

# Speaking Science

## W241 Final Project

Haerang Lee, Aris F, Mumin Khan

March 17th, 2020

## Contents

TODO:	2
Functions	2
Simple Linear Regression	2
Randomization Inference	7

```
d <- fread('speaking_science_data_03-24_clean.csv')
head(d)
```

```
##      start_date      end_date      ip_address duration_in_seconds
## 1: 3/23/2020 20:59 3/23/2020 21:00 196.17.67.134             85
## 2: 3/23/2020 21:00 3/23/2020 21:03 68.105.189.229            185
## 3: 3/23/2020 20:59 3/23/2020 21:03 76.176.54.192             284
## 4: 3/23/2020 21:00 3/23/2020 21:06 107.185.127.204           379
## 5: 3/23/2020 21:04 3/23/2020 21:06 71.6.87.50              163
## 6: 3/23/2020 21:02 3/23/2020 21:06 72.216.72.106           275
##      recorded_date      response_id latitude longitude      mturk_id
## 1: 3/23/2020 21:00 R_3si5y4qwGLzKYPh 34.05440 -118.2440 A1201H00DXBGC3
## 2: 3/23/2020 21:03 R_1jqhwR0mmrLPaoy 36.05881 -115.3104 A12ATVBE1I4567
## 3: 3/23/2020 21:03 R_2ASHk9ILabLrZCB 33.02870 -117.0846 A900V3976AFYF
## 4: 3/23/2020 21:06 R_3n6PLtv1K8spEfC 33.96750 -118.1464 A2C73Y1COWCA51
## 5: 3/23/2020 21:06 R_3kbIZqjBaOkb1pG 37.76880 -122.2620 A10ROXYXMV5MBO
## 6: 3/23/2020 21:06 R_sRUrOCfBuUjTAuB 32.89461 -111.7493 A830LM1ZQC083
##      browser_type browser_version      browser_os browser_resolution
## 1:      Chrome      80.0.3987.149 Windows NT 6.3      1366x768
## 2:      Chrome      80.0.3987.149 Windows NT 10.0     1536x864
## 3:      Chrome      79.0.3945.136      Android 7.0      360x640
## 4:      Chrome      80.0.3987.149 Windows NT 10.0     1920x1080
## 5:      Chrome      80.0.3987.132      Macintosh      1440x900
## 6:      Chrome      80.0.3987.149 Windows NT 10.0     1536x864
##      time_read_intro time_read_article credibility importance q1 q1_correct q2
## 1:          2.607          17.922             5           4 1           0 3
## 2:          7.588          29.649             5           4 3           1 5
## 3:         13.345         166.381             6           7 1           0 3
## 4:          2.854          67.943             7           6 4           0 3
## 5:         18.607          47.366             6           6 2           0 1
## 6:         10.128         163.601             7           7 3           1 1
##      q2_correct q3 q3_correct q4 q4_correct q5 q5_correct q6 q6_correct
```

```
## 1:      1 2      0 1      0 3      0 NA      0
## 2:      0 3      1 1      0 4      1 1      1
## 3:      1 1      0 4      0 4      1 1      1
## 4:      1 3      1 2      0 2      0 NA      0
## 5:      0 1      0 2      0 1      0 1      1
## 6:      0 3      1 1      0 3      0 1      1
##      questions_correct time_answering_questions donation time_donation
## 1:      1      22.689      50      7.384
## 2:      4     125.852      1      4.044
## 3:      3      67.334      1      7.657
## 4:      2     287.135      0      5.991
## 5:      1      49.709     50      7.145
## 6:      3      72.550      0     14.611
##      city state   zip treatment
## 1: Los Angeles   CA 90009      1
## 2:   Las Vegas   NV 89113      0
## 3:   San Diego   CA 92127      0
## 4: Bell Gardens   CA 90201      1
## 5:   Vallejo     CA 94589      1
## 6: Casa Grande   AZ 85122      1
```

