Direct Democracy Analysis

Lorrel Plimier

```
# load packages
library(data.table)
library(foreign)
library(lmtest)
```

```
## Loading required package: zoo
```

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

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

```
library(multiwayvcov)
library(sandwich)
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(pwr)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------- tidyverse 1.3.0 -
```

```
## v tibble  3.0.0     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## v purrr   0.3.3
```

```
## -- Conflicts ------------------------------------------------------------------- tidyverse_conflicts() -
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()
```

```
# library(cobalt)
```

Import Data

```
#Load the raw survey data.
# raw <- fread("rrrockTheVote_final_2020_04_05_16.29.csv")
raw <- fread("rrrockTheVote_final_2020_04_12_22.03P.csv")
#Remove the first two rows that have descriptive data.
raw <- raw[-c(1,2)]
```

```r
#Load the appended data (with columns from python API)
appended <- fread("apr12_cleaned_appended.csv", select = c("responseId", "displayOrder",
                                                           "timeParole", "timeAmb", "timeClinic",
                                                           "state", "country"))

pilot_raw <- fread("rrrockTheVote_pilot_2020_03_09_14.44.csv")
pilot_raw <- pilot_raw[-c(1,2)]

nrow(raw)
```

```
## [1] 441
```

Clean Data and Rename Variables

The following chunk saves and renames only the data that we intend to use. [So far, there is no timing or order of presentation data here.]

```r
#Keep and rename just the variables we want.
final_sm <- raw[, list(progress = as.numeric(Progress),
                       disclaimerTime = as.numeric(time.disclaimer_PageSubmit),
                       age = as.numeric(age),
                       gender,
                       education,
                       income,
                       voting = votingHabits,
                       party,
                       partyDetails = as.factor(party_4_TEXT),
                       partyStrength = party_strength,
                       parole_Control = as.numeric(parole_0.0_1),
                       ambulance_Control = as.numeric(ambulance_0.0_1),
                       clinic_Control = as.numeric(clinic_0.0_1),
                       parole_Tx = as.numeric(parole_1.0_1),
                       ambulance_Tx = as.numeric(ambulance_1.0_1),
                       clinic_Tx = as.numeric(clinic_1.0_1),
                       randomizer = as.numeric(rand),
                       responseId = ResponseId)]

#Merge the appended columns from python API.
final_sm <- merge(final_sm,appended, by = "responseId")
summary(final_sm)
```

```
##   responseId          progress      disclaimerTime          age
##  Length:441         Min.   :  2.00   Min.   :   0.590   Min.   :17.0
##  Class :character   1st Qu.: 98.00   1st Qu.:   3.265   1st Qu.:31.0
##  Mode  :character   Median :100.00   Median :   5.183   Median :38.0
##                     Mean   : 85.14   Mean   :  11.498   Mean   :39.1
##                     3rd Qu.:100.00   3rd Qu.:   8.097   3rd Qu.:48.0
##                     Max.   :100.00   Max.   :1454.242   Max.   :83.0
##                                      NA's   :45         NA's   :38
##     gender           education           income
##  Length:441         Length:441         Length:441
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
```

```
##
##
##     voting             party                                partyDetails
##  Length:441         Length:441                                    :411
##  Class :character   Class :character   Leftist                     :  3
##  Mode  :character   Mode  :character   Progressive                 :  3
##                                        Democratic Socialist        :  2
##                                        Libertarian                 :  2
##                                        Democratic Confereralist:  1
##                                        (Other)                     : 19
##  partyStrength      parole_Control   ambulance_Control clinic_Control
##  Length:441         Min.   :-50.00   Min.   :-50.000   Min.   :-50.00
##  Class :character   1st Qu.:-50.00   1st Qu.:-29.500   1st Qu.: 15.75
##  Mode  :character   Median :-37.00   Median : -8.000   Median : 28.50
##                     Mean   :-28.79   Mean   : -3.005   Mean   : 22.88
##                     3rd Qu.:-21.00   3rd Qu.: 23.500   3rd Qu.: 45.00
##                     Max.   : 50.00   Max.   : 50.000   Max.   : 50.00
##                     NA's   :258      NA's   :254       NA's   :257
##    parole_Tx       ambulance_Tx       clinic_Tx        randomizer
##  Min.   :-50.00   Min.   :-50.000   Min.   :-50.00   Min.   :0.000959
##  1st Qu.:-50.00   1st Qu.:-39.750   1st Qu.: -2.00   1st Qu.:0.234490
##  Median :-41.00   Median :-17.500   Median : 29.50   Median :0.483773
##  Mean   :-28.91   Mean   : -8.532   Mean   : 19.16   Mean   :0.488604
##  3rd Qu.:-20.00   3rd Qu.: 21.750   3rd Qu.: 47.00   3rd Qu.:0.734773
##  Max.   : 50.00   Max.   : 50.000   Max.   : 50.00   Max.   :0.995190
##  NA's   :288      NA's   :287       NA's   :289
##   displayOrder     timeParole          timeAmb          timeClinic
##  Min.   :123.0   Min.   :   0.000   Min.   :  0.000   Min.   :   0.000
##  1st Qu.:132.0   1st Qu.:   9.371   1st Qu.:  9.511   1st Qu.:   7.825
##  Median :213.0   Median :  39.789   Median : 43.873   Median :  34.468
##  Mean   :223.6   Mean   :  57.143   Mean   : 56.321   Mean   :  49.065
##  3rd Qu.:312.0   3rd Qu.:  64.006   3rd Qu.: 75.932   3rd Qu.:  59.995
##  Max.   :321.0   Max.   :3302.782   Max.   :521.231   Max.   :1568.080
##  NA's   :46
##     state              country
##  Length:441         Length:441
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
##
```

```
pilot_sm <- pilot_raw[, list(age = as.numeric(age_1),
                             gender,
                             education,
                             income,
                             voting = votingHabits,
                             party = Q25,
                             partyDatails = as.factor(Q25_4_TEXT),
                             partyStrength = Q28,
                             parole_Control = as.numeric(parole_0.0_1),
                             clinic_Control = as.numeric(clinic_0.0_1),
                             ambulance_Control = as.numeric(ambulance_0.0_1),
```

```
                              parole_Tx = as.numeric(parole_0.2_1),
                              clinic_Tx = as.numeric(clinic_0.2_1),
                              ambulance_Tx = as.numeric(ambulance_0.2_1),
                              randomizer = as.numeric(rand),
                              responseId = ResponseId)]

pilot_sm[, treat := ifelse(randomizer <= 0.5, 0, 1)]

pilot_sm[, parole_Support := ifelse(is.na(parole_Control), parole_Tx, parole_Control)]
pilot_sm[, ambulance_Support := ifelse(is.na(ambulance_Control), ambulance_Tx, ambulance_Control)]
pilot_sm[, clinic_Support := ifelse(is.na(clinic_Control), clinic_Tx, clinic_Control)]

pilot_sm
```

- Create the treatment (`treat`) variable using `rand`, which we used in Qualtics for assignment; `rand` <= 0.5 to control, `rand` > 0.5 to treatment.
- Drop the subjects who never made it into treatment or control (i.e., never made it past the demographics data and disclaimer page.) This is a value of 13% progress in Qualtics.
- Combine the treatment and control outcome for each question type. You can subsequently identify who was in T/C by looking at the `treat` variable.
- Create an attrition variable for people who dropped out after being given the first substantive question (after submitting the disclaimer page.) This shows up as NA for the three new outcome variables.
- Make `party_strength` easier to use by shortening answers to `lean`, `moderate`, `strong`.

```
#Create treatment variable.
final_sm[, treat := ifelse(randomizer <= 0.5, 0, 1)]
#Drop the subjects who never submitted disclaimer page. (random attritors)
final_sm <- final_sm[is.na(disclaimerTime) != T]
#Convert the outcome variable to combine T/C.
final_sm[, parole_Support := ifelse(is.na(parole_Control), parole_Tx, parole_Control)]
final_sm[, ambulance_Support := ifelse(is.na(ambulance_Control), ambulance_Tx, ambulance_Control)]
final_sm[, clinic_Support := ifelse(is.na(clinic_Control), clinic_Tx, clinic_Control)]
#Add attrition variable for each of the six questions
final_sm[, parole_Attrite := ifelse(is.na(parole_Support), 1, 0)]
final_sm[, ambulance_Attrite := ifelse(is.na(ambulance_Support), 1, 0)]
final_sm[, clinic_Attrite := ifelse(is.na(clinic_Support), 1, 0)]
#Shorten/order partyStrength, education, income and voting habits.
final_sm[, partyStrength := ifelse(partyStrength == "I lean '${q://QID75/ChoiceGroup/SelectedChoicesTex
                                   ifelse(partyStrength == "I identify strongly as a '${q://QID75/Choi
final_sm[, education := ifelse(education == "Have not completed high school","1: < high school",
                               ifelse(education == "Obtained a high school degree", "2: high school",
                                      ifelse(education == "Pursued some college studies", "3: some coll
                                             ifelse(education == "Obtained a college degree", "4: colleg
                                                    ifelse(education == "Pursued some post-graduate stu
                                                           ifelse(education == "Obtained a post-graduat

final_sm[, voting := ifelse(voting == "I have never voted for public office", "1: never vote",
                            ifelse(voting == "I vote in presidential elections", "2: every 4 years",
                                   ifelse(voting == "I vote in presidential and midterm elections", "3:
                                          ifelse(voting == "I vote in presidential, midterm, and local
final_sm[, income := ifelse(income == "less than $50,000 per year","1: under $50k",
                            ifelse(income == "more than $50,000 but less than $100,000 per year", "2: $
                                   ifelse(income == "more than $100,000 per year","3: over $100k",""))))
```

```r
#Create bins for timing variables (for later boxplots)
final_sm[,timeParole_bins := ifelse(timeParole <= 20, "1: under 20",
                                    ifelse(timeParole > 20 & timeParole <= 40, "2: 20-40",
                                           ifelse(timeParole > 40 & timeParole <= 60, "3: 40-60",
                                                  ifelse(timeParole > 60, "4: over 60",""))))]
final_sm[,timeAmb_bins := ifelse(timeAmb <= 20, "1: under 20",
                                    ifelse(timeAmb > 20 & timeAmb <= 40, "2: 20-40",
                                           ifelse(timeAmb > 40 & timeAmb <= 60, "3: 40-60",
                                                  ifelse(timeAmb > 60, "4: over 60",""))))]
final_sm[,timeClinic_bins := ifelse(timeClinic <= 20, "1: under 20",
                                    ifelse(timeClinic > 20 & timeClinic <= 40, "2: 20-40",
                                           ifelse(timeClinic > 40 & timeClinic <= 60, "3: 40-60",
                                                  ifelse(timeClinic > 60, "4: over 60",""))))]

#Convert ordered variables to factors.
final_sm[, partyStrength := as.factor(partyStrength)]
final_sm[, education := as.factor(education)]
final_sm[, voting := as.factor(voting)]
final_sm[, income := as.factor(income)]
final_sm[, party := as.factor(party)]
final_sm[, gender := as.factor(gender)]
final_sm[, timeParole_bins := as.factor(timeParole_bins)]
final_sm[, timeAmb_bins := as.factor(timeAmb_bins)]
final_sm[, timeClinic_bins := as.factor(timeClinic_bins)]

summary(final_sm)
```

```
##   responseId           progress      disclaimerTime          age
##  Length:396         Min.   : 15.00   Min.   :   0.590   Min.   :18.00
##  Class :character   1st Qu.:100.00   1st Qu.:   3.265   1st Qu.:31.00
##  Mode  :character   Median :100.00   Median :   5.183   Median :38.00
##                     Mean   : 93.95   Mean   :  11.498   Mean   :39.28
##                     3rd Qu.:100.00   3rd Qu.:   8.097   3rd Qu.:48.00
##                     Max.   :100.00   Max.   :1454.242   Max.   :83.00
##                                                         NA's   :1
##                          gender                  education
##                             :  1                       :  1
##  Female                     :247   1: < high school    :  1
##  Male                       :141   2: high school      : 14
##  Nonbinary/Transgender/Other:  5   3: some college     : 64
##  Prefer not to say          :  2   4: college degree   :106
##                                    5: some post-grad   : 42
##                                    6: graduate degree:168
##               income                   voting            party
##                  :  1                     :  1               :  1
##  1: under $50k   : 70   1: never vote    : 33   Democrat    :233
##  2: $50k to $100k:115   2: every 4 years: 46   Independent:102
##  3: over $100k   :210   3: every 2 years: 66   Other       : 27
##                         4: always vote  :250   Republican  : 33
##
##
##                    partyDetails  partyStrength parole_Control
##                         :369   lean    : 87   Min.   :-50.00
##  Leftist                :  3   moderate:136   1st Qu.:-50.00
```

```
##  Progressive             :  3   strong  :173   Median :-37.00
##  Democratic Socialist    :  2                  Mean   :-28.79
##  Libertarian             :  2                  3rd Qu.:-21.00
##  Democratic Confereralist:  1                  Max.   : 50.00
##  (Other)                 : 16                  NA's   :213
##  ambulance_Control clinic_Control     parole_Tx       ambulance_Tx
##  Min.   :-50.000   Min.   :-50.00   Min.   :-50.00   Min.   :-50.000
##  1st Qu.:-29.500   1st Qu.: 15.75   1st Qu.:-50.00   1st Qu.:-39.750
##  Median : -8.000   Median : 28.50   Median :-41.00   Median :-17.500
##  Mean   : -3.005   Mean   : 22.88   Mean   :-28.91   Mean   : -8.532
##  3rd Qu.: 23.500   3rd Qu.: 45.00   3rd Qu.:-20.00   3rd Qu.: 21.750
##  Max.   : 50.000   Max.   : 50.00   Max.   : 50.00   Max.   : 50.000
##  NA's   :209       NA's   :212      NA's   :243      NA's   :242
##    clinic_Tx        randomizer        displayOrder     timeParole
##  Min.   :-50.00   Min.   :0.001983   Min.   :123.0   Min.   :   0.00
##  1st Qu.: -2.00   1st Qu.:0.223350   1st Qu.:132.0   1st Qu.:  18.39
##  Median : 29.50   Median :0.473187   Median :213.0   Median :  43.93
##  Mean   : 19.16   Mean   :0.481285   Mean   :223.6   Mean   :  63.64
##  3rd Qu.: 47.00   3rd Qu.:0.723002   3rd Qu.:312.0   3rd Qu.:  68.00
##  Max.   : 50.00   Max.   :0.995190   Max.   :321.0   Max.   :3302.78
##  NA's   :244                         NA's   :1
##     timeAmb         timeClinic         state             country
##  Min.   :  0.00   Min.   :   0.00   Length:396         Length:396
##  1st Qu.: 19.67   1st Qu.:  18.10   Class :character   Class :character
##  Median : 48.79   Median :  36.99   Mode  :character   Mode  :character
##  Mean   : 62.72   Mean   :  54.64
##  3rd Qu.: 79.87   3rd Qu.:  63.19
##  Max.   :521.23   Max.   :1568.08
##
##      treat        parole_Support   ambulance_Support clinic_Support
##  Min.   :0.0000   Min.   :-50.00   Min.   :-50.000   Min.   :-50.00
##  1st Qu.:0.0000   1st Qu.:-50.00   1st Qu.:-33.000   1st Qu.:  9.00
##  Median :0.0000   Median :-39.00   Median :-11.000   Median : 29.00
##  Mean   :0.4621   Mean   :-28.84   Mean   : -5.501   Mean   : 21.19
##  3rd Qu.:1.0000   3rd Qu.:-20.75   3rd Qu.: 22.000   3rd Qu.: 46.00
##  Max.   :1.0000   Max.   : 50.00   Max.   : 50.000   Max.   : 50.00
##                   NA's   :60       NA's   :55        NA's   :60
##  parole_Attrite   ambulance_Attrite clinic_Attrite      timeParole_bins
##  Min.   :0.0000   Min.   :0.0000    Min.   :0.0000   1: under 20:105
##  1st Qu.:0.0000   1st Qu.:0.0000    1st Qu.:0.0000   2: 20-40   : 74
##  Median :0.0000   Median :0.0000    Median :0.0000   3: 40-60   : 86
##  Mean   :0.1515   Mean   :0.1389    Mean   :0.1515   4: over 60 :131
##  3rd Qu.:0.0000   3rd Qu.:0.0000    3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000    Max.   :1.0000
##
##       timeAmb_bins    timeClinic_bins
##  1: under 20:102   1: under 20:111
##  2: 20-40   : 62   2: 20-40   :100
##  3: 40-60   : 80   3: 40-60   : 75
##  4: over 60 :152   4: over 60 :110
##
##
##
```

Attrition Analysis

Overall, it looks like we have a greater percentage of attritors for treatment than for control. (18-19% vs. 10-14% depending on the question) This is not surprising to me given that some of the information in the treatment was specific to the US, e.g., names of organizations, etc. I got direct feedback from some that they tried, but couldn't answer the questions because they were not US citizens and didn't live in the US. I would think those people would attrite. I'm not sure what else about the treatment would cause people to attrite other than that it becomes more obvious that there is partisanship in the endorsers/contributors and this may make people feel uncomfortable and want to quit. The text of the measures themselves (ie, the control) is much more dry and unobjectionable on its face. The attrition is not significantly different between the two groups.

```
final_sm[, .(count = .N), keyby = .(parole_Attrite,treat)]
```

```
final_sm[, .(count = .N), keyby = .(ambulance_Attrite,treat)]
```

```
final_sm[, .(count = .N), keyby = .(clinic_Attrite,treat)]
```

```
control_attriters <- final_sm[treat == 0, clinic_Attrite]
treat_attriters <- final_sm[treat == 1, clinic_Attrite]

t.test(control_attriters,treat_attriters)
```

```
##
##  Welch Two Sample t-test
##
## data:  control_attriters and treat_attriters
## t = -0.91242, df = 372.33, p-value = 0.3621
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.10490266  0.03840531
## sample estimates:
## mean of x mean of y
## 0.1361502 0.1693989
```

