Analyzing Data from a Randomized Experiment in R

Andre Gray

June 24, 2025

Contents

1 Estimating the effect of Drug X on Cholesterol

Suppose I want to estimate the effect of a new drug on people's cholesterol levels. I collect a sample of people that may have different baseline characteristics. I assign them randomly to the treatment drug or placebo, and then test their cholesterol at endline. When I compare average endline cholesterol between treated and control, how do I confirm that this "treatment effect" is significant?

1.1 Simulating Randomized Data

Let's first build some data to work with. Cholesterol for person i is our outcome, y_i. Each person comes into the experiment with a baseline cholesterol level b_i, and they get treated $T = 1$ or not $T = 0$. In this simulation, I am going to pick a TRUE effect size of treatment, that is our job to estimate with data. The true treatment effect is -15, that is, the drug lowers your cholesterol on average by 15mg/dL. So the outcome for person i is:

$$y_i = b_i - 15T + \epsilon_i \tag{1}$$

The ϵ_i is a random variable that captures all the residual stuff that may affect a person's cholesterol, including any unobserved variables like family background, etc. We'll assume this is normally distributed, with some mean and standard deviation. Let's create out data in R:

```r
#These are the 2 R packages we will use
library(ggplot2)
library(dplyr)
```

```
##
## 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
```

```r
rm(list = ls())       # Clear variables

#control randomization reproducability
set.seed(123)

# Create a sample size
n <- 100

#give each person a baseline cholesterol level, centered at 200 and a SD of 20
baseline <- rnorm(n,200,20)

# random assignment to treatment (1) or control (0) with 50\% chance of treatment
treatment <- rbinom(n, 1, 0.5)

# Define true effect size (this is the real
#effect of our treatment which we are trying to find)
effectsize <- -15

#set the parameters for the epsilon noise in our data
epsilonmean <- 0
epsilonsd <- 6

# Simulate outcome variable  based on group assignment
outcome <- baseline + effectsize * treatment + rnorm(n, epsilonmean, epsilonsd)

# create a data frame with treatment and outcome data
data <- data.frame(treatment, outcome, baseline)

#view snippet of data
head(data)
```

```
##   treatment  outcome baseline
## 1         0 193.5169 188.7905
## 2         1 185.0107 195.3965
## 3         1 218.1674 231.1742
## 4         1 180.3599 201.4102
## 5         0 201.8690 202.5858
## 6         1 217.6189 234.3013
```

1.2 Checking our Randomization

Did our randomization work? We can test this by comparing the characteristics of our treatment and control group at baseline. The only characteristic we have to work with is baseline, cholesterol. Let's compare the averages across treatment and control.

```
#summarize the mean and SD cholesterol baseline for treatment and control units
balance_table <- data %>%
  group_by(treatment) %>%
  summarize(
    N = n(),
    Mean = mean(baseline),
    SD = sd(baseline)
  ) %>%
  mutate(Group = ifelse(treatment == 1, "Treatment", "Control")) %>%
  select(Group, N, Mean, SD)

# Print balance table
print(balance_table)

## # A tibble: 2 x 4
##   Group         N  Mean    SD
##   <chr>     <int> <dbl> <dbl>
## 1 Control      52  200.  17.7
## 2 Treatment    48  203.  18.9

# Run t-test for baseline difference between treatment and control
ttest <- t.test(baseline ~ treatment, data = data)
print(ttest)

##
##  Welch Two Sample t-test
##
## data:  baseline by treatment
## t = -0.78386, df = 95.927, p-value = 0.4351
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -10.164776   4.409493
## sample estimates:
## mean in group 0 mean in group 1
##        200.4269        203.3045
```

The t-test reveals no significant difference between the groups at baseline. This means our randomization has created balance on observed and unobserved characteristics. Ommitted variable bias is likely not a problem for us.

1.3 Checking the Statistical Power of our Experiment

The Minimum Detectable Effect (MDE) for a two-group randomized experiment with equal sample sizes is calculated as:

$$\text{MDE} = \left(z_{1-\frac{\alpha}{2}} + z_{1-\beta}\right) \cdot \sqrt{\frac{2\sigma^2}{n}}$$

Where:

- α: significance level (e.g., 0.05 for a 95% confidence level)

