MET CS 555 Term Project

Alisha Peermohamed | Fall 2019

1. Assignment Description

Select a small data set from the available public data sets (you can find a list of public data sets here http://www.teymourian.de/public-data-sets-for-data-analytic-projects/).

Describe a research scenario and specify a research question based on data analytic methods that we learned in our class, for example methods like, *one and two sample means, t-test, correlation tests, simple and multiple linear regression, ANOVA and ANCOVA, one and two-Sample Tests for Proportions and logistic regression*.

Clean up your data and reduce it to no more than 500 observations if your data set is large.

2. Research Scenario Description (no more than 200 words)

Describe your research scenario in no more than 200 words. This is a general description of the use case. Similar to our class examples, we first describe the overall scenario and then we specify a specific research question based on it.

Uber and Lyft, two of the most popular ridesharing transportation companies, are constantly being compared.

"Take Uber. It's Cheaper!", "…but, Lyft doesn't surcharge during rush hour"

Especially in the city of Boston, where public transportation means are limited, people are constantly requesting trips and trying to guess which factors will yield the lowest price for their ride. In this project, we take guessing out of the question and will use data and statistical inferencing to explore which factors affect the pricing of your next ride the most.

Using data on various rides in the city of Boston: the time of the ride, the ridesharing company (Uber or Lyft), the weather at the time of the ride, the distance travelled, the pick-up location, the drop-off location, and the type of vehicle requested (UberX, Lyft Shared, etc.), we will explore whether Uber is, in fact, cheaper, whether Lyft does not, in fact, charge a surcharge during rush hour, and whether it actually makes a difference to the cost of your ride if its pouring outside.

3. Describe the data set (no more than 200 words)

Describe briefly the data set. Describe each columns of the data set if you use the column in your analysis. Clean up your data before usage, for example you can remove the outliers. Remove unused columns. If possible provide a Link to the main data set source.

The initial dataset was taken from Kaggle.com (link). The first dataset, *cab_rides.csv*, contains information on ~693,000 rides in Boston, including distance, company, epoch timestamp, source, price, product, and more. The second dataset, *weather.csv*, contains the weather conditions at various epoch times.

I first dropped all unused columns from the *cab_rides.csv* dataset. Using systematic sampling, I limited my dataset to 500 rides. I matched the timestamps in the *cab_rides.csv* dataset to those in the *weather.csv* dataset to extract the temperature and precipitation (in inches) at the time of the ride. The *cab_rides.csv* dataset specifies the 'Product Type' (ie. UberPool, UberX, Lyft Shared, Lux Black, etc.). I grouped similar products across the two companies and mapped them to numerical values. For example, 'UberPool' and 'Lyft Shared', which offer similar products, were mapped to 1. A breakdown of mappings is shown in the table below. Lastly, I removed rows with missing values. I decided to remove precipitation levels as the weather dataset does not differentiate between no rain (0 inches) and missing values (NA). My final dataset includes the following columns for each ride: the company used, the distance (miles), the product level (1 to 5), the pick-up area, and the Temperature. You may find my final, processed dataset, *uber_lyft_dataset.csv*, in the file attached.

Products	Mapping
UberPool, Lyft Shared	1
UberX, Lyft, WAV	2
UberXL, LyftXL	3
UberBlack, Lux Black, Lux	4
UberBlack SUV, Lux Black XL	5

3. Research Question (no more than 100 words)

Describe briefly in one or two sentences the main research question. This is similar to the last sentence of our class examples.

In this study, we will explore the effects of distance travelled, outside temperature, product level used, the ride pick-up location, and the ridesharing company used on the price of the ride. We will determine how much of the variation in the price of the ride is attributable to the above-mentioned factors and which factors have a largest impact on price. We will also explore whether average ride prices differ among companies or among pick-up locations, after controlling for other significant variables.

4. Your solution R code

Copy your R code here. Start from read the data from a data file. Keep the following data read line.

