W271: Lab 4

Tiffany Jaya, Joanna Huang, Robert Deng, Shan He

August 10, 2018

```
# Load libraries
library(car)
library(dplyr)
library(Hmisc)
library(ggplot2)
library(grid)
library(gridExtra)
library(knitr)
library(plm)
# prevent source code from running off the page
opts_chunk$set(tidy.opts=list(width.cutoff=70), tidy=TRUE)
# remove all objects from current workspace
rm(list = ls())
# set seed number to reproduce results
set.seed(1)
# load data
load("data/driving.RData")
```

Description of the Lab

In this lab, you are asked to answer the question **"Do changes in data laws affect data fatalities?"** To do so, you will conduct the tasks specified below using the data set *driving.Rdata*, which 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 come with the dataset.

Exercises:

1. Load the data. Provide a description of the basic structure of the dataset, as we have done in throughout the semester. Conduct a very thorough EDA, which should include both graphical and tabular techniques, on the dataset, including both the dependent variable *totfatrte* and the potential explanatory variables. You need to write a detailed narrative of your observations of your EDA. *Reminder: giving an "output dump" (i.e. providing a bunch of graphs and tables without description and hoping your audience will interpret them) will receive zero point in this exercise.*

```
table(with(data, state, year))
```

```
##
##  1  3  4  5  6  7  8 10 11 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
## 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25
## 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51
## 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25 25
```

```r
str(data)
```

```
## 'data.frame':    1200 obs. of  56 variables:
##  $ year        : int  1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 ...
##  $ state       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ sl55        : num  1 1 1 1 1 ...
##  $ sl65        : num  0 0 0 0 0 ...
##  $ sl70        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ sl75        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ slnone      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ seatbelt    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ minage      : num  18 18 18 18 18 20 21 21 21 21 ...
##  $ zerotol     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ gdl         : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ bac10       : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ bac08       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ perse       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ totfat      : int  940 933 839 930 932 882 1080 1111 1024 1029 ...
##  $ nghtfat     : int  422 434 376 397 421 358 500 499 423 418 ...
##  $ wkndfat     : int  236 248 224 223 237 224 279 300 226 247 ...
##  $ totfatpvm   : num  3.2 3.35 2.81 3 2.83 ...
##  $ nghtfatpvm  : num  1.44 1.56 1.26 1.28 1.28 ...
##  $ wkndfatpvm  : num  0.803 0.89 0.75 0.719 0.72 ...
##  $ statepop    : int  3893888 3918520 3925218 3934109 3951834 3972527 3991569 4015261 4023858 403022
##  $ totfatrte   : num  24.1 24.1 21.4 23.6 23.6 ...
##  $ nghtfatrte  : num  10.84 11.08 9.58 10.09 10.65 ...
##  $ wkndfatrte  : num  6.06 6.33 5.71 5.67 6 ...
##  $ vehicmiles  : num  29.4 27.9 29.9 31 32.9 ...
##  $ unem        : num  8.8 10.7 14.4 13.7 11.1 ...
##  $ perc14_24   : num  18.9 18.7 18.4 18 17.6 ...
##  $ sl70plus    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ sbprim      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ sbsecon     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d80         : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ d81         : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ d82         : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ d83         : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ d84         : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ d85         : int  0 0 0 0 0 1 0 0 0 0 ...
##  $ d86         : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ d87         : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ d88         : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ d89         : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ d90         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d91         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d92         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d93         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d94         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d95         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d96         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d97         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d98         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d99         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d00         : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ d01         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d02         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d03         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ d04         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ vehicmilespc: num  7544 7108 7607 7880 8334 ...
##  - attr(*, "datalabel")= chr ""
##  - attr(*, "time.stamp")= chr "22 Jan 2013 14:09"
##  - attr(*, "formats")= chr  "%8.0g" "%8.0g" "%9.0g" "%9.0g" ...
##  - attr(*, "types")= int  252 251 254 254 254 254 254 251 254 254 ...
##  - attr(*, "val.labels")= chr  "" "" "" "" ...
##  - attr(*, "var.labels")= chr  "1980 through 2004" "48 continental states, alphabetical" "speed limi
##  - attr(*, "version")= int 12
```