Do the same thing as above, but this time with blocking included. I'm setting up a new `block` variable with `1-6` defined below:

1. Democrat + low ed level
2. Democrat + high ed level
3. Republican + low ed level
4. Republican + high ed level
5. Other + low ed level
6. Other + high ed level

```
#First, create two interim variables to simplify the current demographic info.
final_sm[, blockParty := ifelse(party == "Democrat", "dem",
                                ifelse(party == "Republican", "rep", "other"))]
final_sm[, blockEd := ifelse(education == "1: < high school" | education == "2: high school" | educatio
#Create the block variable with the 1-6 as defined in above text.
final_sm[, block := ifelse(blockParty == "dem" & blockEd == "low_ed", 1,
                           ifelse(blockParty == "dem" & blockEd == "high_ed",2,
                                  ifelse(blockParty == "rep" & blockEd == "low_ed",3,
                                         ifelse(blockParty == "rep" & blockEd == "high_ed",4,
                                                ifelse(blockParty == "other" & blockEd == "low_ed",5,6)

final_sm[, .(count = .N), keyby = .(block,parole_Attrite,treat)]
```

```
final_sm[, .(count = .N), keyby = .(block,ambulance_Attrite,treat)]
```

```
final_sm[, .(count = .N), keyby = .(block,clinic_Attrite,treat)]
```

```
final_sm[, .(count = .N), keyby = .(block, treat)]
```

```
final_sm[, .(pct_treat = mean(treat), pct_ctrl = 1-mean(treat)), keyby = .(block)]
```

```
# summary(final_sm )
```

The attrition appears more evenly dispersed among the blocks.

```
block1_attrit_C <- final_sm[treat == 0 & block == 1, clinic_Attrite]
block1_attrit_T <- final_sm[treat == 1 & block == 1, clinic_Attrite]
t.test(block1_attrit_C,block1_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block1_attrit_C and block1_attrit_T
## t = -0.20306, df = 41.475, p-value = 0.8401
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2605256  0.2129066
## sample estimates:
## mean of x mean of y
## 0.1666667 0.1904762
```

```
block2_attrit_C <- final_sm[treat == 0 & block == 2, clinic_Attrite]
block2_attrit_T <- final_sm[treat == 1 & block == 2, clinic_Attrite]
t.test(block2_attrit_C,block2_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block2_attrit_C and block2_attrit_T
## t = -1.893, df = 142.19, p-value = 0.06039
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.189060534  0.004092089
## sample estimates:
##  mean of x  mean of y
## 0.07619048 0.16867470
```

```
block3_attrit_C <- final_sm[treat == 0 & block == 3, clinic_Attrite]
block3_attrit_T <- final_sm[treat == 1 & block == 3, clinic_Attrite]
t.test(block3_attrit_C,block3_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block3_attrit_C and block3_attrit_T
## t = 1, df = 5, p-value = 0.3632
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2617636  0.5950970
## sample estimates:
```

```
## mean of x mean of y
## 0.1666667 0.0000000
```

```
block4_attrit_C <- final_sm[treat == 0 & block == 4, clinic_Attrite]
block4_attrit_T <- final_sm[treat == 1 & block == 4, clinic_Attrite]
t.test(block4_attrit_C,block4_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block4_attrit_C and block4_attrit_T
## t = -0.79881, df = 18.818, p-value = 0.4344
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5881723  0.2633860
## sample estimates:
## mean of x mean of y
## 0.2222222 0.3846154
```

```
block5_attrit_C <- final_sm[treat == 0 & block == 5, clinic_Attrite]
block5_attrit_T <- final_sm[treat == 1 & block == 5, clinic_Attrite]
t.test(block5_attrit_C,block5_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block5_attrit_C and block5_attrit_T
## t = -0.06143, df = 20.627, p-value = 0.9516
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2643259  0.2491744
## sample estimates:
##  mean of x  mean of y
## 0.08333333 0.09090909
```

```
block6_attrit_C <- final_sm[treat == 0 & block == 6, clinic_Attrite]
block6_attrit_T <- final_sm[treat == 1 & block == 6, clinic_Attrite]
t.test(block6_attrit_C,block6_attrit_T)
```

```
##
##  Welch Two Sample t-test
##
## data:  block6_attrit_C and block6_attrit_T
## t = 1.1768, df = 104.67, p-value = 0.242
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.06032838  0.23646873
## sample estimates:
## mean of x mean of y
## 0.2280702 0.1400000
```

Demographic EDA and Covariate Checks

Explore some of the demographic information.

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
```

```
            mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(education,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(gender,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(party,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(partyStrength,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(party,partyStrength,trea
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(income,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(voting,treat)]
```

```
final_sm[, .(count = .N, mean_support_parole = mean(parole_Support, na.rm = T),
             mean_support_ambulance = mean(ambulance_Support, na.rm = T),
             mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = .(block,treat)]
```

```
parole_only <- final_sm[,c("parole_Support", "treat")]
final_sm[, .(count = .N), keyby = .(parole_Support, treat)]
```

```
parole_only[parole_Support > 0, .(count = .N), keyby = .(treat)]
```

```
final_sm[parole_Support > 0, .(count = .N), keyby = .(party, parole_Support)]
```

```
unique(final_sm$state)
```

```
##  [1] "Ohio"               "Florida"            "Illinois"
##  [4] "California"         "Washington"         "Indiana"
##  [7] "Zurich"             "Virginia"           "Arizona"
## [10] "Tennessee"          "Michigan"           "South Carolina"
## [13] "New Jersey"         "North Carolina"     "Nairobi Area"
## [16] "Maryland"           "Washington, D.C."   "New York"
## [19] "Texas"              "Alabama"            "Pennsylvania"
## [22] "Connecticut"        "Juba"               "Massachusetts"
## [25] "Oregon"             "Leinster"           "Gauteng"
## [28] "Montana"            "Wisconsin"          "Iowa"
## [31] "Nouvelle-Aquitaine" "Delaware"           "Colorado"
## [34] "Missouri"           "Georgia"            "Ontario"
## [37] "Vilnius"            "Louisiana"          "Scotland"
## [40] "AnzoÃ¡tegui"        "New Mexico"         "New Hampshire"
## [43] "Western Australia"  "Oklahoma"           "Utah"
## [46] "Minnesota"          "Nevada"             "Tasmania"
## [49] "Kansas"             "Kisumu"             "Kentucky"
```

```
## [52] "Southern Region"    "Tokyo"              "Victoria"
## [55] "ÃŽle-de-France"     "Harare"
```

```
states <- final_sm[, .(num_in_state = .N, mean_support_parole = mean(parole_Support, na.rm = T),
            mean_support_ambulance = mean(ambulance_Support, na.rm = T),
            mean_support_clinic = mean(clinic_Support, na.rm = T)), keyby = state]
states <- states[order(-num_in_state)]
states[, state_bin := ifelse(num_in_state > 10,state,"other")]
statesFinal <- states[, .(state, state_bin)]
statesFinal
```

```
#Merge the state bin back into the main dataset.
final_sm <- merge(final_sm,statesFinal,by = "state")
```

Covariate Balance checks

```
covariate_data <- final_sm[,list(treat,age,gender,education,income,voting,party)]
mod_null <- lm(treat ~ 1, data = na.omit(covariate_data))
mod_covariate <-final_sm[, lm(treat ~ 1 + age + gender + education + income + voting + party)]
summary(mod_covariate)
```

```
##
## Call:
## lm(formula = treat ~ 1 + age + gender + education + income +
##     voting + party)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6539 -0.4401 -0.3108  0.5133  0.7353
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -0.123139   0.509332  -0.242   0.8091
## age                             -0.000957   0.002295  -0.417   0.6769
## genderMale                       0.092597   0.055840   1.658   0.0981 .
## genderNonbinary/Transgender/Other -0.008259   0.232344  -0.036   0.9717
## genderPrefer not to say         -0.504108   0.359773  -1.401   0.1620
## education2: high school          0.681621   0.526520   1.295   0.1963
## education3: some college         0.660449   0.514366   1.284   0.1999
## education4: college degree       0.695034   0.515599   1.348   0.1785
## education5: some post-grad       0.666440   0.517249   1.288   0.1984
## education6: graduate degree      0.657867   0.516466   1.274   0.2035
## income2: $50k to $100k          -0.120248   0.078217  -1.537   0.1250
## income3: over $100k             -0.088775   0.073889  -1.201   0.2303
## voting2: every 4 years          -0.051021   0.117085  -0.436   0.6633
## voting3: every 2 years          -0.119817   0.109783  -1.091   0.2758
## voting4: always vote             0.025777   0.100791   0.256   0.7983
## partyIndependent                 0.051596   0.060750   0.849   0.3962
## partyOther                      -0.093537   0.105941  -0.883   0.3778
## partyRepublican                  0.091768   0.094176   0.974   0.3305
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4998 on 377 degrees of freedom
##   (1 observation deleted due to missingness)
```

```
## Multiple R-squared:  0.04036,    Adjusted R-squared:  -0.002916
## F-statistic: 0.9326 on 17 and 377 DF,  p-value: 0.5354
```

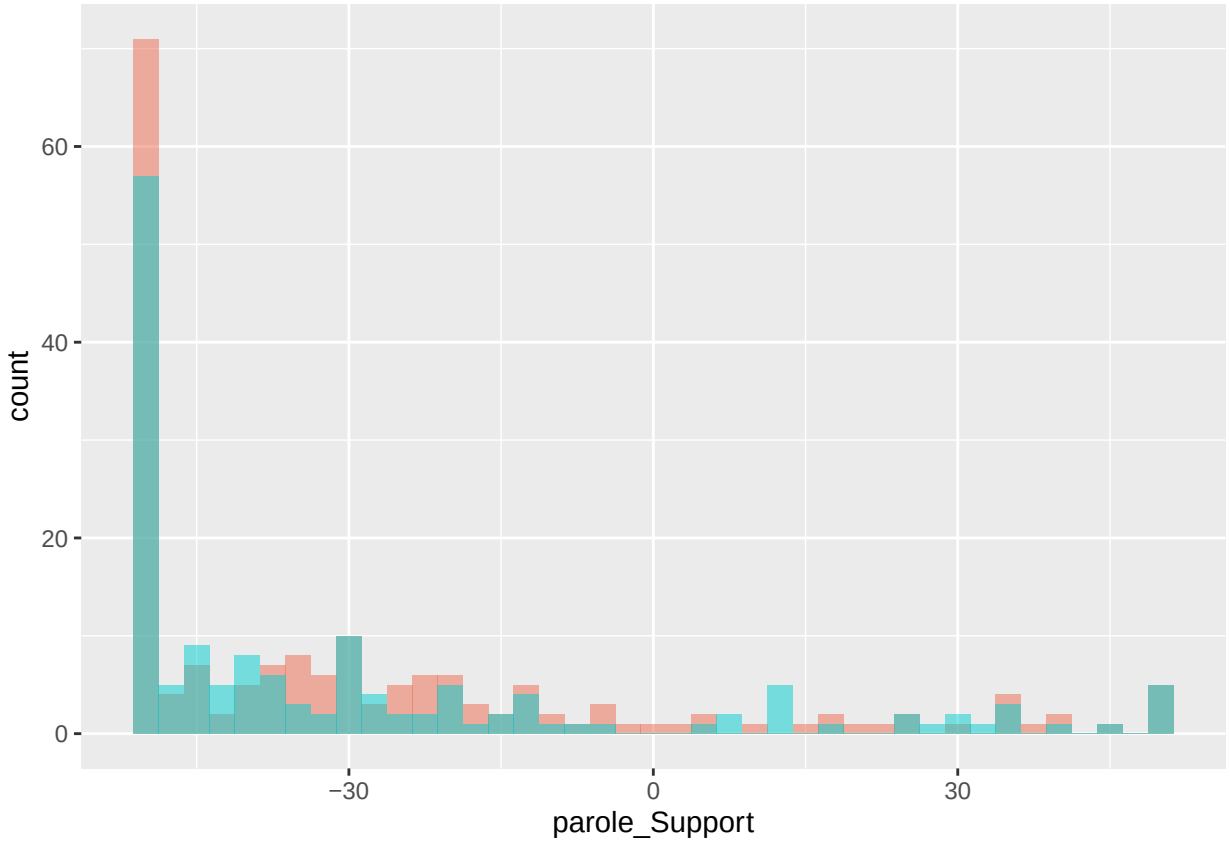
```
anova(mod_covariate, mod_null, test= 'F')
```

Distribution of Data in T/C

```
ggplot(final_sm, aes(x=parole_Support)) +
  geom_histogram(data = final_sm[treat == 0], fill = "coral2", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = final_sm[treat == 1], fill = "darkturquoise", alpha = 0.5, binwidth = 2.5)
```

```
## Warning: Removed 30 rows containing non-finite values (stat_bin).
```

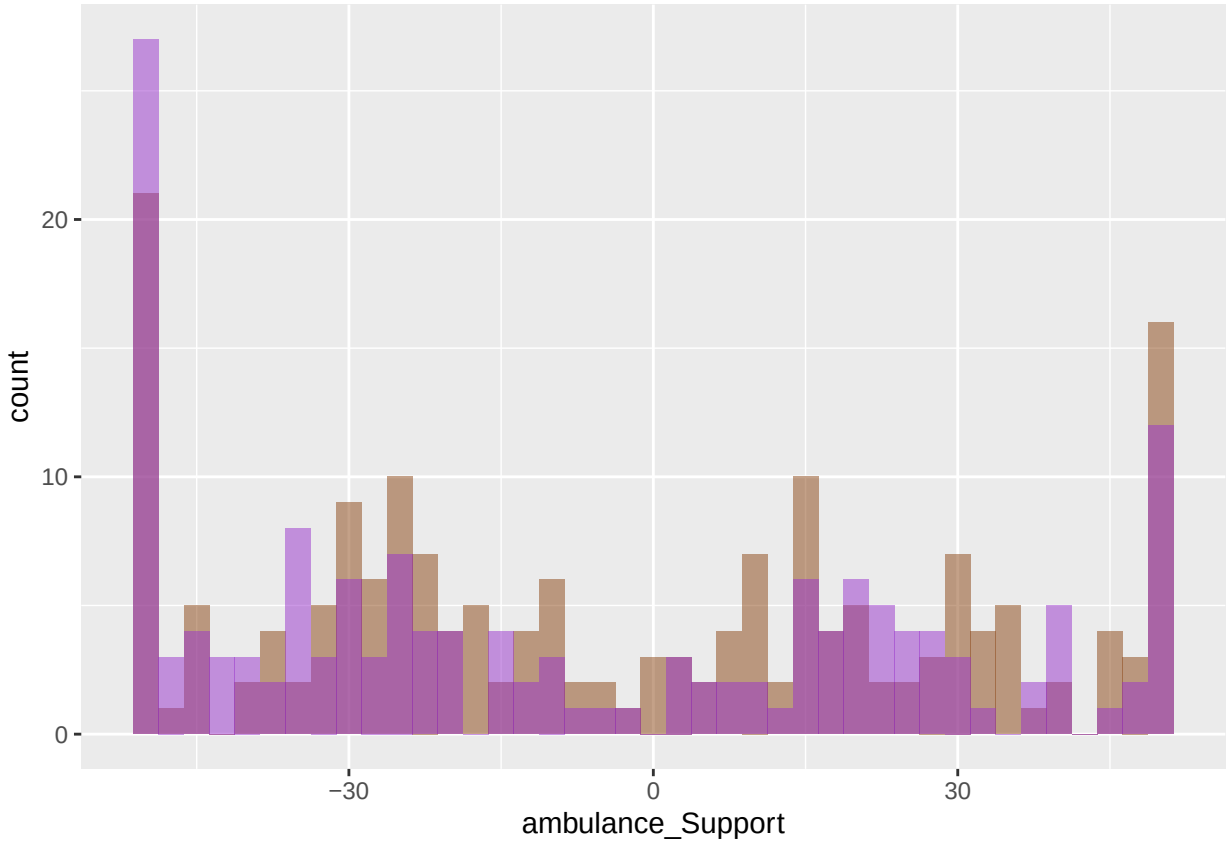
```
## Warning: Removed 30 rows containing non-finite values (stat_bin).
```




```
ggplot(final_sm, aes(x=ambulance_Support)) +
  geom_histogram(data = final_sm[treat == 0], fill = "chocolate4", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = final_sm[treat == 1], fill = "darkorchid", alpha = 0.5, binwidth = 2.5)
```