This is similar to one of our R code examples.

```
# Notes and Comments in BLUE | R Code in BLACK

## DATA PREPROCESSING:
rides <- as.data.frame(read.csv('uber_lyft_dataset.csv', stringsAsFactors =
FALSE, header = TRUE))
options(scipen=999) # prevents scientific notation for time

# Removing unused columns: Destination, Surge Multiplier, ride ID, Product_ID:
drop <- c("destination","surge_multiplier", 'id', 'product_id')
rides_data <- rides_data[ , !(names(rides_data) %in% drop)]

# Removing rows with missing price value:
rides_data <-rides_data %>% drop_na(price)

# Reducing Dataset size to 500 rows using Systematic Sampling:
N = nrow(rides_data)
n = 500
k <- ceiling(N / n)
r <- sample(k, 1)
rows <- seq(r, by = k, length = n)
rides <- rides_data[rows, ]

# Second dataset: Weather data at epoch time and area. Rain column indicates
# inches of Rain.
weather_data <- as.data.frame(read.csv('weather.csv', stringsAsFactors = FALSE,
header = TRUE))
drops <- c('rain', "pressure", 'humidity', 'wind', 'clouds')
weather_data <- weather_data[ , !(names(weather_data) %in% drops)]

#Convert epoch timestamp to Hour of Day:
# truncating epoch time to 10 digits(since Rides data also provided seconds data)
for (i in 1:nrow(rides)) {
  rides[i, 'time_stamp'] = round(rides[i, 'time_stamp']/10^3)
}

#Extracting the hour of the day, month, and hour of the ride
rides$hour <- NA
rides$month <- NA
rides$day <- NA
for (i in 1:nrow(rides)) {
  time <-  rides[i, 'time_stamp']
  z <- as.POSIXlt(time, origin="1970-01-01", tz="EST")
```

```
  hour <- unclass(z)$hour
  month <- unclass(z)$mon
  day <- unclass(z)$mday
  rides[i, 'hour'] = hour
  rides[i, 'month'] = month
  rides[i, 'day'] = day
}

# Extracting the day, month, and hour of weather recording
weather_data$hour <- NA
weather_data$month <- NA
weather_data$day <- NA
for (i in 1:nrow(weather_data)) {
  time_w <- weather_data[i, 'time_stamp']
  x <- as.POSIXlt(time_w, origin="1970-01-01", tz="EST")
  month <- unclass(x)$mon
  day <- unclass(x)$mday
  hour <- unclass(x)$hour
  weather_data[i, 'hour'] = hour
  weather_data[i, 'month'] = month
  weather_data[i, 'day'] = day
}

#connecting ride with temperature at the time of ride
rides$temperature <- NA
for (i in 1:nrow(rides)) {
  hour <- rides[i, 'hour']
  month <- rides[i, 'month']
  day <- rides[i, 'day']
  location <- rides[i, 'source']
  temp_data <- subset(weather_data, weather_data$month == rides[i, 'month']
                      & weather_data$day == rides[i, 'day']
                      & weather_data$hour == rides[i, 'hour']
                      & weather_data$location == rides[i, 'source'])[1,]
  rides[i, 'temperature'] = temp_data$temp
}

#Removing rows with NA temperature values:
rides <-rides %>% drop_na(temperature)

## Encoding Ride Type into Numerical Factors:

# * Grouped Uber and Lyft Rides by their product levels:
#       * (1): UberPool, Shared
#       * (2): UberX, Lyft, WAV
#       * (3): UberXL, Lyft XL
#       * (4): UberBlack, Lux Black, Lux
#       * (5): UberBlack SUV, Lux Black XL

for (i in 1:nrow(rides)) {
```

```
  if (rides[i, 'name'] %in% c('UberPool', 'Shared')) {
    rides[i, 'name'] = 1}
  if (rides[i, 'name'] %in% c('UberX', 'Lyft', 'WAV')) {
    rides[i, 'name'] = 2}
  if (rides[i, 'name'] %in% c('UberXL', 'Lyft XL')) {
    rides[i, 'name'] = 3}
  if (rides[i, 'name'] %in% c('Black', 'Lux Black', 'Lux')) {
    rides[i, 'name'] = 4}
  if (rides[i, 'name'] %in% c('Black SUV', 'Lux Black XL')) {
    rides[i, 'name'] = 5}
}

##Final Datacleaning:
#   Renaming certain columns
names(rides)[6] <- 'Product.Level'
names(rides)[2] <- 'Company'
names(rides)[1] <- 'Distance'
names(rides)[4] <- 'Source'
names(rides)[10] <- 'Temperature'
names(rides)[5] <- 'Price'
#   Dropping Unused columns: timestamp, month, day, hour
drop <- c("time_stamp","month", 'day', 'hour')
rides <- rides[ , !(names(rides) %in% drop)]
#   Re-arranging columns to: Company, Source, Product Level, Distance,
#   temperature.
rides <- rides[,c("Company", "Source", "Product.Level", "Distance",
"Temperature", "Price")]

rides_data <-rides_data %>% drop_na(price)

# Saving cleaned dataset to directory as ‘uber_lyft_dataset.csv’:
# setwd("/Users/alishapeermohamed/Desktop/CS 555/Term Project")
# write.csv(rides, 'uber_lyft_dataset.csv', row.names = FALSE)

#Removing all variables and data in R environment
remove(list = ls())

#################################################################################
## DATA VISUALIZATION AND ANALYSIS – Uber/Lyft Study:

rides <- as.data.frame(read.csv('uber_lyft_dataset.csv', stringsAsFactors =
FALSE, header = TRUE))
options(scipen=10)

## Research Question:
# Explore the effects of various factors on the price of the app-sharing ride:
#     * Distance, Pick-up point, Temperature at the time of the ride, and type of
#       product(UberX, Lyft Lux)

#################################################################################
```

```r
## Two Sample Means T-test:
#       * Testing whether or not the prices of rides leaving from the Financial
#         District are higher than the prices of rides leaving from South Station
#         at a 95% confidence level.

south <- subset(rides, rides$Source == 'South Station')$Price
financial_dis <- subset(rides, rides$Source == 'Financial District')$Price
len_south <- length(south)
len_financial_dis <- length(financial_dis)

# Formal test of Hypothesis:

# Step1:
# Null Hypothesis: mean(south) == mean(financial_dis). The average prices of
# rides leaving from south station is the same as those leaving from the
# financial district.
# Alternate hypothesis: mean(financial_dis) > mean(south). The average prices of
# rides leaving from the financial district is more than those leaving from South
# Station.
# alpha = 0.05

# Step2:
# Select the t-statistic as the appropriate test statistic because the standard
# deviation of the population size is unknown.
# t = (x1bar = x2bar) = (mu1 - mu2) / sqrt((sd1^2/ n1) + (sd2^2/n2))

# Step3:
sd_south <- sd(south)
sd_fin <- sd(financial_dis)
df <- ((sd_south^2/len_south) +
      (sd_fin^2/len_financial_dis))^2/(((sd_south^2/len_south)^2/(len_south-1)) +
      ((sd_fin^2/len_financial_dis)^2/(len_financial_dis-1)))

t_critical <- qt(0.95, df); t_critical #t-critical = 1.66517
# Decision Rule: Reject Null hypothesis (H0) if |t| >= 1.66517
#                Otherwise, do not reject Null hypothesis (H0)

# Step4:
t.test(financial_dis, south, alternative='greater', conf.level=0.95)
# t-statistic = 1.6719; p-value = 0.0493.

# Step5:
# Reject the Null Hypothesis since the p-value from the T-test is less than the
# 0.05. We are 95% confident that the mean prices for rides leaving from South
# station is less than the mean prices of rides leaving from the financial
# district. The average price of the rides leaving from the Financial district is
# $20.92 per ride whereas the average price of Uber rides leaving from South
# Station is $16.93 per ride.

#################################################################################
```

```r
## Correlation Test Between Distance and Price:
#     * Test to determine whether there is a linear association between the price
#       of a ride and the distance of the ride.
#     * Using samples to test the price to distance correlation for the
#       entire population.

r <- cor(rides$Distance, rides$Price); r # r = 0.3641559

# Step1:
# Null Hypothesis: population_corr = 0. There is no linear association between
# price and distance travelled.
# Alternate Hypothesis: population_corr =/= 0. There is a linear association
# between price and distance travelled.
# alpha = 0.05


# Step2:
# t = r(sqrt((n-2)/(1 - r^2)))

# Step3:
df <- length(rides$Price) - 2
# associated right-hand probability of alpha/2 = 0.025
t_critical <- qt(0.975, df=df); t_critical # t_critical = 1.964758

# Decision Rule: Reject Null Hypothesis if |t| >= 1.964758.
#                Else: Do not reject Null Hypothesis.

# Step4:
t <- r*(sqrt((df)/(1 - r^2))); t # t = 8.778396; # p-value < 2.2e-16
cor.test(rides$Distance,rides$Price,alternative='two.sided',method='pearson',
conf.level = 0.95)

# Step 5:
# Reject Null Hypothesis that there is no linear association between confidence
# level and price. We have significant evidence at the 95% confidence interval
# that population_corr =/= 0. There is strong evidence of a significant linear
# association between distance travelled and the price of the ride.

#################################################################################
## Simple Linear Regression on Distance and Price:
#     * Developed a SLR predicting the Price of the Ride based on the Distance
#       travelled.

plot(rides$Distance, rides$Price,
     main = 'Rides Prices for Various Distances Travelled using Uber and Lyft',
     xlab = 'Distance Travelled (in miles)',
     ylab = 'Price of Ride in Dollars ($)',
     col = 'darkcyan', cex.main = 1, pch = 1, cex = 0.8)

SLR <- lm(rides$Price ~ rides$Distance)
```

```
summary(SLR)
abline(a=10.402 , b=3.171, col = 'darkorange', lwd = 2)

distance_bar <- mean(rides$Distance)
sd_distance <- sd(rides$Distance)
price_bar <- mean(rides$Price)
sd_price <- sd(rides$Price)

beta1 <- r*sd_price/sd_distance; beta1
beta0 <- price_bar - beta1*distance_bar ; beta0

# The equation for the Simple Linear Regression between Distance and Price is:
#                       Price = 10.3512 + 3.1899(Distance).

# For every one mile increase in Distance, there is a $3.19 increase in the
# price.
# If a person travelled 0 miles, the average price of the ride will be
# $10.35.

# Formal Inference test for SLR using the ANOVA table:
#     * Test whether there is a linear relationship between distance travelled
#       and the price of the ride.

anova <- anova(SLR); anova
SE_beta1 <- summary(SLR);SE_beta1

# Step 1:
# Null Hypothesis[H0]: beta_distance = 0 (There is no linear association)
# Alternate Hypothesis [H1]: beta_distance =/= 0 (There is a linear association)
# alpha = 0.05

# Step 2:
# Chose the F-statistic as the test statistic: F = MS Reg/MS Res with 1 and
# n-2 = 498 - 2 = 496 degrees of freedom.
#Step 3:
# F distribution with 1, 496 degrees of freedom and alpha = 0.05
F_critical <- qf(0.95, df1 = 1, df2 = 496); F_critical # F_critical = 3.8602
# Decision Rule: Reject H0 if F_statistic >= 3.8601,
#                otherwise can not reject H0.

#Step 4:
# Based on the ANOVA table, F-statistic = 77.06

#Step 5:
# Since the F-statistic > F-critical, 77.06 > 3.8601 and the p-value < 2.2e-16,
# we reject the Null Hypothesis that there is no linear association between the
# distance travelled and the price of the ride. We have significant evidence at
# the alpha = 0.05 level that there is a linear association between distance and
# price.
```

```r
# 95% Confidence Interval of Beta_Distance:
beta1_95confidence <- confint(SLR, level = 0.95)[2,];beta1_95confidence
# For every one mile increase in distance, we are 95% confident that the price of
# the ride will increase from $2.48/ride to $3.90/ride.

r_squared <- r^2; r_squared
# The adjusted R-squared value is 13.26% of the variation in Price is explained
# by changes in the distance.

#################################################################################
## Multiple Linear Regression:
#       * Developed a Multiple Linear Regression to explore the effects of
#       * distance, temperature, & product level together.

# MLR with Company, Product Level, Temperature, and Distance as explanatory
# variables:
MLR <- lm(rides$Price~rides$Product.Level + rides$Temperature + rides$Distance)

# Global F-test: Is there a linear relationship between the price of the ride and
# the distance, temperature, product-level?

# Step1:
# Null Hypothesis: H0: Beta_distance = Beta_Product_level = Beta_Temperature = 0
# (Distance, Product Level, and Temperature are not predictors of annual salary)

# Alternate Hypothesis: H1: Beta_distance =/= Beta_Product_level =/=
#                            Beta_Temperature =/= 0.
# (At least one in Distance, Product Level, and Temperature is a significant
# predictor of annual salary)
# alpha = 0.05

# Step2:
# k = 3
# Chose the F-statistic as the test-statistic with 3 and 494 degrees of freedom.
#Step 3:
qf(.95, df1=3, df2=494) #F(3, 494, 0.05) = F_critical = 2.6229
# Decision Rule: Reject H0 if F >= 2.6229,
#                Otherwise do not reject the null hypothesis.

#Step 4:
summary(MLR)
# F-statistic = 666.9 with p-value < 2.2e-16

#Step5:
# Reject H0 since 666.9 ≥ 2.6229
# We have significant evidence at the α = 0.05 level that Beta_distance =/= 0
# and/or Beta_Product_level =/= 0 and/or Beta_Temperature =/= 0. We are 95%
# confident that there is evidence of a linear association between ride price and
# distance and/or temperature, and/or product level.
```

```
# MLR Inference t-test:
#       * Test the significance of individual attributes: distance, temperature,
#         and product level to gauge the relative contribution of each variable
#         at the alpha = 0.05 level.
#       * Compute the confidence interval for significant variables.

t_critical <- qt(0.95 , df = 494); t_critical #t_critical = 1.6479
# Decision Rule: Reject H0 if |t| >=1.6479
#                Otherwise do not reject H0

# Testing for Temperature at the alpha = 0.05 level:
# The t-statistic of the temperature variable is 1.6479 and p-value is 0.0957. We
# do not have significant evidence at the alpha = 0.05 level that temperature has
# a significant effect on price, after controlling for other variables. That
# being said, since the p-value = 0.0957, we do have evidence at the alpha = 0.10
# that the temperature variable has a significant effect on price. For every one
# degree increase in temperature, there is a $0.05 in the price of the ride.

# Testing for Distance at the alpha = 0.05 level:
# The t-statistic of the distance variable is 16.798 and p-value is <2e-16. We
# have significant evidence at the alpha = 0.05 level that distance has a
# significant effect on price, after controlling for other variables. For every
# one 1 mile increase in distance, the price of the ride increases by $2.93.

conf_dist <- c(2.9328 - (1.6479*0.17459) , 2.9328 + (1.6479*0.17459))
# dis_95%_confidence_interval: [2.645093, 3.220507]
# We are 95% confident that for a one mile increase in distance, the price of the
# ride increases between $2.65 and $3.22 per ride, after controlling for other
# variables in the model.

# Testing for Product.level at the alpha = 0.05 level:
# The t-statistic of the product.level variable is 40.708 and p-value is <2e-16.
# We have significant evidence at the alpha = 0.05 level that product.level has a
# significant effect on price, after controlling for other variables. For every
# one level increase in product level (increasing from UberPool to UberX, or
UberX to UberXL), has a $6.04 increase on the price of the ride.

conf_prodlev <- c(6.03700 - (1.6479*0.14830) , 6.03700 + (1.6479*0.14830))
# product_lev_95%_confidence_interval: [5.792616, 6.281384]
# We are 95% confident that for a one level increase in product level (increasing
# from UberPool to UberX, or UberX to UberXL), the price of the ride increases
# between $5.79 and $6.28 per ride, after controlling for other variables in the
# model.

# R-squared Value:
regss <- sum((fitted(MLR) - mean(rides$Price))^2)