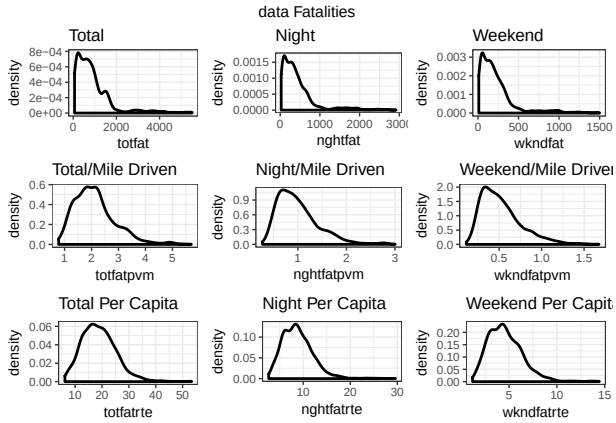
This dataset contains 1200 observations of 56 variables that spans across 25 years from 1980 to 2004. There is a year column, as well as 25 dummy variables corresponding to each year. The dataset contains data for 48 continental U.S. states and each state is numbered from 1 to 51 with 2, 9 and 12 missing. Remaining columns hold variables related to state's data laws, data fatalities, and population demographics. Below we list the variables, their definitions and their values:

data Laws - "sl*" variables correspond to the speed limit mandated by the state in a given year. For example, "sl55" relates to whether or not a state had a 55 mph speed limit. Values range from 0 to 1, with fractions representing the amount of time for which the limit was enforced by law that year given a change in state law. - "seatbelt" consists of values 0, 1 and 2 with 0 being no seatbelt law, 1 being primary (no other violation required to give a ticket), and 2 being secondary (additional violated required for a ticket). Dummy variables, "sbprim" and "sbsecon", also signify the the same thing.

Alcohol-Related data Laws - "minage" is the minimum drinking age, with values from 18-21 and the majority being 21 years old. - "zerotol" indiciates whether or not a state enacted a Zero Tolerance policy for drinking that makes it a criminal offense when drivers under the age of 21 drive with even a small amount of alcohol in their system. Values under this column consist mostly of 0's and 1's with occasional fractions. - "bac" columns denote the blood alcohol content allowed by each state, with "bac08" representing 8% and "bac10" representing 10%. - "perse" relates to "Per se" laws in DUI cases. This law establishes that once an individual is shown to have a BAC at or above the state's allowed percent, that person is considered intoxicated by law. Such laws allow a person to be established as impaired without on-scene evaluation such as sobriety testing. Values range from 0 to 1, with fractions representing the amount of time for which the limit was enforced by law that year given a change in state law. *data Fatalities* - totfat: total data fatalities - nghtfat: total nighttime fatalities - wkndfat: total weekend fatalities - totfatpvm: total fatalities per 100 million miles - nghtfatpvm: nighttime fatalities per 100 million miles - wkndfatpvm: weekend fatalities per 100 million miles - totfatrte: total fatalities per 100,000 population - nghtfatrte: nighttime fatalities per 100,000 population - wkndfatrte: weekend accidents per 100,000 population *Demographics* - gdl: graduated drivers license law - statepop: state population - vehicmiles: vehicle miles traveled, billions - unem: unemployment rate, percent - perc14_24: percent population aged 14 through 24