## TODO:

1. Discussion on compliance (time read)
2. Generalizability
3. Randomization Inference

## Functions

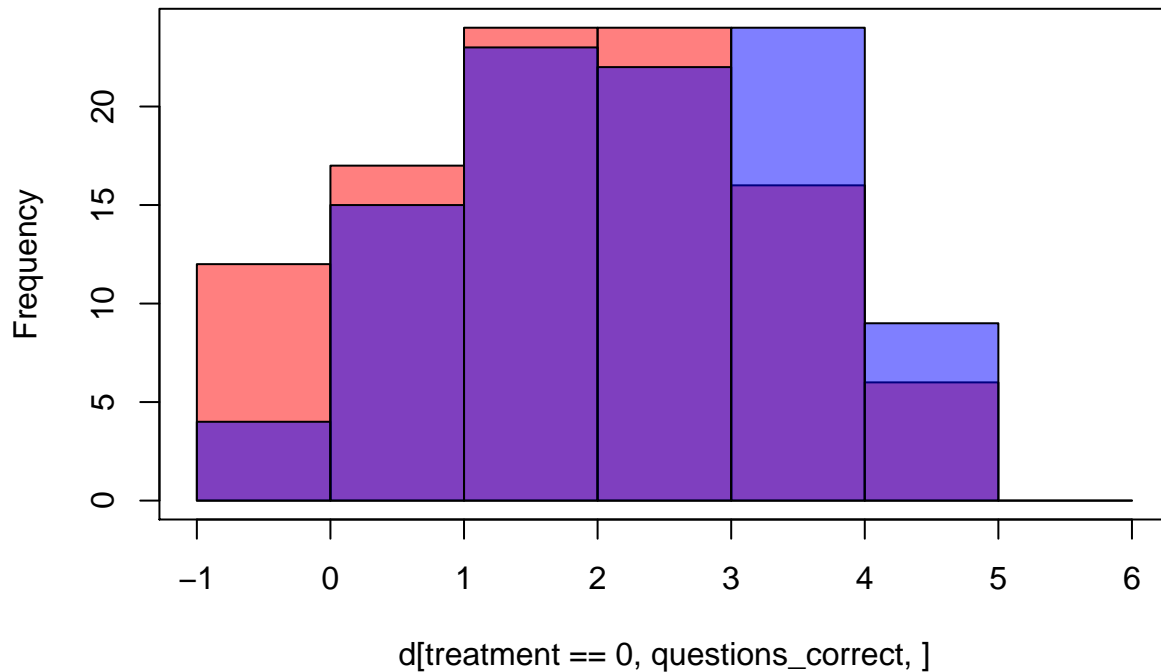
### Simple Linear Regression

```
mod <- lm(questions_correct ~ treatment, data=d)
stargazer(mod, type = "text")
```

```
##
## =====
##                               Dependent variable:
##                               -----
##                               questions_correct
##                               -----
## treatment                      0.430**
##                               (0.197)
##
## Constant                      2.333***
##                               (0.139)
##
## -----
## Observations                   196
## R2                             0.024
## Adjusted R2                   0.019
## Residual Std. Error          1.380 (df = 194)
## F Statistic                   4.746** (df = 1; 194)
```

```
## =====
## Note:                *p<0.1; **p<0.05; ***p<0.01
hist(d[treatment == 0, questions_correct,], col=rgb(1,0,0,0.5), breaks=seq(-1,6, by=1))
hist(d[treatment == 1, questions_correct,], col=rgb(0,0,1,0.5), breaks=seq(-1,6, by=1), add = T)
box()
```