resiss <- sum((rides$Price-fitted(MLR))^2)
totalss <- regss + resiss
fstatistic <- (regss/3)/(resiss/494)
pvalue <- 1-pf(fstatistic , df1=2, df2=97)
```

```
R2 <- regss/totalss; R2

# The R-squared value for the Multiple Linear Regression is 0.8019. This means
# that 80.19% of all variation in the price of the ride can be explained by
# variation in distance, product.level, and the temperature outside.

#################################################################################
## One way ANOVA to compare means across Uber rides and Lyft rides:
#      * Test the hypothesis that the prices for the population of Uber rides is
#        different from the prices of the population of Lyft rides.

rides$Company <- factor(rides$Company, levels = c('Uber', 'Lyft'))
one_way_ANOVA <- aov(rides$Price~rides$Company , data=rides)

# Global F-test for one-way-ANOVA:

# Step1:
# Null Hypothesis: mean(uber) = mean(lyft).
# (The mean price of Uber rides is the same as the mean price of Lyft rides.)
# Alternate hypothesis: mu(uber) =/= mu(lyft).
# (The mean price of Uber rides is not the same as the mean price of Lyft rides.)
# alpha = 0.05

# Step2:
# F-statistic with 1 and 498-2 = 496 degrees of freedom

# Step3:
f_critical <- qf(.95, df1=1, df2=496) #F_critical = 3.8602
# Decision Rule: Reject Null hypothesis if F-statistic >= 3.8602.
#                Otherwise, do not reject H0.

# Step4:
summary(one_way_ANOVA)
# F-statistic = 5.73, p-value = 0.017.
# Step5:
# Since F-statistic (5.73) > F-critical (3.86), We reject the Null hypothesis
# that the mean price of Uber rides is the same as the mean price of Lyft rides.
# We have significant evidence at the α = 0.05 that there is a difference in
# prices between Uber rides and Lyft rides.

# No need for pairwise comparisons as there is only one pair of groups.

# One-Way Anova analysis using a linear regression:

rides$uber <- ifelse(rides$Company == 'Uber', 1, 0)
rides$lyft <- ifelse(rides$Company == 'Lyft',1,0)

one_way_model <- lm(rides$Price ~ rides$uber, data=rides)
summary(one_way_model)
```

```
#     * The regression model equivalent to the one-way ANOVA model, holding lyft
#       as the reference group, is:
#       y = 18.429 + -2.15(group_uber).

# By the fact that our p-value is 0.017, our linear regression confirms that
# there is a significant difference in the price of Uber rides versus lyft rides.
# The beta_uber is -2.1546. So, we say that average Uber price per ride is $2.15
# less than the average Lyft price per ride.

# Adjusting for other variables (ie. distance, temperature, product_level):
install.packages("carData")
install.packages("car")
library(carData)
library(car)
adjust_MLR <- lm(rides$Price~rides$Company+rides$Distance +
rides$Temperature+rides$Product.Level)
Anova(adjust_MLR, type = 3)
summary(adjust_MLR)

# After adjusting for other variables (distance, temperature, product level), we
# can see that although the model passes the Global F-test (indicating that
# at least one of the variables is significant), the 'Company' variable does not
# pass the inference F-test at alpha = 0.05 level. This is shown by the
# fact that the p-value of the Company variable is 0.52 and the F-statistic is
# 0.4145. Thus, after adjusting for other covariants, we are 95% confident that
# the differences that we saw in the one-way ANOVA model were due to other
# variable differences across the Company as opposed to true differences in
# Price attributable only to the Company used.

#Least squares means
install.packages("emmeans")
install.packages('lsmeans')
library(emmeans)
library(lsmeans)
# p-value adjustment:
emmeans(adjust_MLR, specs = "Company" , contr = "pairwise")

# The least square means (adjusted for distance, temperature, and product_level )
# were $17.20 per ride and $17.40 per ride for Uber and lyft respectively.
# However, we do not have significant evidence against the null hypothesis, which
# is: the price of the Uber rides is the same as the price of Lyft rides after
# controlling for other variables in the model.

## One way ANOVA to compare means across pick-up locations:
#      * Test the hypothesis that the prices for rides significantly vary across
#        pick up locations.

rides$Source <- factor(rides$Source, levels = unique(rides$Source))
one_way_ANOVA_s <- aov(rides$Price~rides$Source , data=rides)
```

```
# Global F-test for one-way-ANOVA:

# Step1:
# Null Hypothesis: mean(various pickup points) = mean(various pickup points) = 0.
#     *The mean price of rides is the same across pick up points.
# Alternate hypothesis: mean(various pickup points)=/=mean(various pickup points)
#     * The mean price of rides is not the same across different pick up points.
#     * At least one pair of pick-up points have significantly different ride
#       prices.
# alpha = 0.05

# Step 2:
# F-statistic with 12 and 498-12 = 486 degrees of freedom

# Step 3:
f_critical <- qf(.95, df1=12, df2=486); f_critical #F_critical = 1.77211
# Decision Rule: Reject Null hypothesis if F-statistic >= 1.77211
#                Otherwise, do not reject H0.

# Step 4:
summary(one_way_ANOVA_s)
# F-statistic = 3.005, p-value = 0.0069.

# Step 5:
# We reject the Null hypothesis that the mean price per ride is the same across
# pick up locations. We have significant evidence at the α = 0.05 that there is a
# difference in prices based on pick up location.

#Pairwise Comparison using t-test and Tukey adjustment:

# Null Hypothesis: mean(various pickup points) = mean(various pickup points) = 0
#     * The mean price of rides is the same across pick up points.
# Alternate hypothesis: mean(various pickup points)=/=mean(various pickup points)
#     * The mean price of rides is not the same across different pick up points.
#     * At least one pair of pick-up points have significantly different ride
#       prices.

# alpha = 0.05 | t-statistic with 486 degrees of freedom.

aggregate(rides$Price, by=list(rides$Source), summary)
aggregate(rides$Price, by=list(rides$Source), var)
pairwise.t.test(rides$Price, rides$Source, p.adj='none')
TukeyHSD(one_way_ANOVA_s)

# One-Way Anova analysis using a linear regression:

# The regression model equivalent to the one-way ANOVA model, holding
# NorthEastern University as the reference group, is:
#           y = Beta_intercept + sum(Beta_pickup(group_pickup))
```

```
rides$NEUni <- ifelse(rides$Source == 'Northeastern University', 1, 0)
rides$North <- ifelse(rides$Source == 'North Station', 1, 0)
rides$Fenway <- ifelse(rides$Source == 'Fenway', 1, 0)
rides$Backbay <- ifelse(rides$Source == 'Back Bay', 1, 0)
rides$BU <- ifelse(rides$Source == 'Boston University', 1, 0)
rides$South <- ifelse(rides$Source == 'South Station', 1, 0)
rides$Beacon <- ifelse(rides$Source == 'Beacon Hill', 1, 0)
rides$Hay <- ifelse(rides$Source == 'Haymarket Square', 1, 0)
rides$WestEnd <- ifelse(rides$Source == 'West End', 1, 0)
rides$NorthEnd <- ifelse(rides$Source == 'North End', 1, 0)
rides$Findist <- ifelse(rides$Source == 'Financial District', 1, 0)
rides$Theatre <- ifelse(rides$Source == 'Theatre District', 1, 0)

# holding Northeastern University as reference group
one_way_model_s <- lm(rides$Price ~ rides$North + rides$Fenway +
                                    rides$Backbay + rides$BU + rides$South +
                                    rides$Beacon + rides$Hay + rides$WestEnd
                                    + rides$NorthEnd + rides$Findist +
                                    rides$Theatre , data=rides)
summary(one_way_model_s)

# By the fact that our p-value is 0.00068, our linear regression confirms that
# there is a significant difference in the price of rides across pick up
# locations.

# Adjusting for other variables (ie. distance, temperature, product_level):
library(carData)
library(car)
adjust_MLR_s <- lm(rides$Price~rides$Source+rides$Distance +
rides$Temperature+rides$Product.Level)
Anova(adjust_MLR_s, type = 3)
summary(adjust_MLR_s)


# After adjusting for other variables (distance, temperature, product level), we
# can see that although the model passes the Global F-test (indicating that
# at least one of the variables does not equal zero), the 'Source' variable does
# not pass the inference F-test at the alpha = 0.05 level. This is shown by the
# fact that the p-value of the Source variable is 0.7028 and the F-statistic is
# 0.7363. Thus, after adjusting for other covariants, we are 95% confident that
# the differences that we saw in the one-way ANOVA model were due to other
# variable differences across the pick-up point as opposed to true differences in
# Price attributable only to the Source used.

#Least squares means
library(emmeans)
library(lsmeans)
# p-value adjustment:
emmeans(adjust_MLR_s, specs = "Source" , contr = "pairwise")
```

```
# The least square means (adjusted for distance, temperature, and product.level )
# show that rides leaving from Boston University have the highest price per ride
# of $18.40 per ride and rides leaving from Beacon Hill and the West End have the
# lowest average price per ride of $16.60 per ride. However, we do not have
# significant evidence against the null hypothesis, which is: the price of the
# uber rides is the same as the price of Lyft rides after controlling for other
# variables in the model.

## One way ANOVA to compare means across Product-Level
#      * Test the hypothesis that the prices for rides significantly vary across
#        product level.

rides$Product.Level <- factor(rides$Product.Level, levels =
unique(rides$Product.Level))
one_way_ANOVA_p <- aov(rides$Price~rides$Product.Level , data=rides)

# Global F-test for one-way-ANOVA:

# Null Hypothesis: mu(product_level) = mu(product_level). The mean price of rides
# is the same across different product levels.
# Alternate hypothesis: mu(product_level) =/= mu(product_level). The mean price
# of rides is not the same across different product levels.
# alpha = 0.05

# F-statistic with 5 and 498-5 = 486 degrees of freedom
f_critical <- qf(.95, df1=5, df2=493); f_critical #F_critical = 2.23229
# Decision Rule: Reject Null hypothesis if F-statistic >= 2.23229
#                Otherwise, do not reject H0.
summary(one_way_ANOVA_p)
# F-statistic = 330.4, p-value <2e-16.
# We reject the Null hypothesis that the mean price per ride is the same across
# product levels.We have significant evidence at the α = 0.05 that there is a
# difference in prices based on product level.

#Pairwise Comparison using t-test:

# Null Hypothesis: mu(product_levels) = mu(product_levels) = 0
# Alternate hypothesis: mu(product_levels) =/= mu(product_levels)
# alpha = 0.05 | t-statistic with 493 degrees of freedom.

aggregate(rides$Price, by=list(rides$Product.Level), summary)
aggregate(rides$Price, by=list(rides$Product.Level), var)
pairwise.t.test(rides$Price, rides$Product.Level, p.adj='none')
TukeyHSD(one_way_ANOVA_p)

# One-Way Anova analysis using a linear regression:

# The regression model equivalent to the one-way ANOVA model, holding level 1
# (UberPool, Lyft Shared) as the reference group, is:
#           y = Beta_intercept + sum(Beta_product_level(group_product_level)).
```

```r
rides$one <- ifelse(rides$Product.Level == 1, 1, 0)
rides$two <- ifelse(rides$Product.Level == 2, 1, 0)
rides$three <- ifelse(rides$Product.Level == 3, 1, 0)
rides$four <- ifelse(rides$Product.Level == 4, 1, 0)
rides$five <- ifelse(rides$Product.Level == 5, 1, 0)

# holding UberPool as reference group
one_way_model_p <- lm(rides$Price ~ rides$two + rides$three +
                        rides$four + rides$five , data=rides)
summary(one_way_model_p)
# By the fact that our p-value is <2e-16, our linear regression confirms that
# there is a significant difference in the price of rides across product_levels.

# Adjusting for other variables (ie. distance, temperature, product_level):
adjust_MLR_p <- lm(rides$Price~rides$Product.Level+rides$Distance +
rides$Temperature+rides$Product.Level)
Anova(adjust_MLR_p, type = 3)
summary(adjust_MLR_p)

# After adjusting for other variables (distance, temperature, product level), we
# can see that although the model passes the Global F-test (indicating that
# at least one of the variables does not equal zero). The 'Product Level'
# variable also passes the inference F-test at the alpha = 0.05 level.
# This is shown by the fact that the p-value of the product_level variable is
# <2e-16 and the F-statistic is 547.16. Thus, after adjusting for other
# covariants, we are 95% confident that Product_level has a significant
# effect on the price of the ride after controlling for other covariates.

#Least squares means
# p-value adjustment:
emmeans(adjust_MLR_p, specs = "Product.Level" , contr = "pairwise")

# The least square means (adjusted for distance, temperature, and product.level )
# show that level 1 rides have average price of $7.56, level 2 rides have average
# price of $9.75, level 3 rides have average price of $16.13 rides, level 4 rides
# have average price of $20.50, and level 5 rides have average price
# of $31.52.

# Combined Final Multi linear regression Model: Evaluating the effects of all our
# variables: Company, Source, Distance, Temperature, Product.Level.

adjust_MLR_s_c <- lm(rides$Price~rides$Source+rides$Company+rides$Distance +
rides$Temperature+rides$Product.Level)
Anova(adjust_MLR_s_c, type = 3)
summary(adjust_MLR_s_c)

conf_dist <- c(2.83131 - (1.6479*0.18074) , 2.8131 + (1.6479*0.18074)); conf_dist

# After adjusting for other variables (distance, temperature, product level), we
# can see that although the model passes the Global F-test (indicating that
```

```
# atleast one of the variables does not equal zero). The 'Source' variable does
# not pass the inference F-test at a alpha = 0.05 level. This is shown by the
# fact that the p-value of the Source variable is 0.1904 and the F-statistic is
# 1.3564. Additionally, the Company variable does not pass the inference F-test
# at the alpha = 0.05 level either since the p-value for the Company variable is
# 0.2045 and the F-statistic is 1.6141. Additionally, the product_level does pass
# the inference F-test at the alpha = 0.05 level since the p-value for the
# product_level variable is <2e-16 and the F-statistic is 532.7.

# Thus, after adjusting for other covariants, we are 95% confident that the
# differences that we saw in the one-way ANOVA models were due to distance and
# product level as opposed to true differences in Price attributable to the
# Source, Company used, or temperature.

# The R-squared value is 84.72% which indicates that 84.72% of the variation in
# price is due to the model.

#Least squares means for significant groups
# p-value adjustment:
prod_level_means <- emmeans(adjust_MLR_s_c, specs = "Product.Level" , contr =
"pairwise")

# The least square means (adjusted for distance, temperature, and product.level,
# company, and source) show that level 1 rides have average price of $7.68, level
# 2 rides have average price of $9.57, level 3 rides have average price of
# $16.18 rides, level 4 rides have average price of $20.50, and level 5 rides
# have average price of $31.68.
```