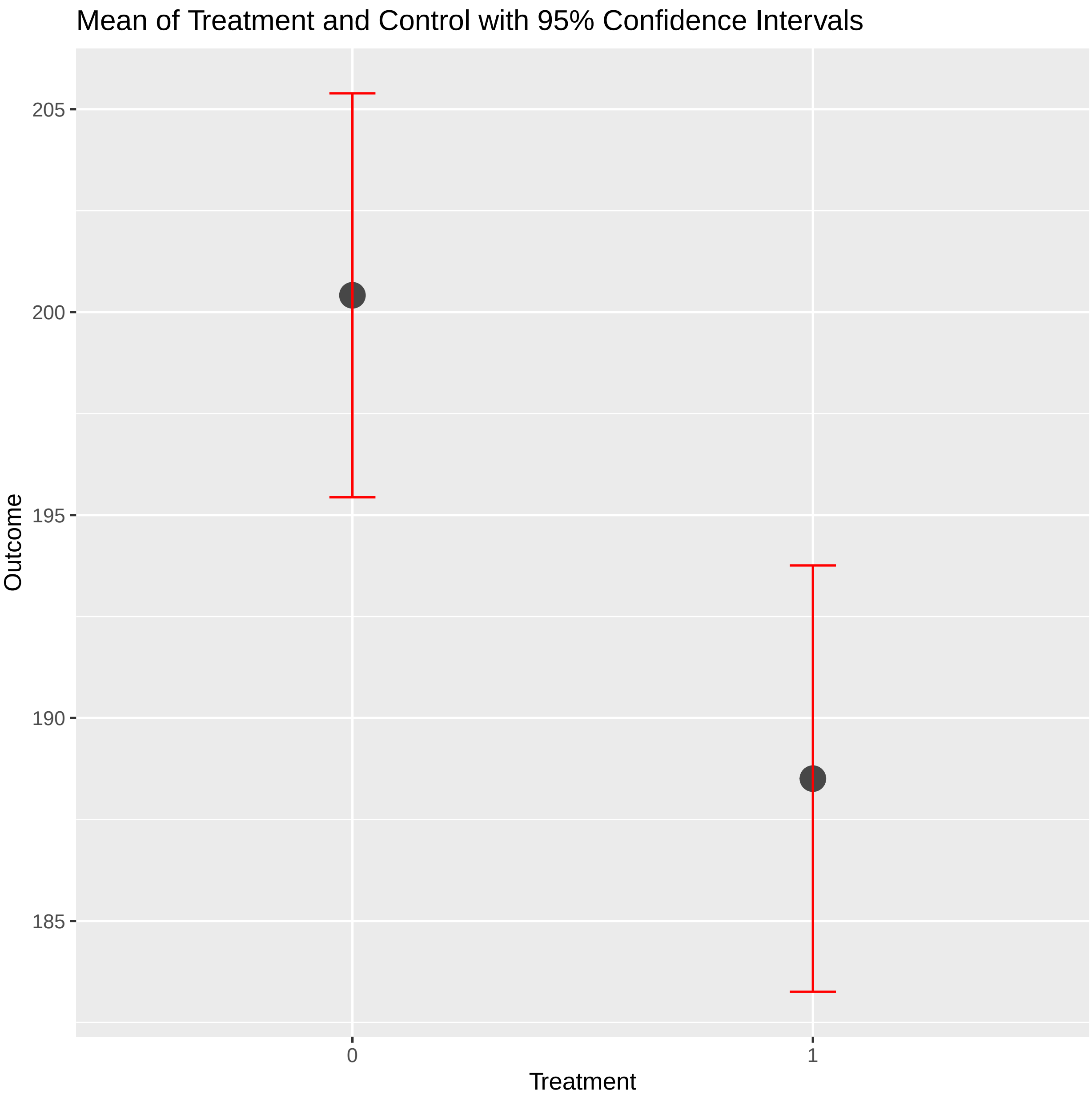
- β: Type II error rate; power is $1 - \beta$ (e.g., 0.2 for 80% power)
- $z_{1-\frac{\alpha}{2}}$: Z-score corresponding to the two-sided significance level (e.g., 1.96 when $\alpha = 0.05$)
- $z_{1-\beta}$: Z-score corresponding to desired power (e.g., 0.84 for 80% power)
- σ^2: variance of the outcome variable
- n: sample size in each group (treatment and control)

```r
#let's calcualte an MDE for our current data. For this we need to find our sigma
#which is the standard deviation of our outcome variable, which we see in our data

# Parameters
alpha <- 0.05          # Significance level
power <- 0.8           # Desired power
n_per_group <- n/2     # Sample size per group
sigma <- sd(data$outcome)

# Get critical values from normal distribution
z_alpha <- qnorm(1 - alpha / 2)
z_beta  <- qnorm(power)

# Calculate MDE
MDE <- (z_alpha + z_beta) * sqrt(2 * sigma^2 / n_per_group)
print(MDE)

## [1] 10.80926

## our MDE is lower than the real effect size, so we're good to go
```