```
## Warning: Removed 26 rows containing non-finite values (stat_bin).
```

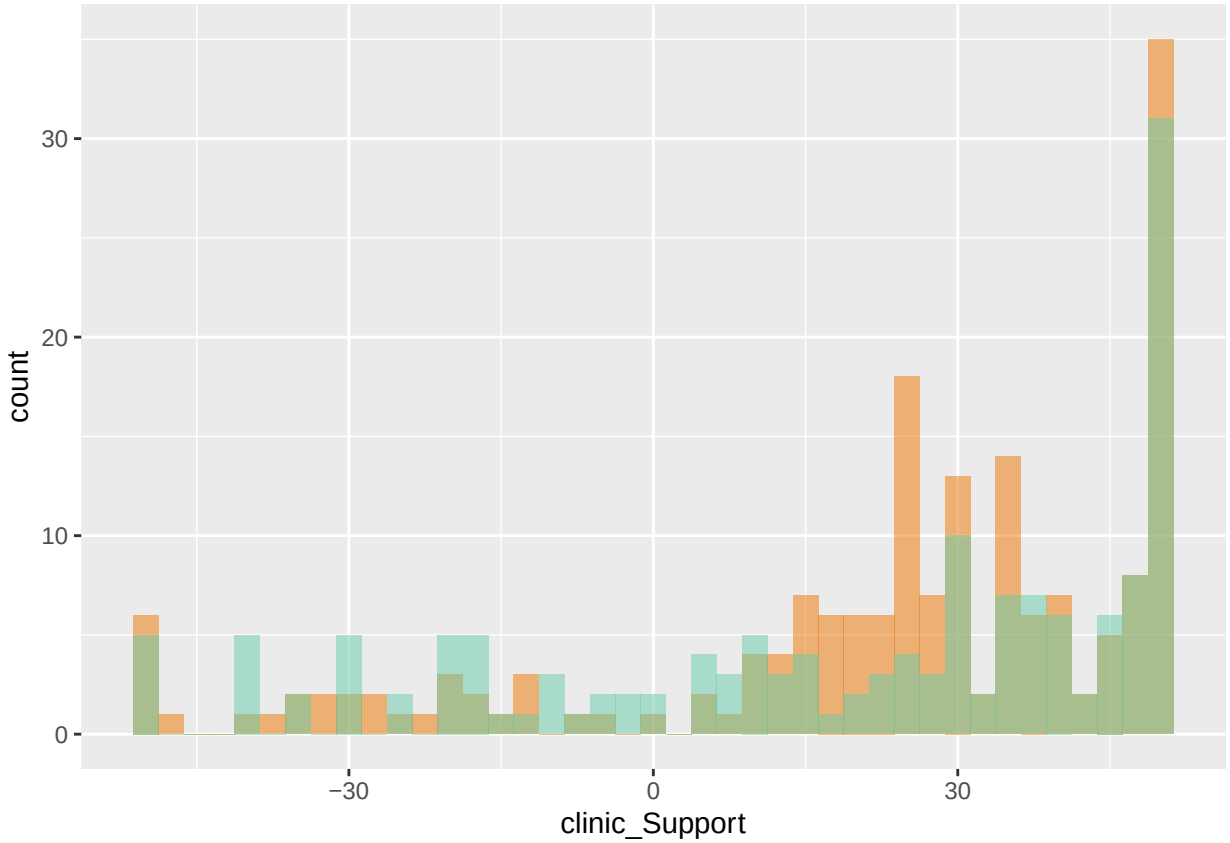
```
## Warning: Removed 29 rows containing non-finite values (stat_bin).
```




```
ggplot(final_sm, aes(x=clinic_Support)) +
  geom_histogram(data = final_sm[treat == 0], fill = "darkorange2", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = final_sm[treat == 1], fill = "aquamarine3", alpha = 0.5, binwidth = 2.5)
```

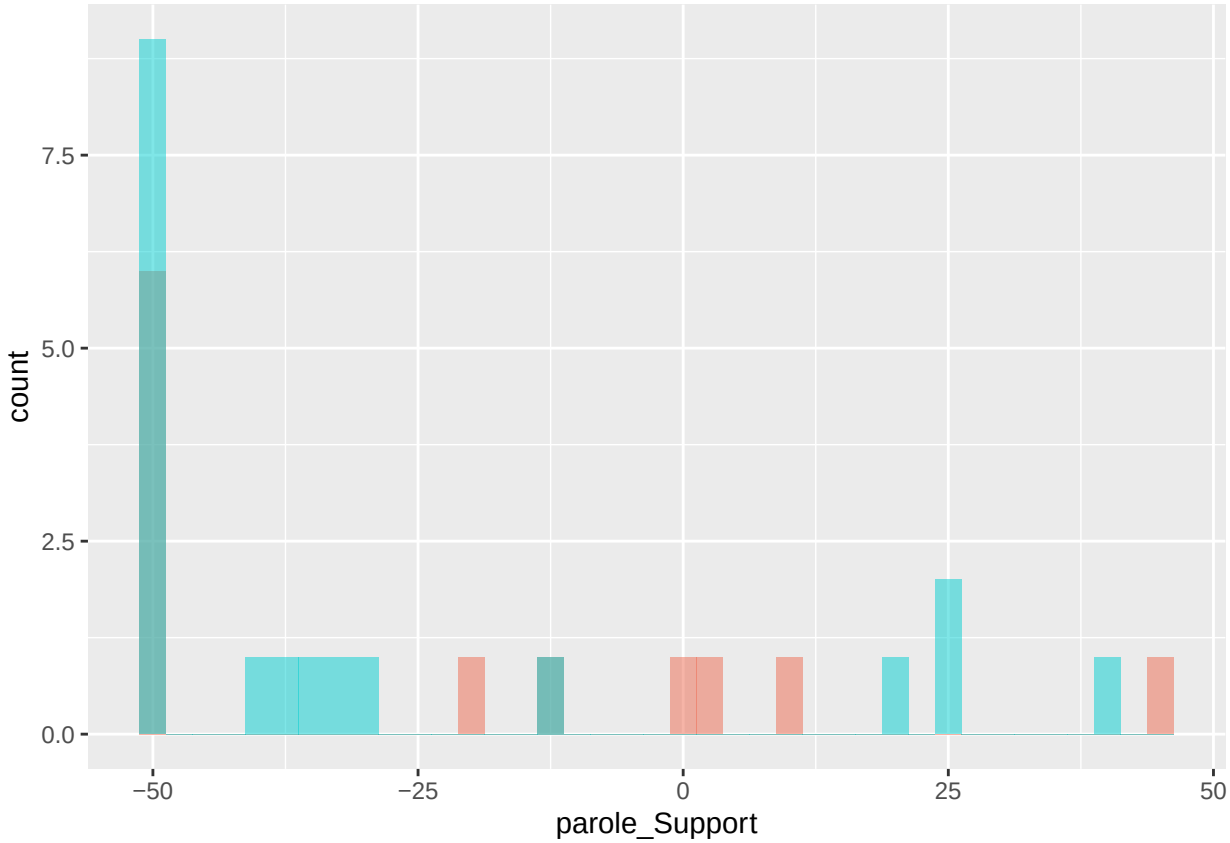
```
## Warning: Removed 29 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 31 rows containing non-finite values (stat_bin).
```



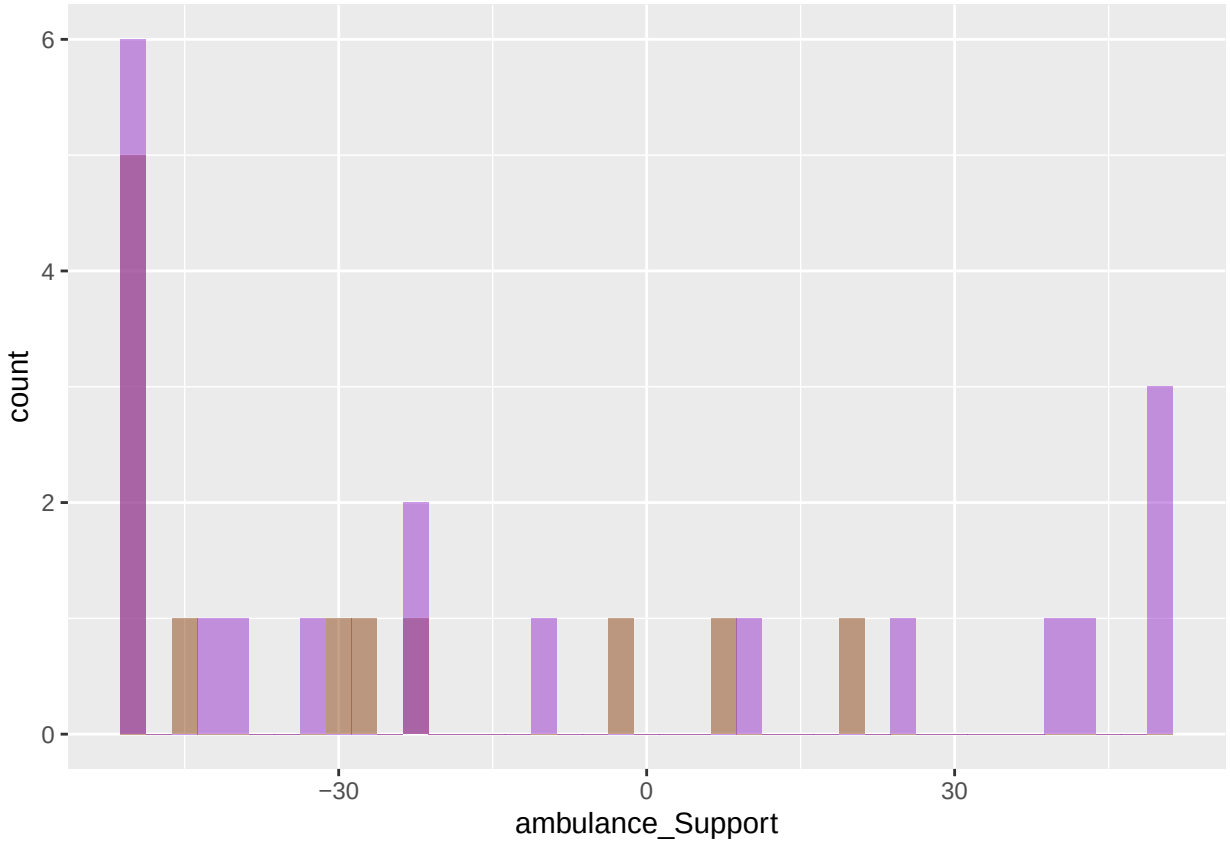

```
ggplot(pilot_sm, aes(x=parole_Support)) +
  geom_histogram(data = pilot_sm[treat == 0], fill = "coral2", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = pilot_sm[treat == 1], fill = "darkturquoise", alpha = 0.5, binwidth = 2.5)
```

```
## Warning: Removed 2 rows containing non-finite values (stat_bin).
```



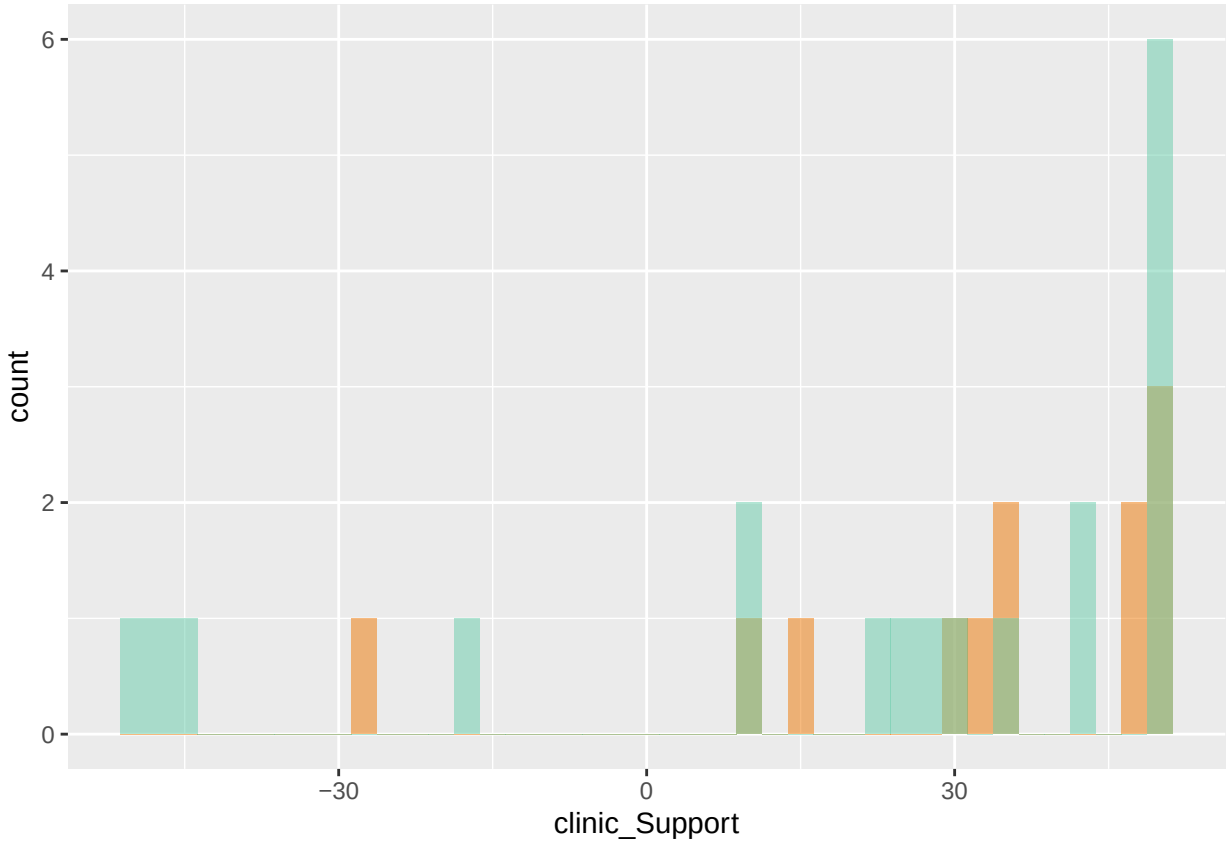

```
ggplot(pilot_sm, aes(x=ambulance_Support)) +
  geom_histogram(data = pilot_sm[treat == 0], fill = "chocolate4", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = pilot_sm[treat == 1], fill = "darkorchid", alpha = 0.5, binwidth = 2.5)
```

```
## Warning: Removed 2 rows containing non-finite values (stat_bin).
```




```
ggplot(pilot_sm, aes(x=clinic_Support)) +
  geom_histogram(data = pilot_sm[treat == 0], fill = "darkorange2", alpha = 0.5, binwidth = 2.5) +
  geom_histogram(data = pilot_sm[treat == 1], fill = "aquamarine3", alpha = 0.5, binwidth = 2.5)
```

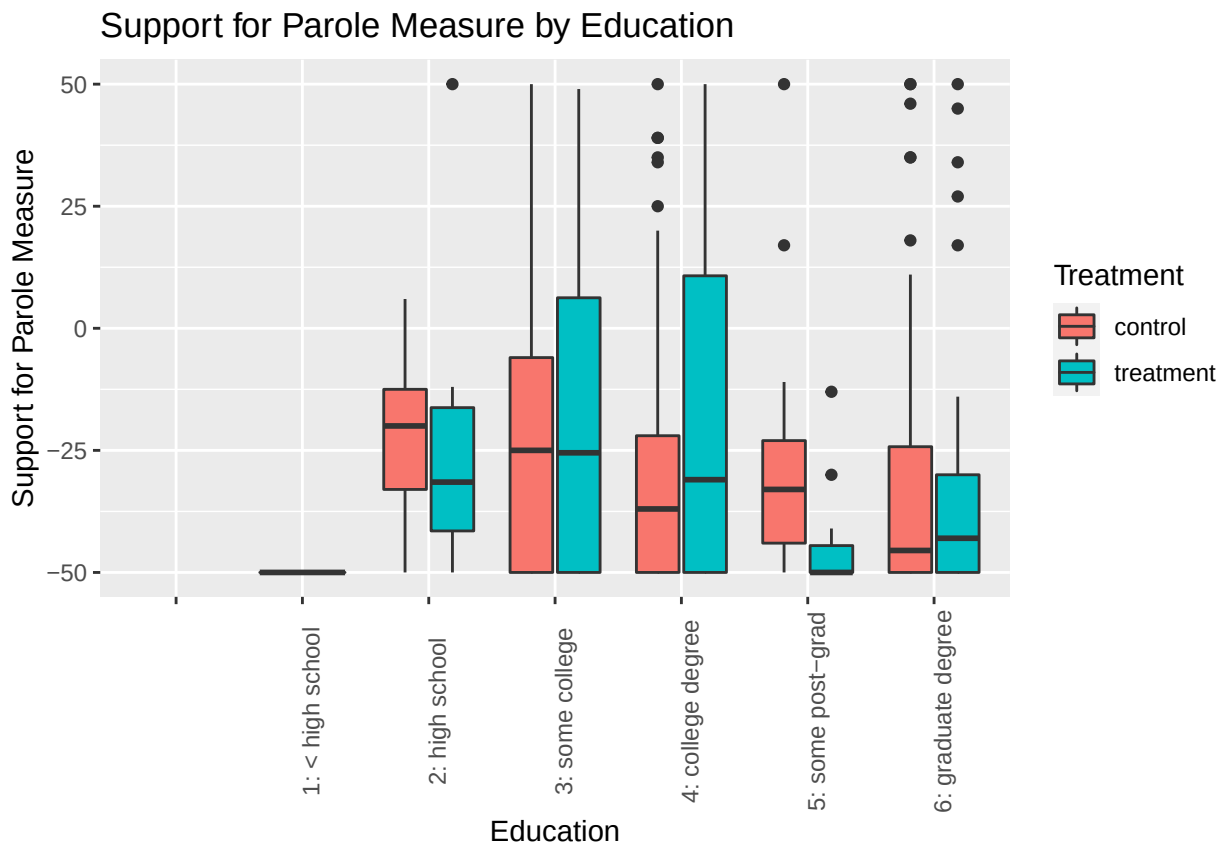
```
## Warning: Removed 2 rows containing non-finite values (stat_bin).
```



See if plotting some of the covariates makes for easier interpretation. These plots will at least show the IQranges.

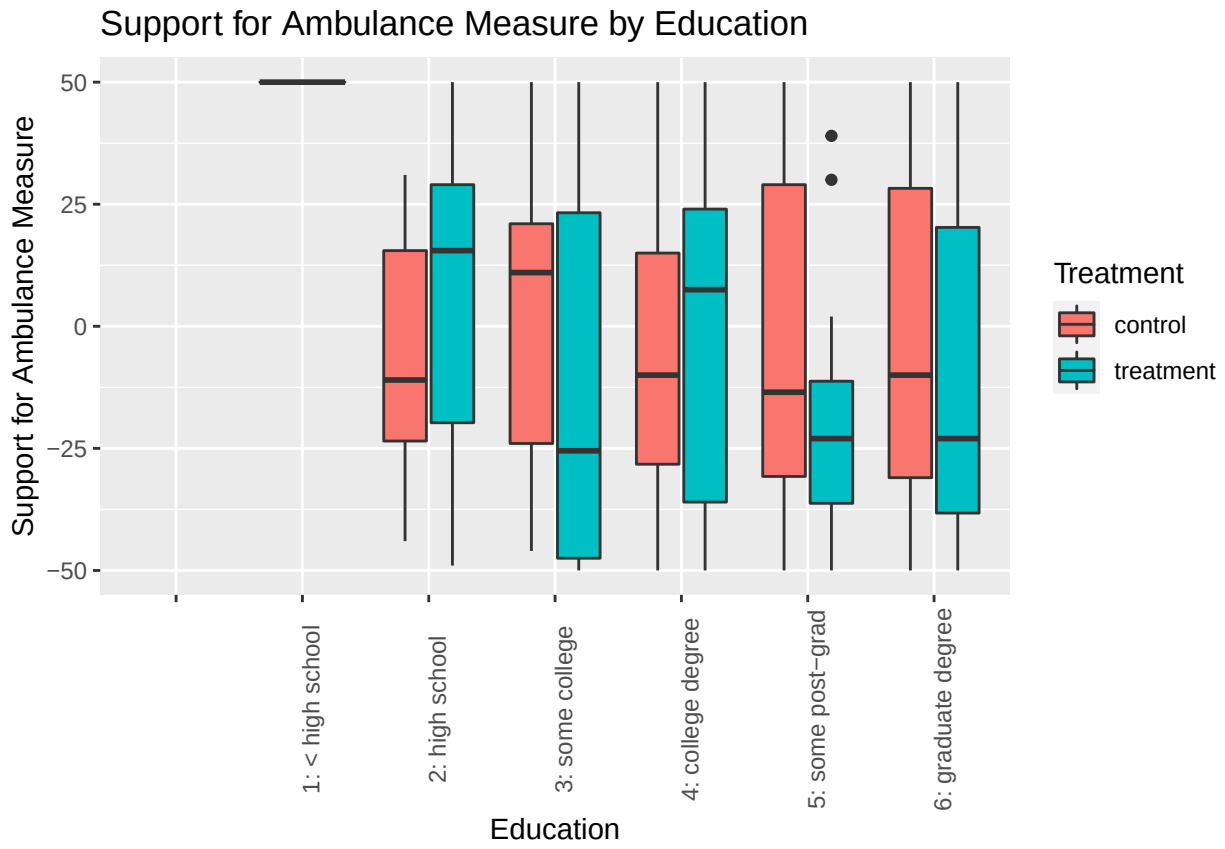
```
ggplot(final_sm, aes(factor(education), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Education") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Education", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