5. Execute your R code, Copy and Paste results here in this Box.

Run your code and copy the output of your code to here.

```
> # CS 555 Term Project - Uber/Lyft Car Rides
> library(tidyr)
> library(anytime)
> options(scipen=999) # prevents scientific notation for time
> rides_data <- as.data.frame(read.csv('cab_rides.csv', stringsAsFactors = FALSE,
header = TRUE))
>
> ##DATA CLEANING:
> drop <- c("destination","surge_multiplier", 'id', 'product_id')
> rides_data <- rides_data[ , !(names(rides_data) %in% drop)]
>
> rides_data <-rides_data %>% drop_na(price)
>
> N = nrow(rides_data)
> n = 500
> k <- ceiling(N / n)
```

```
> r <- sample(k, 1)
> rows <- seq(r, by = k, length = n)
> rides <- rides_data[rows, ]
>
> weather_data <- as.data.frame(read.csv('weather.csv', stringsAsFactors = FALSE,
header = TRUE))
> drops <- c('rain', "pressure", 'humidity', 'wind', 'clouds')
> weather_data <- weather_data[ , !(names(weather_data) %in% drops)]
>
> for (i in 1:nrow(rides)) {
+   rides[i, 'time_stamp'] = round(rides[i, 'time_stamp']/10^3)
+ }
>
> rides$hour <- NA
> rides$month <- NA
> rides$day <- NA
> for (i in 1:nrow(rides)) {
+   time <-  rides[i, 'time_stamp']
+   z <- as.POSIXlt(time, origin="1970-01-01", tz="EST")
+   hour <- unclass(z)$hour
+   month <- unclass(z)$mon
+   day <- unclass(z)$mday
+   rides[i, 'hour'] = hour
+   rides[i, 'month'] = month
+   rides[i, 'day'] = day
+ }
>
> weather_data$hour <- NA
> weather_data$month <- NA
> weather_data$day <- NA
> for (i in 1:nrow(weather_data)) {
+   time_w <- weather_data[i, 'time_stamp']
+   x <- as.POSIXlt(time_w, origin="1970-01-01", tz="EST")
+   month <- unclass(x)$mon
+   day <- unclass(x)$mday
+   hour <- unclass(x)$hour
+   weather_data[i, 'hour'] = hour
+   weather_data[i, 'month'] = month
+   weather_data[i, 'day'] = day
+ }
>
>
> rides$temperature <- NA
> for (i in 1:nrow(rides)) {
+   hour <- rides[i, 'hour']
+   month <- rides[i, 'month']
+   day <- rides[i, 'day']
+   location <- rides[i, 'source']
+   temp_data <- subset(weather_data, weather_data$month == rides[i, 'month']
+                       & weather_data$day == rides[i, 'day']
```

```
+                        & weather_data$hour == rides[i, 'hour']
+                        & weather_data$location == rides[i, 'source'])[1,]
+   rides[i, 'temperature'] = temp_data$temp
+ }
>
>
> rides <-rides %>% drop_na(temperature)
> for (i in 1:nrow(rides)) {
+   if (rides[i, 'name'] %in% c('UberPool', 'Shared')) {
+     rides[i, 'name'] = 1}
+   if (rides[i, 'name'] %in% c('UberX', 'Lyft', 'WAV')) {
+     rides[i, 'name'] = 2}
+   if (rides[i, 'name'] %in% c('UberXL', 'Lyft XL')) {
+     rides[i, 'name'] = 3}
+   if (rides[i, 'name'] %in% c('Black', 'Lux Black', 'Lux')) {
+     rides[i, 'name'] = 4}
+   if (rides[i, 'name'] %in% c('Black SUV', 'Lux Black XL')) {
+     rides[i, 'name'] = 5}
+ }

> ##Final Datacleaning:
> names(rides)[6] <- 'Product.Level'
> names(rides)[2] <- 'Company'
> names(rides)[1] <- 'Distance'
> names(rides)[4] <- 'Source'
> names(rides)[10] <- 'Temperature'
> names(rides)[5] <- 'Price'
>
> drop <- c("time_stamp","month", 'day', 'hour')
> rides <- rides[ , !(names(rides) %in% drop)]
>
> rides <- rides[,c("Company", "Source", "Product.Level", "Distance",
"Temperature", "Price")]
> rides_data <-rides_data %>% drop_na(price)
> rides <-rides %>% drop_na(Temperature)
>
> #write.csv(rides, 'uber_lyft_dataset.csv', row.names = FALSE)
> remove(list = ls())
>################################################################################
> ## DATA VISUALIZATION AND ANALYSIS - RESEARCH SCENARIO:
>
> options(scipen=10)
> rides <- as.data.frame(read.csv('uber_lyft_dataset.csv', stringsAsFactors =
FALSE, header = TRUE))
>
> ## Two Sample Means T-test:
> south <- subset(rides, rides$Source == 'South Station')$Price
> financial_dis <- subset(rides, rides$Source == 'Financial District')$Price
> len_south <- length(south)
> len_financial_dis <- length(financial_dis)
```

