Data Analysis

Julian Gradziel

2023-11-01

1

CODE: firstly, we need to make sure the working directory is set correctly (see #). We then load in the necessary libraries and data for the following tasks: Using the `library()` function, we load the `dplyr`, `ggplot2`, and the `margins` packages. Using the `load()` function, we load in the data set.

```
library(dplyr)
```

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

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(margins)
setwd("C:/Users/julia/Downloads/Empirical data analysis/Data")
load("admissions.Rdata")
```

Task 1

a.

CODE: By using the `head()` function, we can display the first few parts of the data (if not specified by n, the first 6)

```
head(admissions)
```

```
## # A tibble: 6 x 4
##   admit   gre   gpa  rank
##   <dbl> <dbl> <dbl> <dbl>
## 1     0   380  3.61     3
## 2     1   660  3.67     3
## 3     1   800  4        1
## 4     1   640  3.19     4
## 5     0   520  2.93     4
## 6     1   760  3        2
```

b)

CODE: To compare the GPA's of those accepted to the schools and those not accepted, we had to seperate the values in the already existing data using indexing, and subsetting (gpa dependent on whether the student was or was not accepted to a graduate school). We saved those subset in `adm` and `nadm` respectively, for further analysis. Then we calculate the variance of both those subsets using the built in `var()` function, the mean using `mean()` and the median using `median()`. We show the results using the `cat()` function.

```
adm <- admissions$gpa[admissions$admit == 1]
nadm <- admissions$gpa[admissions$admit == 0]

cat("The variance of GPA for admitted students is ",
    var(adm),"\ncompared to", var(nadm),"for not admitted ones")
```

```
## The variance of GPA for admitted students is  0.1370311
## compared to 0.1422293 for not admitted ones
```

```
cat("The mean GPA for admitted students is ",
    mean(adm),"\ncompared to", mean(nadm),"for not admitted ones")
```

```
## The mean GPA for admitted students is  3.489213
## compared to 3.3437 for not admitted ones
```

```
cat("The median GPA for admitted students is ", median(adm),
    "\ncompared to", median(nadm),"for not admitted ones")
```

```
## The median GPA for admitted students is  3.54
## compared to 3.34 for not admitted ones
```

INTERPRETATION:

The variance in GPA for admitted students is slightly lower than that for not admitted students, suggesting that GPA's of admitted students are more consistent or clustered around the mean GPA compared to those of not admitted students.

The mean GPA for admitted students (3.489213) is higher than the mean GPA for not admitted students (3.3437), indicating that, on average, students who were admitted to graduate school have a higher GPA compared to those who were not admitted.

Admitted students have a higher median GPA compared to not admitted students, indicating that half of the admitted students have a GPA greater than 3.54, compared to 3.34 for not admitted ones. It would seem that the GPA has an influence on being admitted. However, this sort of analysis is not reliable, and results should be taken with a grain of salt.

Task 2

a.

CODE: We added a new column to the dataframe by creating a new vector named `adj_gre` which is a twin vector of `gre`, but 10 times smaller (divided by 10), and assigning it to the admissions dataframe.

```
admissions$adj_gre <- admissions$gre/10
```

b.

CODE: We estimated a logit model by using the `glm()` function. Simply using `lm()` wouldn't be sufficient as we would not be able to add `family = 'binomial'` as an argument, which is needed for a logit model. Within the `glm()` function we examine the relationship between the given vectors from the "admissions" dataset. We stored the results in `logit`, and retrieved a summary of those using the `summary()` function.

```
logit = glm(
  admit ~ rank + gpa + adj_gre,
  data = admissions,
  family = 'binomial'
)
summary(logit)
```

```
##
## Call:
## glm(formula = admit ~ rank + gpa + adj_gre, family = "binomial",
##     data = admissions)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.44955    1.13285  -3.045  0.00233 **
## rank        -0.56003    0.12714  -4.405 1.06e-05 ***
## gpa          0.77701    0.32748   2.373  0.01766 *
## adj_gre      0.02294    0.01092   2.101  0.03564 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 499.98  on 399  degrees of freedom
## Residual deviance: 459.44  on 396  degrees of freedom
## AIC: 467.44
##
## Number of Fisher Scoring iterations: 4
```

INTERPRETATION:

All three variables matter for admission. The results are significant (Assuming a 95% significance level). We can therefore draw the conclusion that the better the GPA and GRE the better the chances to get admitted (since the estimates are >0), and that the better the rank of graduate schools, the worse the chances of getting in are (estimate <0).

Task 3

a.

CODE: We used the `margins()` function from the margins Package to compute the marginal effect.

```
margins(logit)
```

```
## Average marginal effects

## glm(formula = admit ~ rank + gpa + adj_gre, family = "binomial",     data = admissions)

##     rank    gpa  adj_gre
##  -0.1094 0.1518 0.004481
```

INTERPRETATION:

Assuming no other influences then mentioned above, a 1 point increase of GPA leads to you having a ~15.2% higher chance of getting admitted.

2

first we install (#) and load the necessary packages:

```
#install.packages("ggplot2")
library(ggplot2)
#install.packages("dplyr")
library(dplyr)
#install.packages("estimatr")
library(estimatr)
```

we then load in the dataset (make sure to set the wd correctly beforehand), and check if it loaded correctly using the `head()`function:

```
setwd("C:/Users/julia/Downloads/Empirical data analysis/Data")
data <- read.csv("ebay.csv")
head(data)
```

```
##   treatnr blind dsr round seller_quality_choice buyer_cf_notpos auction_price
## 1       3     0   1     1                    53               1           141
## 2       2     1   0     1                    70               1           171
## 3       3     0   1     1                    95               0           171
## 4       3     0   1     1                    85               0           181
## 5       2     1   0     1                   100               1           211
## 6       1     0   0     1                    70               1           175
##   matching_group_id
## 1                57
## 2                32
## 3                24
## 4                22
## 5                65
## 6                12
```

Task 1

Does the average quality provided by the seller differ in different treatments?

Descriptive:

We first expanded the data by categorizing the `treatnr` variable into factors representing different treatments: 'CF', 'Blind', 'CF + DSR'. We make sure that CF is the base level, using `relevel()`. We create two additional factor variables: `is_blind` & `is_dsr`, based on `blind` and `dsr` respectively. Lastly, we compute summary statistics for 'seller_quality_choice' within each treatment group, including minimum, median, mean, standard deviation, standard error of the mean, and quartiles.The results are stored in `data_summary` for further analysis

```
data <- data %>% mutate(treatment = factor(recode(treatnr, "1" = "CF","2" = "Blind","3" = "CF + DSR")))
data$treatment <- relevel(data$treatment, "CF")
data <- data %>% mutate(is_blind = factor(recode(blind, "0" = "Not blind",
"1" = "Blind")))
data <- data %>% mutate(is_dsr = factor(recode(dsr, "0" = "Not DSR","1" = "DSR")))

