```
# showing the data:
head(data)
```

```
##       V1 session_starts treatment   team handed_out passed_by
##    <int>          <num>     <int> <char>      <int>     <int>
## 1:     0              7         1     js          1         4
## 2:     1            127         1     js          3        11
## 3:     2            247         1     js          5        18
## 4:     3            367         1     js          7        29
## 5:     4            487         1     js          6        17
## 6:     5            607         0     js          0        20
##    session_within_session success_rate   day person session
##                     <int>        <num> <int> <char>   <num>
## 1:                      0    0.2500000     0              0
## 2:                      1    0.2727273     0              0
## 3:                      2    0.2777778     0              0
## 4:                      3    0.2413793     0              0
## 5:                      4    0.3529412     0              0
## 6:                      0    0.0000000     0              1
```

```
model_not_interacted = data[, lm(success_rate ~ treatment + factor(team) + factor(per
son))]
model_interacted_person = data[, lm(success_rate ~ treatment*factor(person))]
model_interacted_team = data[, lm(success_rate ~ treatment*factor(team))]

stargazer(model_not_interacted, model_interacted_person, model_interacted_team, type
= "text")
```

```
##
## ================================================================================
======
##                                          Dependent variable:
##                          ---------------------------------------------------------
------
##                                              success_rate
##                                (1)                  (2)                  (3)
## --------------------------------------------------------------------------------
------
## treatment                     0.007                0.008               -0.050
##                              (0.039)              (0.075)              (0.062)
##
## factor(team)vw               0.383***                                  -0.004
##                              (0.082)                                   (0.058)
##
## factor(person)j               0.121*               0.044
##                              (0.070)              (0.094)
##
## factor(person)s               