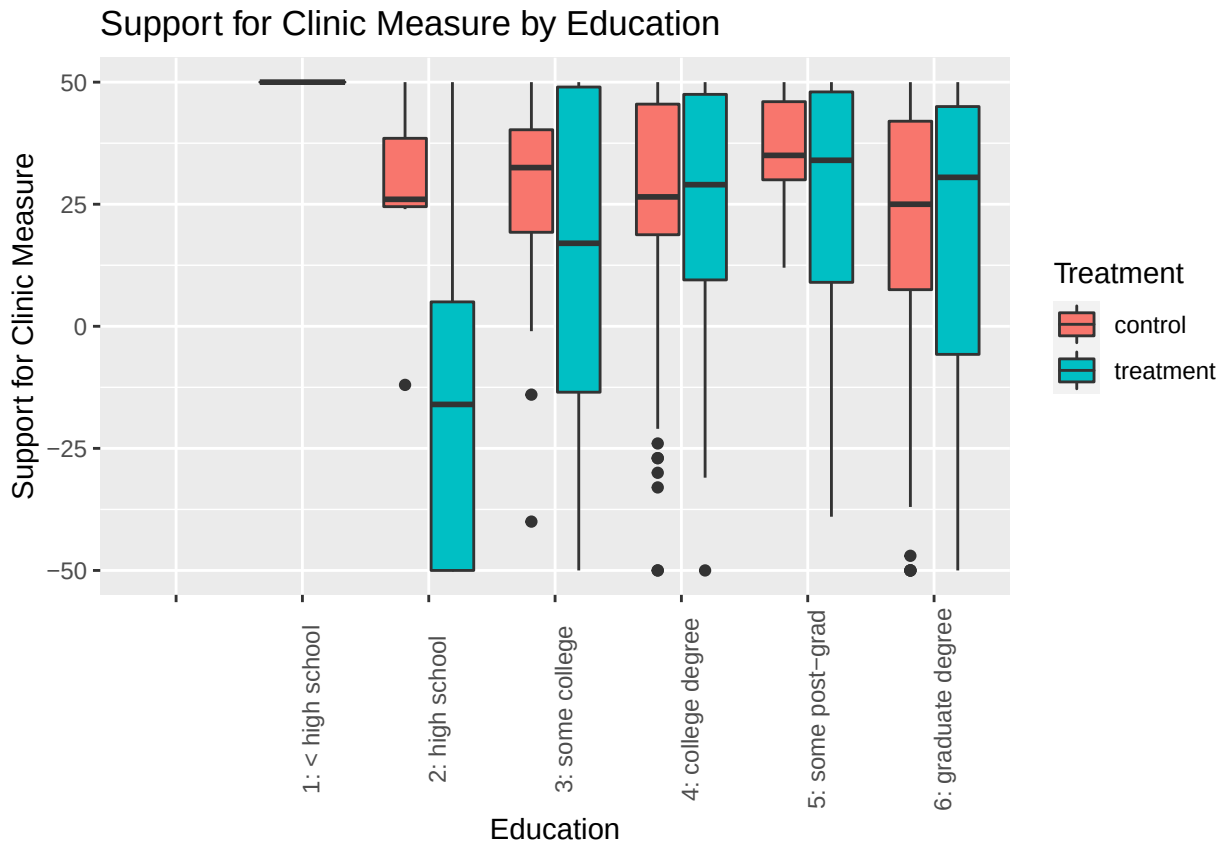

```
ggplot(final_sm, aes(factor(education), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by Education") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Education", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```



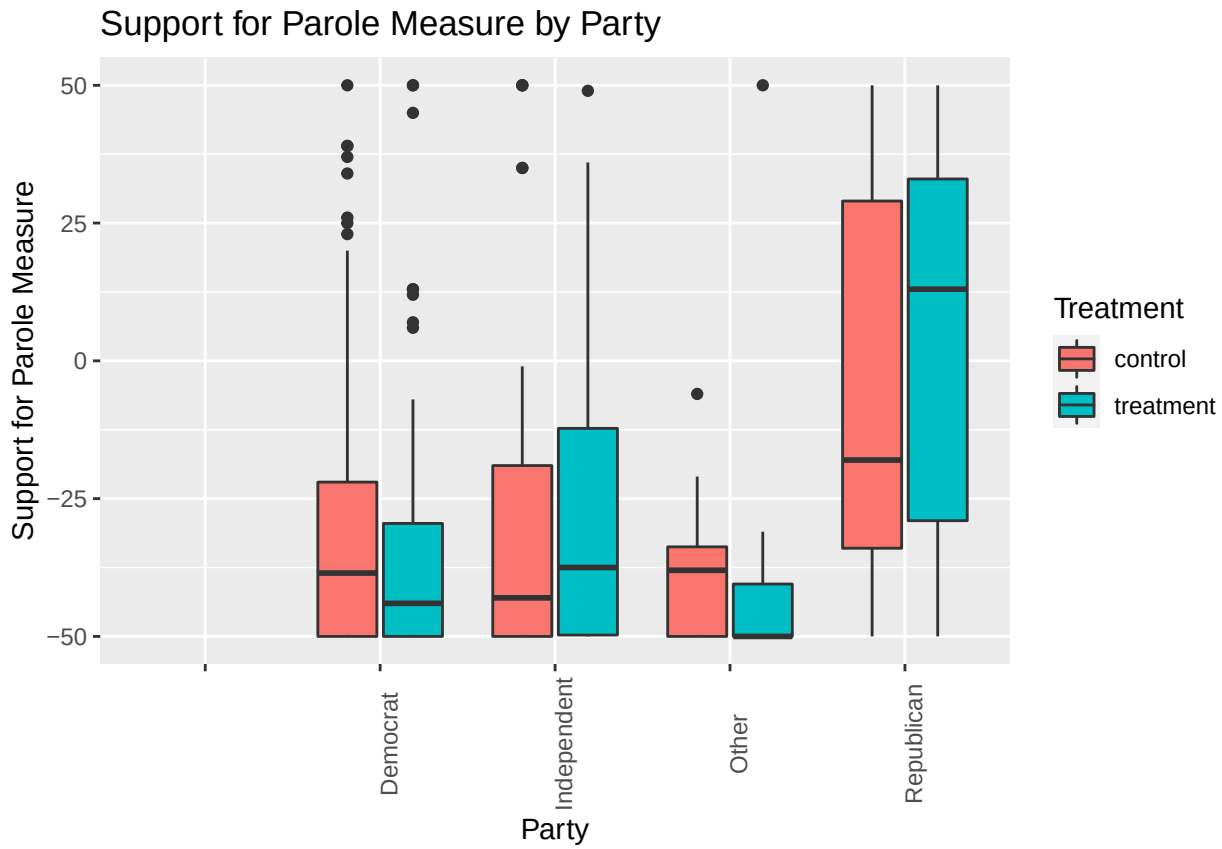

```
ggplot(final_sm, aes(factor(education), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Education") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Education", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



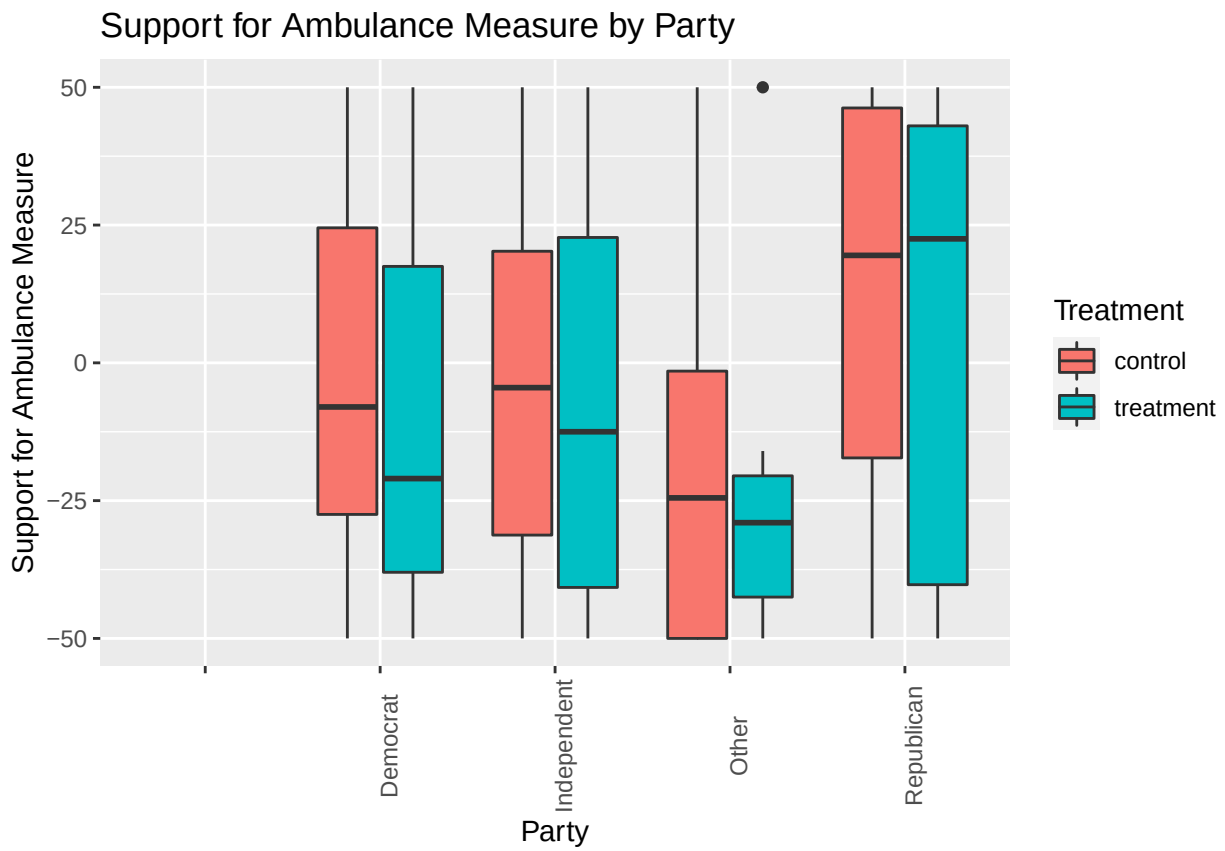

```
ggplot(final_sm, aes(factor(party), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Party") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Party", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



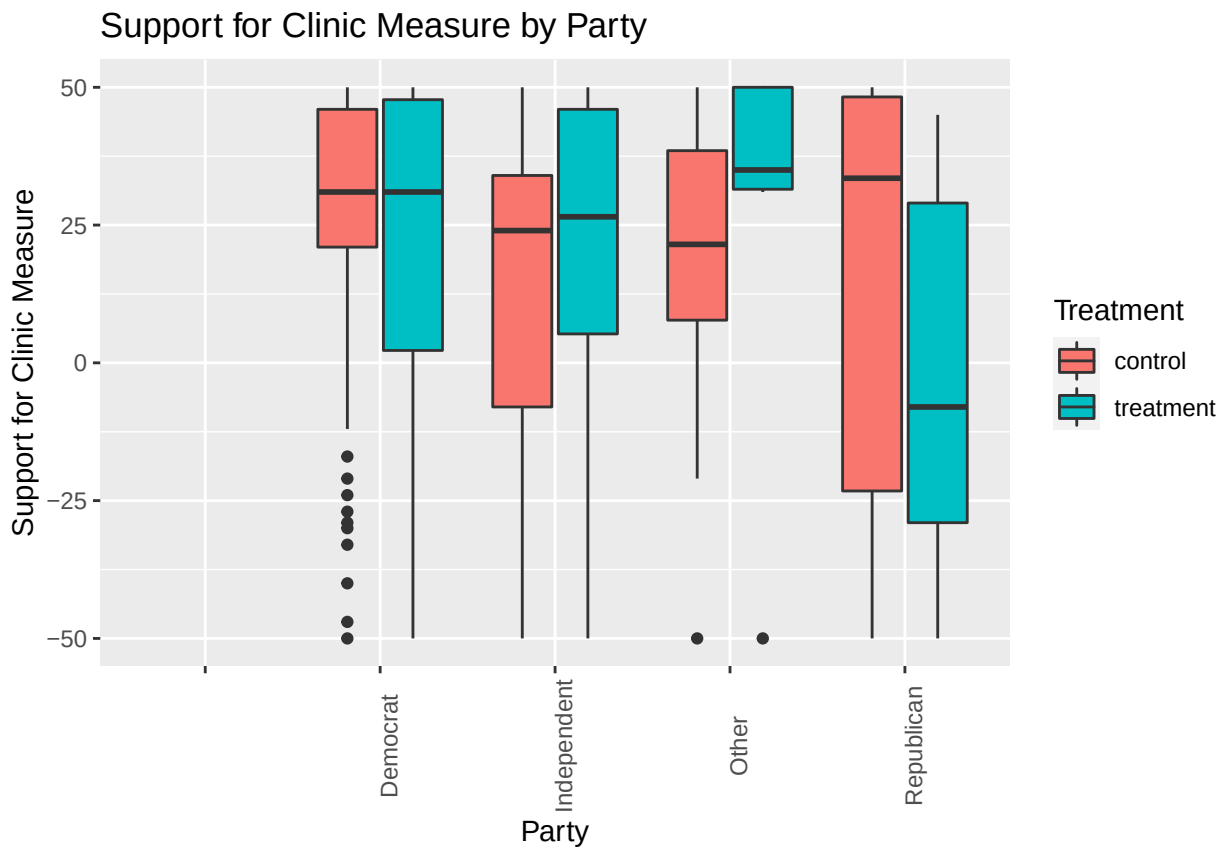

```
ggplot(final_sm, aes(factor(party), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by Party") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Party", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```