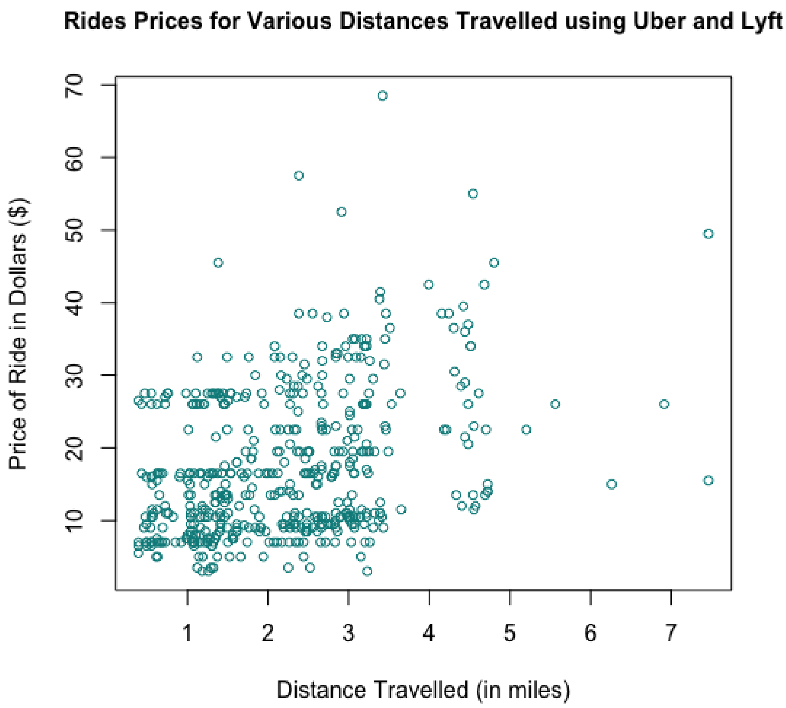
```
> sd_south <- sd(south)
> sd_fin <- sd(financial_dis)
> df <- ((sd_south^2/len_south) +
(sd_fin^2/len_financial_dis))^2/(((sd_south^2/len_south)^2/(len_south-1)) +
((sd_fin^2/len_financial_dis)^2/(len_financial_dis-1)))
> t_critical <- qt(0.95, df);t_critical #t-critical = 1.66517
[1] 1.665172
> t.test(financial_dis, south, alternative='greater', conf.level=0.95)

      Welch Two Sample t-test

data:  financial_dis and south
t = 1.6719, df = 75.923, p-value = 0.04933
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.01607659        Inf
sample estimates:
mean of x mean of y
 20.91667  16.93056
> r <- cor(rides$Distance, rides$Price); r # r = 0.3641559
[1] 0.3667035
> df <- length(rides$Price) - 2
> t_critical <- qt(0.975, df=df); t_critical # t_critical = 1.964758
[1] 1.964758
>
> ## Correlation Test Between Distance and Price:
> t <- r*(sqrt((df)/(1 - r^2))); t # t = 8.778396; # p-value < 2.2e-16.
[1] 8.778396
> cor.test(rides$Distance , rides$Price , alternative='two.sided',
method='pearson', conf.level = 0.95)

      Pearson's product-moment correlation

data:  rides$Distance and rides$Price
t = 8.7784, df = 496, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2881204 0.4403806
sample estimates:
      cor
0.3667035
> ## Simple Linear Regression on Distance and Price:
> plot(rides$Distance, rides$Price,
+      main = 'Rides Prices for Various Distances Travelled using Uber and Lyft',
+      xlab = 'Distance Travelled (in miles)',
+      ylab = 'Price of Ride in Dollars ($)',
            +      col = 'darkcyan', cex.main = 1, pch = 1, cex = 0.8)
```




```
> SLR <- lm(rides$Price ~ rides$Distance)
> summary(SLR)

Call:
lm(formula = rides$Price ~ rides$Distance)

Residuals:
    Min      1Q  Median      3Q     Max
-18.648  -7.297  -1.911   5.401  47.239

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     10.3512     0.8959  11.554   <2e-16 ***
rides$Distance   3.1899     0.3634   8.778   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.385 on 496 degrees of freedom
Multiple R-squared:  0.1345, Adjusted R-squared:  0.1327
F-statistic: 77.06 on 1 and 496 DF,  p-value: < 2.2e-16

> abline(a=10.402 , b=3.171, col = 'darkorange', lwd = 2)
```


```
> distance_bar <- mean(rides$Distance)
> sd_distance <- sd(rides$Distance)
> price_bar <- mean(rides$Price)
> sd_price <- sd(rides$Price)
> beta1 <- r*sd_price/sd_distance;beta1
[1] 3.189919
> beta0 <- price_bar - beta1*distance_bar;beta0
[1] 10.35121
> anova <- anova(SLR); anova

Analysis of Variance Table

Response: rides$Price
                Df Sum Sq Mean Sq F value    Pr(>F)
rides$Distance   1   6787  6786.6   77.06 < 2.2e-16 ***
Residuals      496  43682    88.1
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> SE_beta1 <- summary(SLR);SE_beta1

Call:
lm(formula = rides$Price ~ rides$Distance)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-18.648  -7.297  -1.911   5.401  47.239

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     10.3512     0.8959  11.554   <2e-16 ***
rides$Distance   3.1899     0.3634   8.778   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.385 on 496 degrees of freedom
Multiple R-squared:  0.1345, Adjusted R-squared:  0.1327
F-statistic: 77.06 on 1 and 496 DF,  p-value: < 2.2e-16
> F_critical <- qf(0.95, df1 = 1, df2 = 496); F_Critical # F_critical = 3.8602
[1] 3.860275
> beta1_95confidence <- confint(SLR, level = 0.95)[2,];beta1_95confidence
   2.5 %   97.5 %
2.475960 3.903879
> r_squared <- r^2; r_squared
[1] 0.1344715
>
> ## MLR with Company, Product Level, Temeperature, and Distance as explanatory
variables:
> MLR <- lm(rides$Price~rides$Product.Level + rides$Temperature + rides$Distance)
> qf(.95, df1=3, df2=494) #F(3, 494, 0.05) = F_critical = 2.6229
[1] 2.622952
> summary(MLR)

Call:
lm(formula = rides$Price ~ rides$Product.Level + rides$Temperature +
    rides$Distance)

Residuals:
    Min      1Q  Median      3Q     Max
-10.958  -2.539  -0.573   1.578  48.100

Coefficients:
                    Estimate Std. Error t value          Pr(>|t|)
(Intercept)         -9.56363    1.28923  -7.418 0.000000000000522 ***
rides$Product.Level  6.03700    0.14830  40.708           < 2e-16 ***
rides$Temperature    0.05013    0.03003   1.669            0.0957 .
rides$Distance       2.93284    0.17459  16.798           < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.498 on 494 degrees of freedom
Multiple R-squared:  0.802,  Adjusted R-squared:  0.8008
F-statistic: 666.9 on 3 and 494 DF,  p-value: < 2.2e-16

> t_critical <- qt(0.95 , df = 494); t_critical #t_critical = 1.6479
```

```
[1] 1.647944
> conf_dist <- c(2.9328 - (1.6479*0.17459) , 2.9328 + (1.6479*0.17459)); ;
conf_dist
[1] 2.645093 3.220507
> conf_prodlev <- c(6.03700 - (1.6479*0.14830) , 6.03700 + (1.6479*0.14830));
conf_prodlev
[1] 5.792616 6.281384
> regss <- sum((fitted(MLR) - mean(rides$Price))^2)
> resiss <- sum((rides$Price-fitted(MLR))^2)
> totalss <- regss + resiss
> R2 <- regss/totalss; R2
[1] 0.8019829
>
> ## One way ANOVA to compare means across Uber rides and Lyft rides:
> rides$Company <- factor(rides$Company, levels = c('Uber', 'Lyft'))
> one_way_ANOVA <- aov(rides$Price~rides$Company , data=rides)
> f_critical <- qf(.95, df1=1, df2=496); f_critical #F_critical = 3.8602
[1] 3.860275
> summary(one_way_ANOVA)
               Df Sum Sq Mean Sq F value Pr(>F)  
rides$Company   1    576   576.4    5.73  0.017 *
Residuals     496  49893   100.6                 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> t_critical <- qt (.975 , df =496); t_critical #t_critical = 1.964758.
[1] 1.964758
>
> pairwise.t.test(rides$Price, rides$Company, p.adj='none')

      Pairwise comparisons using t tests with pooled SD 

data:  rides$Price and rides$Company 

     Uber 
Lyft 0.017

P value adjustment method: none 
>
> ## One-Way Anova analysis using a linear regression:
> rides$uber <- ifelse(rides$Company == 'Uber', 1, 0)
> rides$lyft <- ifelse(rides$Company == 'Lyft',1,0)
> one_way_model <- lm(rides$Price ~ rides$uber, data=rides)
> summary(one_way_model)

Call:
lm(formula = rides$Price ~ rides$uber, data = rides)

Residuals:
    Min      1Q  Median      3Q     Max 
```

```
-15.429  -7.429  -1.929   7.571  52.225

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  18.4294     0.6529  28.229   <2e-16 ***
rides$uber   -2.1546     0.9001  -2.394    0.017 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.03 on 496 degrees of freedom
Multiple R-squared:  0.01142, Adjusted R-squared:  0.009428
F-statistic:  5.73 on 1 and 496 DF,  p-value: 0.01704
>
> # Adjusting for other variables ANCOVA (ie. distance, temperature,
product_level):
> install.packages("carData")
trying URL 'https://cran.rstudio.com/bin/macosx/el-
capitan/contrib/3.6/carData_3.0-3.tgz'
Content type 'application/x-gzip' length 1815539 bytes (1.7 MB)
==================================================
downloaded 1.7 MB


The downloaded binary packages are in
      /var/folders/jg/zys1slm143g7n5mnm31mkv280000gn/T//RtmpSzKteU/downloaded_pac
kages
> install.packages("car")
trying URL 'https://cran.rstudio.com/bin/macosx/el-capitan/contrib/3.6/car_3.0-
5.tgz'
Content type 'application/x-gzip' length 1561232 bytes (1.5 MB)
==================================================
downloaded 1.5 MB


The downloaded binary packages are in
      /var/folders/jg/zys1slm143g7n5mnm31mkv280000gn/T//RtmpSzKteU/downloaded_pac
kages
> library(carData)
> library(car)
> adjust_MLR <- lm(rides$Price~rides$Company+rides$Distance +
rides$Temperature+rides$Product.Level)
> Anova(adjust_MLR, type = 3)
Anova Table (Type III tests)

Response: rides$Price
                    Sum Sq  Df   F value            Pr(>F)
(Intercept)           1082   1   53.4412 0.000000000001083 ***
rides$Company            8   1    0.4145           0.51998
rides$Distance        5653   1  279.1036         < 2.2e-16 ***
rides$Temperature       57   1    2.7995           0.09493 .
```

```
rides$Product.Level  32700   1 1614.4594        < 2.2e-16 ***
Residuals             9985 493
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary(adjust_MLR)
Call:
lm(formula = rides$Price ~ rides$Company + rides$Distance + rides$Temperature +
    rides$Product.Level)

Residuals:
    Min      1Q  Median      3Q     Max
-11.025  -2.500  -0.609   1.630  47.986

Coefficients:
                    Estimate Std. Error t value         Pr(>|t|)
(Intercept)         -9.47906    1.29666  -7.310 0.00000000000108 ***
rides$CompanyLyft   -0.26454    0.41088  -0.644           0.5200
rides$Distance       2.92524    0.17510  16.706          < 2e-16 ***
rides$Temperature    0.05028    0.03005   1.673           0.0949 .
rides$Product.Level  6.05381    0.15067  40.180          < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.5 on 493 degrees of freedom
Multiple R-squared:  0.8021, Adjusted R-squared:  0.8005
F-statistic: 499.7 on 4 and 493 DF,  p-value: < 2.2e-16
>
> install.packages("emmeans")
trying URL 'https://cran.rstudio.com/bin/macosx/el-
capitan/contrib/3.6/emmeans_1.4.3.01.tgz'
Content type 'application/x-gzip' length 1417347 bytes (1.4 MB)
==================================================
downloaded 1.4 MB


The downloaded binary packages are in
      /var/folders/jg/zys1slm143g7n5mnm31mkv280000gn/T//RtmpSzKteU/downloaded_pac
kages
> install.packages('lsmeans')
trying URL 'https://cran.rstudio.com/bin/macosx/el-
capitan/contrib/3.6/lsmeans_2.30-0.tgz'