0.026               -0.009
##                              (0.071)              (0.101)
##
## factor(person)v              -0.408***            -0.156*
##                              (0.081)              (0.080)
##
## factor(person)w              -0.320***            -0.011
##                              (0.081)              (0.080)
##
## treatment:factor(person)j                         -0.042
##                                                   (0.146)
##
## treatment:factor(person)s                         -0.124
##                                                   (0.146)
##
## treatment:factor(person)v                          0.068
##                                                   (0.112)
##
## treatment:factor(person)w                         -0.038
##                                                   (0.112)
##
## treatment:factor(team)vw                                                0.069
##                                                                        (0.082)
##
## Constant                     0.271***             0.365***             0.335***
##                              (0.045)              (0.052)              (0.042)
##
## --------------------------------------------------------------------------------
------
## Observations                   179                  179                  179
## R2                            0.144                0.046                0.007
## Adjusted R2                   0.114               -0.005               -0.010
## Residual Std. Error      0.256 (df = 172)     0.273 (df = 169)     0.274 (df =
175)
## F Statistic          4.809*** (df = 6; 172) 0.898 (df = 9; 169) 0.432 (df =
3; 175)
```

```
## ================================================================================
======
## Note:                                                          *p<0.1; **p<0.05; ***
p<0.01
```

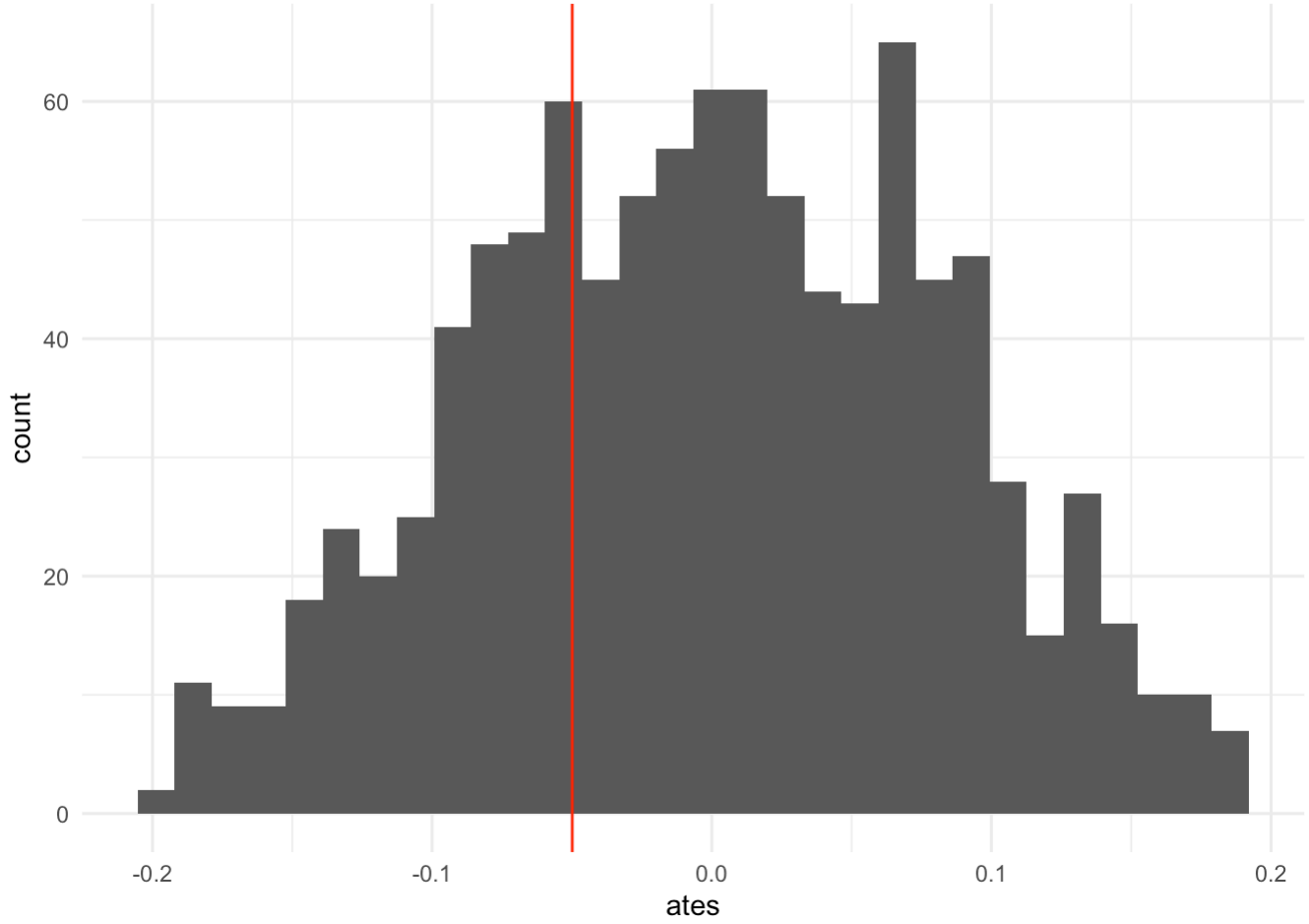
```
for (pers in data[, unique(person)]) {
  ate = data[(treatment == 1 & person == pers ), mean(success_rate)] - data[(treatmen
t == 0 & person == pers ), mean(success_rate)]
  print(paste("The ATE for person", pers, "is", ate))
}
```

```
## [1] "The ATE for person  is 0.00753246958875192"
## [1] "The ATE for person v is 0.0759355185148529"
## [1] "The ATE for person w is -0.0303782389108476"
## [1] "The ATE for person j is -0.033988633127565"
## [1] "The ATE for person s is -0.116818783250665"
```

