Key Components

```r
library(tidyverse)
library(boot)

manhattan_data <- readr::read_csv("hospital.csv") %>%
  janitor::clean_names() %>%
  filter(hospital_county == "Manhattan") %>%
  select(age_group, total_charges)

f_boot <- readRDS("boot_obj.RDS")
f_log_boot <- readRDS("log_boot_obj.RDS")
```

Problem trying to be addressed

We are trying to see if the mean total medical costs differs between age groups in Manhattan. This tests the following hypotheses:

- H_0 : There is no difference in true mean total medical costs between age groups in Manhattan.
- H_A : There some difference in true mean total medical costs between age groups in Manhattan.

Methods used

In order to test the hypotheses, ANOVA methods are used.

Traditional

Since the data is non-negative and heavily right skewed, a `log1p` transformation is used. This yields the following table:

```
aov(
  log1p(total_charges) ~ age_group,
  data = manhattan_data
) %>%
  anova()
```

```
Analysis of Variance Table

Response: log1p(total_charges)
              Df Sum Sq Mean Sq F value    Pr(>F)
age_group      4  63714 15928.5   17595 < 2.2e-16 ***
Residuals 395712 358242     0.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Traditional methods yield an extremely high F-statistic, which would cause H_0 to be rejected.

Permutation based

The permutation based method uses the following process:

1. Compute ANOVA model and F-statistic for the original data.
2. Run the following N times:

   a. Generate a random permutation of the response and keep the predictor(s) in the same order. These are used to make the permuted data.

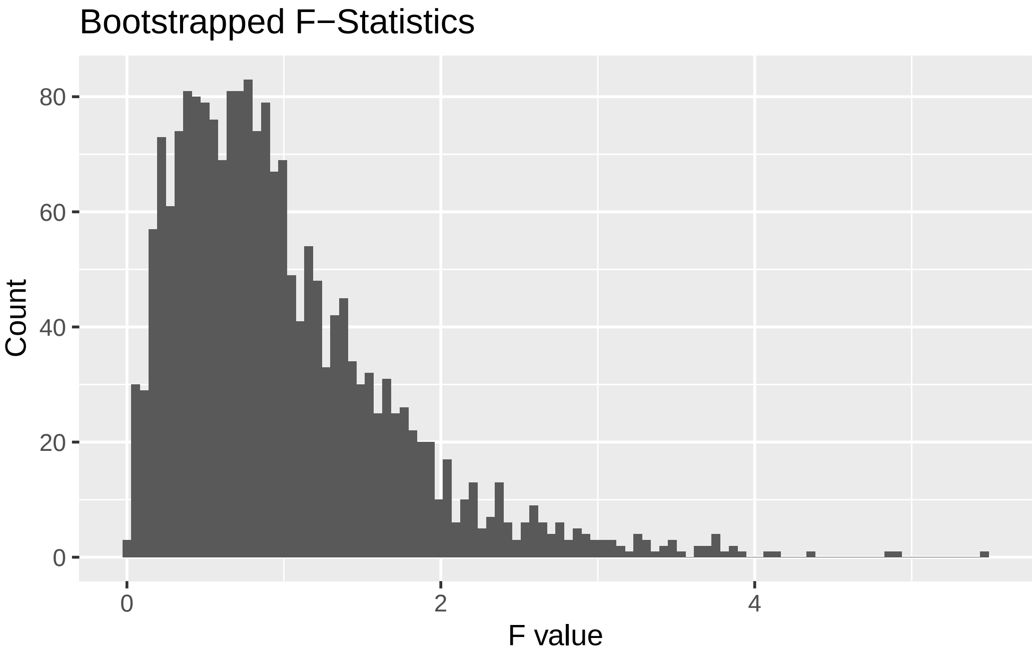
   b. Compute ANOVA model and F-statistic for permuted data.

   c. Save permuted F-statistic.

3. Compute the P-value as the proportion of permuted F-statistics greater than the original.

In this case, we let $N = 2000$.

```
ggplot(mapping = aes(f_boot$t)) +
  geom_histogram(bins = 100) +
  labs(
    title = "Bootstrapped F-Statistics",
    x = "F value",
    y = "Count"
  )
```



original F-value was 1771.0185321, which is higher than any of the F-statistics. This indicates a Permuted P-value of $\frac{1}{2001}$, which is small enough to reject H_0.

Advantages and Disadvantage of Permutation ANOVA

In this case, both methods indicated that there was some difference in mean total costs between age groups.

Traditional ANOVA methods require the following assumptions (from page 428):

- Observations are assigned to fixed groups.
- Residuals are i.i.d. following a normal distribution and shared variance.

The Permutation based ANOVA does not require the those assumptions (pages 64 and 428), making it more robust when the residuals are not normally distributed.

```
aov(
  total_charges ~ age_group,
  data = manhattan_data
) %>%
  car::powerTransform() %>%
  summary()
```

```
bcPower Transformation to Normality
   Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
Y1   -0.0725       -0.07      -0.0745      -0.0704

Likelihood ratio test that transformation parameter is equal to 0
 (log transformation)
                           LRT df       pval
LR test, lambda = (0) 4728.893  1 < 2.22e-16

Likelihood ratio test that no transformation is needed
                          LRT df       pval
LR test, lambda = (1) 1012531  1 < 2.22e-16
```

This was useful in this case as the Box-Cox method indicated that the response required a power transformation of -0.07 for normality.

The disadvantage of the permutation based ANOVA, is that it essentially requires N ANOVA models to be fit, which can take up a lot of time and space. In this case, fitting 2000 ANOVA models took approximately 8 minutes.

Implementation

Traditional ANOVA

```
anova_mod <- aov(
  log1p(total_charges) ~ age_group,
  data = manhattan_data
)

anova(anova_mod)
```

```
Analysis of Variance Table

Response: log1p(total_charges)
              Df Sum Sq Mean Sq F value    Pr(>F)
age_group      4  63714 15928.5   17595 < 2.2e-16 ***
Residuals 395712 358242     0.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Permuation

```
# set seed for reproducible results
set.seed(0)

# function to calculate F statistics
f_stat <- function(data, idx) {
  anova(

    # generate data with groups the same but response shuffled
    tibble(
      age_group = data$age_group,
      total_charges = data$total_charges[idx]
    ) |>

      # compute ANOVA with permuted data
      aov(total_charges ~ age_group, data = _)

  # extract F statistic
  )$F[1]
}

# use boot package
f_boot <- boot(

  # specify function to compute statistic
  statistic = f_stat,

  data = manhattan_data,

  # run 2000 times
  R = 2000,
```

```r

  # specify permutation method
  sim = "permutation",

  # use 6 cpu cores
  parallel = "multicore",
  ncpus = 6
)
```

```r
# compute p-value from permutation
p_value <- (sum(f_boot$t >= f_boot$t0) + 1) / 2001
p_value
```

```
[1] 0.0004997501
```