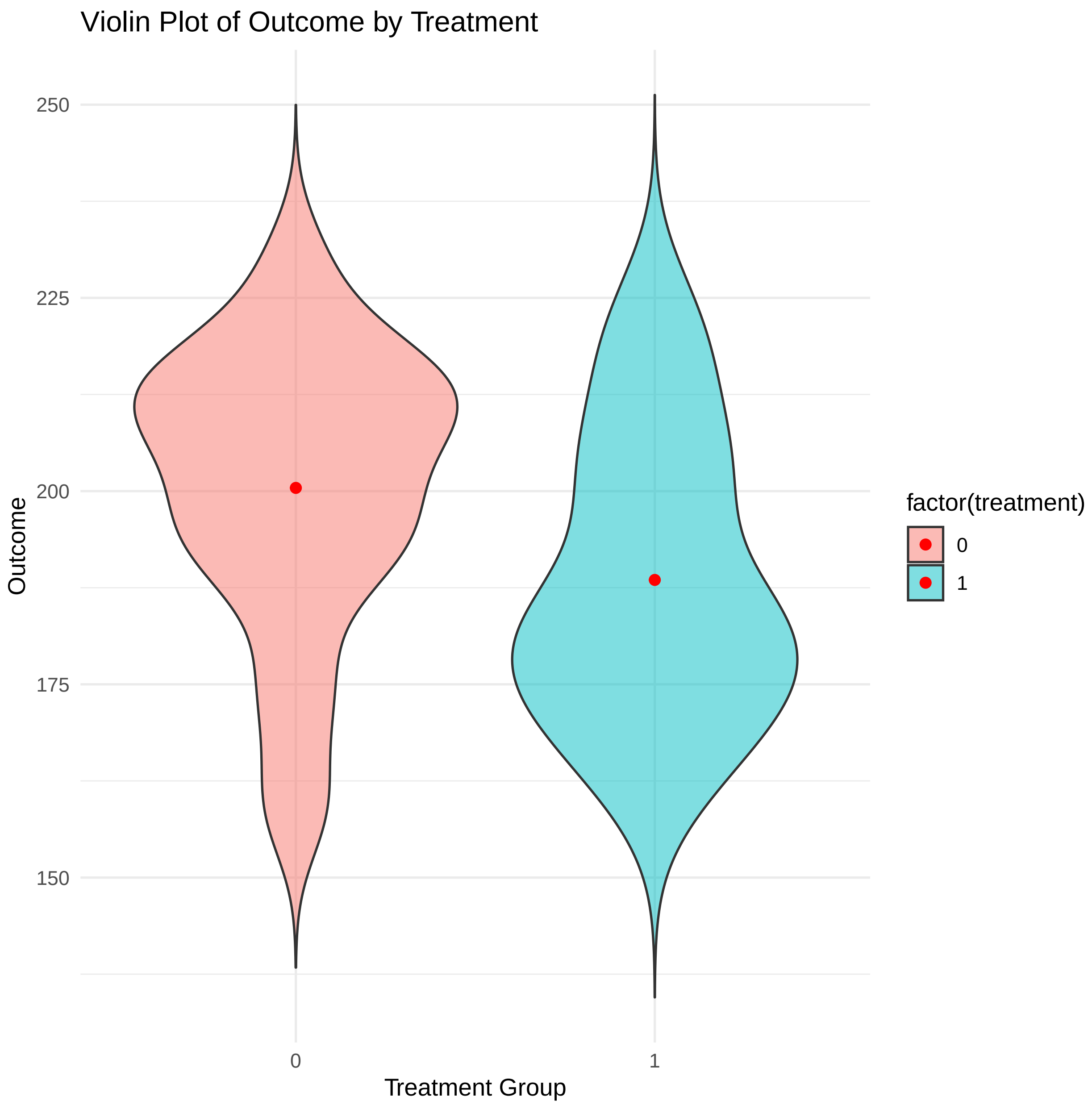
1.4 Visualize Data

Let's graph our data, with some bars that reflect our confidence in our mean estimates. We'll generate confidence intervals around our means. Confidence intervals at a 95% level mean that 95% of the time this experiment is run, the mean value will be within that confidence range.For a normal distribution, about 95% of the values lie within 1.96 standard deviations.

