1 1970-12-01          1120328
## 2 1971-01-01          1125491
## 3 1971-02-01          1128168
## 4 1971-03-01          1130920
## 5 1971-04-01          1139069
## 6 1971-05-01          1144752
```

```r
miles_data$miles <- miles_data$M12MTVUSM227NFWA
miles_data <- na.omit(miles_data)
miles_data$DATE <- as.Date(miles_data$DATE)
```

```r
# Extracting data for 2018 and pandemic period

# get population to convert to per capita
# miles are measured in millions, multiple by 1 million
#take log to reflect the "within" model vehicle miles per capita input

USA_pop <- 333000000

miles_data$miles_pc_log <- log((miles_data$miles*1000000)/USA_pop)

miles_2018 <- subset(miles_data, year(DATE) == 2018)

miles_pandemic <- subset(miles_data, year(DATE) >= 2020)

# Finding corresponding months in 2018 for each month in the pandemic period
corresponding_months_2018 <- as.Date(paste("2018", format(miles_pandemic$DATE, "%m"), "01", sep = "-"))

# Extracting miles driven for corresponding months in 2018
miles_2018_corresponding <- miles_2018[miles_2018$DATE %in% corresponding_months_2018, ]
```

```r
# Merging monthly miles data for 2018 and pandemic period
# Extract month from DATE column
miles_2018_corresponding$month <- format(miles_2018_corresponding$DATE, "%m")
miles_pandemic$month <- format(miles_pandemic$DATE, "%m")
```

```
# Merge by month
merged_miles <- merge(miles_2018_corresponding, miles_pandemic, by = "month", suffixes = c("_2018", "_pa

head(merged_miles)
```

```
##   month  DATE_2018 M12MTVUSM227NFWA_2018 miles_2018 miles_pc_log_2018
## 1    01 2018-01-01               3214482    3214482          9.175034
## 2    01 2018-01-01               3214482    3214482          9.175034
## 3    01 2018-01-01               3214482    3214482          9.175034
## 4    01 2018-01-01               3214482    3214482          9.175034
## 5    01 2018-01-01               3214482    3214482          9.175034
## 6    02 2018-02-01               3216597    3216597          9.175692
##   DATE_pandemic M12MTVUSM227NFWA_pandemic miles_pandemic miles_pc_log_pandemic
## 1    2020-01-01                   3273691        3273691              9.193286
## 2    2024-01-01                   3261680        3261680              9.189610
## 3    2023-01-01                   3209343        3209343              9.173434
## 4    2022-01-01                   3151184        3151184              9.155146
## 5    2021-01-01                   2867733        2867733              9.060890
## 6    2020-02-01                   3284595        3284595              9.196611
```

```
# Calculating percentage change in monthly miles driven
merged_miles$percentage_change <- ((merged_miles$miles_pc_log_pandemic - merged_miles$miles_pc_log_2018)

# Month with largest decrease in driving during the pandemic
max_decrease_month_pandemic <- merged_miles[which.min(merged_miles$percentage_change), "DATE_pandemic"]

# Month with largest increase in driving during the pandemic
max_increase_month_pandemic <- merged_miles[which.max(merged_miles$percentage_change), "DATE_pandemic"]

# Output results

cat("decrease:", format(max_decrease_month_pandemic, "%B %Y"), "\n")
```

```
## decrease: February 2021
```

```
cat("Percentage decrease:", round(min(merged_miles$percentage_change), 2), "%\n\n")
```

```
## Percentage decrease: -1.39 %
```

```
cat("increase:", format(max_increase_month_pandemic, "%B %Y"), "\n")
```

```
## increase: February 2020
```

```
cat("Percentage increase:", round(max(merged_miles$percentage_change), 2), "%\n\n")
```

```
## Percentage increase: 0.23 %
```

Answer

Month with largest decrease in driving during the pandemic: February 2021 Percentage decrease: -1.39 %

Month with largest increase in driving during the pandemic: February 2020 Percentage increase: 0.23 %

Now, use these changes in driving to make forecasts from your models.

- Suppose that the number of miles driven per capita, increased by as much as the COVID boom. Using the FE estimates, what would the consequences be on the number of traffic fatalities? Please interpret the estimate.

- Suppose that the number of miles driven per capita, decreased by as much as the COVID bust. Using the FE estimates, what would the consequences be on the number of traffic fatalities? Please interpret the estimate.