data_summary <- data %>%
  group_by(treatnr) %>%
  summarise(minimum = min(seller_quality_choice),
            q1 = quantile(seller_quality_choice, 0.25),
            median = median(seller_quality_choice),
            mean = mean(seller_quality_choice),
            standard_dev = sd(seller_quality_choice),
            standard_err = sd(seller_quality_choice)/sqrt(length(seller_quality_choice)),
            q3 = quantile(seller_quality_choice, 0.75),
            maximum = max(seller_quality_choice))
data_summary
```

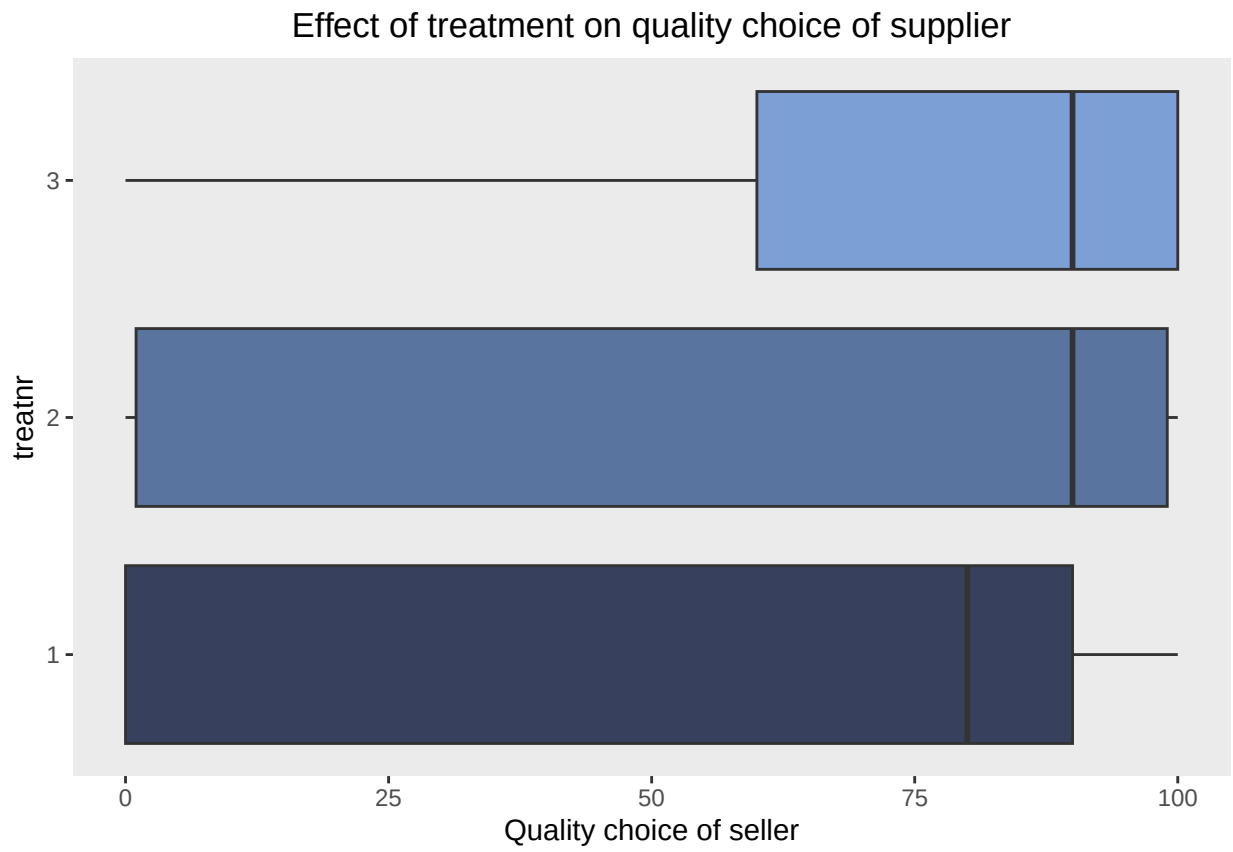
```
## # A tibble: 3 x 9
##   treatnr minimum    q1 median  mean standard_dev standard_err    q3 maximum
##     <int>   <int> <dbl>  <dbl> <dbl>        <dbl>        <dbl> <dbl>   <int>
## 1       1       0     0     80  58.5         40.0         1.41    90     100
## 2       2       0     1     90  65.6         40.8         1.44    99     100
## 3       3       0    60     90  72.1         38.0         1.34   100     100
```

INTERPRETATION: All treatments have the same lowest quality (0), Treatment 1 has the lowest median quality (80), and Treatment 2 shows the widest spread in quality, as indicated by its larger standard deviation. The standard errors are quite low in all treatments, suggesting precise mean estimates. Overall, Treatment 3 has the highest mean quality among the three treatments, indicating a potential higher seller quality choice caused by this treatment. To know more, we have to conduct more in depth analysis.

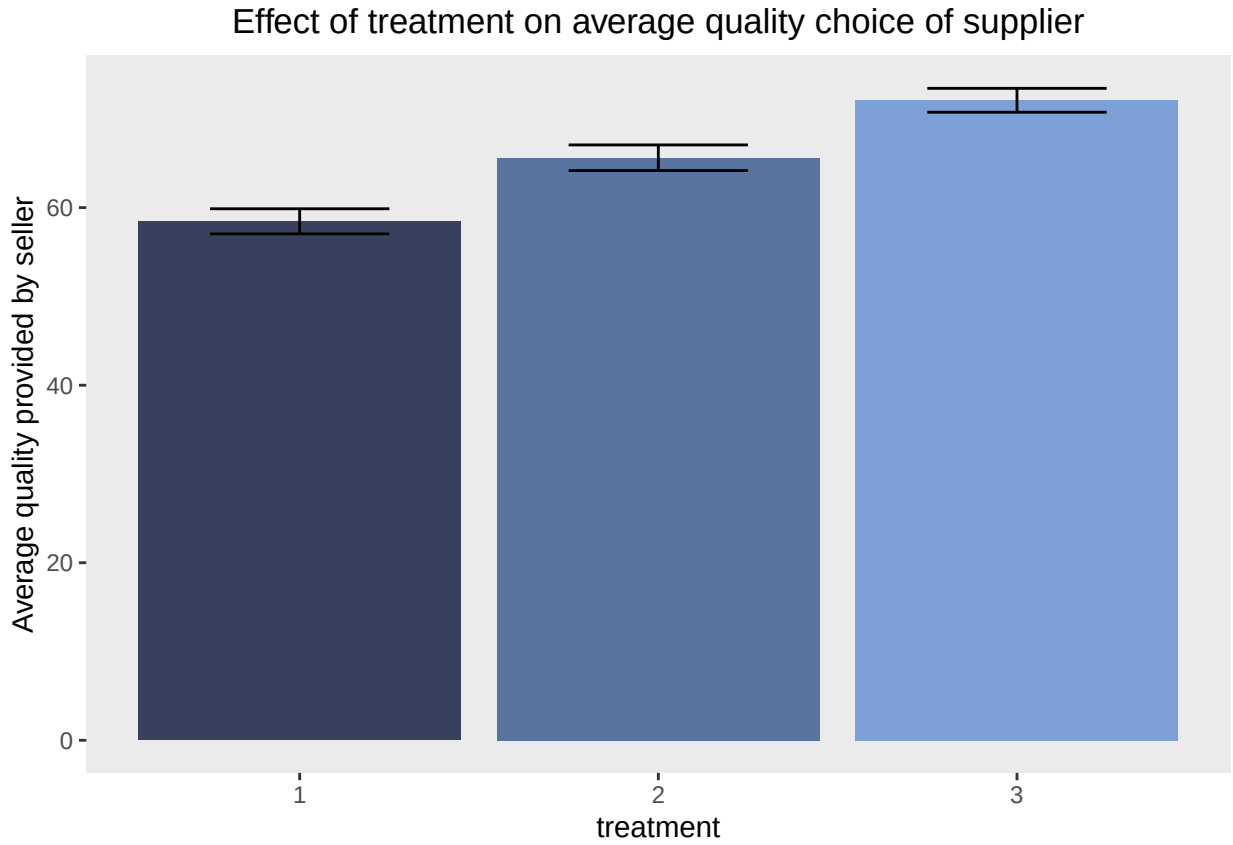
Visualisation:

The first plot utilizes `geom_boxplot()` to display the distribution of seller quality choices for each treatment group. Each treatment (`CF`, `Blind`, `CF + DSR`) is represented on the y-axis, and seller quality choices are on the x-axis. The second plot uses `geom_bar()` to show the mean quality choice for each treatment, with error bars indicating the standard error of the mean.

```
ggplot(data, aes(x = seller_quality_choice, y = treatnr)) +
  geom_boxplot(aes(group=treatnr), fill=c("#37415e", "#59749e", "#7c9fd6"))+
  xlab("Quality choice of seller") +
  ggtitle("Effect of treatment on quality choice of supplier")+
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank(),
  panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```




```
data_summary %>% ggplot(aes(x = treatnr, y = mean, fill = treatnr)) +
  geom_bar(position=position_dodge(), stat="identity", fill=c("#37415e",
  "#59749e", "#7c9fd6")) +
  geom_errorbar(aes(ymin=mean-standard_err, ymax=mean+standard_err), width=.5,
  position=position_dodge(.9)) +
  xlab("treatment") +
  ylab("Average quality provided by seller") +
  ggtitle("Effect of treatment on average quality choice of supplier") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank(),
  panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```



INTERPRETATION: These Graphs visually represent what we already interpreted in the previous step. We can see that treatments 1 and 2 have a wide spread of quality. Treatment 3 seems less variant.

Statistical tests:

In the first part, we check whether it is likely that the data follows a normal distribution using the `shapiro.test()` function, see our conclusion:

```
stest <- rep(0,3)
for(i in 1:3){
stest[i] <- shapiro.test(data$seller_quality_choice[data$treatnr==i])$p.value
if(stest[i]<0.05){
  cat("\ntreatmeant",i,"seller quality choices are unlikely to be N distributed")
}else {
  cat("\ntreatment",i,"seller quality choices are likely to be N distributed")
}}
```

```
##
## treatmeant 1 seller quality choices are unlikely to be N distributed
## treatmeant 2 seller quality choices are unlikely to be N distributed
## treatmeant 3 seller quality choices are unlikely to be N distributed
```

We now conduct two Wilcoxon rank-sum tests, comparing mean quality choices between different treatments. We use this test, instead of the t.test, because our data is not normally distributed. Lastly, we fit a linear regression model using `lm_robust()` to investigate the relationship between treatment and seller quality choices while considering the clustering effect of `matching_group_id`.

```r
DSR <- data %>% filter(is_blind == "Not blind") %>%
  group_by(treatment, matching_group_id) %>%
  summarise(mean_quality = mean(seller_quality_choice, na.rm = TRUE),
  mean_price = mean(auction_price, na.rm = TRUE))

blind <- data %>% filter(is_dsr == "Not DSR") %>%
  group_by(treatment, matching_group_id) %>%
  summarise(mean_quality = mean(seller_quality_choice, na.rm = TRUE),
  mean_price = mean(auction_price, na.rm = TRUE))


wilcox.test(mean_quality ~ treatment, data = DSR)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  mean_quality by treatment
## W = 11, p-value = 0.02813
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(mean_quality ~ treatment, data = blind)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  mean_quality by treatment
## W = 30, p-value = 0.8785
## alternative hypothesis: true location shift is not equal to 0
```

```r
m1 <- lm_robust(seller_quality_choice ~ treatment , clusters = matching_group_id, data = data)
summary(m1)
```

```
##
## Call:
## lm_robust(formula = seller_quality_choice ~ treatment, data = data,
##     clusters = matching_group_id)
##
## Standard error type:  CR2
##
## Coefficients:
##                   Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper DF
## (Intercept)         58.457      4.287 13.6348 2.687e-06   48.319    68.60  7
## treatmentBlind       7.161      8.378  0.8548 4.070e-01  -10.807    25.13 14
## treatmentCF + DSR   13.635      5.491  2.4833 2.630e-02    1.859    25.41 14
##
## Multiple R-squared:  0.01939 ,   Adjusted R-squared:  0.01857
## F-statistic: 3.095 on 2 and 23 DF,  p-value: 0.06454
```

INTERPRETATION: For the subset of data excluding blindness (DSR), the test resulted in a p-value of 0.02813, indicating a statistically significant difference in mean quality choices between treatments when compared to the baseline treatment (CF). For the subset of data excluding DSR (blind), the test yielded

a p-value of 0.8785, suggesting no significant difference in mean quality choices among treatments when compared to the baseline treatment (CF).

LM: The intercept represents the mean seller quality choice for the CF treatment group. The coefficient estimate of 58.457 indicates that the mean quality choice for CF treatment is 58.457. This is statistically significant (p>0.05)

The coefficient for the Blind treatment group is 7.161. It signifies that, on average, the seller quality choice in the Blind treatment group is higher by 7.161 than in the CF treatment group. However, this difference is not statistically significant (p = 0.407).

The coefficient for the CF + DSR treatment group is 13.635. It suggests that, on average, the seller quality choice in the CF + DSR treatment group is higher by 13.635 than in the CF treatment group. Importantly, this difference is statistically significant (p = 0.0263), indicating that the seller quality choice in the CF + DSR treatment group significantly differs from the CF treatment group.

The two methods come to the same conclusion.

Task 2

Does the average auction price achieved differ in different treatments?

Descriptive:

The code works similarily to the one in Task 1. The only difference is the variable (`auction_price`) and the filtering out of NA's using the `is.na()` function.