```
for (tm in data[, unique(team)]) {
  ate = data[(treatment == 1 & team == tm ), mean(success_rate)] - data[(treatment ==
0 & team == tm ), mean(success_rate)]
  print(paste("The ATE for team", tm, "is", ate))
}
```

```
## [1] "The ATE for team js is -0.0499080368127092"
## [1] "The ATE for team vw is 0.0192059636860158"
```

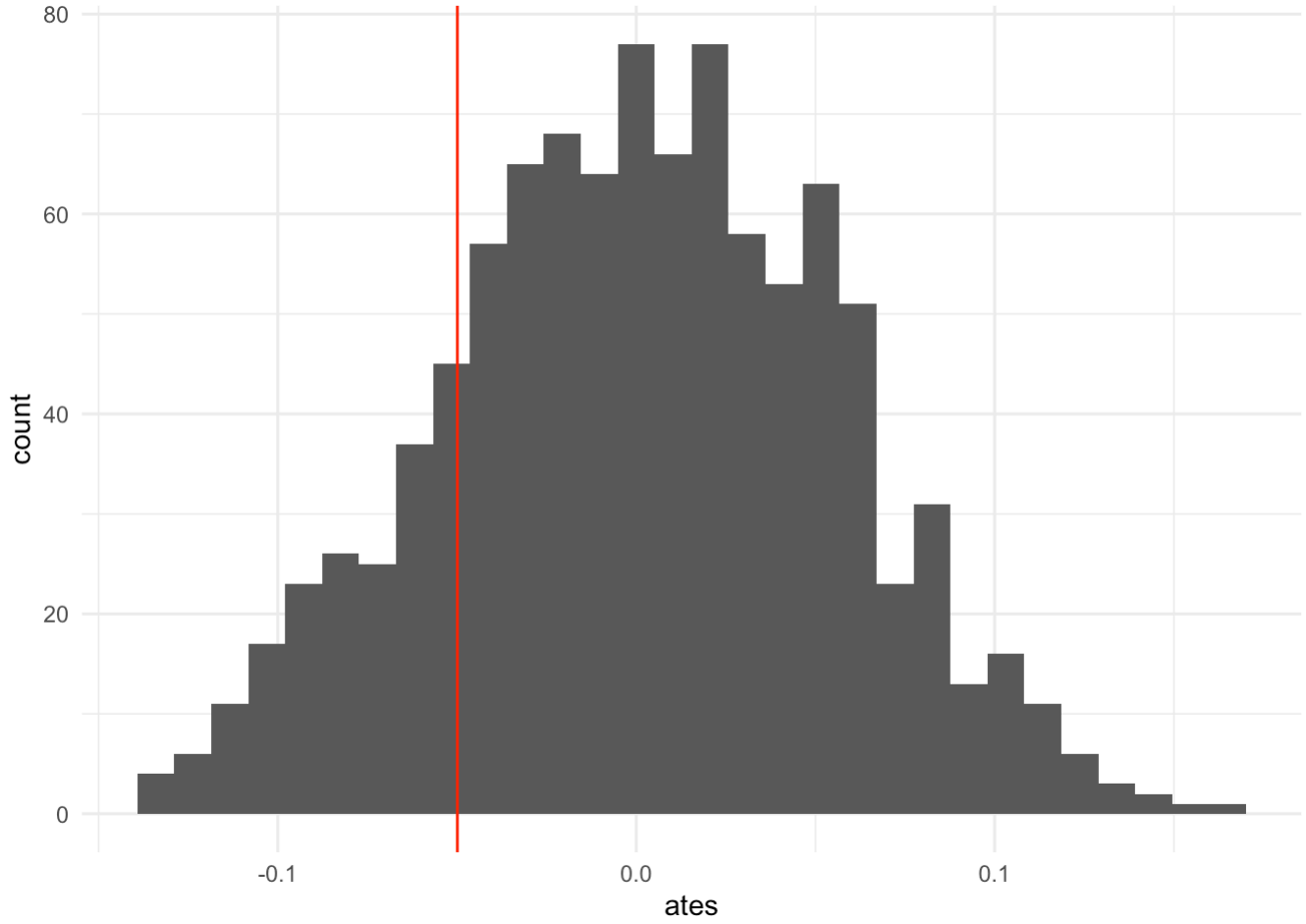
```
# This is making sure, that each minute within a session is assigned to the same trea
tment
rel_data = data[team == "js"]
real_ate = mean(rel_data[treatment == 1, success_rate]) - mean(rel_data[treatment ==
0, success_rate])
sessions = unique(rel_data[, .(session, day)])
ates = c()
n = nrow(sessions)
for (i in 1:1000) {
  treatments <- if (n %% 2 == 0) {
    rep(c(0, 1), length.out = n)
  } else {
    c(rep(c(0, 1), length.out = n - 1), sample(c(0, 1), 1))  # If odd, add one extra
random assignment
  }
  sessions[, treatment_ri := sample(treatments)]
  # merge by session and day
  merged = merge(rel_data, sessions, by = c("session", "day"))
  ate = mean(merged[treatment_ri == 1, success_rate]) - mean(merged[treatment_ri ==
0, success_rate])
  ates = c(ates, ate)
}
# plot hist with vertical line for real_ate and print p value
ggplot() + geom_histogram(aes(x = ates), bins = 30) + geom_vline(xintercept = real_at
e, color = "red") + theme_minimal()
```




```
print(paste("The p-value is", sum(abs(ates) >= abs(real_ate)) / length(ates)))
```

```
## [1] "The p-value is 0.604"
```

```
# This will assign every minute at random but at least being balanced
rel_data = data[team == "js"]
real_ate = mean(rel_data[treatment == 1, success_rate]) - mean(rel_data[treatment ==
0, success_rate])
ates = c()
n = nrow(rel_data)
for (i in 1:1000) {
  treatments <- if (n %% 2 == 0) {
    rep(c(0, 1), length.out = n)
  } else {
    c(rep(c(0, 1), length.out = n - 1), sample(c(0, 1), 1))  # If odd, add one extra
random assignment
  }
  rel_data[, treatment_ri := sample(treatments)]
  # merge by session and day
  ate = mean(rel_data[treatment_ri == 1, success_rate]) - mean(rel_data[treatment_ri
== 0, success_rate])
  ates = c(ates, ate)
}
ggplot() + geom_histogram(aes(x = ates), bins = 30) + geom_vline(xintercept = real_at
e, color = "red") + theme_minimal()
```




```
(p_value = sum(abs(ates) <= abs(real_ate)) / length(ates))
```

```
## [1] 0.621
```