**Histogram of d[treatment == 0, questions\_correct, ]**



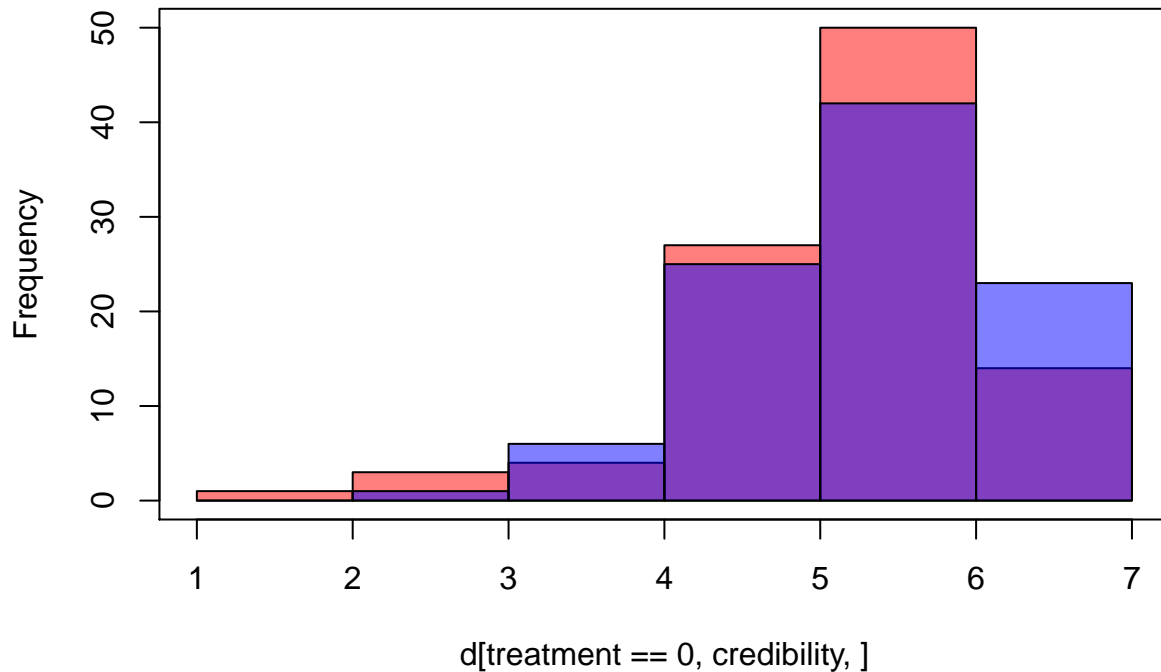
```
#Non parametric test
mod <- lm(credibility ~ treatment, data=d)
summary(mod)
```

```
##
## Call:
## lm(formula = credibility ~ treatment, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.6566 -0.6566  0.1753  0.3434  1.3434
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.65657    0.09306  60.781  <2e-16 ***
## treatment    0.16818    0.13229   1.271   0.205
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.926 on 194 degrees of freedom
```

```
## Multiple R-squared:  0.008262,   Adjusted R-squared:  0.00315
## F-statistic: 1.616 on 1 and 194 DF,  p-value: 0.2052
```

```
hist(d[treatment == 0, credibility,], col=rgb(1,0,0,0.5), breaks=seq(1,7, by=1))
hist(d[treatment == 1, credibility,], col=rgb(0,0,1,0.5), breaks=seq(1,7, by=1), add = T)
box()
```

## Histogram of d[treatment == 0, credibility, ]

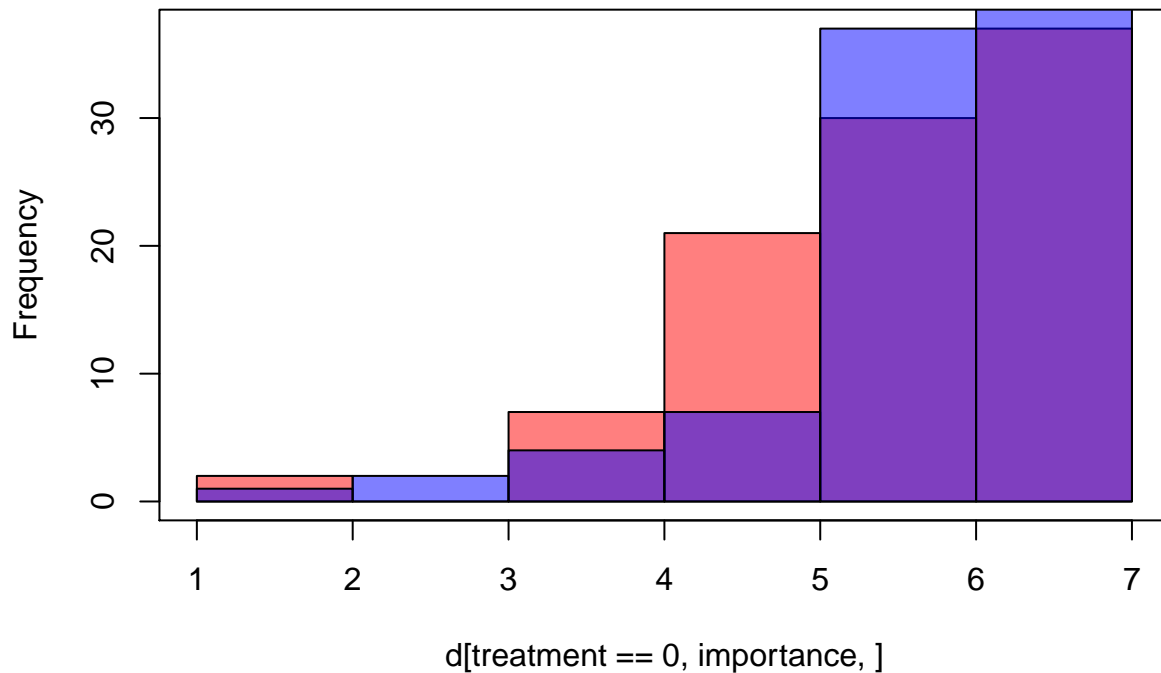


```
mod <- lm(importance ~ treatment, data=d)
stargazer(mod, type = "text")
```

```
##
## =====
##                               Dependent variable:
##                               -----
##                               importance
## -----
## treatment                     0.289*
##                               (0.162)
##
## Constant                     5.918***
##                               (0.114)
## -----
## Observations                  194
## R2                            0.016
## Adjusted R2                   0.011
## Residual Std. Error          1.126 (df = 192)
```

```
## F Statistic          3.190* (df = 1; 192)
## =====
## Note:                *p<0.1; **p<0.05; ***p<0.01
hist(d[treatment == 0, importance,], col=rgb(1,0,0,0.5), breaks=seq(1,7, by=1))
hist(d[treatment == 1, importance,], col=rgb(0,0,1,0.5), breaks=seq(1,7, by=1), add = T)
box()
```