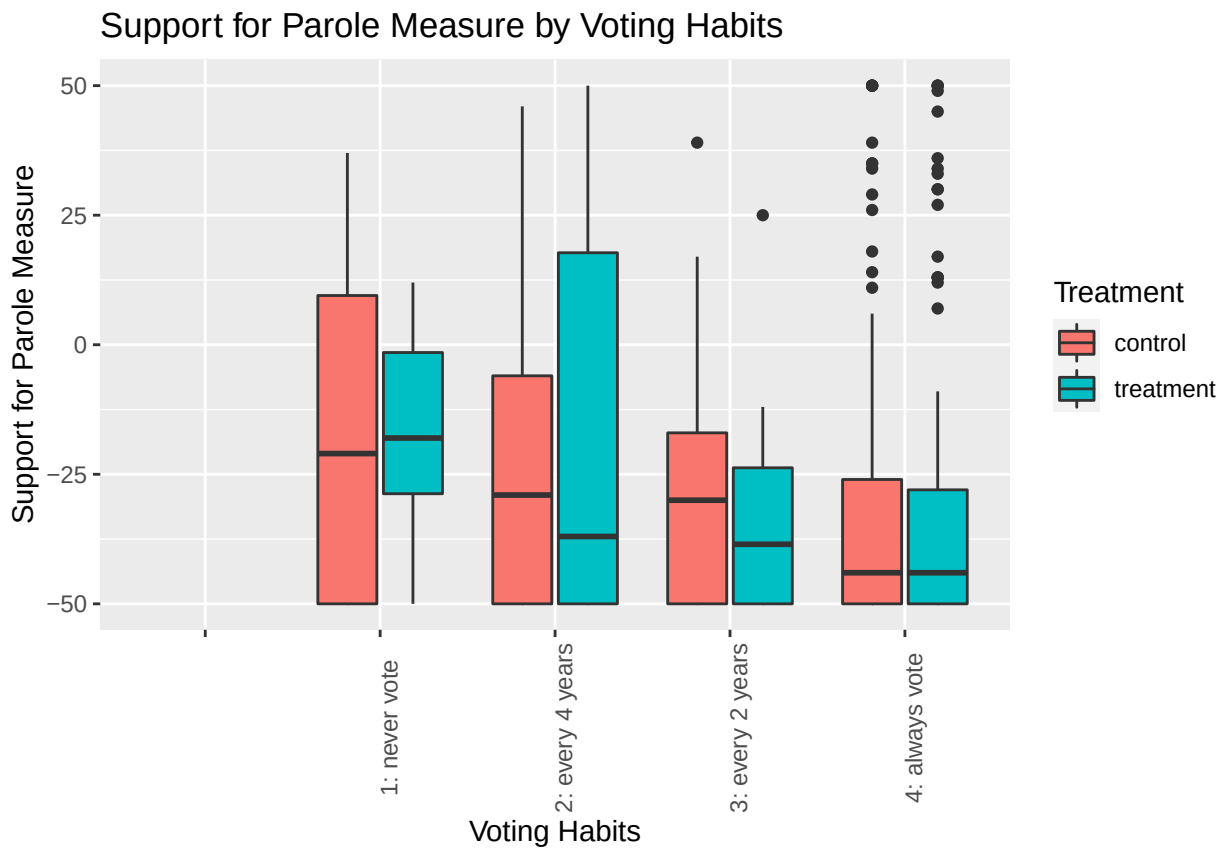

```
ggplot(final_sm, aes(factor(party), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Party") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Party", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



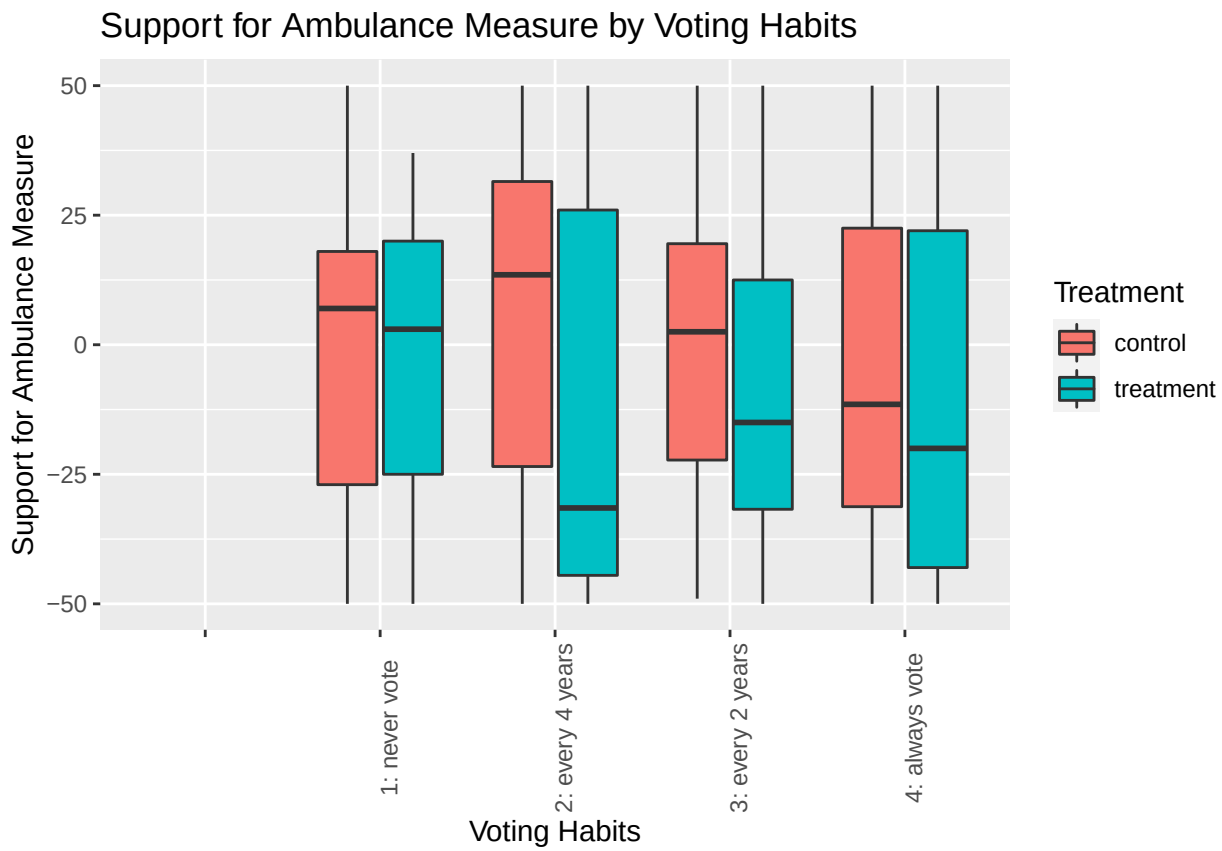

```
ggplot(final_sm, aes(factor(voting), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Voting Habits") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Voting Habits", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



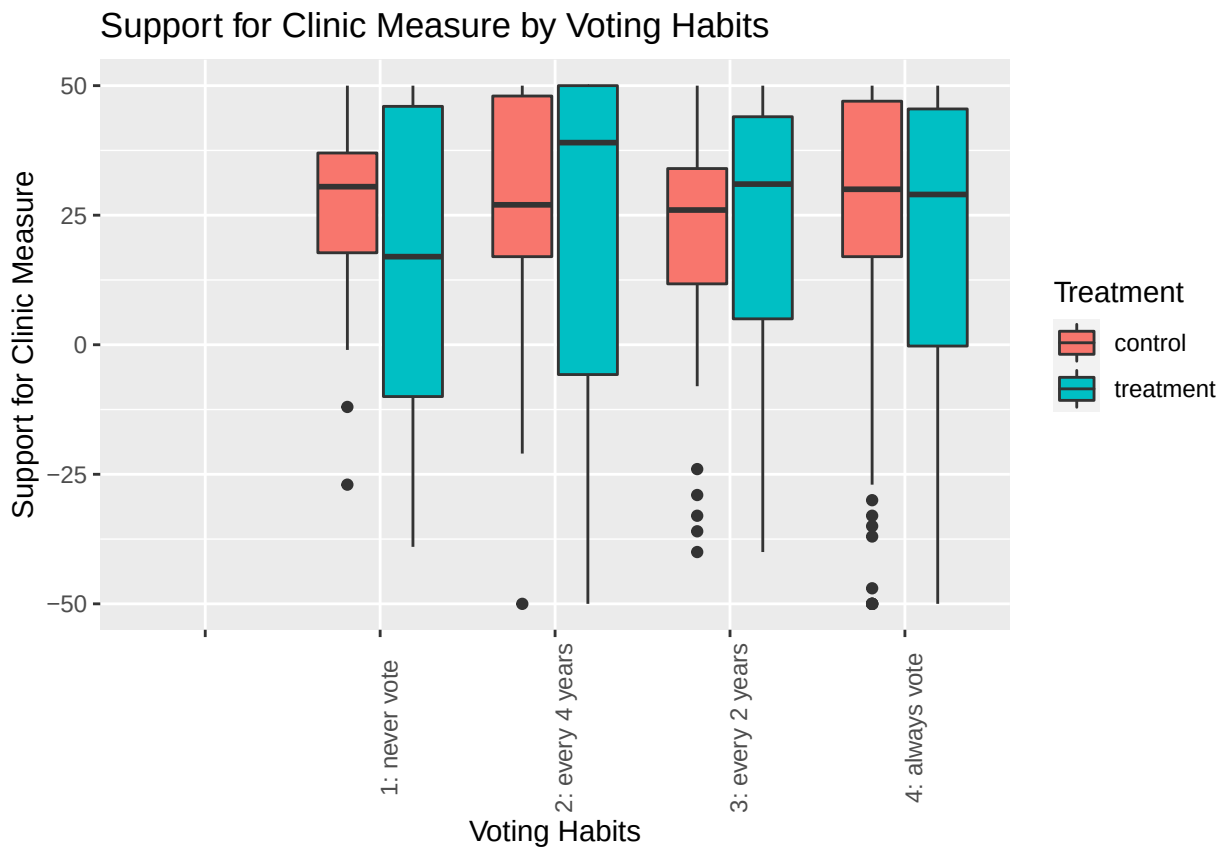

```
ggplot(final_sm, aes(factor(voting), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by Voting Habits") + theme(axis.text.x  = element_text(angle=90))
  labs(x = "Voting Habits", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```



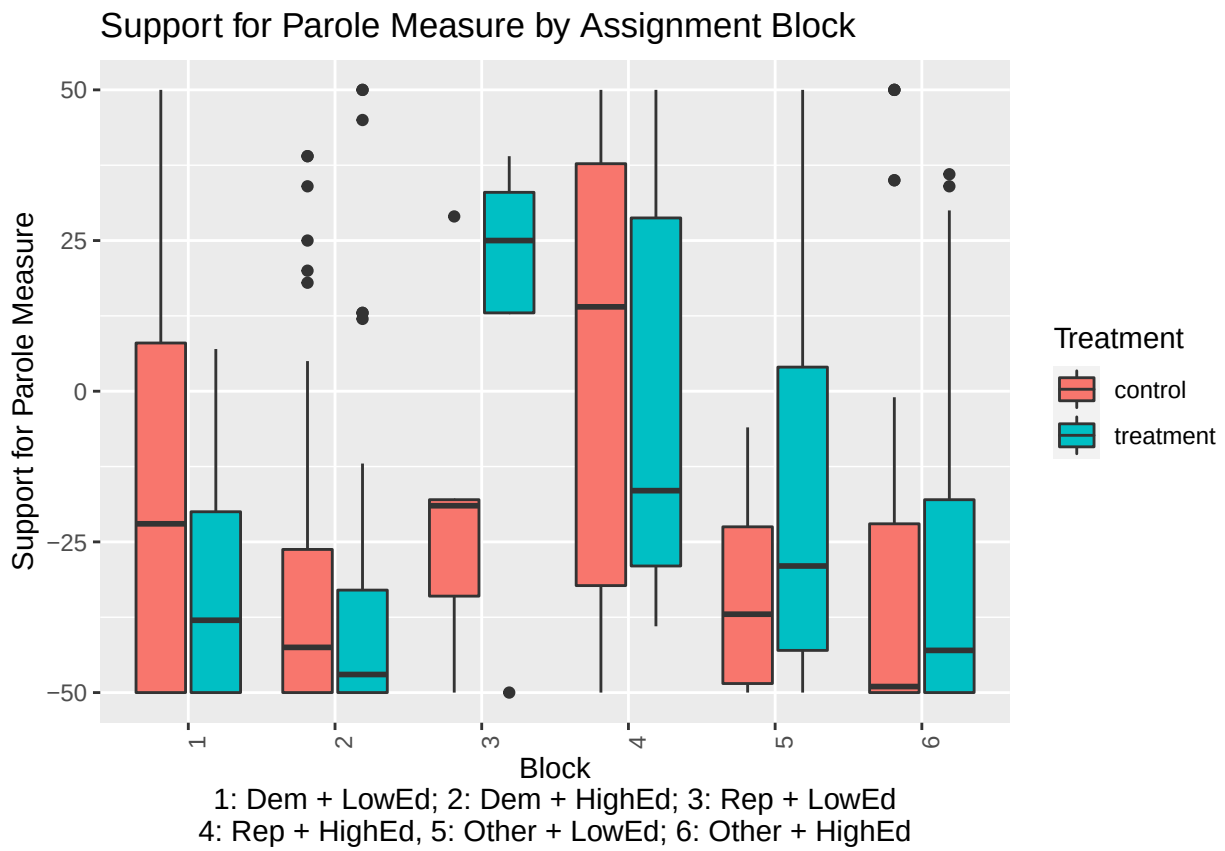

```
ggplot(final_sm, aes(factor(voting), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Voting Habits") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Voting Habits", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



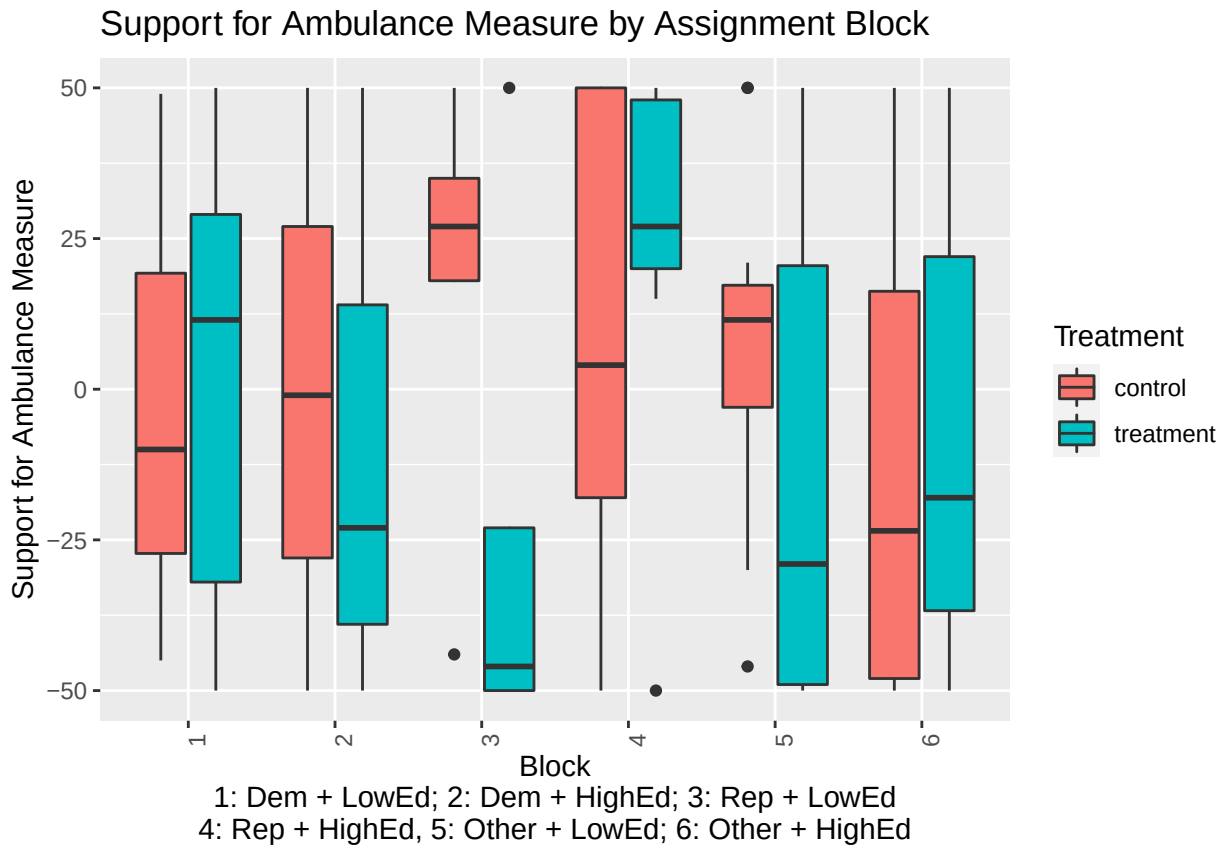

```
ggplot(final_sm, aes(factor(block), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Assignment Block") + theme(axis.text.x  = element_text(angle=90))
  labs(x = "Block\n1: Dem + LowEd; 2: Dem + HighEd; 3: Rep + LowEd\n4: Rep + HighEd, 5: Other + LowEd;
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



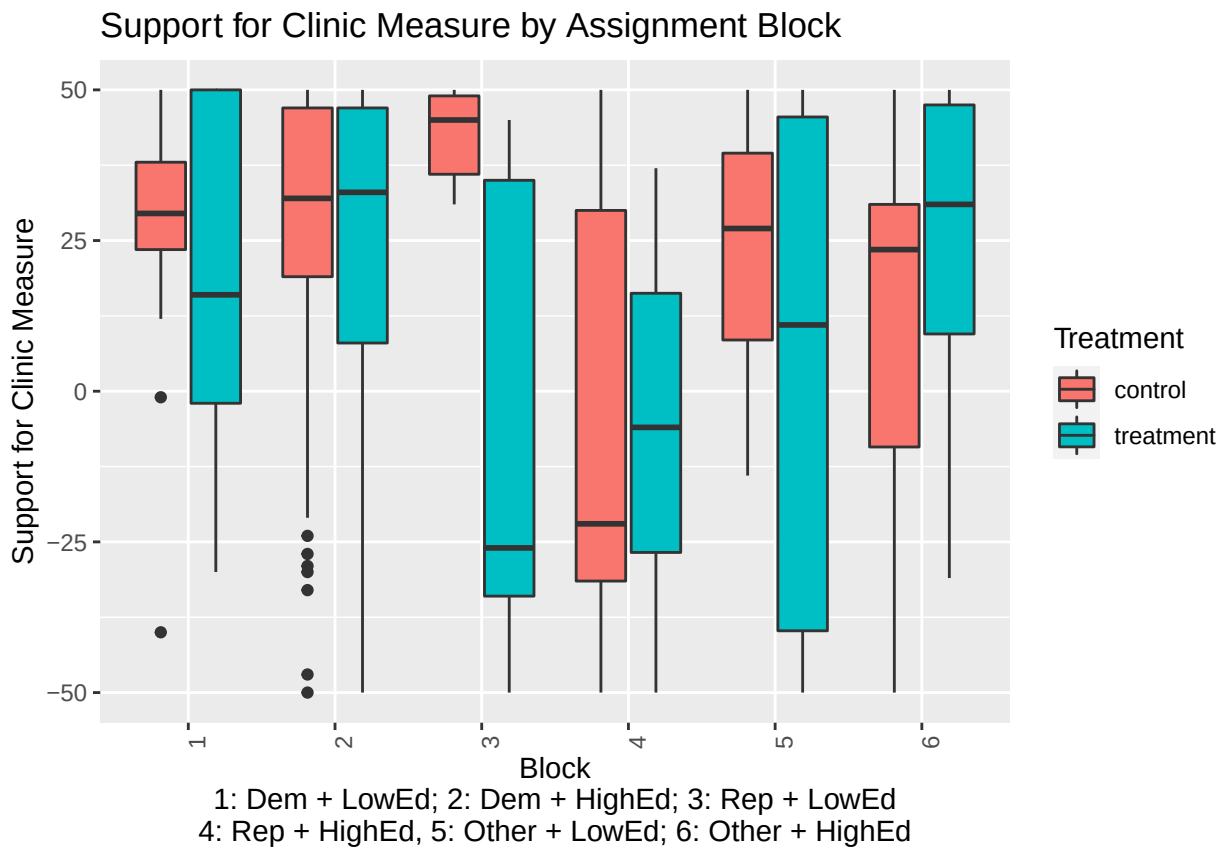