```
data_summary2 <- data %>% subset(!is.na(auction_price)) %>%
  group_by(treatnr) %>%
  summarise(minimum = min(auction_price),
            q1 = quantile(auction_price, 0.25),
            median = median(auction_price),
            mean = mean(auction_price),
            standard_dev = sd(auction_price),
            standard_err = sd(auction_price)/sqrt(length(auction_price)),
            q3 = quantile(auction_price, 0.75),
            maximum = max(auction_price))
data_summary2
```

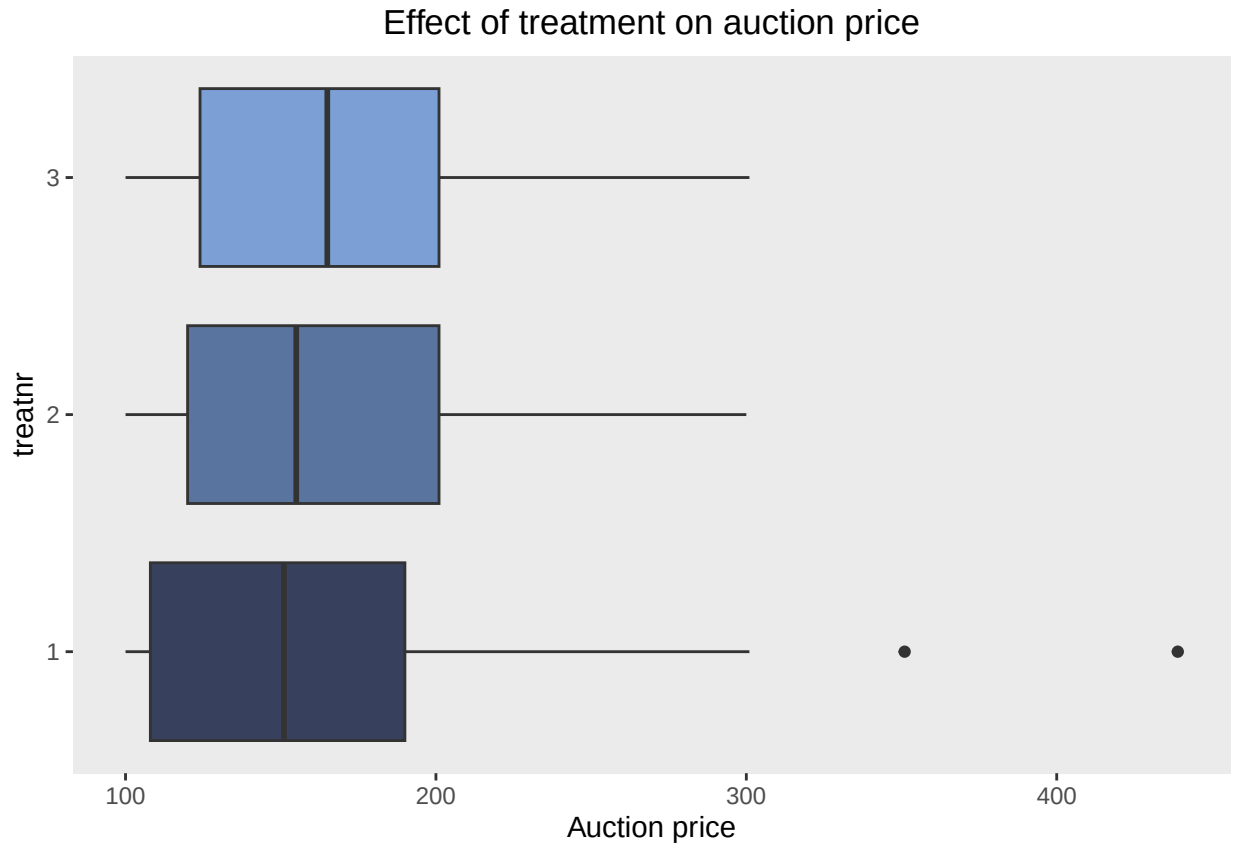
```
## # A tibble: 3 x 9
##   treatnr minimum    q1 median  mean standard_dev standard_err    q3 maximum
##     <int>   <int> <dbl>  <int> <dbl>        <dbl>        <dbl> <dbl>   <int>
## 1       1     100   108    151  155.         48.9         1.79   190     439
## 2       2     100   120    155  163.         50.4         1.83   201     300
## 3       3     100   124    165  168.         49.6         1.78   201     301
```

INTERPRETATION: All treatments have the same minimum price (100), Treatment 1 has the lowest median price (108), and Treatment 2 shows the widest spread in quality, as indicated by its larger standard deviation. The standard errors are quite low in all treatments, but higher than in the quality interpretation. Overall, Treatment 3 has the highest mean price among the three treatments, indicating a potentially higher price choice caused by this treatment. The interpretation is nearly identical to the one in task 1. To know more, we have to conduct more in depth analysis.

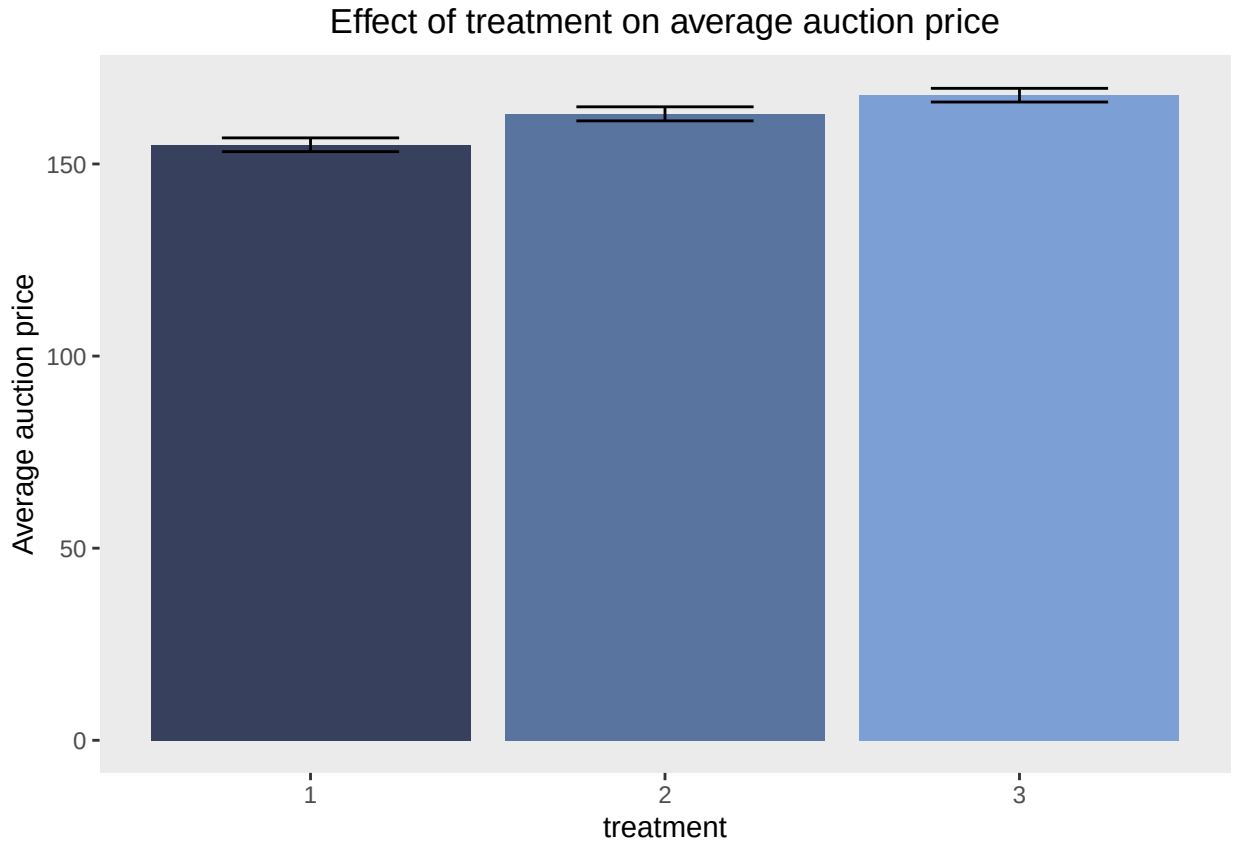
Visualisation:

again, we can recycle the code from Task 1, changing only the variables and descriptions.