**Histogram of d[treatment == 0, importance, ]**



```
mod <- lm(time_read_article ~ treatment, data=d)
summary(mod)
```

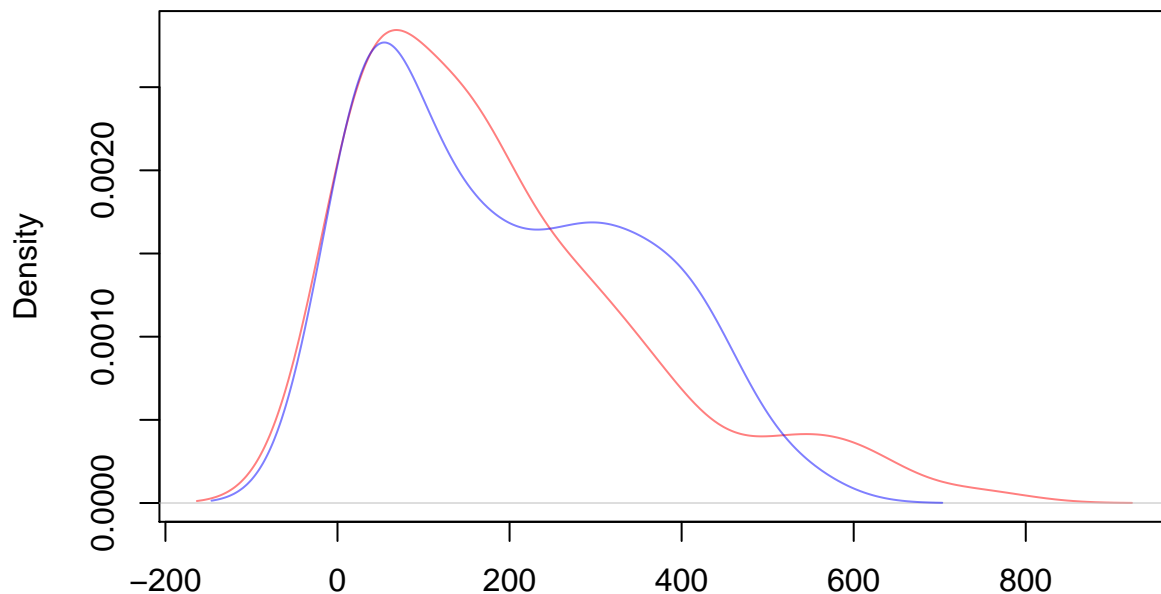
```
##
## Call:
## lm(formula = time_read_article ~ treatment, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -179.58 -138.20  -32.26   98.13  555.57
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   188.318     15.971  11.792  <2e-16 ***
## treatment       7.592     22.702   0.334    0.738
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.9 on 194 degrees of freedom
```

```
## Multiple R-squared:  0.0005761, Adjusted R-squared:  -0.004576
## F-statistic: 0.1118 on 1 and 194 DF,  p-value: 0.7384
```

```
d1 <- density(d[treatment == 0, time_read_article,])
d2 <- density(d[treatment == 1, time_read_article,])
```

```
plot(d1, col=rgb(1,0,0,0.5))
lines(d2, col=rgb(0,0,1,0.5))
```