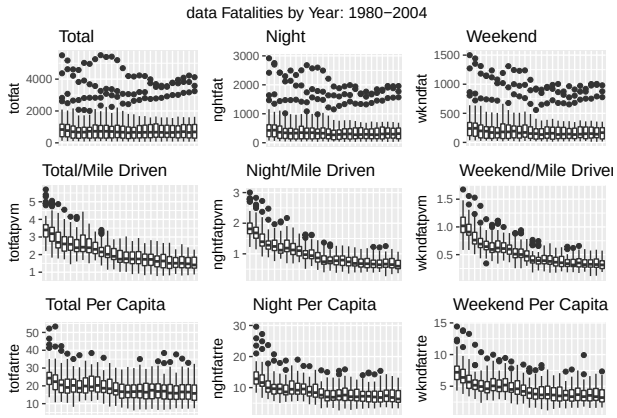
```
d1 <- data %>% ggplot(aes(totfat)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Total")
d2 <- data %>% ggplot(aes(nghtfat)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Night")
d3 <- data %>% ggplot(aes(wkndfat)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Weekend")
d4 <- data %>% ggplot(aes(totfatpvm)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Total/Mile Driven")
d5 <- data %>% ggplot(aes(nghtfatpvm)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Night/Mile Driven")
d6 <- data %>% ggplot(aes(wkndfatpvm)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Weekend/Mile Driven")
d7 <- data %>% ggplot(aes(totfatrte)) + geom_density(kernel = "gaussian",
```

```
    size = 1) + theme_bw() + ggtitle("Total Per Capita")
d8 <- data %>% ggplot(aes(nghtfatrte)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Night Per Capita")
d9 <- data %>% ggplot(aes(wkndfatrte)) + geom_density(kernel = "gaussian",
    size = 1) + theme_bw() + ggtitle("Weekend Per Capita")
grid.arrange(d1, d2, d3, d4, d5, d6, d7, d8, d9, nrow = 3, ncol = 3, top = quote("data Fatalities"))
```



All fatality variables skew right, with the skew most apparent when not normalized by population or by mileage.

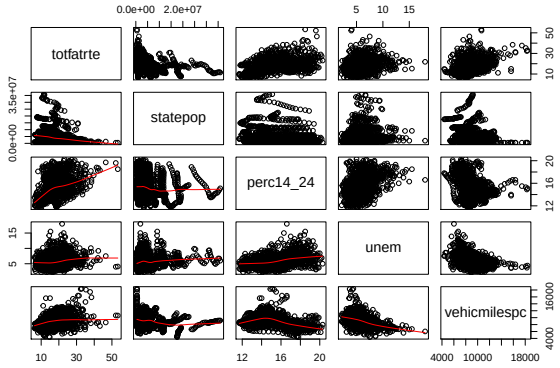
```
t <- theme(axis.title.x = element_blank(), axis.text.x = element_blank(),
    axis.ticks.x = element_blank())
b1 <- data %>% ggplot(aes(factor(year), totfat)) + geom_boxplot() + ggtitle("Total") +
    t
b2 <- data %>% ggplot(aes(factor(year), nghtfat)) + geom_boxplot() + ggtitle("Night") +
    t
b3 <- data %>% ggplot(aes(factor(year), wkndfat)) + geom_boxplot() + ggtitle("Weekend") +
    t
b4 <- data %>% ggplot(aes(factor(year), totfatpvm)) + geom_boxplot() +
    ggtitle("Total/Mile Driven") + t
b5 <- data %>% ggplot(aes(factor(year), nghtfatpvm)) + geom_boxplot() +
    ggtitle("Night/Mile Driven") + t
b6 <- data %>% ggplot(aes(factor(year), wkndfatpvm)) + geom_boxplot() +
    ggtitle("Weekend/Mile Driven") + t
b7 <- data %>% ggplot(aes(factor(year), totfatrte)) + geom_boxplot() +
    ggtitle("Total Per Capita") + t
b8 <- data %>% ggplot(aes(factor(year), nghtfatrte)) + geom_boxplot() +
    ggtitle("Night Per Capita") + t
b9 <- data %>% ggplot(aes(factor(year), wkndfatrte)) + geom_boxplot() +
    ggtitle("Weekend Per Capita") + t

grid.arrange(b1, b2, b3, b4, b5, b6, b7, b8, b9, nrow = 3, ncol = 3, top = quote("data Fatalities by Yea
```



The number of fatalities across total, night and weekend look mostly unchanged over the years. However, when looking at normalized fatality rates, there are obvious downward trends at the national level.