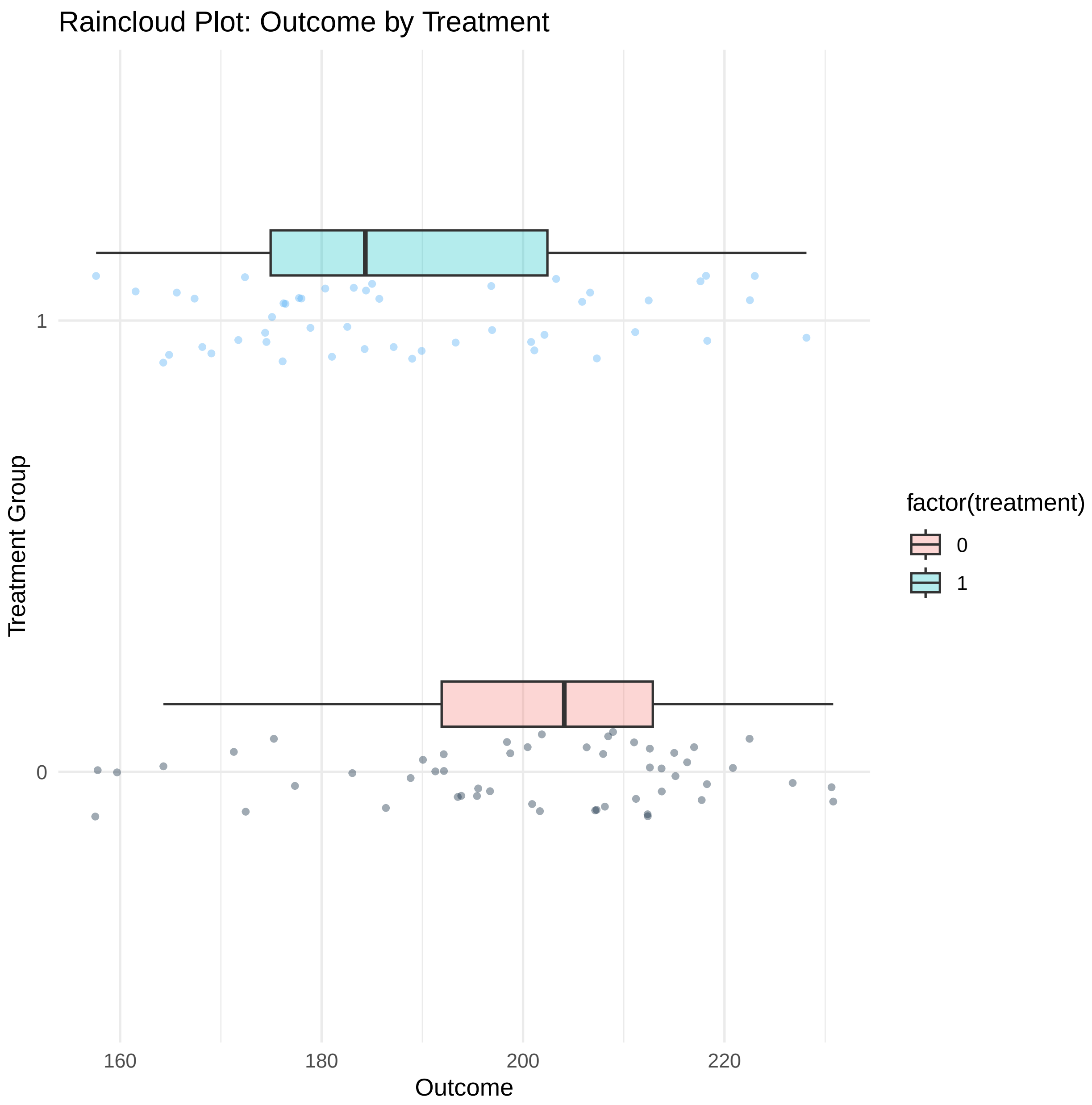
```r
#calculate means and standard errors by treatment (se = sqrt(variance/N) for each group)
# this collapse command comes from package dplyr
bar_data <- data %>%
  group_by(treatment) %>%
    summarize(y = mean(outcome),
              se = sqrt(var(outcome)/length(outcome)))

#calculate confidence intervals using a 1.96 threshold

bar_data$lowerci <- bar_data$y - 1.96*bar_data$se
bar_data$upperci <- bar_data$y + 1.96*bar_data$se

# Create the dot plot graph with confidence intervals using ggplot2
ggplot(bar_data, aes(x = factor(treatment), y = y)) +
  geom_point(stat = "identity", fill = "blue", alpha = 0.7, size=5) +
  geom_errorbar(aes(ymin = lowerci, ymax = upperci), width = 0.1, color = "red") +
  labs(title = "Mean of Treatment and Control with 95% Confidence Intervals",
       x = "Treatment",
       y = "Outcome")
```



```
#there are many other kinds of visualization we could create
ggplot(data, aes(x = factor(treatment), y = outcome, fill = factor(treatment))) +
  geom_violin(trim = FALSE, alpha = 0.5) +
  stat_summary(fun = mean, geom = "point", shape = 20, size = 3, color = "red") +
  labs(title = "Violin Plot of Outcome by Treatment",
       x = "Treatment Group",
       y = "Outcome") +
  theme_minimal()
```



```
ggplot(data, aes(x = factor(treatment), y = outcome, fill = factor(treatment))) +
  geom_jitter(aes(color = treatment), width = 0.1, size = 1, alpha = 0.4, show.legend = FALSE) +
  geom_boxplot(width = 0.1, outlier.shape = NA, alpha = 0.3, position = position_nudge(x = 0.15)) +
  labs(title = "Raincloud Plot: Outcome by Treatment",
       x = "Treatment Group",
       y = "Outcome") +
  theme_minimal() +
  coord_flip()  # Optional: horizontal layout
```