```
ggplot(final_sm, aes(factor(block), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by Assignment Block") + theme(axis.text.x  = element_text(angle=
  labs(x = "Block\n1: Dem + LowEd; 2: Dem + HighEd; 3: Rep + LowEd\n4: Rep + HighEd, 5: Other + LowEd; 
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```




```
ggplot(final_sm, aes(factor(block), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Assignment Block") + theme(axis.text.x  = element_text(angle=90))
  labs(x = "Block\n1: Dem + LowEd; 2: Dem + HighEd; 3: Rep + LowEd\n4: Rep + HighEd, 5: Other + LowEd; 6
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

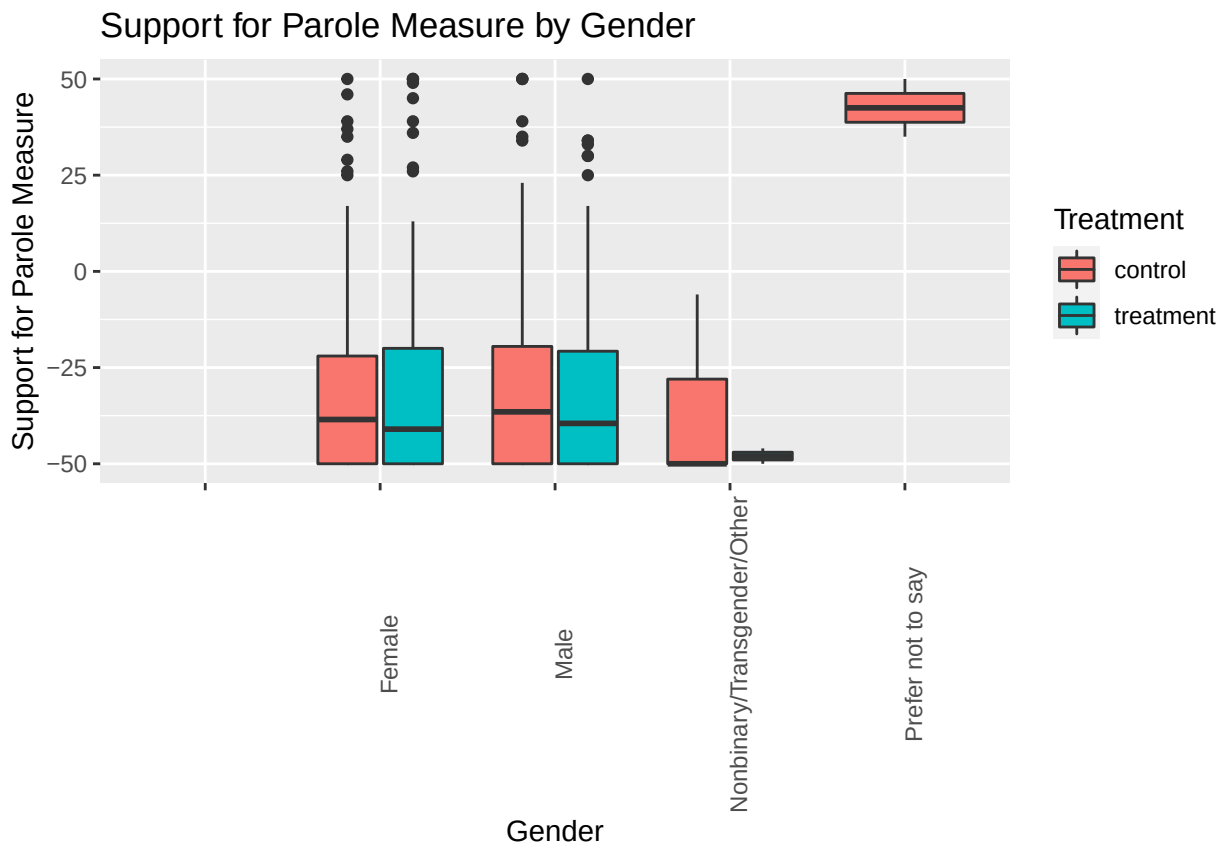
```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



For Ambulance and Clinic, the treatment caused an opposite effect for low education level vs. high education level

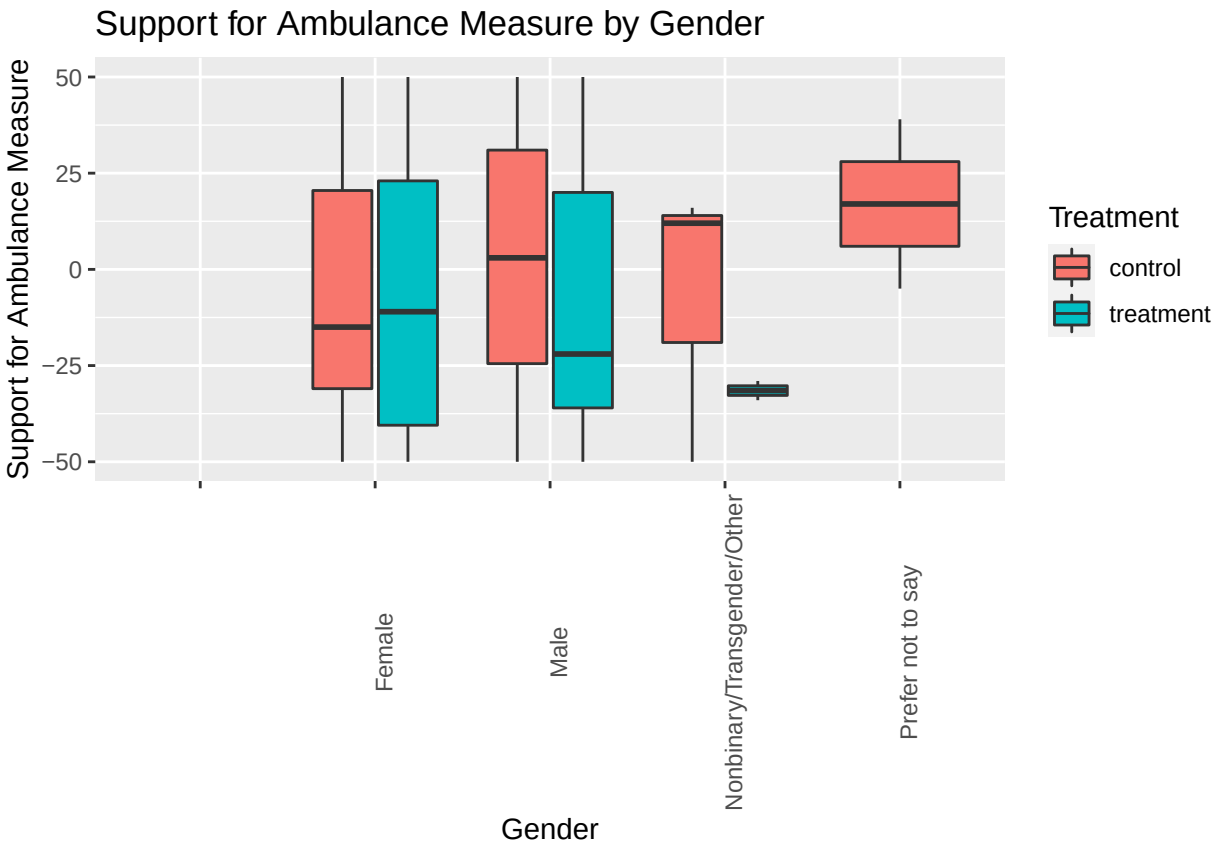
```
ggplot(final_sm, aes(factor(gender), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Gender") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Gender", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



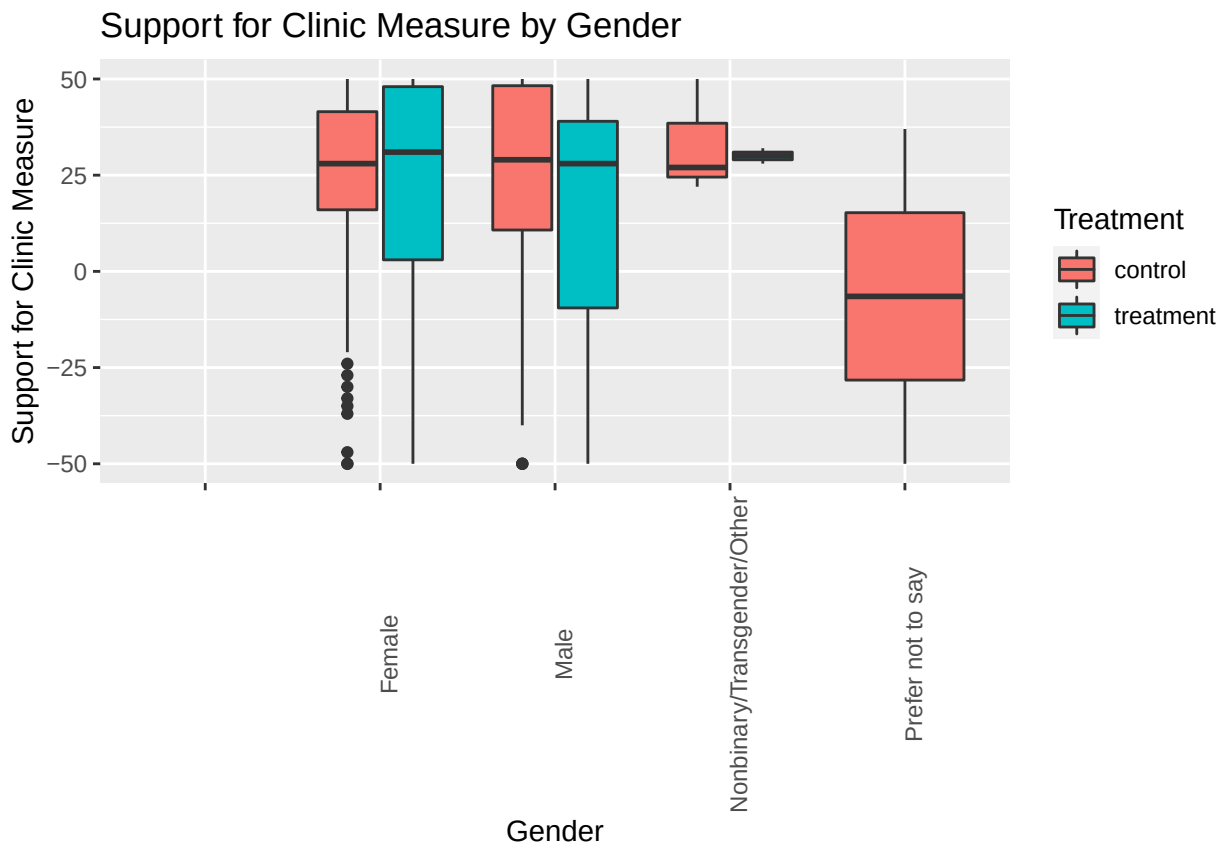

```
ggplot(final_sm, aes(factor(gender), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by Gender") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Gender", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```



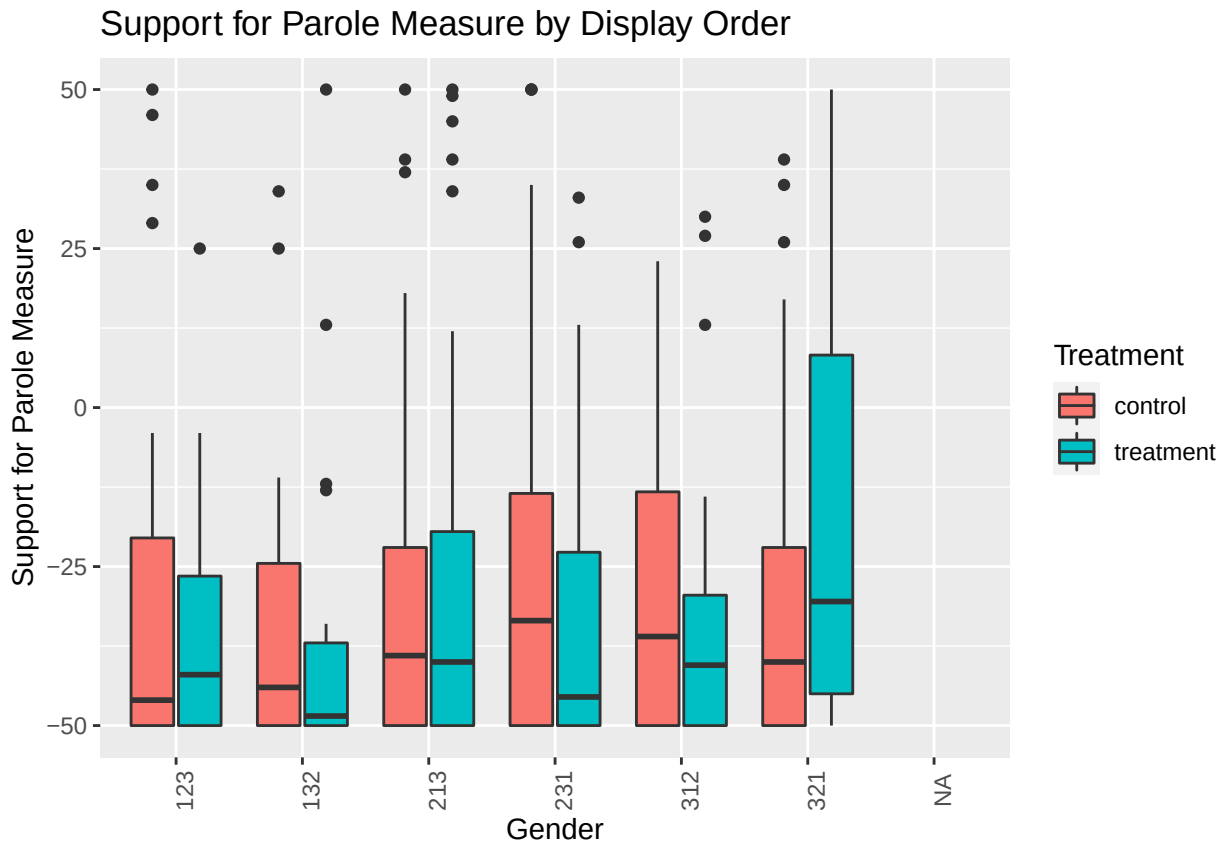

```
ggplot(final_sm, aes(factor(gender), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Gender") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Gender", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



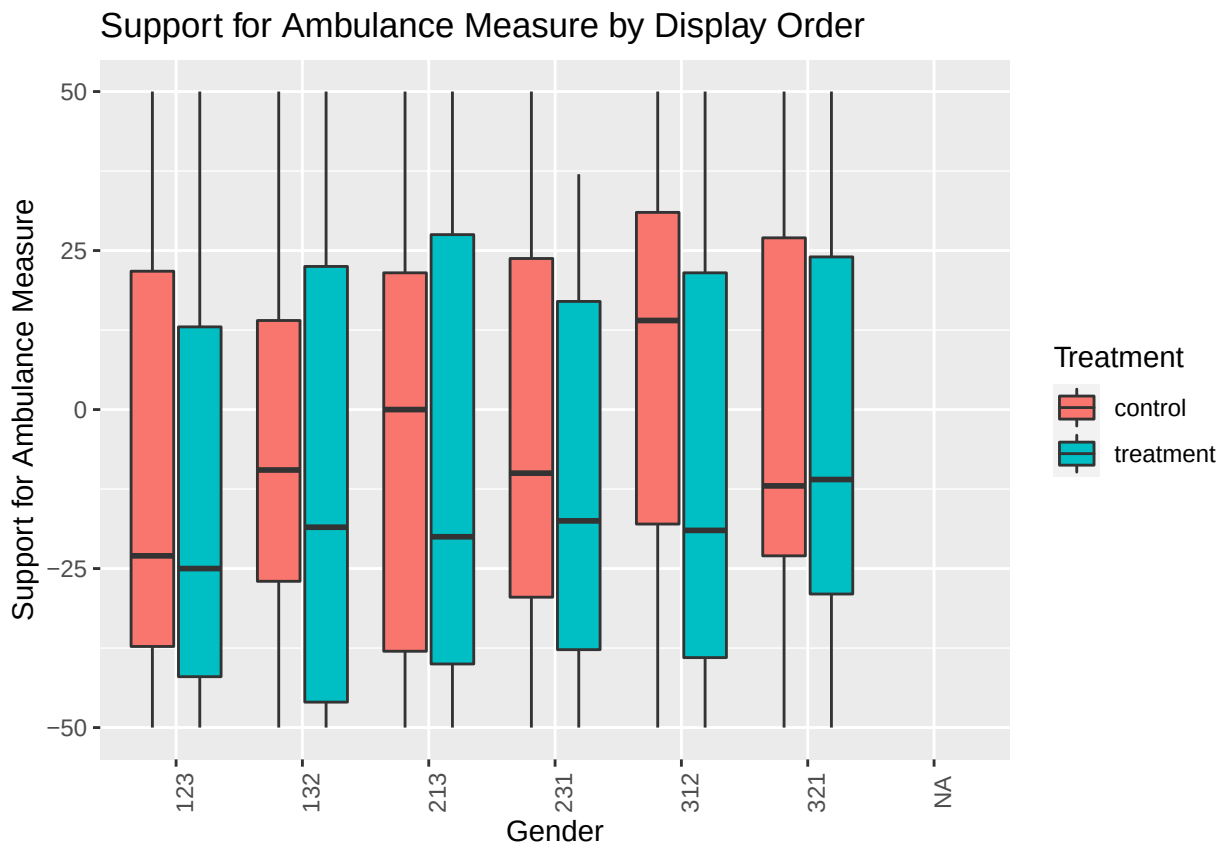