```
ggplot(data, aes(x = auction_price, y = treatnr)) +
  geom_boxplot(aes(group=treatnr), fill=c("#37415e", "#59749e", "#7c9fd6"))+
  xlab("Auction price") +
  ggtitle("Effect of treatment on auction price")+
theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank(),
panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```




```
data_summary2 %>% ggplot(aes(x = treatnr, y = mean, fill = treatnr)) +
  geom_bar(position=position_dodge(), stat="identity", fill=c("#37415e",
  "#59749e", "#7c9fd6")) +
  geom_errorbar(aes(ymin=mean-standard_err, ymax=mean+standard_err), width=.5,
  position=position_dodge(.9))+
  xlab("treatment") +
  ylab("Average auction price") +
  ggtitle("Effect of treatment on average auction price") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank(),
  panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```



INTERPRETATION: Again, these Graphs visually represent what we already interpreted in the previous step. However, in the boxplot, we can see outliers in the base treatment. The medians of treatments 1 and 2 are higher, but the distribution looks similar. In the barplot, all means are different, but the difference is small.

Statistical tests:

We once again conduct shapiro tests:

```
stest <- rep(0,3)
for(i in 1:3){
stest[i] <- shapiro.test(data$auction_price[data$treatnr==i])$p.value
if(stest[i]<0.05){
  cat("\ntreatnr",i,"is unlikely to be N distributed")
}else {
  cat("\ntreatnr",i,"is likely to be N distributed")
}}
```

```
##
## treatnr 1 is unlikely to be N distributed
## treatnr 2 is unlikely to be N distributed
## treatnr 3 is unlikely to be N distributed
```

And conduct the wilcoxon test, since our data is not normally distributed. This code is also recycled from Task 1.

```r
wilcox.test(mean_price ~ treatment, data = DSR)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  mean_price by treatment
## W = 16, p-value = 0.1049
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(mean_price ~ treatment, data = blind)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  mean_price by treatment
## W = 22, p-value = 0.3282
## alternative hypothesis: true location shift is not equal to 0
```

```r
m2 <- lm_robust( auction_price ~ treatment , clusters = matching_group_id ,data = data)
summary(m2)
```

```
##
## Call:
## lm_robust(formula = auction_price ~ treatment, data = data, clusters = matching_group_id)
##
## Standard error type:  CR2
##
## Coefficients:
##                   Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper
## (Intercept)        154.996      6.149 25.2083 4.210e-08  140.442   169.55
## treatmentBlind       8.045     10.198  0.7889 4.434e-01  -13.841    29.93
## treatmentCF + DSR   12.905      7.270  1.7751 9.770e-02   -2.693    28.50
##                       DF
## (Intercept)        6.965
## treatmentBlind    13.908
## treatmentCF + DSR 13.946
##
## Multiple R-squared:  0.01135 ,   Adjusted R-squared:  0.01049
## F-statistic: 1.585 on 2 and 23 DF,  p-value: 0.2266
```

INTERPRETATION: For the subset excluding blindness (`DSR`), the test resulted in a p-value of 0.1049. This suggests that there is no statistically significant difference in mean auction prices between the treatment (CF + DSR) compared to the baseline treatment (CF) for this subset. For the subset excluding DSR (`blind`), the test yielded a p-value of 0.3282. Similarly, there is no significant difference in mean auction prices between treatment (blind) and the baseline treatment (CF) for this subset.

The intercept (CF treatment) is 154.996, representing the mean auction price for the CF treatment group. The coefficient for the Blind treatment group is 8.045, and for the CF + DSR treatment group, it is 12.905. However, neither of these differences are statistically significant. Again, both tests yield the same conclusion, only this time, it implies that there is no strong evidence to suggest significant differences in mean auction prices between these treatment groups and the CF treatment group.

Task 3

Do the treatments influence the probability to leave negative feedback in different treatments?

Descriptive:

Once again we compute summary statistics and store them, removing all NA's from the data.

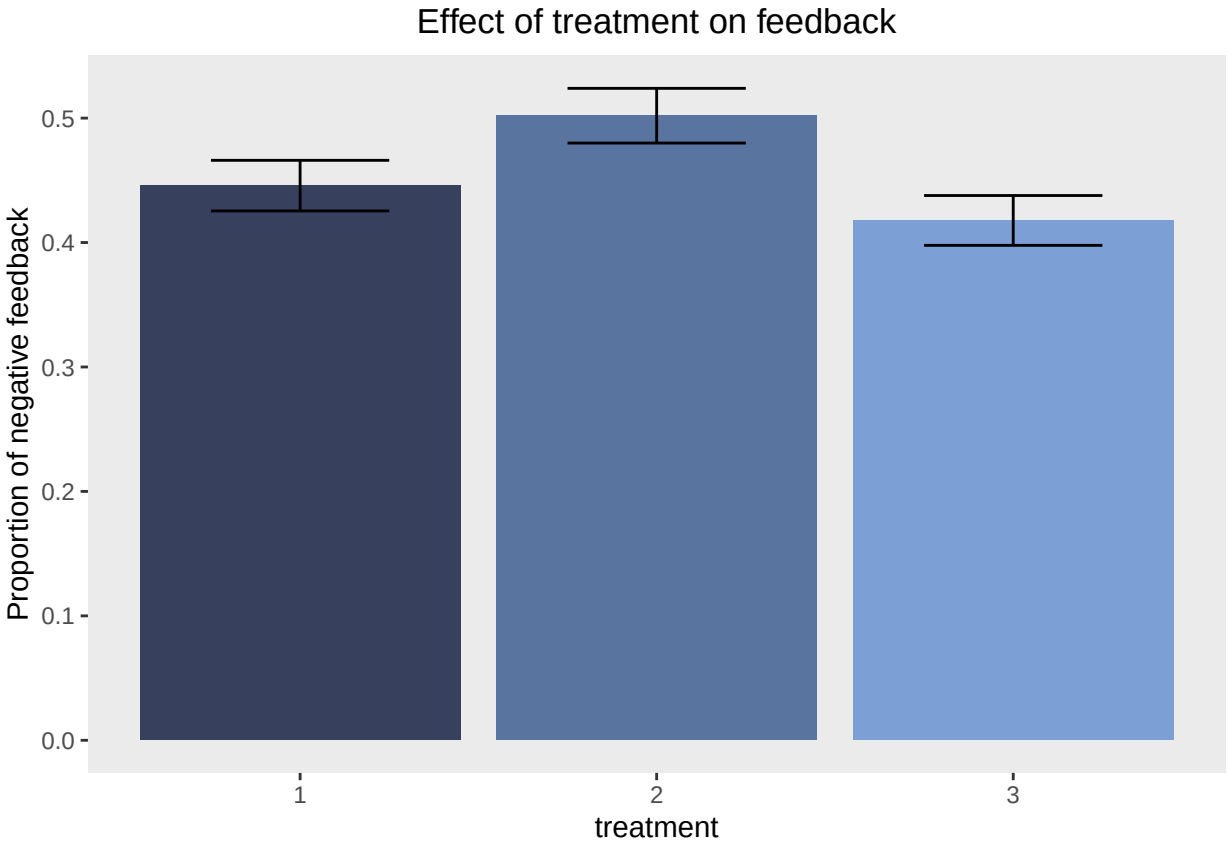
```
data_summary3 <- data %>% subset(!is.na(buyer_cf_notpos)) %>%
  group_by(treatnr) %>%
  summarise(mean = mean(buyer_cf_notpos),
            standard_dev = sd(buyer_cf_notpos),
            standard_err = sd(buyer_cf_notpos)/sqrt(length(buyer_cf_notpos)))
data_summary3
```

```
## # A tibble: 3 x 4
##   treatnr  mean standard_dev standard_err
##     <int> <dbl>        <dbl>        <dbl>
## 1       1 0.446        0.497       0.0203
## 2       2 0.502        0.500       0.0220
## 3       3 0.418        0.494       0.0200
```

INTERPRETATION: All treatments have the different means. Since the data is binominal, we can interpret the mean as proportions. Treatment 2, again, shows the widest spread in quality, as indicated by its larger standard deviation. The standard errors are extremely low in all treatments. Overall, Treatment 2 has the highest mean among the three treatments, indicating a potentially higher proportion to leave negative feedback caused by this treatment. To know more, we have to conduct more in depth analysis.

Visualisation:

```
data_summary3 %>% ggplot(aes(x = treatnr, y = mean, fill = treatnr)) +
  geom_bar(position=position_dodge(), stat="identity", fill=c("#37415e",
  "#59749e", "#7c9fd6")) +
  geom_errorbar(aes(ymin=mean-standard_err, ymax=mean+standard_err), width=.5,
  position=position_dodge(.9)) +
  xlab("treatment") +
  ylab("Proportion of negative feedback") +
  ggtitle("Effect of treatment on feedback") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank(),
  panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```



INTERPRETATION: Again, the Graph visually represent what we already interpreted in the previous step. All proportions are different, and it seems that difference (except between 1 and 3) is quite big.

Statistical tests:

```
m3 <- glm(buyer_cf_notpos ~ treatment , data = data, family = binomial(link = "logit"))
summary(m3)
```

```
##
## Call:
## glm(formula = buyer_cf_notpos ~ treatment, family = binomial(link = "logit"),
##     data = data)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -0.2179     0.0822  -2.651  0.00804 **
## treatmentBlind      0.2256     0.1203   1.875  0.06081 .
## treatmentCF + DSR  -0.1141     0.1163  -0.981  0.32654
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2375.9  on 1724  degrees of freedom
## Residual deviance: 2367.8  on 1722  degrees of freedom
```

```
##   (675 observations deleted due to missingness)
## AIC: 2373.8
##
## Number of Fisher Scoring iterations: 4
```

INTERPRETATION: The intercept of -0.2179 represents the log-odds of the probability of buyer_cf_notpos being 1 (positive outcome) when the treatment is CF. When the log-odds are negative, it means that the odds of the event happening are less than 1 The coefficient for the Blind treatment group is 0.2256. This positive value suggests an increase in the log-odds of buyer_cf_notpos being 1 compared to the CF treatment. However, this difference is statistically insignificant at confidence level .95 (p = 0.06081). The coefficient for the CF + DSR treatment group is -0.1141. This negative value indicates a decrease in the log-odds of buyer_cf_notpos being 1 compared to the CF treatment. This difference is not statistically significant (p = 0.32654). There is therefore no statistical evidence of treatments influencing the probability to leave negative feedback