1.5 Test for Treatment Effect

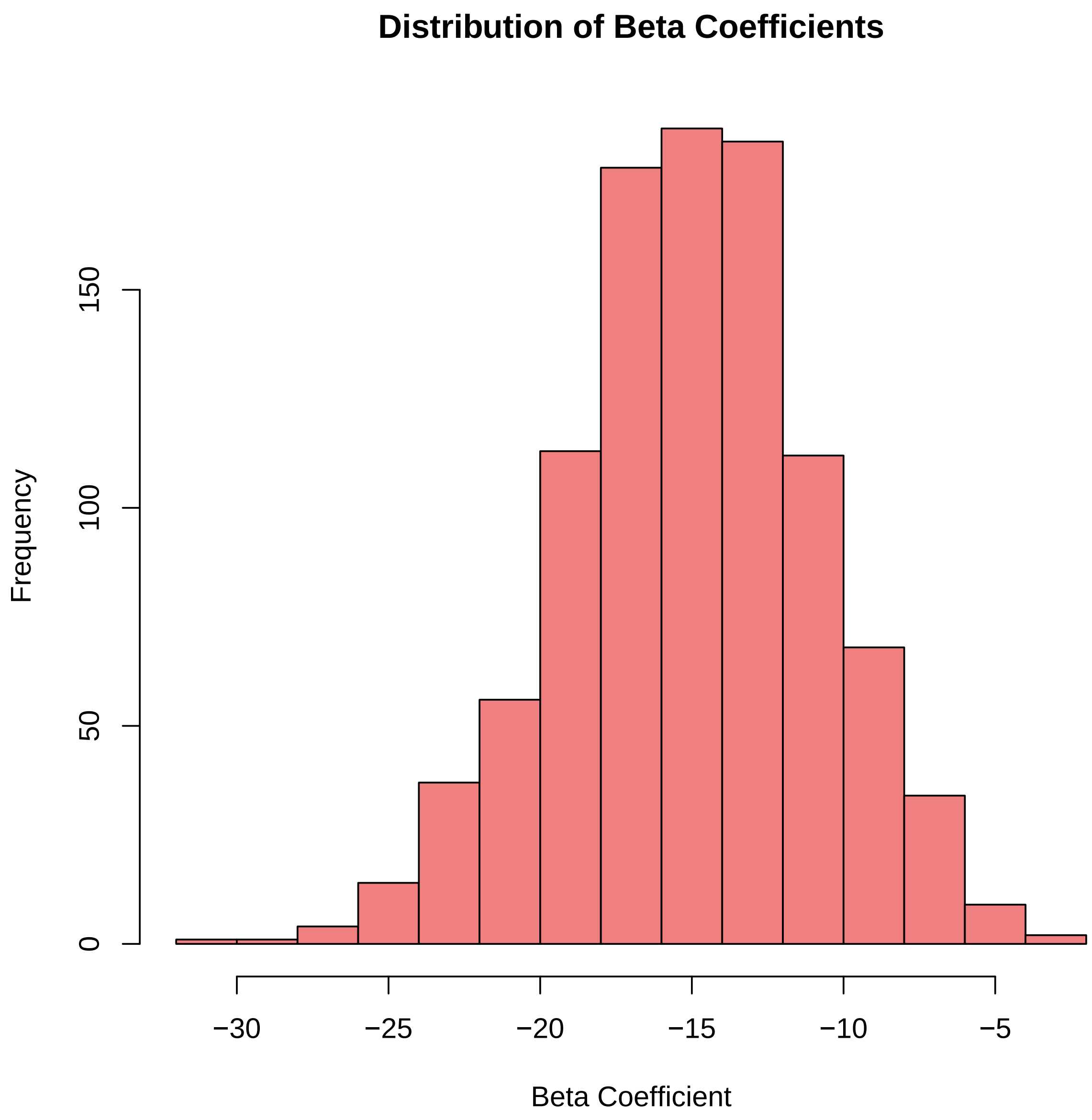
It looks like there was a statistically significant treatment effect, how do we prove it? We can use t-tests or regressions to compare the means to each other and get a p-value.

```
#T-test two sample comparison of treatment and control
t_test_result <- t.test(outcome ~ treatment, data = data)
print(t_test_result)

##
##  Welch Two Sample t-test
##
## data:  outcome by treatment
## t = 3.2256, df = 97.128, p-value = 0.001714
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##   4.581194 19.235922
## sample estimates:
## mean in group 0 mean in group 1
##        200.4151        188.5065

# We can see our test is statistically significant if our p-value
# is below 0.01 (.05 depending on your threshold)
# Our 95\% confidence interval does not overlap with zero

##Alternatively, we can run a test in the form of a regression, where
# Y is our outcome and X is a dummy variable that takes 1 or 0 depending on treatment

# run linear regression of outcome on treatment
model <- lm(outcome ~ treatment )
model_summary <- summary(model)
print(model_summary)

##
## Call:
## lm(formula = outcome ~ treatment)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.891 -12.140  -0.659  13.342  39.634
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  200.415      2.556  78.397   <2e-16 ***
## treatment    -11.909      3.690  -3.227   0.0017 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.43 on 98 degrees of freedom
## Multiple R-squared:  0.09607, Adjusted R-squared:  0.08685
## F-statistic: 10.42 on 1 and 98 DF,  p-value: 0.001699

# we can see how close our treatment coefficient is to the "true value", and check if
# our p-value is under a threshold (either .05 or .01)

##This is pretty good, but we can also increase the precision
#of our estimate by adding in controls to our regression.
#In this case, we could condition on baseline cholesterol.
model <- lm(outcome ~ treatment + baseline)
model_summary <- summary(model)
print(model_summary)

##
## Call:
## lm(formula = outcome ~ treatment + baseline)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2104 -4.1253 -0.6309  3.2909 19.2010
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.93045    6.26696   1.265    0.209
## treatment   -14.67217    1.12805 -13.007   <2e-16 ***
## baseline      0.96037    0.03103  30.954   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.618 on 97 degrees of freedom
## Multiple R-squared:  0.9169, Adjusted R-squared:  0.9152
## F-statistic: 535.2 on 2 and 97 DF,  p-value: < 2.2e-16

#Now we've gotten very close to estimating the magnitude of the true effect.
```