Let's also look at the scatterplot matrix between key continuous variable and our dependent variable *totfatrte*

```
pairs(~totfatrte + statepop + perc14_24 + unem + vehicmilespc, data = data,
    lower.panel = panel.smooth)
```

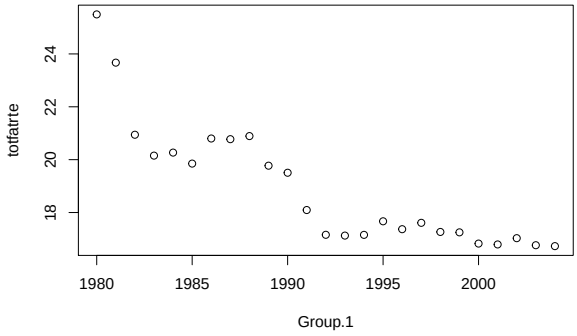


It doesn't seem like these variables need transformation since we didn't observe any nonlinear correaltion.

2. How is the our dependent variable of interest *totfatrte* defined? What is the average of this variable in each of the years in the time period covered in this dataset? Estimate a very simple regression model of totfatrte on dummy variables for the years 1981 through 2004. What does this model explain? Describe what you find in this model. Did driving become safer over this period? Please provide a detailed explanation.

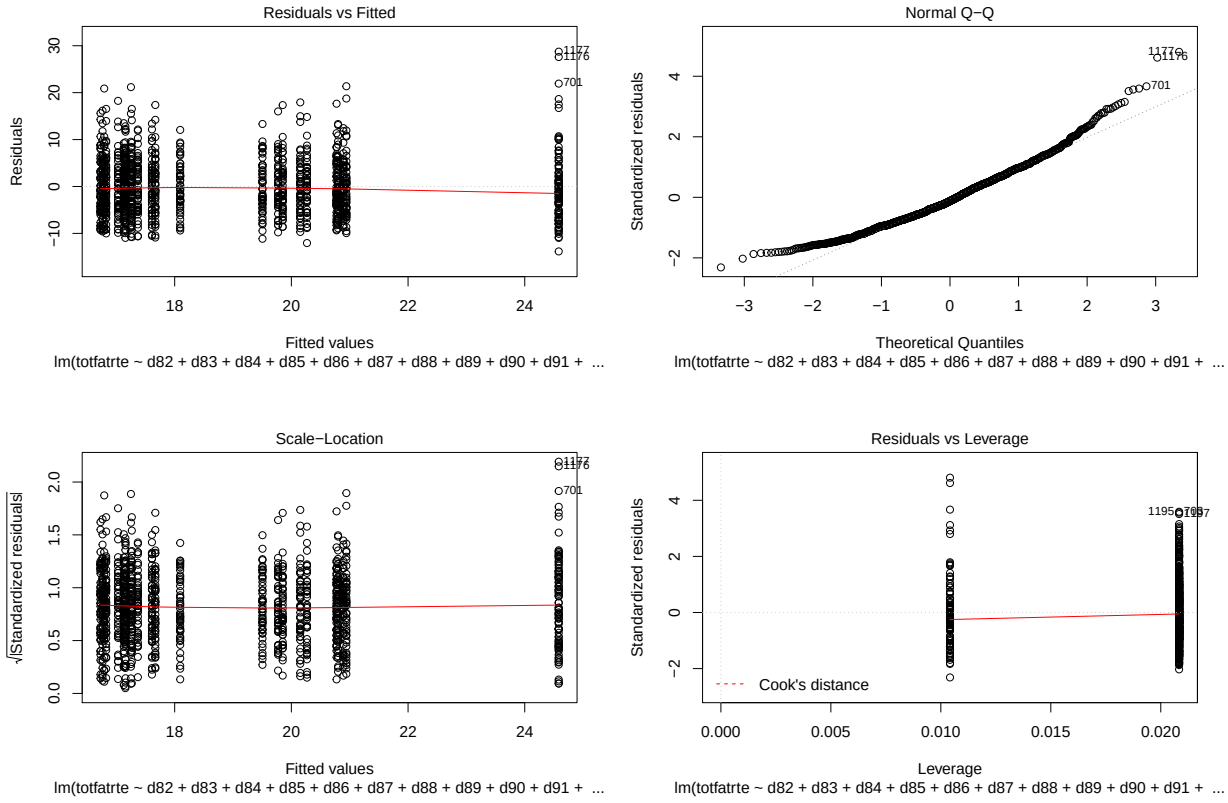
The dependent variable of interest *totfatrte* is total fatalities per 100,000 population.

```
plot(aggregate(data["totfatrte"], list(data$year), mean))
```