3

```
library(dplyr)
library(ggplot2)
setwd("C:/Users/julia/Downloads/Empirical data analysis/Data")
mlda = read.csv("MLDA RD.csv")
head(mlda)
```

```
##    agecell      all internal external   alcohol homicide  suicide      mva
## 1 19.06849 92.82540 16.61759 76.20782 0.6391380 16.31682 11.20371 35.82933
## 2 19.15068 95.10074 18.32768 76.77306 0.6774093 16.85996 12.19337 35.63926
## 3 19.23288 92.14429 18.91105 73.23324 0.8664426 15.21925 11.71581 34.20565
## 4 19.31507 88.42776 16.10177 72.32598 0.8673084 16.74282 11.27501 32.27896
## 5 19.39726 88.70494 17.36352 71.34141 1.0191631 14.94773 10.98431 32.65097
## 6 19.47945 90.19179 17.87211 72.31968 1.1713219 15.64282 12.16663 32.72144
##      drugs externalother
## 1 3.872425      8.534373
## 2 3.236511      8.655786
## 3 3.202071      8.513741
## 4 3.280689      8.258285
## 5 3.548198      8.417533
## 6 3.211689      7.972546
```

```
length(which(is.na(mlda)))
```

```
## [1] 18
```

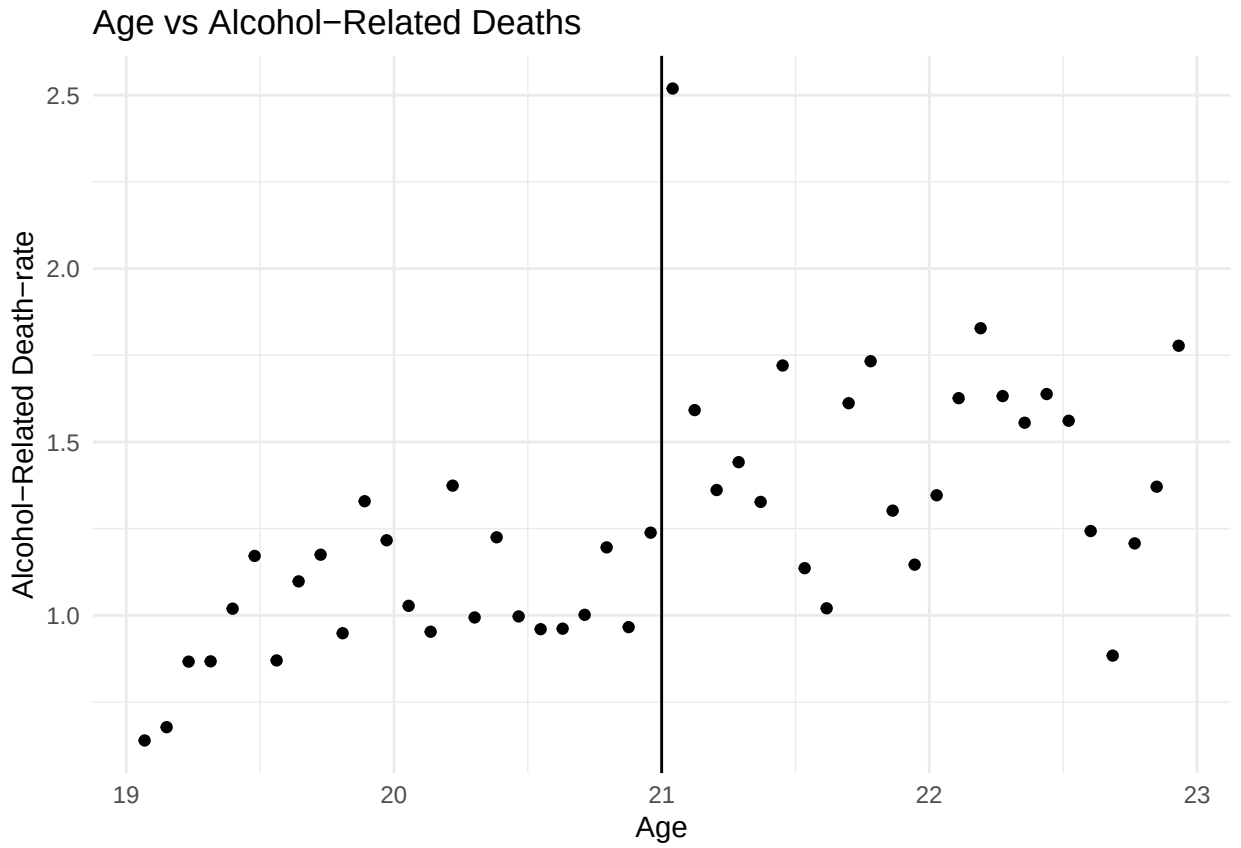
```
mlda <- na.omit(mlda)
```

We choose alcohol

1. Visualize the Data:

We create a graph with functions from **ggplot2**. As we chose alcohol, we plot a scatter plot **(geom_point())** with **"age"** on the x-axis (independent variable) and **"alcohol"** on the y-axis (dependent variable). With the **title** function we name the whole graph and the axes. The **theme_minimal()** function is used to apply a minimalistic theme to the plot.

```
ggplot(mlda, aes(x = agecell, y = alcohol)) +
  geom_point() +
  geom_vline(xintercept = 21) +
  labs(title = "Age vs Alcohol-Related Deaths", x = "Age", y = "Alcohol-Related Death-rate") +
  theme_minimal()
```



2. Estimate the Linear Regression Discontinuity (RD) Model:

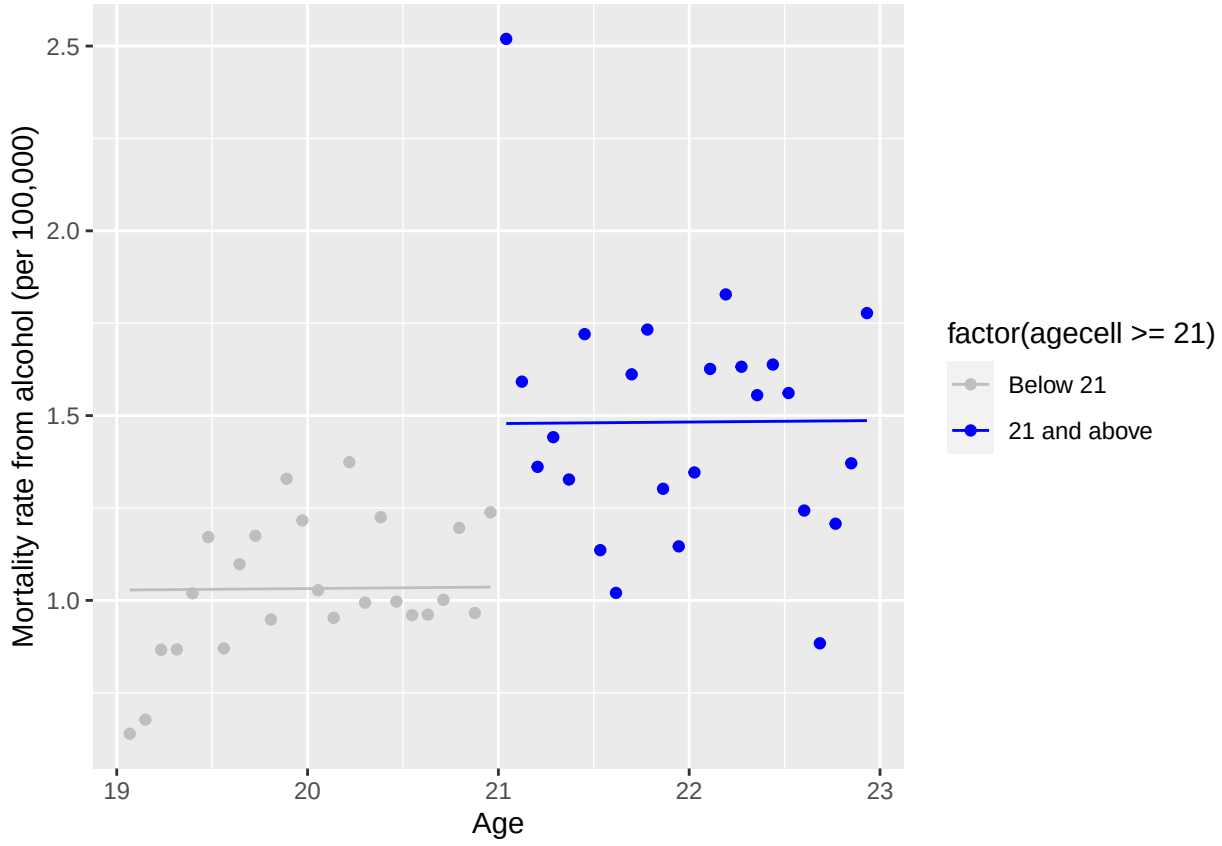
This model investigates the relationship between alcohol consumption (dependent variable) and age (independent variable), particularly focusing on the discontinuity at age 21. The coefficient for **I((agecell >= 21) * 1)** is 0.442357, significant at the 0.01 level, suggesting an increase in alcohol consumption at age 21.

```
linear_model <- lm(alcohol ~ agecell + I((agecell >= 21)*1), data = mlda)
summary(linear_model)
```

```
##
```

```
## Call:
## lm(formula = alcohol ~ agecell + I((agecell >= 21) * 1), data = mlda)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60156 -0.15476 -0.03526  0.14982  1.04062
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             0.950120   1.382869   0.687  0.49557
## agecell                 0.004097   0.069040   0.059  0.95294
## I((agecell >= 21) * 1) 0.442357   0.157224   2.814  0.00724 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2721 on 45 degrees of freedom
## Multiple R-squared:  0.4222, Adjusted R-squared:  0.3965
## F-statistic: 16.44 on 2 and 45 DF,  p-value: 4.369e-06
```

```
ggplot(mlda, aes(x = agecell, y = alcohol, color = factor(agecell >= 21))) +
  geom_point() +
  geom_line(aes(y = linear_model$fitted.values)) +
  scale_color_manual(values = c("grey", "blue"), labels = c("Below 21", "21 and above")) +
  ylab("Mortality rate from alcohol (per 100,000)") +
  xlab("Age")
```