1.6 What does statistical significance mean?

A 95% confidence interval means that if we ran the same experiment 1000 times, we'd expect that 950 of our constructed confidence intervals would contain the ”true effect”. A p-value of 0.05 means that is we ran our experiment 1000 times, we would observe our effect size only 5% of the time under the null hypothesis. Let's run a simulation to show this.

```
# define the number of simulations
n_simulations <- 1000

# store beta coefficients from each simulation
beta_coefficients <- numeric(n_simulations)

#ci storage
lower_ci <- numeric(n_simulations)
upper_ci <- numeric(n_simulations)

# Monte Carlo simulation
for (i in 1:n_simulations) {

#give each person a baseline cholesterol level, centered at 200 and a SD of 20
baselinesim <- rnorm(n,200,20)

# random assignment to treatment (1) or control (0) with 50\% chance of treatment
treatmentsim <- rbinom(n, 1, 0.5)

# Simulate outcome variable  based on group assignment
outcomesim <- baselinesim + effectsize * treatmentsim + rnorm(n, epsilonmean, epsilonsd)


  datasim <- data.frame(treatmentsim, outcomesim,baselinesim)

  # run linear regression of outcome on treatment
  model <- lm(outcomesim ~ treatmentsim,datasim)

  # store the beta coefficient for the treatment variable
  beta_coefficients[i] <- coef(model)[2]  # coef for treatment is the second element
  mean_difference <- mean(outcomesim[treatmentsim==1]) - mean(outcomesim[treatmentsim==0])
  se <- sqrt(var(outcomesim[treatmentsim==1])/length(outcomesim[treatmentsim==1]) + var(outcomesim[trea

  lower_ci[i] <- mean_difference - 1.96 * se
  upper_ci[i] <- mean_difference + 1.96 * se
}

# Create a histogram of beta coefficients
# see how they are a normal distribution around our "true effect"?
hist(beta_coefficients, breaks = 20,
main = "Distribution of Beta Coefficients",
xlab = "Beta Coefficient", col = "lightcoral")
```



```
#how many CIs contain the true effect?
ci_list <- as.data.frame(cbind(lower_ci, upper_ci))
head(ci_list)
```

```
##    lower_ci   upper_ci
## 1 -21.67160  -5.594393
## 2 -23.08883  -7.291830
## 3 -21.52830  -5.416589
## 4 -21.21043  -4.187267
## 5 -23.48730  -7.947086
## 6 -25.98957 -11.257253
```

```
ci_list$true_effect <- ifelse(ci_list$upper_ci > effectsize & ci_list$lower_ci < effectsize, 1,0 )

print(mean(ci_list$true_effect))
```

```
## [1] 0.947
```

```
## now let's run the simulation assuming the null
#hypothesis is true (treatment effect is zero).

# define the number of simulations
n_simulations <- 1000

# store beta coefficients from each simulation
beta_coefficients <- numeric(n_simulations)

# Monte Carlo simulation
for (i in 1:n_simulations) {

#give each person a baseline cholesterol level,
#centered at 200 and a SD of 20
baselinesim <- rnorm(n,200,20)

# random assignment to treatment (1) or control (0) with 50\% chance of treatment
treatmentsim <- rbinom(n, 1, 0.5)

# Simulate outcome variable  based on group assignment
outcomesim <- baselinesim + 0 * treatmentsim + rnorm(n, epsilonmean, epsilonsd)

  datasim <- data.frame(treatmentsim, outcomesim,baselinesim)

  # run linear regression of outcome on treatment
  model <- lm(outcomesim ~ treatmentsim,datasim)

  # store the beta coefficient for the treatment variable
  beta_coefficients[i] <- coef(model)[2]  # coef for treatment is the second element

}


#what fraction of our beta coefficients are as large as
#our observed effect size in the first regression we ran?
model <- lm(outcome ~ treatment )
model_summary <- summary(model)
```