Content type 'application/x-gzip' length 43500 bytes (42 KB)
==================================================
downloaded 42 KB


The downloaded binary packages are in
      /var/folders/jg/zys1slm143g7n5mnm31mkv280000gn/T//RtmpSzKteU/downloaded_pac
kages
> library(emmeans)
```

```
Welcome to emmeans.
NOTE -- Important change from versions <= 1.41:
    Indicator predictors are now treated as 2-level factors by default.
    To revert to old behavior, use emm_options(cov.keep = character(0))
> library(lsmeans)
The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.
> emmeans(adjust_MLR, specs = "Company" , contr = "pairwise")
$emmeans
 Company emmean    SE  df lower.CL upper.CL
 Uber      17.4 0.280 493     16.9     18.0
 Lyft      17.2 0.296 493     16.6     17.7

Confidence level used: 0.95

$contrasts
 contrast    estimate    SE  df t.ratio p.value
 Uber - Lyft    0.265 0.411 493 0.644   0.5200

> ## One way ANOVA to compare means across pick-up location:
> rides$Source <- factor(rides$Source, levels = unique(rides$Source))
> one_way_ANOVA_s <- aov(rides$Price~rides$Source , data=rides)
> f_critical <- qf(.95, df1=12, df2=486); f_critical #F_critical = 1.77211
[1] 1.77211
> summary(one_way_ANOVA_s)
              Df Sum Sq Mean Sq F value  Pr(>F)
rides$Source  11   3214  292.18   3.005 0.00069 ***
Residuals    486  47255   97.23
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> aggregate(rides$Price, by=list(rides$Source), summary)
                Group.1   x.Min. x.1st Qu. x.Median   x.Mean x.3rd Qu.   x.Max.
1  Northeastern University  3.00000  10.87500 16.50000 19.18182  26.00000 57.50000
2        Boston University  7.00000  11.00000 16.25000 20.95455  28.12500 55.00000
3                 West End  5.00000  10.12500 13.75000 16.74559  25.25000 35.00000
4            North Station  5.00000  10.25000 16.00000 16.78431  23.50000 35.00000
5            South Station  3.00000   9.00000 16.50000 16.93056  23.37500 38.50000
6                North End  3.50000   9.00000 11.50000 14.86667  16.50000 38.50000
7                   Fenway  5.00000  10.50000 16.50000 18.18750  21.75000 68.50000
8              Beacon Hill  3.50000   9.00000 14.50000 17.09000  27.50000 32.50000
9       Financial District  3.00000  10.50000 21.00000 20.91667  27.50000 49.50000
10                Back Bay  3.50000   9.50000 13.50000 18.11538  26.50000 45.50000
11         Haymarket Square  5.00000   7.00000  9.00000 10.93421  13.12500 26.00000
12        Theatre District  5.00000   9.00000 16.00000 16.43023  22.50000 45.50000
> aggregate(rides$Price, by=list(rides$Source), var)
                Group.1         x
1  Northeastern University 125.11734
2        Boston University 157.67230
3                 West End  81.93400
4            North Station  68.94255
```

```
5           South Station  90.98790
6               North End  77.70909
7                  Fenway 150.75403
8             Beacon Hill  80.57847
9      Financial District 132.58435
10               Back Bay 104.92713
11       Haymarket Square  28.67799
12       Theatre District  77.00692
> pairwise.t.test(rides$Price, rides$Source, p.adj='none')

        Pairwise comparisons using t tests with pooled SD 

data:  rides$Price and rides$Source 

                   Northeastern University Boston University West End North Station South Station North End Fenway
Boston University  0.39951                 -                 -        -             -             -         -     
West End           0.27978                 0.06218           -        -             -             -         -     
North Station      0.23791                 0.04037           0.98586  -             -             -         -     
South Station      0.31018                 0.07001           0.93751  0.94571       -             -         -     
North End          0.03954                 0.00376           0.40213  0.34215       0.34972       -         -     
Fenway             0.66446                 0.22770           0.55298  0.52834       0.60006       0.14593   -     
Beacon Hill        0.30527                 0.05855           0.87521  0.87628       0.94106       0.27305   0.62319
Financial District 0.41515                 0.98580           0.06733  0.04485       0.07573       0.00442   0.23877
Back Bay           0.62312                 0.19109           0.55409  0.52600       0.60338       0.13273   0.97555
Haymarket Square   0.00018                 0.0000057         0.01287  0.00585       0.00921       0.07089   0.00229
Theatre District   0.19377                 0.03288           0.88923  0.86238       0.82238       0.45751   0.44564
                   Beacon Hill Financial District Back Bay Haymarket Square
Boston University  -           -                  -        -               
West End           -           -                  -        -               
North Station      -           -                  -        -               
South Station      -           -                  -        -               
North End          -           -                  -        -               
Fenway             -           -                  -        -               
Beacon Hill        -           -                  -        -               
Financial District 0.06433     -                  -        -               
Back Bay           0.62666     0.20203            -        -               
Haymarket Square   0.00389     0.0000077          0.00149  -               
Theatre District   0.74781     0.03649            0.43999  0.01263         

P value adjustment method: none 
> 
> TukeyHSD(one_way_ANOVA_s)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = rides$Price ~ rides$Source, data = rides)

$`rides$Source`
                                                  diff         lwr         upr     p adj
Boston University-Northeastern University   1.77272727  -5.1314480  8.6769026 0.9995178
West End-Northeastern University           -2.43622995  -9.8306592  4.9581993 0.9953223
North Station-Northeastern University      -2.39750446  -9.0605614  4.2655525 0.9901564
South Station-Northeastern University      -2.25126263  -9.5289024  5.0263772 0.9973097
North End-Northeastern University          -4.31515152 -11.1808632  2.5505601 0.6486969
Fenway-Northeastern University             -0.99431818  -8.5179688  6.5293324 0.9999994
Beacon Hill-Northeastern University        -2.09181818  -8.7856645  4.6020281 0.9970556
Financial District-Northeastern University  1.73484848  -5.2510359  8.7207329 0.9996496
Back Bay-Northeastern University           -1.06643357  -8.1884596  6.0555924 0.9999979
Haymarket Square-Northeastern University   -8.24760766 -15.4191397 -1.0760757 0.0096494
Theatre District-Northeastern University   -2.75158562  -9.6957855  4.1926142 0.9787502
West End-Boston University                 -4.20895722 -11.6033864  3.1854720 0.7776881
North Station-Boston University            -4.17023173 -10.8332887  2.4928253 0.6548155
South Station-Boston University            -4.02398990 -11.3016297  3.2536499 0.8087611
North End-Boston University                -6.08787879 -12.9535904  0.7778329 0.1398717
Fenway-Boston University                   -2.76704545 -10.2906961  4.7566052 0.9882277
Beacon Hill-Boston University              -3.86454545 -10.5583918  2.8293009 0.7613722
Financial District-Boston University       -0.03787879  -7.0237632  6.9480056 1.0000000
```

```
Back Bay-Boston University                    -2.83916084  -9.9611868  4.2828652 0.9777378
Haymarket Square-Boston University           -10.02033493 -17.1918669 -2.8488029 0.0003517
Theatre District-Boston University            -4.52431290 -11.4685127  2.4198870 0.5943801
North Station-West End                         0.03872549  -7.1310914  7.2085424 1.0000000
South Station-West End                         0.18496732  -7.5593201  7.9292547 1.0000000
North End-West End                            -1.87892157  -9.2374502  5.4796070 0.9995427
Fenway-West End                                1.44191176  -6.5340061  9.4178296 0.9999852
Beacon Hill-West End                           0.34441176  -6.8540273  7.5428508 1.0000000
Financial District-West End                    4.17107843  -3.2996998 11.6418566 0.7987190
Back Bay-West End                              1.36979638  -6.2284403  8.9680331 0.9999856
Haymarket Square-West End                     -5.81137771 -13.4560371  1.8332817 0.3445629
Theatre District-West End                     -0.31535568  -7.7471696  7.1164582 1.0000000
South Station-North Station                    0.14624183  -6.9030654  7.1955491 1.0000000
North End-North Station                       -1.91764706  -8.5408402  4.7055461 0.9985193
Fenway-North Station                           1.40318627  -5.8998275  8.7062001 0.9999724
Beacon Hill-North Station                      0.30568627  -6.1391776  6.7505501 1.0000000
Financial District-North Station               4.13235294  -2.6153335 10.8800394 0.6856552
Back Bay-North Station                         1.33107089  -5.5574664  8.2196082 0.9999708
Haymarket Square-North Station                -5.85010320 -12.7898123  1.0896059 0.1970099
Theatre District-North Station                -0.35408117  -7.0586023  6.3504399 1.0000000
North End-South Station                       -2.06388889  -9.3050490  5.1772713 0.9987180
Fenway-South Station                           1.25694444  -6.6108203  9.1247092 0.9999959
Beacon Hill-South Station                      0.15944444  -6.9189722  7.2378611 1.0000000