The average total fatality rate shows a downward trend from 25 per 100,000 residents in 1980 to 17 in 2004. There's a steep drop from 1980 to 1983 and then from 1988 to 1992.

```
m1 <- lm(totfatrte ~ d82 + d83 + d84 + d85 + d86 + d87 + d88 + d89 + d90 +
    d91 + d92 + d93 + d94 + d95 + d96 + d97 + d98 + d99 + d00 + d01 + d02 +
    d03 + d04, data = data)
summary(m1)
plot(m1)
```



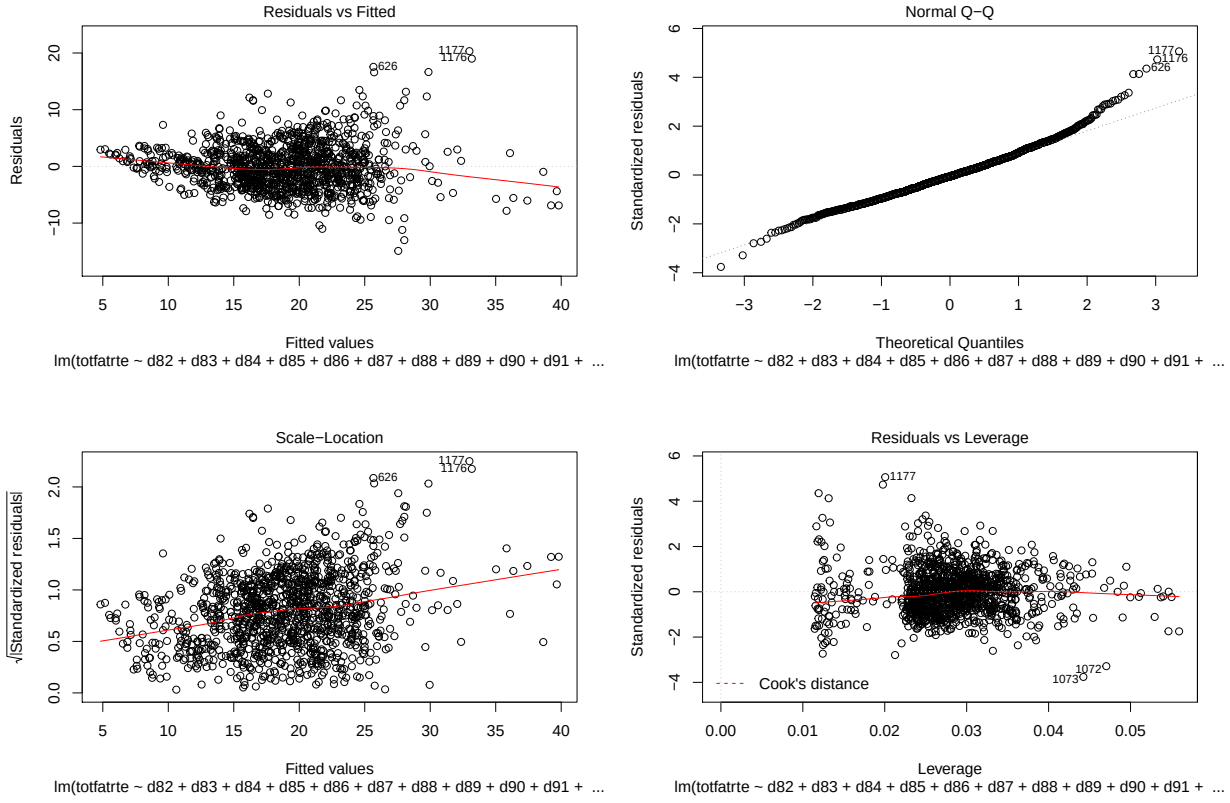
[Shan: I took out d81 because it will give the model high collinearity. d81 is linearly dependent with all the rest year dummies]