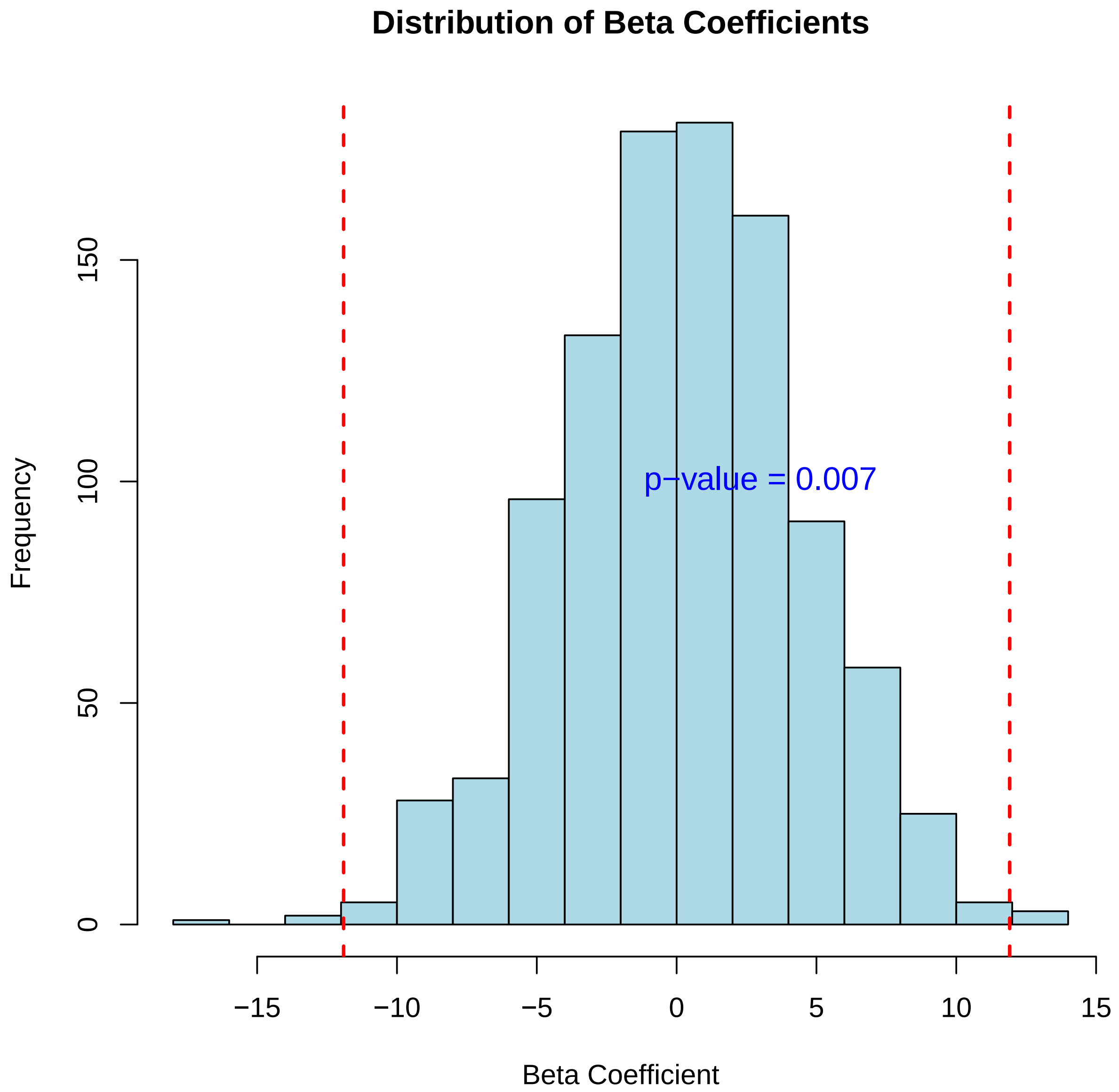
```
print(model_summary)
```

```
##
## Call:
## lm(formula = outcome ~ treatment)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.891 -12.140  -0.659  13.342  39.634
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  200.415      2.556  78.397   <2e-16 ***
## treatment    -11.909      3.690  -3.227   0.0017 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.43 on 98 degrees of freedom
## Multiple R-squared:  0.09607, Adjusted R-squared:  0.08685
## F-statistic: 10.42 on 1 and 98 DF,  p-value: 0.001699
```

```
observed_effect <- coef(model)[2]
#number of estimated betas that are greater than observed effect
p_value <- mean(abs(beta_coefficients) >= abs(observed_effect))
print(p_value)
```

```
## [1] 0.007
```

```
hist(beta_coefficients, breaks = 20,
main = "Distribution of Beta Coefficients",
xlab = "Beta Coefficient", col = "lightblue")

# Mark the observed effect and its mirror
abline(v = c(-abs(observed_effect), abs(observed_effect)), col = "red", lwd = 2, lty = 2)

# Add text with p-value
text(3, 100, paste("p-value =", round(p_value, 4)), col = "blue", cex = 1.2)
```