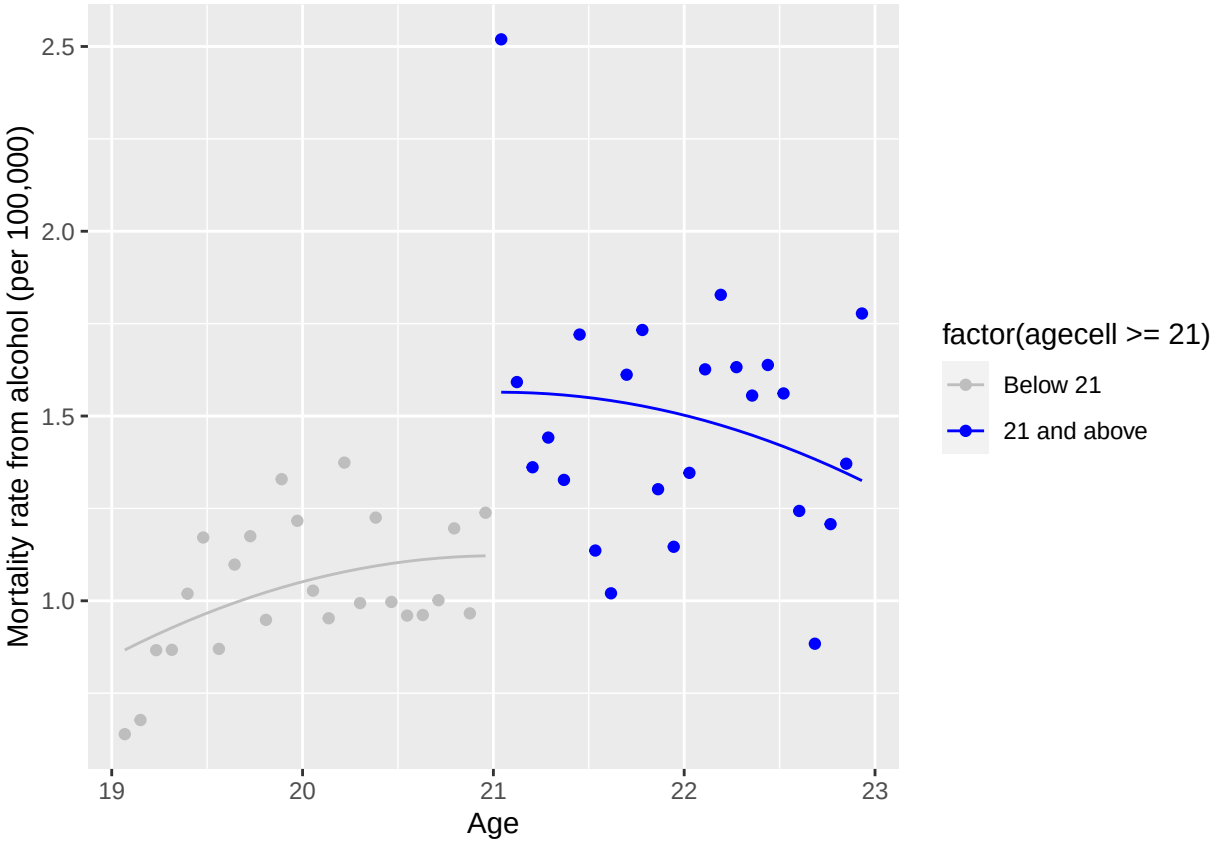
3. Estimate the RD Model with Quadratic Age Term:

The model investigates the same relationship as before, but includes a quadratic term (age squared). All coefficients have much better p-values than the previous model which indicates that this model is more statistically significant, since every independent variable is statistically significant at $p < 0.05$. The negative coefficient of the quadratic term suggests a diminishing effect of age on the response variable. The positive coefficient for **I((agecell >= 21) * 1)** indicates an increase in alcohol-related deaths at age 21, possibly due to the legal drinking age.

```
quadratic_model <- lm(alcohol ~ agecell + I(agecell^2) + I((agecell >= 21)*1), data = mlda)
summary(quadratic_model)
```

```
##
## Call:
## lm(formula = alcohol ~ agecell + I(agecell^2) + I((agecell >=
##     21) * 1), data = mlda)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52147 -0.15299 -0.03654  0.14912  0.95484
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -28.18142   14.48034  -1.946  0.05804 .
## agecell                  2.78671    1.37885   2.021  0.04939 *
## I(agecell^2)            -0.06625    0.03279  -2.020  0.04945 *
## I((agecell >= 21) * 1)   0.44236    0.15210   2.908  0.00568 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2633 on 44 degrees of freedom
## Multiple R-squared:  0.4712, Adjusted R-squared:  0.4352
## F-statistic: 13.07 on 3 and 44 DF,  p-value: 3.088e-06
```

```
ggplot(mlda, aes(x = agecell, y = alcohol, color = factor(agecell >= 21))) +
  geom_point() +
  geom_line(aes(y = quadratic_model$fitted.values)) +
  scale_color_manual(values = c("grey", "blue"), labels = c("Below 21", "21 and above")) +
  ylab("Mortality rate from alcohol (per 100,000)") +
  xlab("Age")
```



4. **Estimate a Model with Interaction and Non-linearity:**

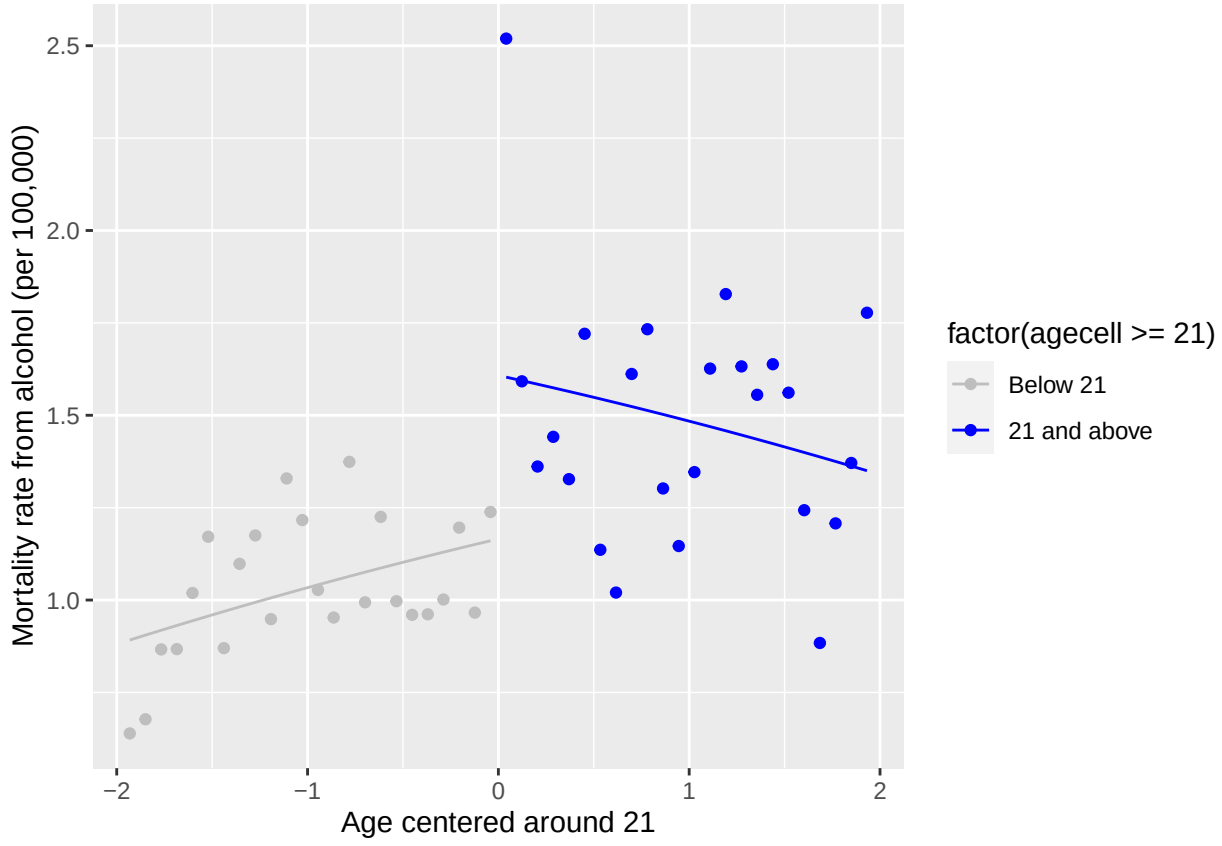
The model investigates the same relationship as before, but by making a new vector in the data set ($centered_age) we can put a non-linear term into the model i.e. **I(centered_age^2)**. We include it to explore how the effect of age on alcohol consumption changes around the age of 21 and whether this relationship is non-linear. The results indicate that the interaction term is not statistically significant, suggesting that the effect of age on alcohol consumption does not significantly change at the age of 21. The non-linear term is also not significant, implying that the relationship between age and alcohol consumption is approximately linear within the range of ages in the dataset.

```
mlda$centered_age <- mlda$agecell - mean(mlda$agecell)
interaction_model <- lm(alcohol ~ centered_age * I((agecell >= 21)*1) + I(centered_age^2),
data = mlda)
summary(interaction_model)
```

```
##
## Call:
## lm(formula = alcohol ~ centered_age * I((agecell >= 21) * 1) +
##     I(centered_age^2), data = mlda)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51374 -0.15661 -0.03285  0.15929  0.91618
##
## Coefficients:
```

```
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         1.16544    0.13863   8.407 1.27e-10 ***
## centered_age                        0.12139    0.27882   0.435  0.66547
## I((agecell >= 21) * 1)              0.44236    0.15352   2.881  0.00615 **
## I(centered_age^2)                  -0.01048    0.13283  -0.079  0.93746
## centered_age:I((agecell >= 21) * 1) -0.23459    0.54109  -0.434  0.66679
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2657 on 43 degrees of freedom
## Multiple R-squared:  0.4735, Adjusted R-squared:  0.4246
## F-statistic: 9.669 on 4 and 43 DF,  p-value: 1.143e-05
```

```
ggplot(mlda, aes(x = centered_age, y = alcohol, color = factor(agecell >= 21))) +
  geom_point() +
  geom_line(aes(y = interaction_model$fitted.values)) +
  scale_color_manual(values = c("grey", "blue"), labels = c("Below 21", "21 and above")) +
  ylab("Mortality rate from alcohol (per 100,000)") +
  xlab("Age centered around 21")
```



5. **How does the coefficient for the MLDA change in the different models? How can you interpret this?**

In the three models, the coefficient for MLDA (I((agecell >= 21) * 1)) remains consistent at approximately 0.44236, suggesting a stable effect of reaching the legal drinking age on alcohol-related deaths. This consistency across different model complexities, from linear to quadratic and interaction models, indicates the

robustness of the MLDA effect, irrespective of the inclusion of age non-linearities and interaction terms. The unchanged MLDA coefficient, despite varying complexities, supports a significant, direct impact of reaching the legal drinking age on alcohol-related death outcomes.