```
# Print the summary of the fixed effects model
summary(within_model)
```

```
## Oneway (individual) effect Within Model
## 
## Call:
## plm(formula = log(total_fatalities_rate) ~ year_test_fac + factor(bac) + 
##     factor(sl70plus) + factor(perse_binary) + factor(sbprim_binary) + 
##     factor(sbsec_binary) + factor(gdl_binary) + perc14_24 + log(unem) + 
##     log(vehicmilespc), data = data, effect = "individual", model = "within", 
##     index = c("state", "year_test_fac"))
## 
## Balanced Panel: n = 48, T = 25, N = 1200
## 
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max. 
## -0.3809194 -0.0517516  0.0041331  0.0535470  0.2912838 
## 
## Coefficients:
##                          Estimate Std. Error  t-value  Pr(>|t|)    
## year_test_fac1981      -0.0629920  0.0180829  -3.4835  0.000514 ***
## year_test_fac1982      -0.1368192  0.0190106  -7.1970 1.126e-12 ***
## year_test_fac1983      -0.1721909  0.0196020  -8.7844 < 2.2e-16 ***
## year_test_fac1984      -0.2071493  0.0203281 -10.1903 < 2.2e-16 ***
## year_test_fac1985      -0.2307212  0.0213486 -10.8073 < 2.2e-16 ***
## year_test_fac1986      -0.1923439  0.0229726  -8.3727 < 2.2e-16 ***
## year_test_fac1987      -0.2380879  0.0249768  -9.5323 < 2.2e-16 ***
## year_test_fac1988      -0.2696855  0.0273541  -9.8591 < 2.2e-16 ***
## year_test_fac1989      -0.3426135  0.0292016 -11.7327 < 2.2e-16 ***
## year_test_fac1990      -0.3535455  0.0303472 -11.6500 < 2.2e-16 ***
## year_test_fac1991      -0.3895487  0.0310366 -12.5513 < 2.2e-16 ***
## year_test_fac1992      -0.4505470  0.0321027 -14.0345 < 2.2e-16 ***
## year_test_fac1993      -0.4677952  0.0327017 -14.3049 < 2.2e-16 ***
## year_test_fac1994      -0.5033180  0.0335512 -15.0015 < 2.2e-16 ***
## year_test_fac1995      -0.5000318  0.0346860 -14.4159 < 2.2e-16 ***
## year_test_fac1996      -0.5531910  0.0366731 -15.0844 < 2.2e-16 ***
## year_test_fac1997      -0.5754564  0.0376269 -15.2938 < 2.2e-16 ***
## year_test_fac1998      -0.6300575  0.0385917 -16.3262 < 2.2e-16 ***
## year_test_fac1999      -0.6497906  0.0390355 -16.6461 < 2.2e-16 ***
## year_test_fac2000      -0.6827377  0.0396002 -17.2408 < 2.2e-16 ***
## year_test_fac2001      -0.6538599  0.0396768 -16.4797 < 2.2e-16 ***
## year_test_fac2002      -0.6158438  0.0399246 -15.4252 < 2.2e-16 ***
## year_test_fac2003      -0.6188054  0.0401019 -15.4308 < 2.2e-16 ***
## year_test_fac2004      -0.6553754  0.0412336 -15.8942 < 2.2e-16 ***
## factor(bac)0.1          0.0042323  0.0106432   0.3977  0.690962    
## factor(bac)None         0.0204314  0.0144299   1.4159  0.157082    
## factor(sl70plus)1       0.0720100  0.0113558   6.3412 3.302e-10 ***
## factor(perse_binary)1  -0.0534343  0.0097673  -5.4707 5.529e-08 ***
## factor(sbprim_binary)1 -0.0404917  0.0150058  -2.6984  0.007072 ** 
## factor(sbsec_binary)1   0.0056980  0.0110141   0.5173  0.605019    
## factor(gdl_binary)1    -0.0154092  0.0122313  -1.2598  0.207998    
## perc14_24               0.0202582  0.0041644   4.8646 1.311e-06 ***
## log(unem)              -0.1926117  0.0171865 -11.2072 < 2.2e-16 ***
## log(vehicmilespc)       0.6784007  0.0507510  13.3672 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    31.924
## Residual Sum of Squares: 8.6994
## R-Squared:      0.7275
## Adj. R-Squared: 0.70776
## F-statistic: 87.7863 on 34 and 1118 DF, p-value: < 2.22e-16
```