1.7 Other Tests

There are other statistical tests we can run.

- ANOVA: for cases when we have more than 2 groups
- Mann-Whitney – when normality is violated
- Fisher's Exact test – when our sample sizes are small
- chi square test – when our outcomes are categorical or proportions
- paired t-test – when we have two measurements of related data, like pre/post a test, or similar subject under different conditions. (Ex., treatment-control comparison on adjacent mondays)

Let's see an example of an ANOVA test when we have two treatment groups and a control.

```
#let's produce data for a second treatment group
#in our drug experiment with a different true effect
treatment2 <- rbinom(n, 1, 0.5)
baseline2 <- rnorm(n,200,20)

# Define true effect size (this is the real effect
#of our treatment which we are trying to find)
effectsize2 <- -20

# Simulate outcome variable  based on group assignment
outcome2 <- baseline2 + effectsize2 * treatment2 + rnorm(n, epsilonmean, epsilonsd)

data2 <- data.frame(treatment2, outcome2,baseline2)
data2$treatment <- ifelse(data2$treatment2==1, 2,0)
data2$outcome <-data2$outcome2
data2$baseline <-data2$baseline2


anova_dataset <- rbind(data, data2[c("treatment", "outcome", "baseline")])

#now we have a dataset with treatments labelled 1,2,0
table(anova_dataset$treatment)

##
##   0   1   2
## 107  48  45

#fit anova model -- we see a sig difference between groups
anova_result <- aov(outcome ~ treatment, data = anova_dataset)

# Summary of the ANOVA
summary(anova_result)

##              Df Sum Sq Mean Sq F value   Pr(>F)
## treatment     1  16721   16721   44.42 2.57e-10 ***
## Residuals   198  74530     376
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#we can also regress both treatments as dummies
anova_dataset$treatmentfactor <- factor(anova_dataset$treatment)

model <- lm(outcome ~ treatmentfactor + baseline, data = anova_dataset)

# View results
summary(model)

##
## Call:
## lm(formula = outcome ~ treatmentfactor + baseline, data = anova_dataset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.1797  -4.0471  -0.0853   3.4778  19.2099
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        6.9370     4.4135   1.572    0.118
## treatmentfactor1 -15.3396     1.0251 -14.964   <2e-16 ***
## treatmentfactor2 -19.4881     1.0474 -18.606   <2e-16 ***
## baseline           0.9685     0.0219  44.222   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.885 on 196 degrees of freedom
## Multiple R-squared:  0.9256, Adjusted R-squared:  0.9245
## F-statistic: 813.1 on 3 and 196 DF,  p-value: < 2.2e-16
```