6. **How does that compare to the effect we found on all deaths in the lecture?**

The results for deaths from alcohol are signficiant. Similarly to what we found in class for the total deaths. It suggests that the alcohol restriction lift (legal drinking age) has an impact on the mortality rate.

4

Firstly, we load the necessary packages:

```
#install.packages("estimatr")
#install.packages("ggplot2")
#install.packages("dplyr")
library(estimatr)
library(ggplot2)
library(dplyr)
```

We then load in the dataset (make sure to set the wd correctly beforehand), and check if it loaded correctly using the `head()`function:

```
setwd("C:/Users/julia/Downloads/Empirical data analysis/Data")
guns_data <- read.csv("guns.csv")
head(guns_data)
```

```
##   year   vio  mur   rob incarc_rate   pb1064   pw1064   pm1029      pop
## 1   77 414.4 14.2  96.8          83 8.384873 55.12291 18.17441 3.780403
## 2   78 419.1 13.3  99.1          94 8.352101 55.14366 17.99408 3.831838
## 3   79 413.3 13.2 109.5         144 8.329575 55.13586 17.83934 3.866248
## 4   80 448.5 13.2 132.1         141 8.408386 54.91259 17.73420 3.900368
## 5   81 470.5 11.9 126.5         149 8.483435 54.92513 17.67372 3.918531
## 6   82 447.7 10.6 112.0         183 8.514000 54.89621 17.51052 3.925229
##     avginc    density stateid shall
## 1 9.563149 0.07455240       1     0
## 2 9.932000 0.07556673       1     0
## 3 9.877028 0.07624532       1     0
## 4 9.541428 0.07682881       1     0
## 5 9.548351 0.07718658       1     0
## 6 9.478919 0.07731851       1     0
```

Regression of log(vio) against shall:

```
# Adding a log_vio column to the data column
guns_data$log_vio <- log(guns_data$vio)

# Running the regression model
```

```r
model_1 <- lm_robust(log_vio ~ shall, data = guns_data, clusters = stateid)


# Printing the summary
summary(model_1)
```

```
##
## Call:
## lm_robust(formula = log_vio ~ shall, data = guns_data, clusters = stateid)
##
## Standard error type:  CR2
##
## Coefficients:
##             Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper    DF
## (Intercept)    6.135    0.07911  77.545 6.656e-48   5.9754    6.294 43.07
## shall         -0.443    0.16026  -2.764 9.878e-03  -0.7709   -0.115 28.63
##
## Multiple R-squared:  0.08664 ,   Adjusted R-squared:  0.08586
## F-statistic:  7.64 on 1 and 50 DF,  p-value: 0.007973
```

We can see that both coefficient estimates are statistically significant, moreover, there exists a negative relationship between the estimate for shall and logvio.

Regression of log(vio) against shall, incarc rate, density, avginc, pop, pb1064, pw1064, and pm1029:

```r
# Running the regression with additional control variables
model_2 <- lm_robust(log_vio ~ shall + incarc_rate + density + avginc + pop + pb1064 + pw1064 + pm1029,

# Printing the summary
summary(model_2)
```

```
##
## Call:
## lm_robust(formula = log_vio ~ shall + incarc_rate + density +
##     avginc + pop + pb1064 + pw1064 + pm1029, data = guns_data,
##     clusters = stateid)
##
## Standard error type:  CR2
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)   CI Lower  CI Upper     DF
## (Intercept)  2.981738  2.3408107  1.27381 0.216047 -1.8732515  7.836728 21.965
## shall       -0.368387  0.1200544 -3.06850 0.004571 -0.6137078 -0.123066 29.606
## incarc_rate  0.001613  0.0008284  1.94664 0.096763 -0.0003867  0.003612  6.364
## density      0.026688  0.0569029  0.46902 0.670951 -0.1541043  0.207481  3.009
## avginc       0.001205  0.0265682  0.04536 0.964330 -0.0546954  0.057106 17.635
## pop          0.042710  0.0148866  2.86901 0.033456  0.0048894  0.080530  5.204
## pb1064       0.080853  0.0816292  0.99049 0.330907 -0.0867960  0.248501 26.462
```

```
## pw1064       0.031201  0.0367887  0.84810 0.404077 -0.0443977  0.106799 26.157
## pm1029       0.008871  0.0398255  0.22274 0.826103 -0.0744634  0.092205 19.072
##
## Multiple R-squared:  0.5643 ,    Adjusted R-squared:  0.5613
## F-statistic: 4.686 on 8 and 50 DF,  p-value: 0.000254
```

Apart from the coefficient estimates for shall and pop, the results are not statistically significant at significance level = 0.05. There exists a positive relationship between the estimate for pop and logvio, and a negative one for shall.

Interpret the coefficient on shall in regressions (1) and (2). Are the estimates large or small in a real–world sense?

As mentioned in the first model, both coefficient estimates are statistically significant, moreover, there exists a negative relationship between the estimate for shall and logvio. This means that allowing citizens to carry concealed weapons decreases the number of violent crimes committed by 44.3%. This is very large in a real world sense.

In the second model, as previously mentioned; Apart from the coefficient estimates for shall (which is now slightly lower, at -0.37) and pop, the results are not statistically significant at significance level = 0.05. This means that allowing citizens to carry concealed weapons decreases the number of violent crimes committed by 37%. This is very large in a real world sense. And, that there exists a positive relationship between the estimate for pop and logvio, and a negative one for shall. A 1-unit larger population causes an increase of violent crimes commited by 4.27%, holding other variables constant. Also large IRL terms. It is also important to note that this model provides a much better adjusted R squared (>.56), indicating a much better model fit.

Does adding the control variables in regression (2) change the estimated effect of a shall issue (or shall carry) law in the regression?

Yes, Its effect was reduced by the addition of other variables. As mentioned in the point above, the estimate remains significant, but it suggest only a 36.83% decrease in violence when carrying concealed weapons is allowed.