```
ggplot(final_sm, aes(factor(displayOrder), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Display Order") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Gender", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



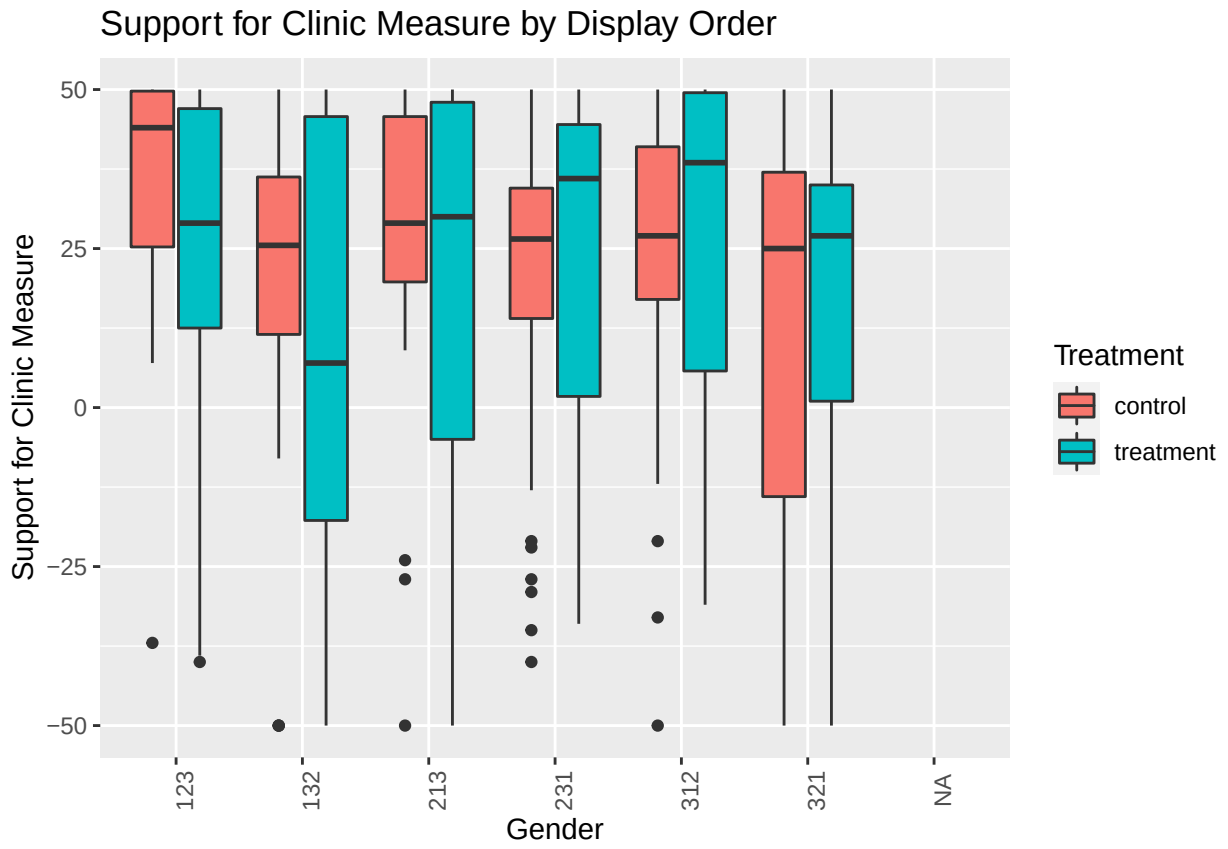

```
ggplot(final_sm, aes(factor(displayOrder), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat)))
ggtitle("Support for Ambulance Measure by Display Order") + theme(axis.text.x  = element_text(angle=90))
  labs(x = "Gender", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```




```
ggplot(final_sm, aes(factor(displayOrder), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Display Order") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Gender", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```

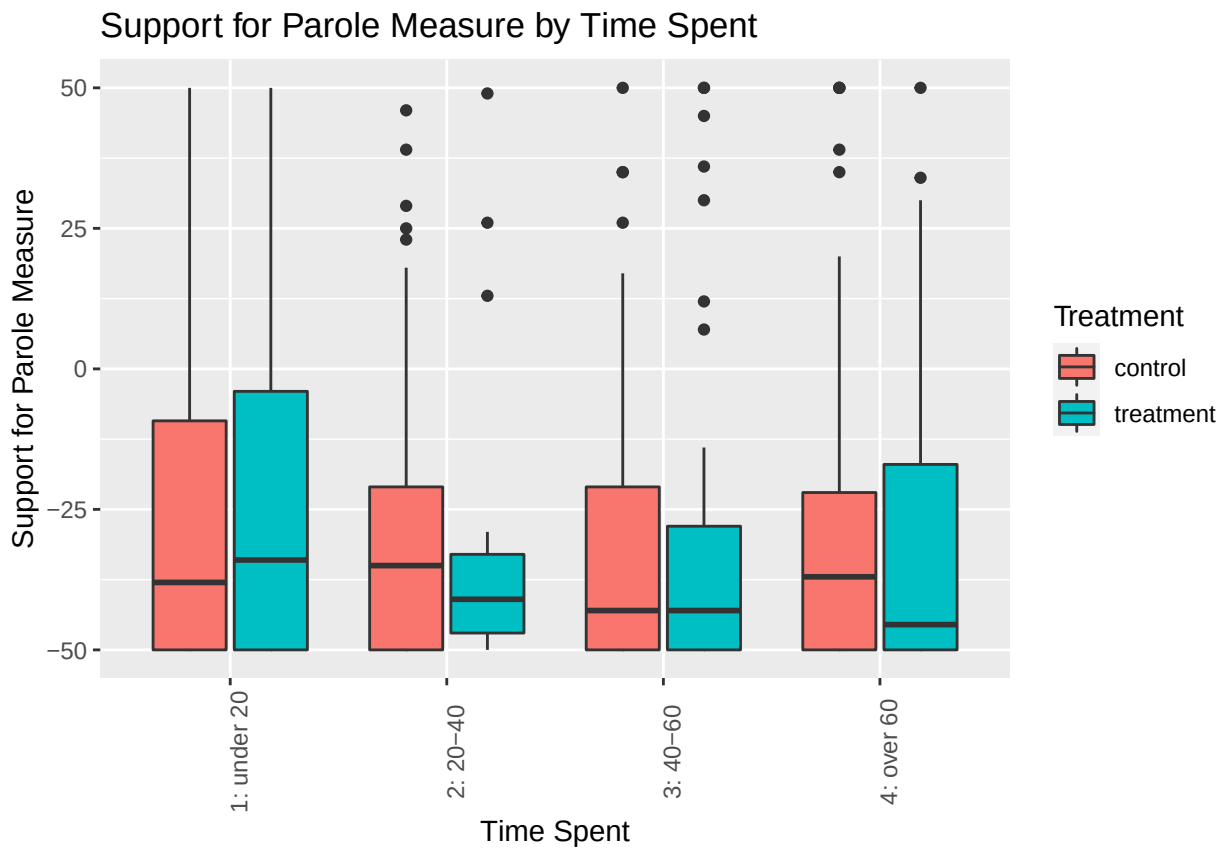



```
summary(final_sm$timeParole)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   18.39   43.93   63.64   68.00 3302.78
```

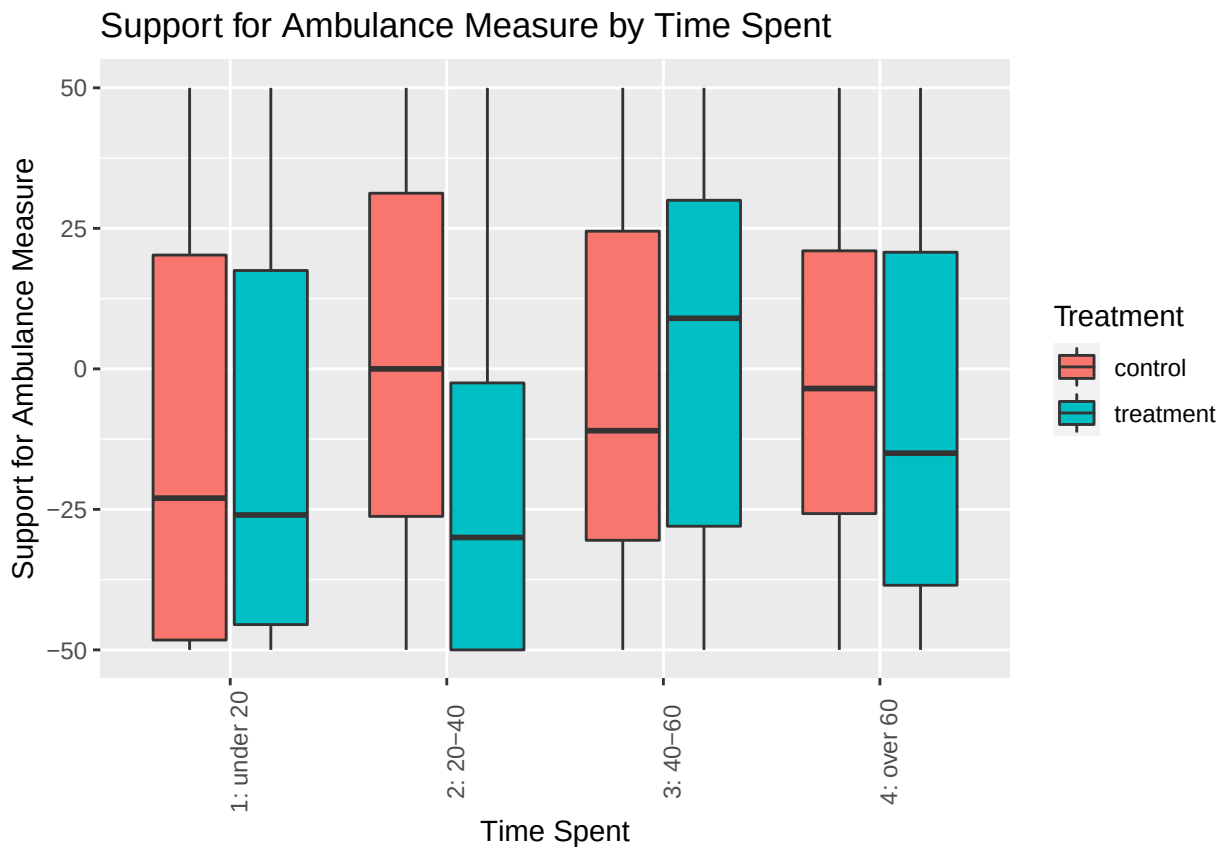
```
ggplot(final_sm, aes(factor(timeParole_bins), parole_Support)) + geom_boxplot(aes(fill = factor(treat)))
ggtitle("Support for Parole Measure by Time Spent") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Time Spent", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



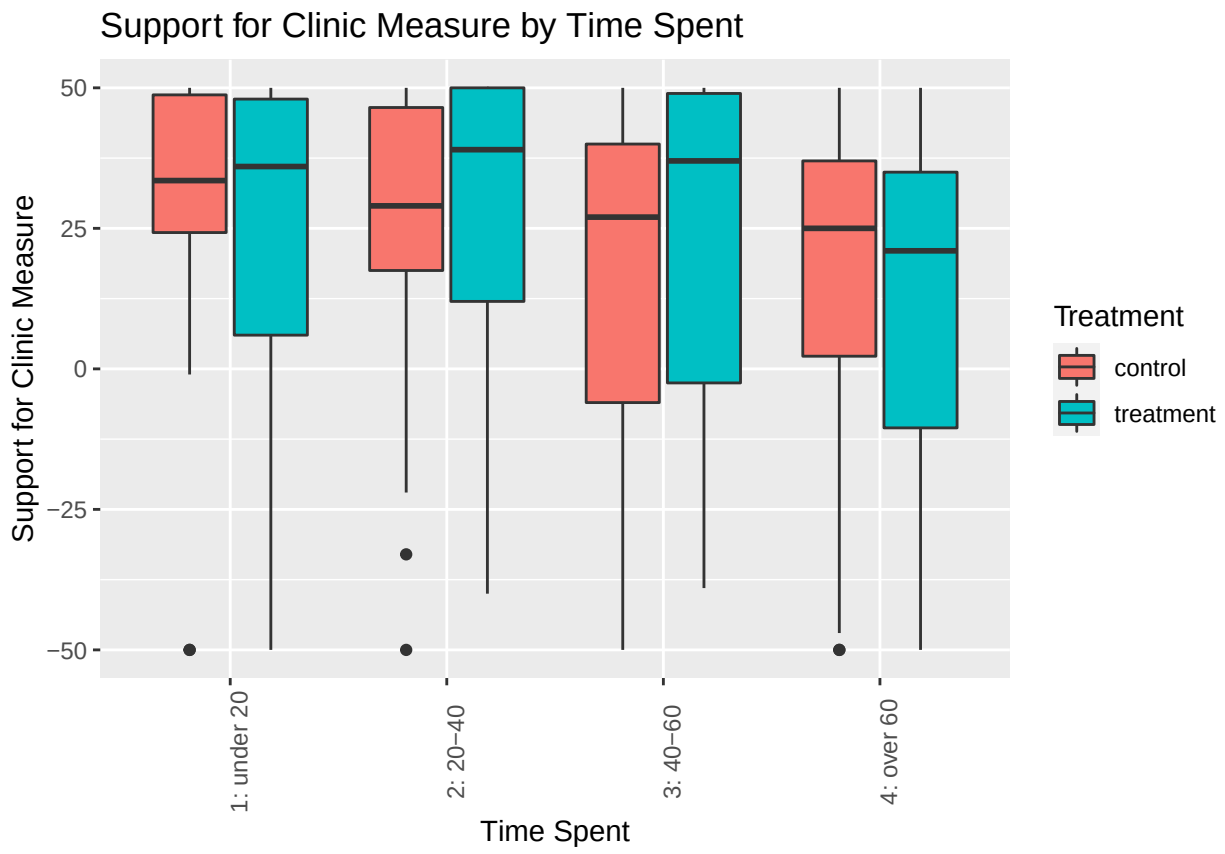

```
ggplot(final_sm, aes(factor(timeAmb_bins), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat)))
ggtitle("Support for Ambulance Measure by Time Spent") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Time Spent", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```




```
ggplot(final_sm, aes(factor(timeClinic_bins), clinic_Support)) + geom_boxplot(aes(fill = factor(treat)))
ggtitle("Support for Clinic Measure by Time Spent") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Time Spent", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```

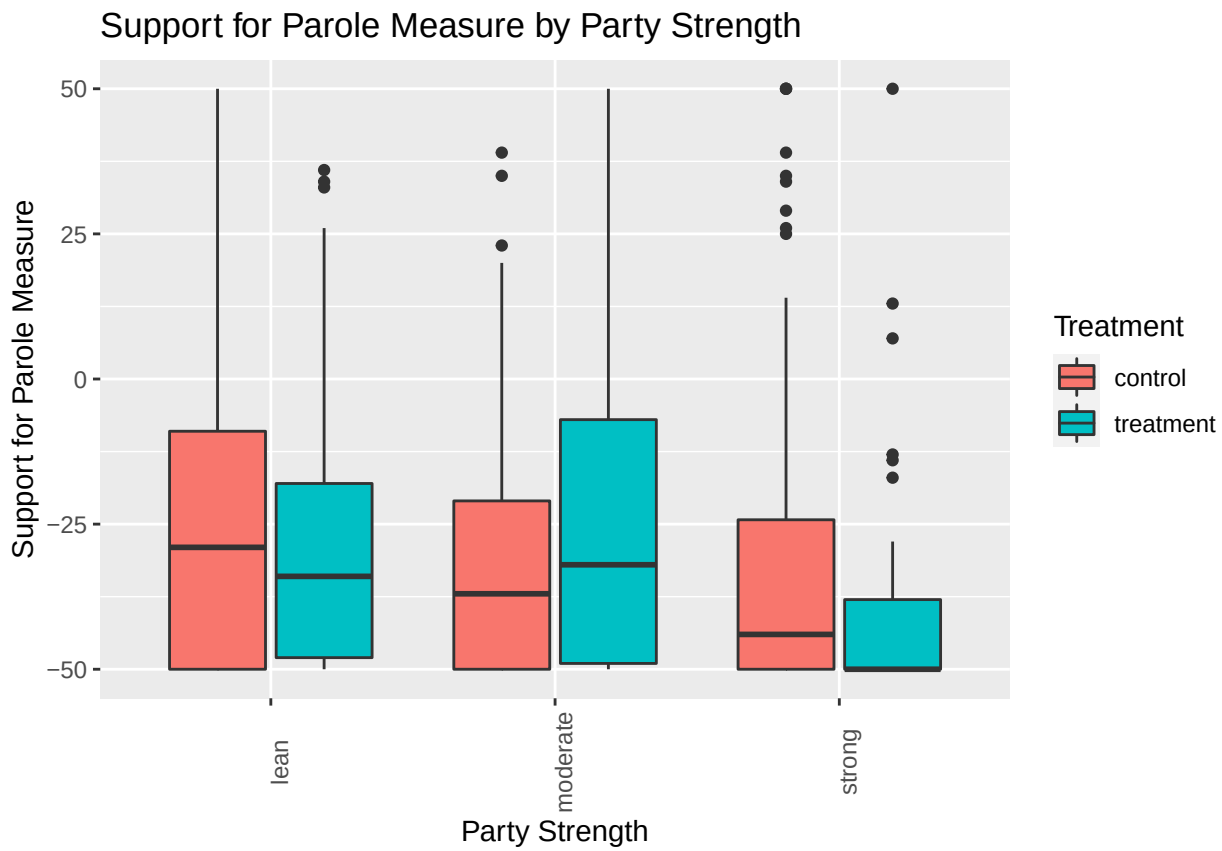