**density.default(x = d[treatment == 0, time\_read\_article, ])**



N = 99 Bandwidth = 59.85

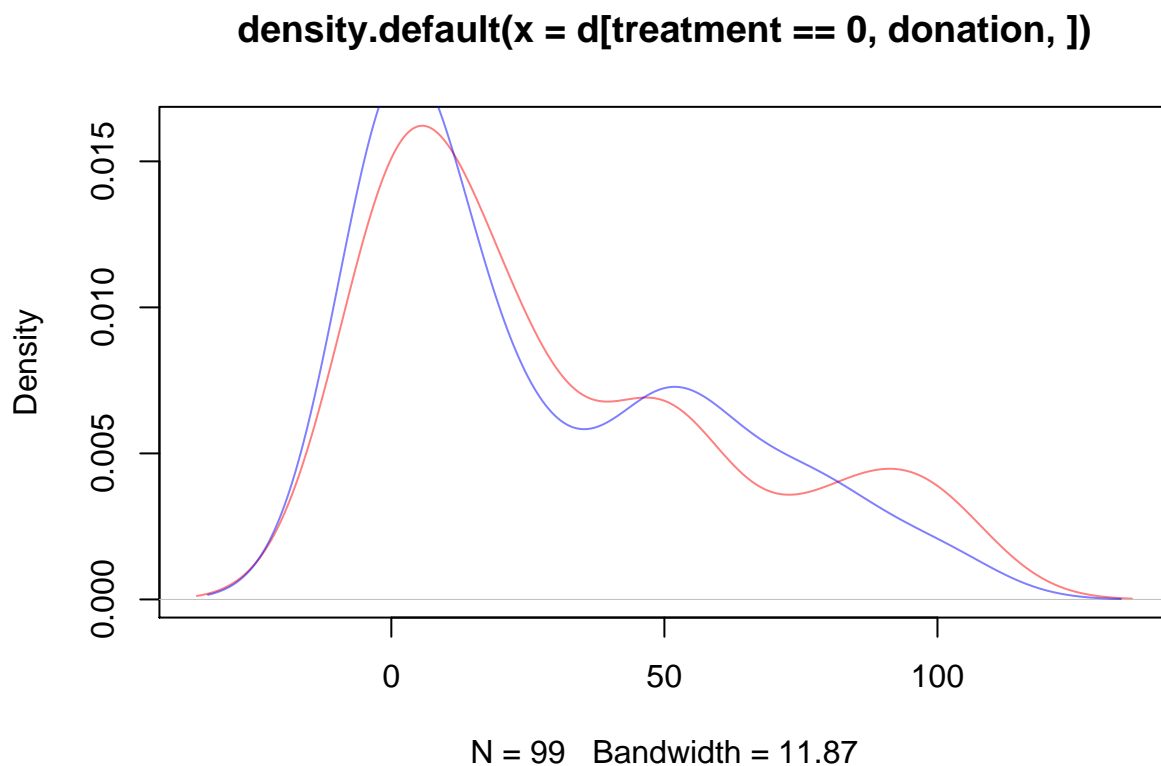
```
mod <- lm(donation ~ treatment + time_donation, data=d)
summary(mod)
```

```
##
## Call:
## lm(formula = donation ~ treatment + time_donation, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -32.17  -27.72  -11.42   21.25   72.90
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   32.9374     4.2204   7.804 3.69e-13 ***
## treatment     -3.2382     4.6070  -0.703   0.483
## time_donation -0.1283     0.1586  -0.809   0.420
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 32.11 on 193 degrees of freedom
## Multiple R-squared:  0.006508,    Adjusted R-squared:  -0.003787
## F-statistic: 0.6321 on 2 and 193 DF,  p-value: 0.5326

d1 <- density(d[treatment == 0, donation,])
d2 <- density(d[treatment == 1, donation,])

plot(d1, col=rgb(1,0,0,0.5))
lines(d2, col=rgb(0,0,1,0.5))
```