```r
# Get the Fixed Effects estimate
fe_estimate <- coef(within_model)["log(vehicmilespc)"]

# Maximum increase and decrease in miles driven during the pandemic
max_increase_percentage <- max(merged_miles$percentage_change)

max_decrease_percentage <- min(merged_miles$percentage_change)

# Calculate the estimated increase in traffic fatalities during the COVID boom
fatalities_increase_boom <- fe_estimate * (max_increase_percentage/ 100)

# Interpret the estimate for the COVID boom
cat("fatalities, COVID boom:", (exp(fatalities_increase_boom) - 1)*100, "%", "\n")
```

```
## fatalities, COVID boom: 0.1547862 %
```

```r
# Calculate the estimated decrease in traffic fatalities during the COVID bust
fatalities_decrease_bust <- fe_estimate * (max_decrease_percentage/100)

# Interpret the estimate for the COVID bust

cat("fatalities, COVID bust:", (exp(fatalities_decrease_bust) - 1)*100, "%", "\n")
```

```
## fatalities, COVID bust: -0.9365583 %
```

Answer

We multiplied the FE estimate by the percent increase in driving during the covid boom to find the impact on traffic fatalities. Estimated increase in traffic fatalities during the COVID boom: 0.1547862 %

We multiplied the FE estimate by the percent increase in driving during the covid bust to find the impact on traffic fatalities. " Estimated decrease in traffic fatalities during the COVID bust: -0.9365583 %

8 (5 points) Evaluate Error

If there were serial correlation or heteroskedasticity in the idiosyncratic errors of the model, what would be the consequences on the estimators and their standard errors? Is there any serial correlation or heteroskedasticity?

Answer We have evidence to suggest that there is heteroscedasticity present in the errors of our fixed effects model after conducting the Breusch-Pagan test on the Fixed Effects model and getting a p-value of 1.795e-06. This means that the variance of the errors in our model is not constant across all levels of the independent variables, which may affect the reliability of our analysis. Heteroskedasticity occurs when the variance of the errors in predicting traffic fatality rates varies across different conditions, violating the assumption of homoscedasticity. This means that the coefficient estimates may not be as precise as they could be if the errors were homoskedastic. Additionally, the standard errors of the coefficient estimates can be biased.

Underestimated standard errors can lead to an increased likelihood of Type I errors, while overestimated standard errors can result in a higher likelihood of Type II errors.

```r
library(zoo)
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library (lmtest)
# Breusch-Pagan Test for Heteroskedasticity
bptest(within_model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  within_model
## BP = 86.572, df = 34, p-value = 1.795e-06
```

Answer We have evidence to suggest that there is serial correlation in idiosyncratic errors after conducting the Breusch-Godfrey/Wooldridge test for serial correlation in panel models and getting a p-value of 2.2e-16. Serial correlation means that the errors in predicting traffic fatality rate are not independent across observations. Positive serial correlation can cause coefficient estimates to be less variable than expected, potentially leading to an underestimation of the true effects of variables such as laws and driving behavior on traffic fatalities. Conversely, negative serial correlation can result in coefficient estimates that are more variable, potentially leading to overestimation of these effects. When serial correlation is present, standard errors tend to be underestimated. This means that the confidence intervals around our coefficient estimates may be too narrow, leading to an increased risk of Type I errors, where we mistakenly conclude that there is a significant effect when there is not.

```r
pbgtest(within_model)
```

```
##
##  Breusch-Godfrey/Wooldridge test for serial correlation in panel models
##
## data:  log(total_fatalities_rate) ~ year_test_fac + factor(bac) + factor(sl70plus) +  ...
## chisq = 243.23, df = 25, p-value < 2.2e-16
## alternative hypothesis: serial correlation in idiosyncratic errors
```