This model suggests that time was a statistically significant variable at the 1% signifiance level every year with negative coefficients. Since the baseline is 1981, we see strong evidence that drivers become safer over

the years.

3. Expand your model in *Exercise 2* by adding variables *bac08, bac10, perse, sbprim, sbsecon, sl70plus, gdl, perc14_24, unem, vehicmilespc*, and perhaps *transformations of some or all of these variables*. Please 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. How are the variables *bac8* and *bac10* defined? Interpret the coefficients on *bac8* and *bac10*. Do *per se laws* have a negative effect on the fatality rate? What about having a primary seat belt law? (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.)

```
m2 <- lm(totfatrte ~ d82 + d83 + d84 + d85 + d86 + d87 + d88 + d89 + d90 +
    d91 + d92 + d93 + d94 + d95 + d96 + d97 + d98 + d99 + d00 + d01 + d02 +
    d03 + d04 + bac08 + bac10 + perse + sbprim + sbsecon + sl70plus + gdl +
    perc14_24 + unem + vehicmilespc, data = data)
summary(m2)
plot(m2)
```



No transformation was needed for the continuous variables since 1) they are normalized and 2) they don't demonstrate obvious non-linear relationship with the dependent variable.

bac8 and *bac10* are defined to denote the blood alcohol content allowed by each state, with "bac08" representing 8% and "bac10" representing 10%. They are not strictly binary and the fraction denotes the fraction of time that the policy was imposed, just like the other "binary" variables" for the other data laws.

The coefficient for bac8 is -2.50 and bac10 -1.14. This means that with all other variables held constant, the fatality rate drops by 2.50 if 8% alchohol content is allowed and it's enforced 100% of the time. Similarly, the fatality rate drops by 1.14 if 10% alchohol content is allowed and it's enforced 100% of the time. Both coefficients are statistically significant.

Per se law seemed to have a significant negative effect on fatality rate with 95% confidence level. But the primamry seat belt law doesn't seem to have a significant impact of the fatality rate.