Do the results change when you add time and state fixed effects? If so, which set of regression results is more credible, and why?

```
guns_data$year = as.factor(guns_data$year)
guns_data$stateid = as.factor(guns_data$stateid)

# Running the regression with fixed effects
model_3 <- lm_robust(log_vio ~ shall + incarc_rate + density + avginc + pop + pb1064 + pw1064 + pm1029 +

# Printing the summary
summary(model_3)
```

```
##
## Call:
## lm_robust(formula = log_vio ~ shall + incarc_rate + density +
##     avginc + pop + pb1064 + pw1064 + pm1029 + year + stateid,
##     data = guns_data, clusters = stateid)
##
## Standard error type:  CR2
##
## Coefficients:
##               Estimate Std. Error  t value  Pr(>|t|)   CI Lower   CI Upper
## (Intercept)  3.972e+00  1.1166265  3.55714 3.306e-03  1.5695653  6.3744296
## shall       -2.799e-02  0.0417197 -0.67099 5.070e-01 -0.1129422  0.0569551
## incarc_rate  7.599e-05  0.0002141  0.35489 7.295e-01 -0.0003965  0.0005484
## density     -9.155e-02  0.1321412 -0.69286 5.357e-01 -0.4991751  0.3160653
## avginc       9.586e-04  0.0172810  0.05547 9.565e-01 -0.0360447  0.0379620
## pop         -4.754e-03  0.0175198 -0.27138 8.011e-01 -0.0558240  0.0463151
## pb1064       2.919e-02  0.0514145  0.56766 5.872e-01 -0.0912739  0.1496462
## pw1064       9.250e-03  0.0245167  0.37730 7.096e-01 -0.0416402  0.0601404
## pm1029       7.333e-02  0.0552650  1.32680 2.036e-01 -0.0440207  0.1906715
## year78       5.853e-02  0.0164152  3.56536 8.120e-04  0.0255538  0.0914985
## year79       1.639e-01  0.0249317  6.57592 2.837e-08  0.1138644  0.2140328
## year80       2.171e-01  0.0342239  6.34283 6.759e-08  0.1483109  0.2858410
## year81       2.173e-01  0.0403427  5.38524 2.134e-06  0.1361453  0.2983650
## year82       1.946e-01  0.0484854  4.01426 2.182e-04  0.0970339  0.2922317
## year83       1.586e-01  0.0620474  2.55684 1.425e-02  0.0334473  0.2838426
## year84       1.930e-01  0.0804623  2.39849 2.154e-02  0.0300410  0.3559356
## year85       2.445e-01  0.0964208  2.53552 1.602e-02  0.0484884  0.4404645
## year86       3.241e-01  0.1139181  2.84494 7.876e-03  0.0915702  0.5566106
## year87       3.244e-01  0.1309392  2.47722 1.961e-02  0.0560355  0.5926946
## year88       3.867e-01  0.1465135  2.63963 1.388e-02  0.0854872  0.6879952
## year89       4.422e-01  0.1611064  2.74486 1.119e-02  0.1099656  0.7744630
## year90       5.430e-01  0.2053550  2.64443 1.458e-02  0.1179348  0.9681610
## year91       5.959e-01  0.2142101  2.78206 1.082e-02  0.1518913  1.0400000
## year92       6.275e-01  0.2277108  2.75576 1.158e-02  0.1551045  1.0999299
## year93       6.497e-01  0.2358862  2.75447 1.173e-02  0.1598579  1.1396251
## year94       6.354e-01  0.2450950  2.59254 1.687e-02  0.1261720  1.1446655
## year95       6.277e-01  0.2545151  2.46619 2.220e-02  0.0989030  1.1564633
## year96       5.713e-01  0.2656975  2.15035 4.321e-02  0.0191248  1.1235600
## year97       5.501e-01  0.2738410  2.00889 5.745e-02 -0.0190332  1.1192640
## year98       4.933e-01  0.2871975  1.71760 1.006e-01 -0.1039758  1.0905569
## year99       4.329e-01  0.2989902  1.44780 1.626e-01 -0.1894387  1.0551942
## stateid2    -1.474e-01  0.1361603 -1.08254 2.993e-01 -0.4425290  0.1477319
## stateid4     1.394e-01  0.1669806  0.83457 4.388e-01 -0.2787911  0.5575048
## stateid5    -1.565e-01  0.0952384 -1.64331 1.657e-01 -0.4070214  0.0940095
## stateid6     5.385e-01  0.5016158  1.07361 3.314e-01 -0.7455749  1.8226513
## stateid8    -1.261e-01  0.2673082 -0.47157 6.489e-01 -0.7342284  0.4821207
## stateid9    -1.074e-01  0.3054950 -0.35145 7.349e-01 -0.8193559  0.6046247
## stateid10    3.334e-02  0.1267105  0.26315 7.996e-01 -0.2631345  0.3298224
## stateid11    1.988e+00  1.3332004  1.49141 2.188e-01 -1.9062476  5.8829342
## stateid12    8.598e-01  0.2569968  3.34562 7.019e-03  0.2902905  1.4293355
## stateid13    3.073e-02  0.0571574  0.53772 6.030e-01 -0.0972958  0.1587653
## stateid15   -1.010e+00  0.5460900 -1.84964 1.240e-01 -2.4169583  0.3968133
## stateid16   -6.831e-01  0.2495535 -2.73729 3.765e-02 -1.3101544 -0.0560441
## stateid17    4.953e-01  0.2446165  2.02473 6.584e-02 -0.0379566  1.0285200
```

```
## stateid18   -2.361e-01  0.2489740 -0.94825 3.751e-01 -0.8266900  0.3545119
## stateid19   -7.003e-01  0.2688948 -2.60446 4.773e-02 -1.3903875 -0.0102631
## stateid20   -2.824e-01  0.2020330 -1.39797 2.166e-01 -0.7898989  0.2250284
## stateid21   -4.660e-01  0.2222564 -2.09661 8.274e-02 -1.0152410  0.0832708
## stateid22    2.727e-01  0.0859163  3.17418 1.761e-02  0.0656219  0.4798063
## stateid23   -1.142e+00  0.2966888 -3.84811 1.001e-02 -1.8839090 -0.3994716
## stateid24    4.707e-01  0.1432108  3.28666 4.391e-03  0.1683519  0.7730194
## stateid25    2.664e-01  0.3296973  0.80812 4.469e-01 -0.5212524  1.0541225
## stateid26    2.886e-01  0.2213182  1.30402 2.155e-01 -0.1910628  0.7682695
## stateid27   -6.150e-01  0.2860691 -2.14989 7.215e-02 -1.3040965  0.0740625
## stateid28   -5.105e-01  0.1713579 -2.97916 1.200e-02 -0.8857003 -0.1353047
## stateid29    1.584e-01  0.1885565  0.83995 4.344e-01 -0.3075768  0.6243317
## stateid30   -1.027e+00  0.2146720 -4.78229 4.055e-03 -1.5664364 -0.4868124
## stateid31   -5.439e-01  0.2316086 -2.34824 6.409e-02 -1.1336567  0.0459115
## stateid32    3.236e-01  0.1935423  1.67220 1.229e-01 -0.1028826  0.7501649
## stateid33   -1.309e+00  0.3246329 -4.03134 5.762e-03 -2.0876740 -0.5297383
## stateid34    2.104e-01  0.2968123  0.70893 4.986e-01 -0.4743885  0.8952251
## stateid35    2.877e-01  0.1418102  2.02848 8.460e-02 -0.0519134  0.6272324
## stateid36    6.948e-01  0.3531583  1.96740 8.299e-02 -0.1129602  1.5025710
## stateid37   -7.169e-02  0.0774960 -0.92504 3.728e-01 -0.2401433  0.0967693
## stateid38   -2.073e+00  0.2353527 -8.80677 8.875e-05 -2.6415548 -1.5038410
## stateid39   -8.862e-02  0.2909897 -0.30455 7.665e-01 -0.7301089  0.5528683
## stateid40   -9.372e-02  0.1077802 -0.86956 4.212e-01 -0.3641148  0.1766730
## stateid41    3.654e-02  0.2595560  0.14076 8.928e-01 -0.6037360  0.6768077
## stateid42   -1.519e-01  0.3331889 -0.45575 6.575e-01 -0.8853952  0.5816917
## stateid44   -2.263e-01  0.2871372 -0.78802 4.682e-01 -0.9762737  0.5237361
## stateid45    2.759e-01  0.0762759  3.61736 1.565e-02  0.0789839  0.4728504
## stateid46   -1.192e+00  0.1869004 -6.37962 9.107e-04 -1.6576534 -0.7270521
## stateid47    9.298e-02  0.1461496  0.63622 5.430e-01 -0.2460156  0.4319826
## stateid48    1.200e-01  0.3108442  0.38614 7.108e-01 -0.6138524  0.8539094
## stateid49   -7.211e-01  0.2711418 -2.65958 3.028e-02 -1.3526706 -0.0895743
## stateid50   -1.326e+00  0.3008329 -4.40845 4.603e-03 -2.0635640 -0.5888497
## stateid51   -5.622e-01  0.1368802 -4.10690 4.904e-04 -0.8465546 -0.2777531
## stateid53   -1.219e-01  0.2529472 -0.48206 6.408e-01 -0.6902117  0.4463402
## stateid54   -9.411e-01  0.2675085 -3.51790 1.622e-02 -1.6232896 -0.2588466
## stateid55   -8.292e-01  0.2573630 -3.22184 1.808e-02 -1.4588163 -0.1995497
## stateid56   -6.536e-01  0.2549140 -2.56404 4.115e-02 -1.2712990 -0.0359218
##                  DF
## (Intercept) 13.549
## shall       32.305
## incarc_rate 10.786
## density      3.176
## avginc      14.249
## pop          3.566
## pb1064       7.335
## pw1064      21.665
## pm1029      15.688
## year78      49.911
## year79      49.701
## year80      49.293
## year81      48.107
## year82      45.949
## year83      42.216
## year84      37.580
```

```
## year85      33.825
## year86      30.411
## year87      27.741
## year88      25.839
## year89      24.358
## year90      22.707
## year91      22.164
## year92      21.865
## year93      21.482
## year94      21.314
## year95      21.340
## year96      21.208
## year97      21.205
## year98      20.996
## year99      20.709
## stateid2    12.591
## stateid4     5.481
## stateid5     4.647
## stateid6     5.070
## stateid8     8.674
## stateid9     7.542
## stateid10    7.386
## stateid11    3.549
## stateid12   10.418
## stateid13    9.625
## stateid15    4.963
## stateid16    5.413
## stateid17   11.947
## stateid18    6.892
## stateid19    5.028
## stateid20    5.420
## stateid21    5.765
## stateid22    6.399
## stateid23    5.502
## stateid24   16.864
## stateid25    6.663
## stateid26   12.603
## stateid27    6.418
## stateid28   11.491
## stateid29    5.766
## stateid30    5.398
## stateid31    5.162
## stateid32   10.887
## stateid33    6.530
## stateid34    7.976
## stateid35    6.593
## stateid36    8.395
## stateid37   12.258
## stateid38    6.322
## stateid39   10.858
## stateid40    5.444
## stateid41    5.807
## stateid42   10.976
## stateid44    4.749
```

```
## stateid45    4.928
## stateid46    5.603
## stateid47    7.740
## stateid48    7.054
## stateid49    7.566
## stateid50    5.958
## stateid51   21.317
## stateid53    9.427
## stateid54    5.136
## stateid55    6.004
## stateid56    6.253
##
## Multiple R-squared:  0.9562 ,    Adjusted R-squared:  0.953
## F-statistic:    NA on 80 and 50 DF,  p-value: NA
```

Once again, our results have changed. The **shall** estimate is now not only very small, but also statistically insignificant. We can not make inferences on violence based on it. The adjusted R-squared, however, is incredibly high, meaning this model is a very good fit, and therefore the most credible out of the three models we created. The good fit of the data can be attributed to the inclusion of time and state fixed effects: By including fixed effects, the model accounts for unobserved heterogeneity across states and over time. This helps reduce omitted variable bias and provides more credible estimates of the relationships between the variables of interest.

Based on your analysis, what conclusions would you draw about the effects of concealed weapon laws on violent crime rates?

The most robust and credible model, with time and state fixed effects, does not provide sufficient evidence to support a conclusion about the impact of concealed weapon laws on violent crime rates. Therefore, the relationship between shall-carry laws and violent crime appears to be contingent on other factors, and no definitive conclusions can be drawn based on the analysis.