Financial District-South Station               3.98611111  -3.3690901 11.3413123 0.8283904
Back Bay-South Station                         1.18482906  -6.2997991  8.6694572 0.9999962
Haymarket Square-South Station                -5.99634503 -13.5280962  1.5354061 0.2743336
Theatre District-South Station                -0.50032300  -7.8159443  6.8152983 1.0000000
Fenway-North End                               3.32083333  -4.1675362 10.8092029 0.9512599
Beacon Hill-North End                          2.22333333  -4.4308336  8.8775003 0.9947212
Financial District-North End                   6.05000000  -0.8978731 12.9978731 0.1590566
Back Bay-North End                             3.24871795  -3.8360272 10.3334631 0.9386111
Haymarket Square-North End                    -3.93245614 -11.0669660  3.2020537 0.8119400
Theatre District-North End                     1.56356589  -5.3423932  8.4695250 0.9998574
Beacon Hill-Fenway                            -1.09750000  -8.4286159  6.2336159 0.9999979
Financial District-Fenway                      2.72916667  -4.8695347 10.3278680 0.9903037
Back Bay-Fenway                               -0.07211538  -7.7961646  7.6519338 1.0000000
Haymarket Square-Fenway                       -7.25328947 -15.0230097  0.5164308 0.0936416
Theatre District-Fenway                       -1.75726744  -9.3176638  5.8031289 0.9998155
Financial District-Beacon Hill                 3.82666667  -2.9514247 10.6047580 0.7867773
Back Bay-Beacon Hill                           1.02538462  -5.8929386  7.9437078 0.9999981
Haymarket Square-Beacon Hill                  -6.15578947 -13.1250658  0.8134869 0.1438481
Theatre District-Beacon Hill                  -0.65976744  -7.3948883  6.0753534 1.0000000
Back Bay-Financial District                   -2.80128205 -10.0025457  4.3999816 0.9815877
Haymarket Square-Financial District           -9.98245614 -17.2326848 -2.7322275 0.0004742
Theatre District-Financial District           -4.48643411 -11.5118775  2.5390093 0.6250470
Haymarket Square-Back Bay                     -7.18117409 -14.5626705  0.2003223 0.0651163
Theatre District-Back Bay                     -1.68515206  -8.8459851  5.4756809 0.9997918
Theatre District-Haymarket Square              5.49602203  -1.7140505 12.7060946 0.3402601
>
> # One-Way Anova analysis using a linear regression:
> rides$NEUni <- ifelse(rides$Source == 'Northeastern University', 1, 0)
> rides$North <- ifelse(rides$Source == 'North Station', 1, 0)
> rides$Fenway <- ifelse(rides$Source == 'Fenway', 1, 0)
> rides$Backbay <- ifelse(rides$Source == 'Back Bay', 1, 0)
> rides$BU <- ifelse(rides$Source == 'Boston University', 1, 0)
> rides$South <- ifelse(rides$Source == 'South Station', 1, 0)
> rides$Beacon <- ifelse(rides$Source == 'Beacon Hill', 1, 0)
> rides$Hay <- ifelse(rides$Source == 'Haymarket Square', 1, 0)
```

```
> rides$WestEnd <- ifelse(rides$Source == 'West End', 1, 0)
> rides$NorthEnd <- ifelse(rides$Source == 'North End', 1, 0)
> rides$Findist <- ifelse(rides$Source == 'Financial District', 1, 0)
> rides$Theatre <- ifelse(rides$Source == 'Theatre District', 1, 0)
> # holding Northeastern University as reference group
> one_way_model_s <- lm(rides$Price ~ rides$North + rides$Fenway +
+                                     rides$Backbay + rides$BU + rides$South +
+                                     rides$Beacon + rides$Hay + rides$WestEnd
+                                     + rides$NorthEnd + rides$Findist +
rides$Theatre , data=rides)
> summary(one_way_model_s)
Call:
lm(formula = rides$Price ~ rides$North + rides$Fenway + rides$Backbay +
    rides$BU + rides$South + rides$Beacon + rides$Hay + rides$WestEnd +
    rides$NorthEnd + rides$Findist + rides$Theatre, data = rides)

Residuals:
    Min      1Q  Median      3Q     Max
-17.917  -7.431  -2.136   6.759  50.312

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     19.1818     1.4865  12.904  < 2e-16 ***
rides$North     -2.3975     2.0289  -1.182 0.237907
rides$Fenway    -0.9943     2.2909  -0.434 0.664463
rides$Backbay   -1.0664     2.1686  -0.492 0.623115
rides$BU         1.7727     2.1023   0.843 0.399513
rides$South     -2.2513     2.2160  -1.016 0.310180
rides$Beacon    -2.0918     2.0383  -1.026 0.305270
rides$Hay       -8.2476     2.1837  -3.777 0.000178 ***
rides$WestEnd   -2.4362     2.2516  -1.082 0.279785
rides$NorthEnd  -4.3152     2.0906  -2.064 0.039540 *
rides$Findist    1.7348     2.1272   0.816 0.415149
rides$Theatre   -2.7516     2.1145  -1.301 0.193771
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.861 on 486 degrees of freedom
Multiple R-squared:  0.06368,Adjusted R-squared:  0.04249
F-statistic: 3.005 on 11 and 486 DF,  p-value: 0.0006895
>
> # Adjusting for other variables (ie. distance, temperature, product_level):
> adjust_MLR_s <- lm(rides$Price~rides$Source+rides$Distance +
rides$Temperature+rides$Product.Level)
>
> Anova(adjust_MLR_s, type = 3)
Anova Table (Type III tests)

Response: rides$Price
                    Sum Sq  Df   F value         Pr(>F)
```

```
(Intercept)            761   1   37.3605 0.000000002026 ***
rides$Source           157  11    0.7028         0.7363
rides$Distance        3998   1  196.3045      < 2.2e-16 ***
rides$Temperature       47   1    2.3141         0.1289
rides$Product.Level  32947   1 1617.8291      < 2.2e-16 ***
Residuals             9836 483
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> summary(adjust_MLR_s)
Call:
lm(formula = rides$Price ~ rides$Source + rides$Distance + rides$Temperature +
    rides$Product.Level)

Residuals:
    Min      1Q  Median      3Q     Max
-11.141  -2.404  -0.578   1.767  48.282

Coefficients:
                               Estimate Std. Error t value      Pr(>|t|)
(Intercept)                    -9.34348    1.52863  -6.112 0.00000000203 ***
rides$SourceBoston University   1.32653    0.96353   1.377         0.169
rides$SourceWest End           -0.55794    1.05709  -0.528         0.598
rides$SourceNorth Station      -0.35606    0.94096  -0.378         0.705
rides$SourceSouth Station       0.55253    1.04187   0.530         0.596
rides$SourceNorth End           0.25037    1.00156   0.250         0.803
rides$SourceFenway              0.11920    1.05055   0.113         0.910
rides$SourceBeacon Hill        -0.47524    0.95503  -0.498         0.619
rides$SourceFinancial District  0.90548    0.97885   0.925         0.355
rides$SourceBack Bay            0.79278    1.01459   0.781         0.435
rides$SourceHaymarket Square   -0.28667    1.08032  -0.265         0.791
rides$SourceTheatre District    0.04471    1.00475   0.044         0.965
rides$Distance                  2.83119    0.20207  14.011       < 2e-16 ***
rides$Temperature               0.04614    0.03033   1.521         0.129
rides$Product.Level             6.02759    0.14986  40.222       < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.513 on 483 degrees of freedom
Multiple R-squared:  0.8051,  Adjusted R-squared:  0.7995
F-statistic: 142.5 on 14 and 483 DF,  p-value: < 2.2e-16
>
> emmeans(adjust_MLR_s, specs = "Source" , contr = "pairwise")
$emmeans
 Source                   emmean    SE  df lower.CL upper.CL
 Northeastern University    17.1 0.703 483     15.7     18.5
 Boston University          18.4 0.703 483     17.1     19.8
 West End                   16.6 0.777 483     15.0     18.1
 North Station              16.8 0.633 483     15.5     18.0
 South Station              17.7 0.755 483     16.2     19.1
 North End                  17.4 0.683 483     16.0     18.7
 Fenway                     17.2 0.806 483     15.6     18.8
 Beacon Hill                16.6 0.640 483     15.4     17.9
```

```
 Financial District        18.0 0.703 483     16.6     19.4
 Back Bay                  17.9 0.724 483     16.5     19.3
 Haymarket Square          16.8 0.770 483     15.3     18.3
 Theatre District          17.2 0.694 483     15.8     18.5

Confidence level used: 0.95

$contrasts
 contrast                                     estimate    SE  df t.ratio p.value
 Northeastern University - Boston University   -1.3265 0.964 483 -1.377  0.9675
 Northeastern University - West End             0.5579 1.057 483  0.528  1.0000
 Northeastern University - North Station        0.3561 0.941 483  0.378  1.0000
 Northeastern University - South Station       -0.5525 1.042 483 -0.530  1.0000
 Northeastern University - North End           -0.2504 1.002 483 -0.250  1.0000
 Northeastern University - Fenway              -0.1192 1.051 483 -0.113  1.0000
 Northeastern University - Beacon Hill          0.4752 0.955 483  0.498  1.0000
 Northeastern University - Financial District  -0.9055 0.979 483 -0.925  0.9989
 Northeastern University - Back Bay            -0.7928 1.015 483 -0.781  0.9998
 Northeastern University - Haymarket Square     0.2867 1.080 483  0.265  1.0000
 Northeastern University - Theatre District    -0.0447 1.005 483 -0.044  1.0000
 Boston University - West End                   1.8845 1.058 483  1.781  0.8278
 Boston University - North Station              1.6826 0.943 483  1.784  0.8259
 Boston University - South Station              0.7740 1.041 483  0.743  0.9999
 Boston University - North End                  1.0762 1.002 483  1.074  0.9956
 Boston University - Fenway                     1.2073 1.053 483  1.147  0.9923
 Boston University - Beacon Hill                1.8018 0.953 483  1.891  0.7646
 Boston University - Financial District         0.4211 0.978 483  0.430  1.0000