4. Reestimate the model from *Exercise 3* using a fixed effects (at the state level) model. How do the coefficients on *bac08, bac10, perse, and sbprim* compare with the pooled OLS estimates? Which set of estimates do you think is more reliable? What assumptions are needed in each of these models? Are these assumptions reasonable in the current context?

```
data.panel <- plm.data(data, c("state", "year"))

m3.fe <- plm(totfatrte ~ d82 + d83 + d84 + d85 + d86 + d87 + d88 + d89 +
    d90 + d91 + d92 + d93 + d94 + d95 + d96 + d97 + d98 + d99 + d00 + d01 +
    d02 + d03 + d04 + bac08 + bac10 + perse + sbprim + sbsecon + sl70plus +
    gdl + perc14_24 + unem + vehicmilespc, data = data.panel, model = "within")

summary(m3.fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = totfatrte ~ d82 + d83 + d84 + d85 + d86 + d87 +
##     d88 + d89 + d90 + d91 + d92 + d93 + d94 + d95 + d96 + d97 +
##     d98 + d99 + d00 + d01 + d02 + d03 + d04 + bac08 + bac10 +
##     perse + sbprim + sbsecon + sl70plus + gdl + perc14_24 + unem +
##     vehicmilespc, data = data.panel, model = "within")
##
## Balanced Panel: n=48, T=25, N=1200
##
## Residuals :
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -7.6876679 -1.0325795  0.0068316  0.9496057 14.0209108
##
## Coefficients :
##                 Estimate  Std. Error  t-value  Pr(>|t|)
## d82          -2.22321874  0.38632353  -5.7548 1.119e-08 ***
## d83          -2.68990862  0.40088536  -6.7099 3.087e-11 ***
## d84          -3.46249353  0.41297938  -8.3842 < 2.2e-16 ***
## d85          -3.92034415  0.43487479  -9.0149 < 2.2e-16 ***
## d86          -2.84456705  0.46963313  -6.0570 1.891e-09 ***
## d87          -3.48232453  0.51042298  -6.8224 1.463e-11 ***
## d88          -3.93813818  0.56028011  -7.0289 3.613e-12 ***
## d89          -5.29070925  0.60093386  -8.8041 < 2.2e-16 ***
## d90          -5.37740343  0.62608949  -8.5889 < 2.2e-16 ***
## d91          -6.04479364  0.64239267  -9.4098 < 2.2e-16 ***
## d92          -6.89078521  0.66370931 -10.3822 < 2.2e-16 ***
## d93          -7.21492792  0.67809127 -10.6401 < 2.2e-16 ***
## d94          -7.63274603  0.69817926 -10.9324 < 2.2e-16 ***
## d95          -7.38732434  0.72197164 -10.2322 < 2.2e-16 ***
## d96          -7.72641487  0.76286585 -10.1281 < 2.2e-16 ***
## d97          -7.82978738  0.78812686  -9.9347 < 2.2e-16 ***
## d98          -8.47500690  0.80312827 -10.5525 < 2.2e-16 ***
## d99          -8.60530254  0.81426306 -10.5682 < 2.2e-16 ***
## d00          -9.12491737  0.82711742 -11.0322 < 2.2e-16 ***
## d01          -8.75514501  0.84364721 -10.3777 < 2.2e-16 ***
## d02          -8.01962252  0.85271345  -9.4048 < 2.2e-16 ***
```

```
## d03          -8.04738896  0.86081013  -9.3486 < 2.2e-16 ***
## d04          -8.45876376  0.88351466  -9.5740 < 2.2e-16 ***
## bac08        -1.42650736  0.39637457  -3.5989 0.0003336 ***
## bac10        -1.05770861  0.27031636  -3.9129 9.671e-05 ***
## perse        -1.13843446  0.23524932  -4.8393 1.486e-06 ***
## sbprim       -1.22821353  0.34460425  -3.5641 0.0003805 ***
## sbsecon      -0.35242029  0.25356009  -1.3899 0.1648391
## sl70plus     -0.08639032  0.27071594  -0.3191 0.7496966
## gdl          -0.40782268  0.29418493  -1.3863 0.1659376
## perc14_24     0.21291082  0.09536071   2.2327 0.0257667 *
## unem         -0.58520311  0.06080157  -9.6248 < 2.2e-16 ***
## vehicmilespc  0.00092814  0.00011161   8.3161 2.614e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    12134
## Residual Sum of Squares: 4589.5
## R-Squared:      0.62177
## Adj. R-Squared: 0.59473
## F-statistic: 55.7422 on 33 and 1119 DF, p-value: < 2.22e-16
```

The coefficients for bac08 and bac10 became less negative yet those for perse and sbprim become more pronounced. The p-value for perse and sbprim also decreased, implying strong statistical significance (at 99.9% confidence level).