## Randomization Inference

Testing the sharp null hypothesis that the treatment has no effect for anyone.

```
n <- 10000

# Initialize randomization inference
ate_questions_correct <- rep(NA, n)
ate_credibility       <- rep(NA, n)
ate_importance        <- rep(NA, n)
ate_time_read_article <- rep(NA, n)
ate_donation          <- rep(NA, n)

for(i in 1:n){
  d_ri <- na.omit(d)
```

```

d_ri$treatment <- sample(d_ri$treatment)

# Is there any way to do this with a loop and col.names(ri)? I hate R...
ate_questions_correct[i] <- d_ri[, .('group_mean' = mean(questions_correct)), by=treatment][, diff(gr
ate_credibility[i] <- d_ri[, .('group_mean' = mean(credibility)), by=treatment][, diff(gr
ate_importance[i] <- d_ri[, .('group_mean' = mean(importance)), by=treatment][, diff(gr
ate_time_read_article[i] <- d_ri[, .('group_mean' = mean(time_read_article)), by=treatment][, diff(gr
ate_donation[i] <- d_ri[, .('group_mean' = mean(donation)), by=treatment][, diff(gr

}

ri <- data.table(
  questions_correct = ate_questions_correct,
  credibility = ate_credibility,
  importance = ate_importance,
  time_read_article = ate_time_read_article,
  donation = ate_donation
)

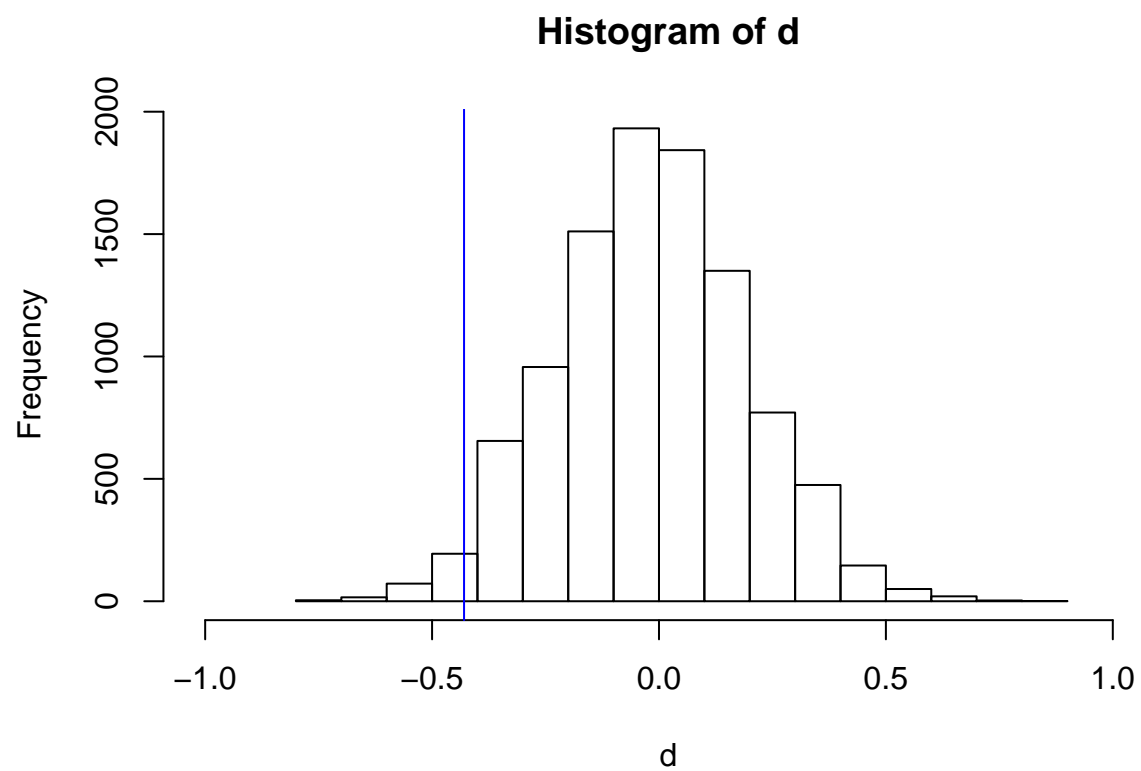
visualize_ri <- function(d, ate) {
  hist(d, xlim = c(min(d)-0.25, max(d)+0.25))
  abline(v=ate, col='blue', lwd = 1)
}

# Actual ATE's
ate_questions_correct <- d[, .('group_mean' = mean(questions_correct)), by=treatment][, diff(group_mean
ate_credibility <- d[, .('group_mean' = mean(credibility)), by=treatment][, diff(group_mean
ate_importance <- d[, .('group_mean' = mean(importance)), by=treatment][, diff(group_mean
ate_time_read_article <- d[, .('group_mean' = mean(time_read_article)), by=treatment][, diff(group_mean
ate_donation <- d[, .('group_mean' = mean(donation)), by=treatment][, diff(group_mean

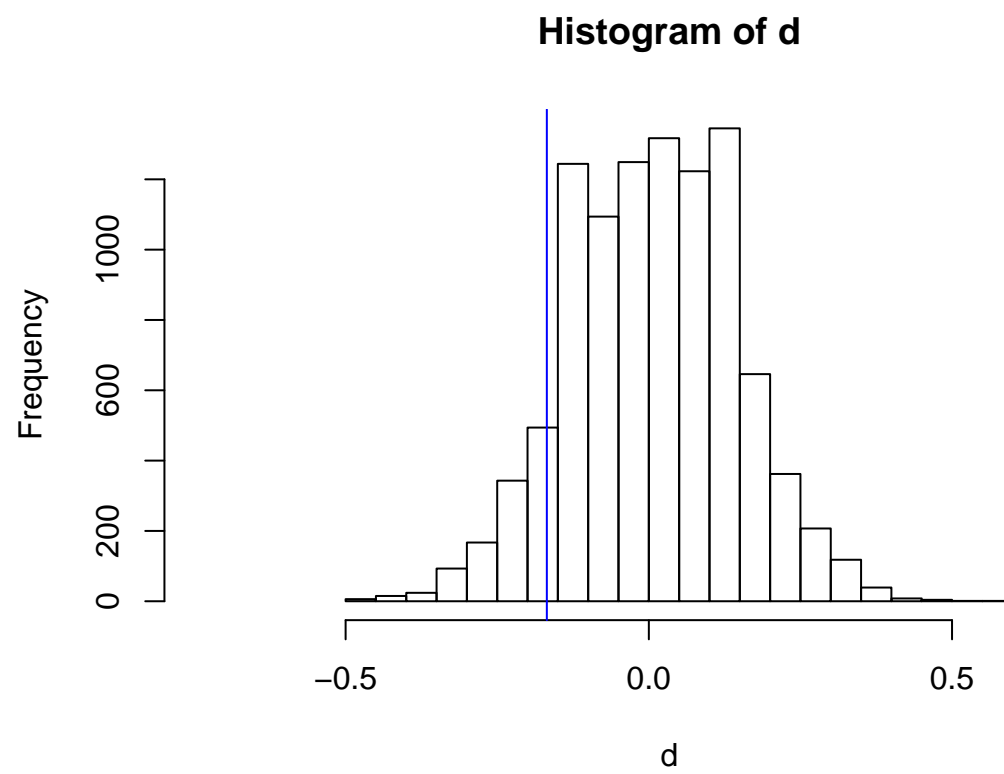
visualize_ri(ri$questions_correct, ate_questions_correct)