```
summary(final_sm$timeParole)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   18.39   43.93   63.64   68.00 3302.78
```

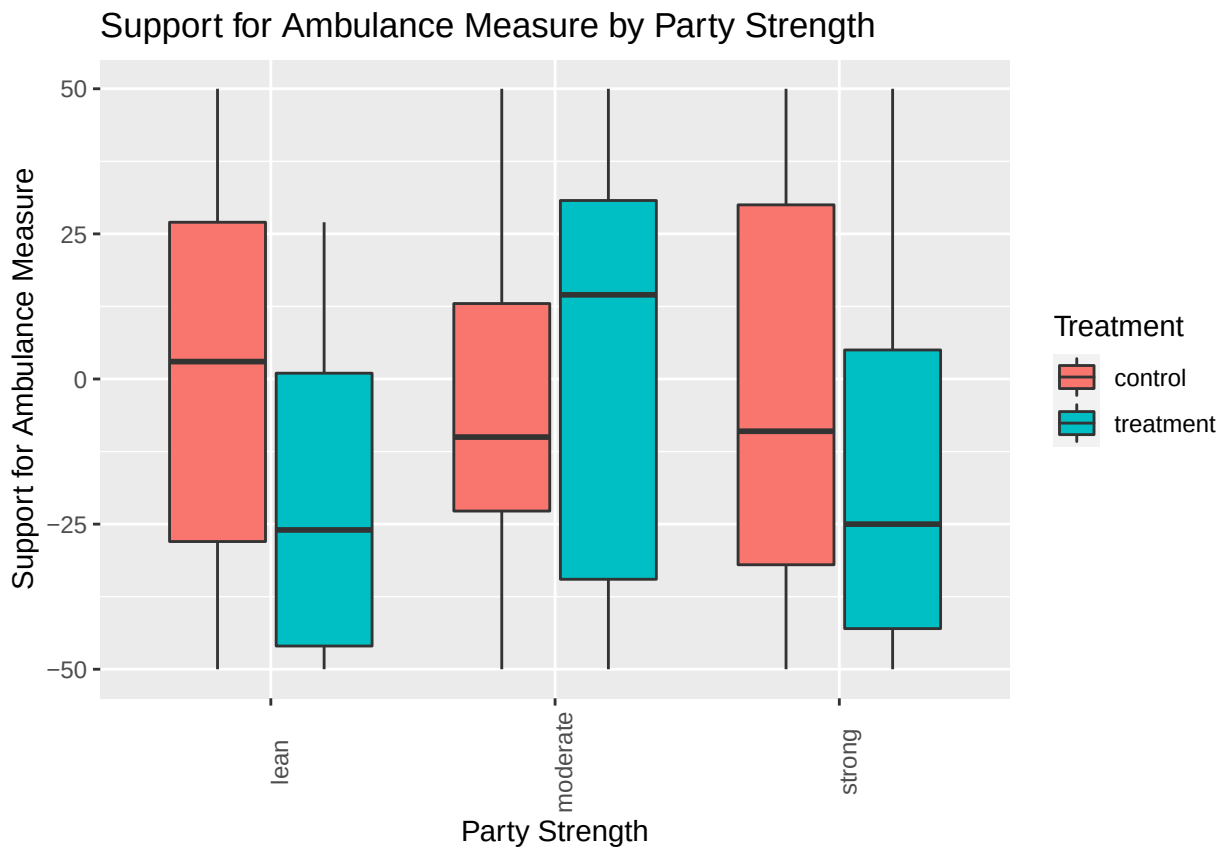
```
ggplot(final_sm, aes(factor(partyStrength), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by Party Strength") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Party Strength", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



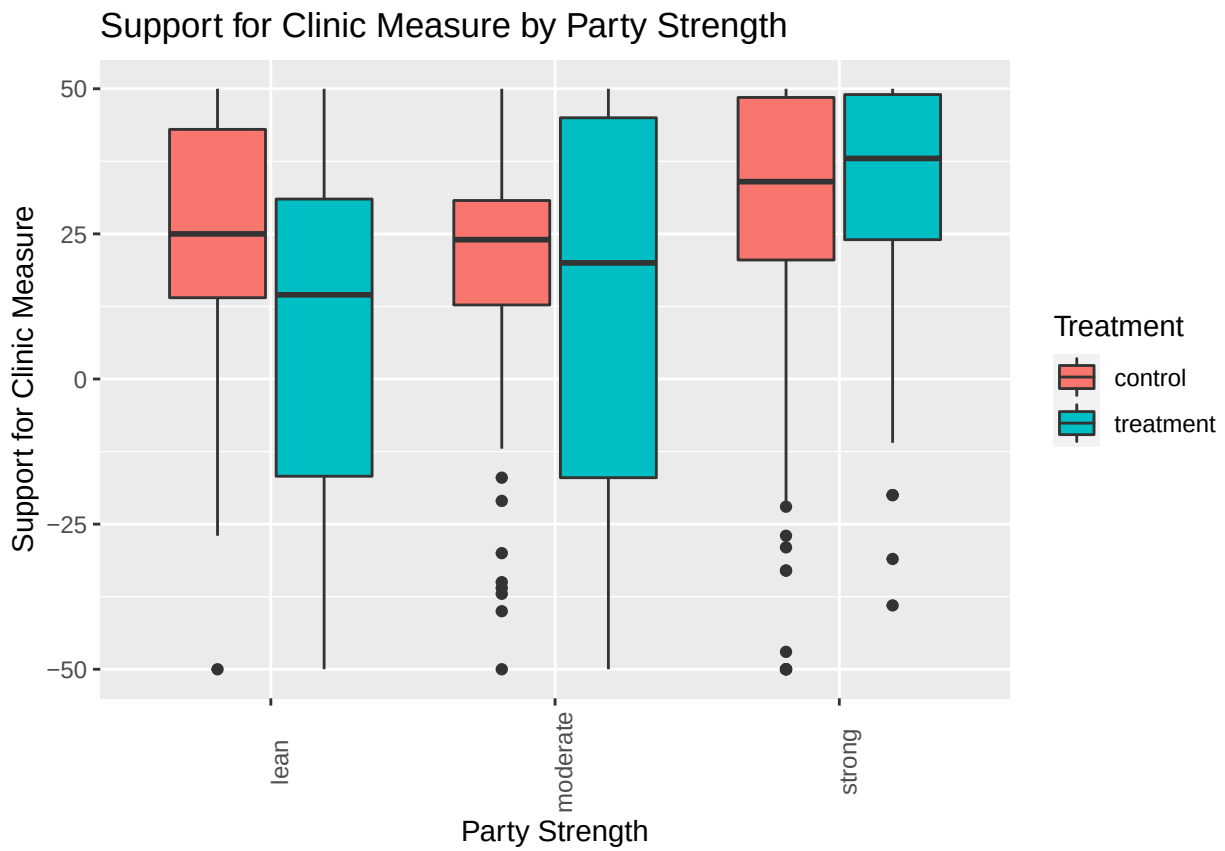

```
ggplot(final_sm, aes(factor(partyStrength), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat)))
ggtitle("Support for Ambulance Measure by Party Strength") + theme(axis.text.x  = element_text(angle=90)
  labs(x = "Party Strength", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```



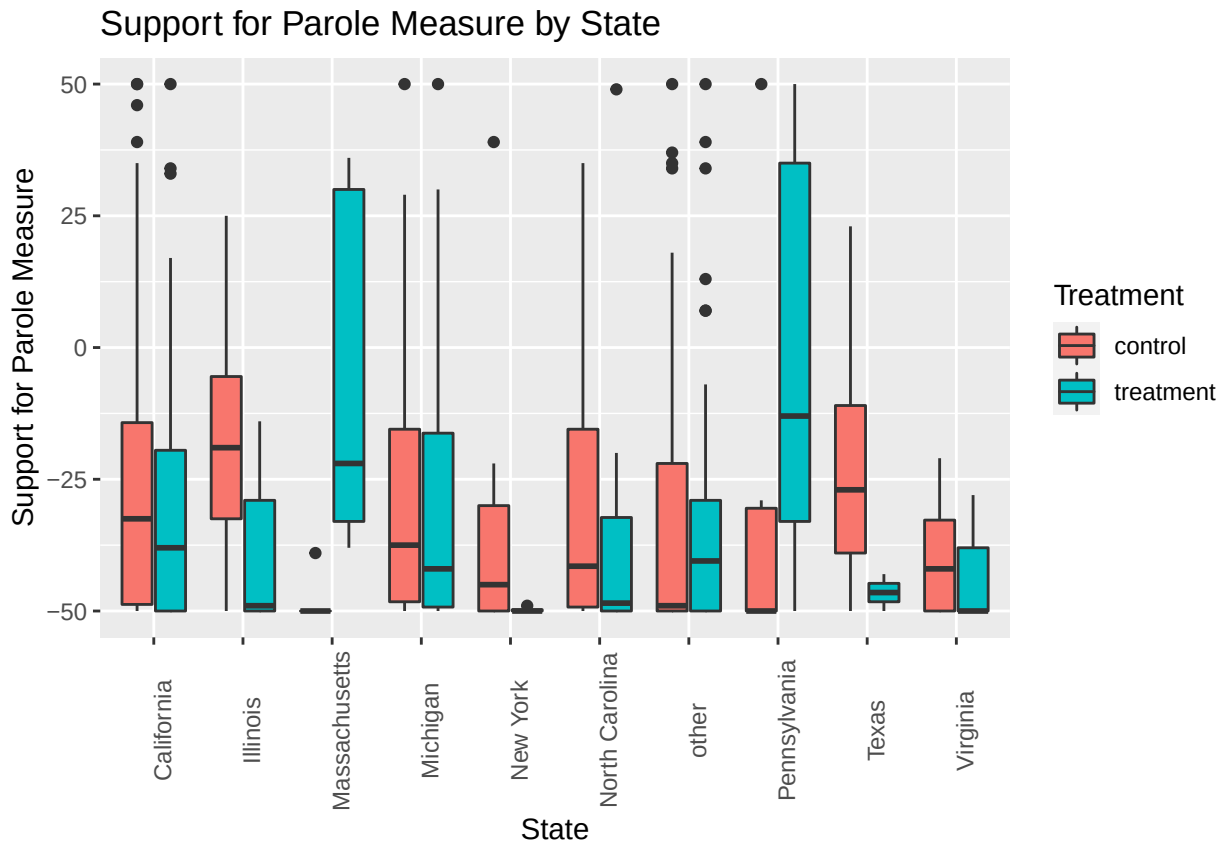

```
ggplot(final_sm, aes(factor(partyStrength), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by Party Strength") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "Party Strength", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



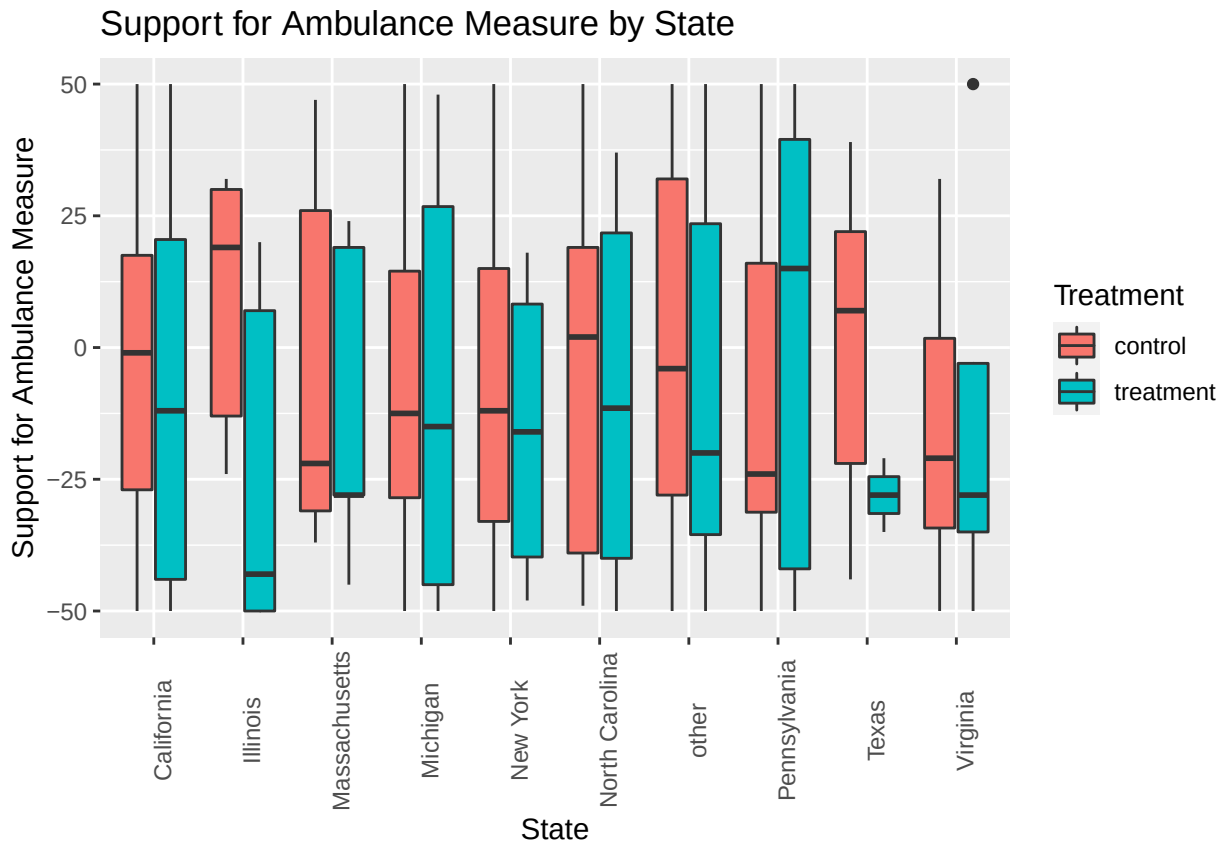

```
ggplot(final_sm, aes(factor(state_bin), parole_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Parole Measure by State") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "State", y = "Support for Parole Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```



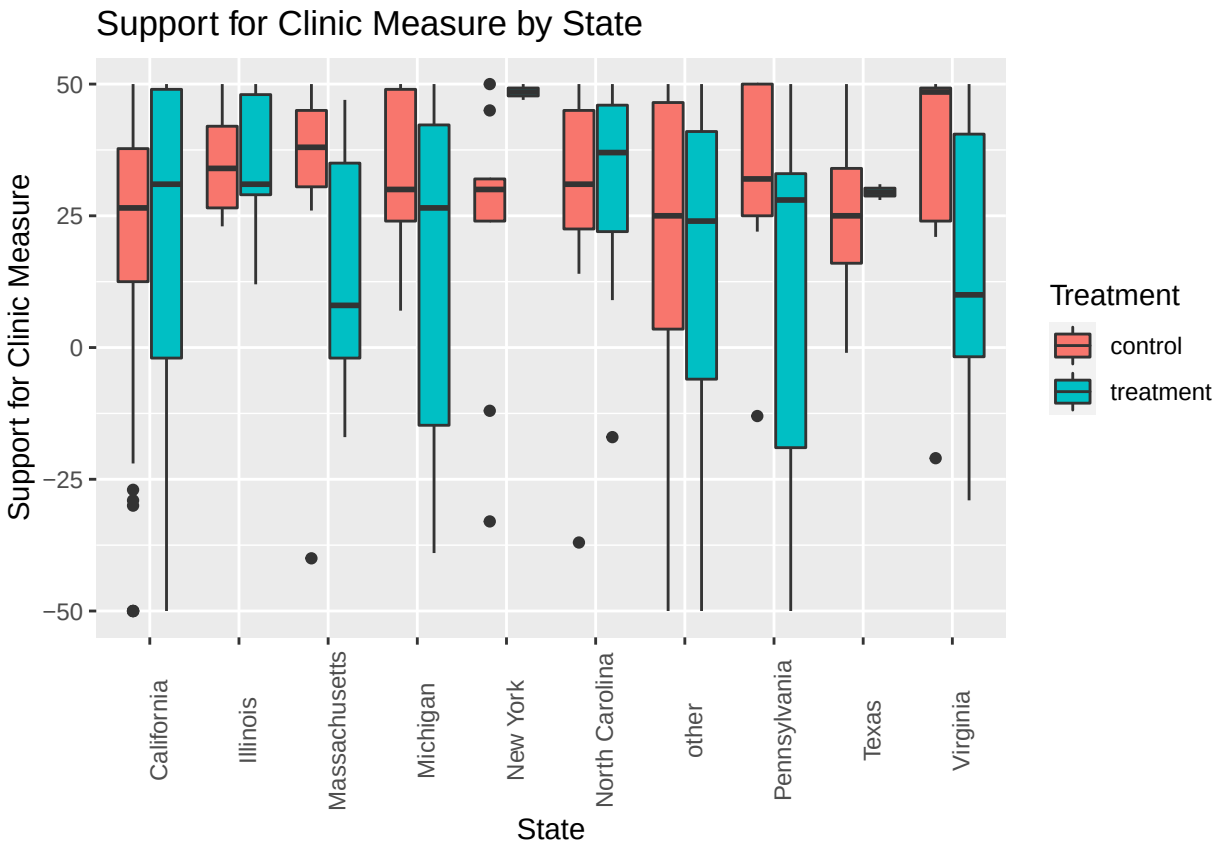

```
ggplot(final_sm, aes(factor(state_bin), ambulance_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Ambulance Measure by State") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "State", y = "Support for Ambulance Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 55 rows containing non-finite values (stat_boxplot).
```




```
ggplot(final_sm, aes(factor(state_bin), clinic_Support)) + geom_boxplot(aes(fill = factor(treat))) +
ggtitle("Support for Clinic Measure by State") + theme(axis.text.x  = element_text(angle=90)) +
  labs(x = "State", y = "Support for Clinic Measure") +
scale_fill_discrete(name="Treatment", labels=c("control", "treatment"))#, breaks=c(0, 1, 2))
```

```
## Warning: Removed 60 rows containing non-finite values (stat_boxplot).
```


Support for Clinic Measure by State
Support for Clinic Measure
50
25
0
−25
−50
California
Illinois
Massachusetts
Michigan
New York
North Carolina
other
Pennsylvania
Texas
Virginia
State
Treatment
control
treatment