I think the exstimes from this fixed effects model are more reliable since it controls for not just year-level but also state-level unobserved effect.

On top of the typical CLM assumptions, the pooled effects assumption is that the individual-specific effects are uncorrelated with the independent variables for the estimates to be unbiased.

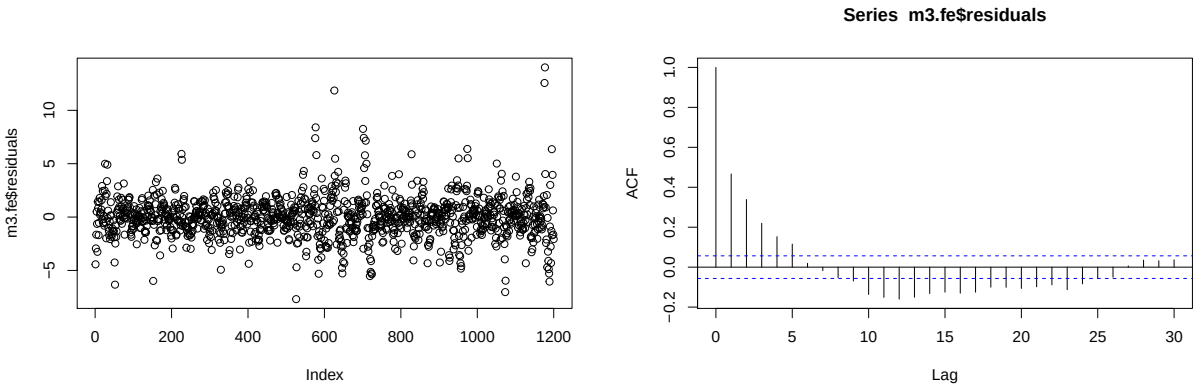
But the fixed effect model is robust even when the individual-specific effects are correlated with the independent variables.

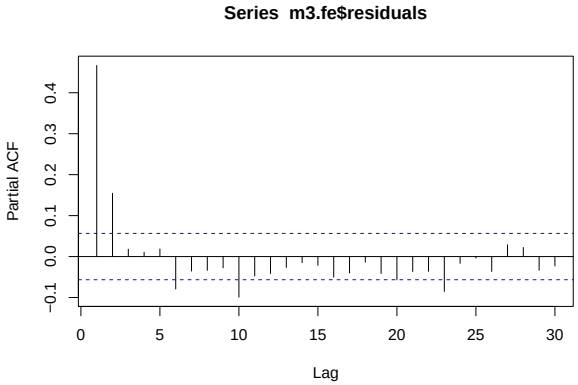
5. Suppose that *vehicmilespc*, the number of miles driven per capita, increases by 1,000. Using the FE estimates, what is the estimated effect on totfatrte? Please interpret the estimate as if explaining to a layperson.

Since the coefficient is 0.000928, an increase by 1,000 indicates that, with all other variables held constant, the fatality rate will go up by 0.928. In simple English, with all other variables being the same, a 1000 miles increase in the number of miles driven per captia will increase the fatality rate by 0.928.

6. If there is serial correlation or heteroskedasticity in the idiosyncratic errors of the model, what would be the consequences on the coefficient estimates and their standard errors?

```
plot(m3.fe$residuals)
acf(m3.fe$residuals)
pacf(m3.fe$residuals)
```





If there is existence of heteroskedasticity, then our estimates would not be consistent. From the plot of the residuals, our model seems robust against heteroskedasticity.

If there existed postive serial correlation in the model errors, we would have underestimated the p-values of our estimates. From the acf and pacf plots, we see evidence of postive serial correlation between our residuals.