```

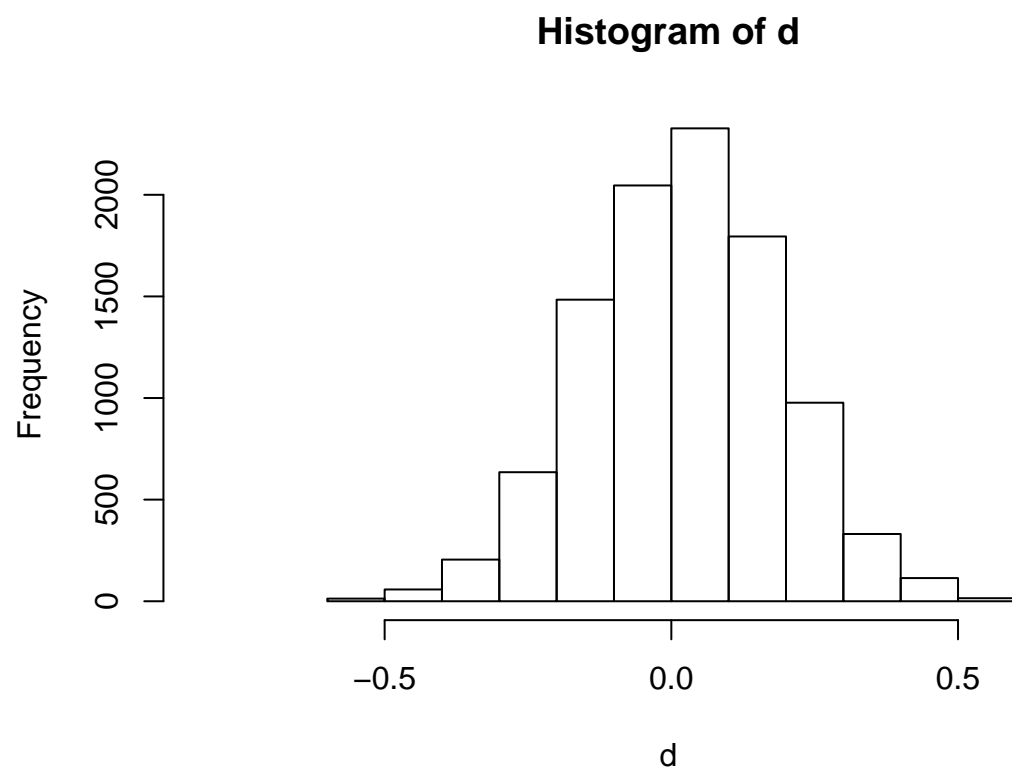




```
visualize_ri(ri$credibility, ate_credibility)
```

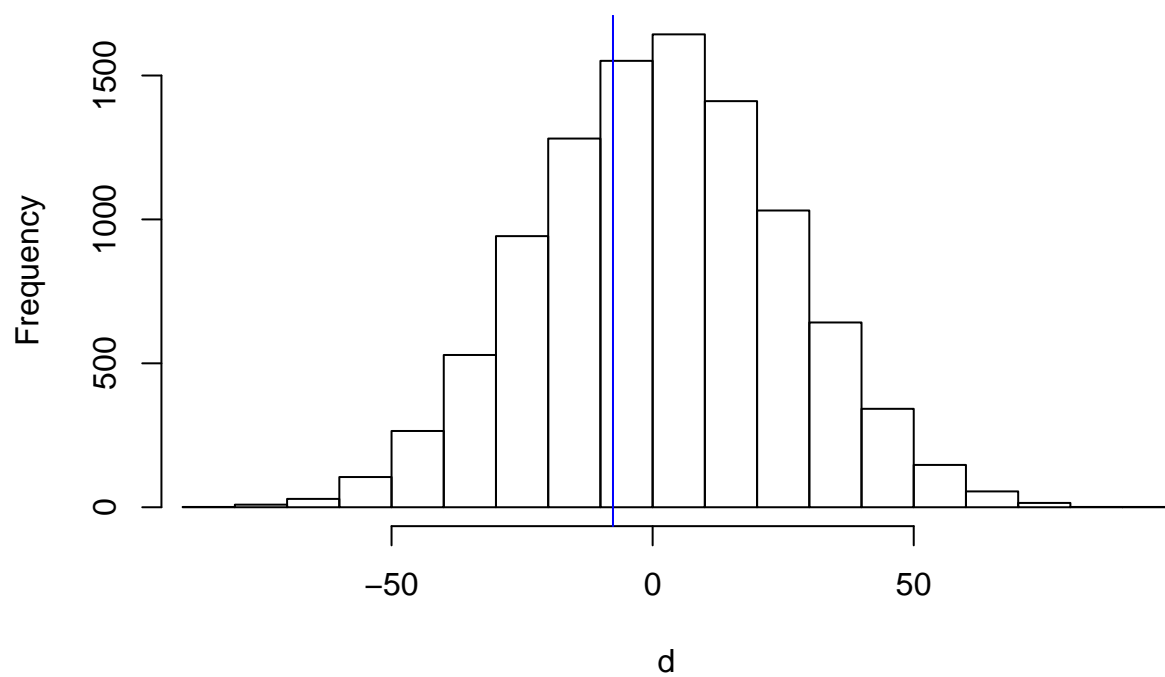


```
visualize_ri(ri$importance, ate_importance)
```



```
visualize_ri(ri$time_read_article, ate_time_read_article)
```

**Histogram of d**



```
visualize_ri(ri$donation, ate_donation)
```

**Histogram of d**