 Boston University - Back Bay                   0.5338 1.015 483  0.526  1.0000
 Boston University - Haymarket Square           1.6132 1.081 483  1.493  0.9422
 Boston University - Theatre District           1.2818 1.005 483  1.276  0.9818
 West End - North Station                      -0.2019 1.003 483 -0.201  1.0000
 West End - South Station                      -1.1105 1.080 483 -1.028  0.9970
 West End - North End                          -0.8083 1.029 483 -0.786  0.9998
 West End - Fenway                             -0.6771 1.125 483 -0.602  1.0000
 West End - Beacon Hill                        -0.0827 1.006 483 -0.082  1.0000
 West End - Financial District                 -1.4634 1.052 483 -1.391  0.9649
 West End - Back Bay                           -1.3507 1.059 483 -1.275  0.9818
 West End - Haymarket Square                   -0.2713 1.083 483 -0.250  1.0000
 West End - Theatre District                   -0.6027 1.037 483 -0.581  1.0000
 North Station - South Station                 -0.9086 0.987 483 -0.920  0.9989
 North Station - North End                     -0.6064 0.935 483 -0.649  1.0000
 North Station - Fenway                        -0.4753 1.021 483 -0.465  1.0000
 North Station - Beacon Hill                    0.1192 0.902 483  0.132  1.0000
 North Station - Financial District            -1.2615 0.945 483 -1.336  0.9741
 North Station - Back Bay                      -1.1488 0.962 483 -1.194  0.9893
 North Station - Haymarket Square              -0.0694 1.002 483 -0.069  1.0000
 North Station - Theatre District              -0.4008 0.942 483 -0.425  1.0000
 South Station - North End                      0.3022 1.011 483  0.299  1.0000
 South Station - Fenway                         0.4333 1.111 483  0.390  1.0000
 South Station - Beacon Hill                    1.0278 0.988 483  1.041  0.9967
 South Station - Financial District            -0.3530 1.037 483 -0.340  1.0000
 South Station - Back Bay                      -0.2402 1.044 483 -0.230  1.0000
 South Station - Haymarket Square               0.8392 1.066 483  0.788  0.9997
 South Station - Theatre District               0.5078 1.020 483  0.498  1.0000
 North End - Fenway                             0.1312 1.069 483  0.123  1.0000
 North End - Beacon Hill                        0.7256 0.934 483  0.777  0.9998
 North End - Financial District                -0.6551 0.992 483 -0.660  1.0000
 North End - Back Bay                          -0.5424 0.992 483 -0.547  1.0000
 North End - Haymarket Square                   0.5370 1.001 483  0.536  1.0000
 North End - Theatre District                   0.2057 0.963 483  0.213  1.0000
 Fenway - Beacon Hill                           0.5944 1.033 483  0.575  1.0000
```

```
 Fenway - Financial District                          -0.7863 1.061 483 -0.741  0.9999
 Fenway - Back Bay                                    -0.6736 1.087 483 -0.620  1.0000
 Fenway - Haymarket Square                             0.4059 1.137 483  0.357  1.0000
 Fenway - Theatre District                             0.0745 1.074 483  0.069  1.0000
 Beacon Hill - Financial District                     -1.3807 0.952 483 -1.451  0.9525
 Beacon Hill - Back Bay                               -1.2680 0.965 483 -1.313  0.9772
 Beacon Hill - Haymarket Square                       -0.1886 0.998 483 -0.189  1.0000
 Beacon Hill - Theatre District                       -0.5199 0.942 483 -0.552  1.0000
 Financial District - Back Bay                         0.1127 1.011 483  0.111  1.0000
 Financial District - Haymarket Square                 1.1922 1.065 483  1.119  0.9937
 Financial District - Theatre District                 0.8608 0.996 483  0.864  0.9994
 Back Bay - Haymarket Square                           1.0794 1.051 483  1.027  0.9970
 Back Bay - Theatre District                           0.7481 1.000 483  0.748  0.9998
 Haymarket Square - Theatre District                  -0.3314 1.017 483 -0.326  1.0000

P value adjustment: tukey method for comparing a family of 12 estimates

> rides$Product.Level <- factor(rides$Product.Level, levels =
unique(rides$Product.Level))
> one_way_ANOVA_p <- aov(rides$Price~rides$Product.Level , data=rides)
> f_critical <- qf(.95, df1=5, df2=493); f_critical #F_critical = 2.23229
[1] 2.232298
> summary(one_way_ANOVA_p)
                     Df Sum Sq Mean Sq F value Pr(>F)
rides$Product.Level   4  36757    9189   330.4 <2e-16 ***
Residuals           493  13712      28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> aggregate(rides$Price, by=list(rides$Product.Level), summary)
  Group.1     x.Min. x.1st Qu.  x.Median    x.Mean x.3rd Qu.    x.Max.
1       2  7.000000  7.500000  9.500000  9.798800 10.500000 36.500000
2       1  3.000000  5.500000  7.000000  7.171233  9.000000 12.000000
3       5 26.000000 27.500000 29.500000 31.811881 34.000000 57.500000
4       4 10.500000 16.500000 19.500000 20.218447 22.750000 34.000000
5       3  9.000000 12.000000 16.000000 16.348958 18.000000 68.500000
> aggregate(rides$Price, by=list(rides$Product.Level), var)
  Group.1         x
1       2 10.508970
2       1  4.459855
3       5 43.804257
4       4 25.174852
5       3 54.100630
> pairwise.t.test(rides$Price, rides$Product.Level, p.adj='none')

      Pairwise comparisons using t tests with pooled SD

data:  rides$Price and rides$Product.Level

  2       1       5       4
1 0.00078 -       -       -
5 < 2e-16 < 2e-16 -       -
4 < 2e-16 < 2e-16 < 2e-16 -
3 < 2e-16 < 2e-16 < 2e-16 0.00000034
```

```
P value adjustment method: none
> TukeyHSD(one_way_ANOVA_p)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = rides$Price ~ rides$Product.Level, data = rides)

$`rides$Product.Level`
          diff        lwr         upr     p adj
1-2  -2.627567  -4.754538  -0.5005966 0.0069220
5-2  22.013081  20.081185  23.9449776 0.0000000
4-2  10.419647   8.498152  12.3411409 0.0000000
3-2   6.550158   4.590633   8.5096834 0.0000000
5-1  24.640648  22.422463  26.8588336 0.0000000
4-1  13.047214  10.838082  15.2563454 0.0000000
3-1   9.177725   6.935436  11.4200146 0.0000000
4-5 -11.593435 -13.615434  -9.5714349 0.0000000
3-5 -15.462923 -17.521097 -13.4047487 0.0000000
3-4  -3.869488  -5.917902  -1.8210748 0.0000033

> rides$one <- ifelse(rides$Product.Level == 1, 1, 0)
> rides$two <- ifelse(rides$Product.Level == 2, 1, 0)
> rides$three <- ifelse(rides$Product.Level == 3, 1, 0)
> rides$four <- ifelse(rides$Product.Level == 4, 1, 0)
> rides$five <- ifelse(rides$Product.Level == 5, 1, 0)
>
> # holding UberPool as reference group
> one_way_model_p <- lm(rides$Price ~ rides$two + rides$three +
+                          rides$four + rides$five , data=rides)
> summary(one_way_model_p)

Call:
lm(formula = rides$Price ~ rides$two + rides$three + rides$four +
    rides$five, data = rides)

Residuals:
   Min     1Q Median     3Q    Max
-9.718 -3.340 -0.299  1.829 52.151

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.1712     0.6173  11.618  < 2e-16 ***
rides$two     2.6276     0.7769   3.382 0.000776 ***
rides$three   9.1777     0.8190  11.206  < 2e-16 ***
rides$four   13.0472     0.8069  16.170  < 2e-16 ***
rides$five   24.6406     0.8102  30.414  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 5.274 on 493 degrees of freedom
Multiple R-squared:  0.7283,	Adjusted R-squared:  0.7261 
F-statistic: 330.4 on 4 and 493 DF,  p-value: < 2.2e-16

> adjust_MLR_p <- lm(rides$Price~rides$Product.Level+rides$Distance +
rides$Temperature+rides$Product.Level)
> Anova(adjust_MLR_p, type = 3)
Anova Table (Type III tests)

Response: rides$Price
                    Sum Sq  Df  F value   Pr(>F)    
(Intercept)             54   1   3.2992 0.06992 .  
rides$Product.Level  35545   4 547.1666 < 2e-16 ***
rides$Distance        5645   1 347.5901 < 2e-16 ***
rides$Temperature       26   1   1.6054 0.20575    
Residuals             7974 491                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary(adjust_MLR_p)

Call:
lm(formula = rides$Price ~ rides$Product.Level + rides$Distance + 
    rides$Temperature + rides$Product.Level)

Residuals:
   Min     1Q Median     3Q    Max 
-9.765 -1.985 -0.587  1.198 48.822 

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)           2.05425    1.13096   1.816 0.069922 .  
rides$Product.Level1 -2.18315    0.59441  -3.673 0.000266 ***
rides$Product.Level5 21.76960    0.54023  40.297  < 2e-16 ***
rides$Product.Level4 10.74931    0.53657  20.033  < 2e-16 ***
rides$Product.Level3  6.38694    0.54745  11.667  < 2e-16 ***
rides$Distance        2.92236    0.15675  18.644  < 2e-16 ***
rides$Temperature     0.03417    0.02697   1.267 0.205747    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.03 on 491 degrees of freedom
Multiple R-squared:  0.842,	Adjusted R-squared:  0.8401 
F-statistic: 436.1 on 6 and 491 DF,  p-value: < 2.2e-16
> 
> emmeans(adjust_MLR_p, specs = "Product.Level" , contr = "pairwise")
$emmeans
 Product.Level emmean    SE  df lower.CL upper.CL
 2               9.75 0.361 491     9.04    10.46
 1               7.56 0.472 491     6.64     8.49
 5              31.52 0.402 491    30.73    32.31
```

```
 4              20.50 0.398 491    19.71    21.28
 3              16.13 0.412 491    15.32    16.94

Confidence level used: 0.95

$contrasts
 contrast estimate    SE  df t.ratio p.value
 2 - 1        2.18 0.594 491   3.673 0.0025
 2 - 5      -21.77 0.540 491 -40.297 <.0001
 2 - 4      -10.75 0.537 491 -20.033 <.0001
 2 - 3       -6.39 0.547 491 -11.667 <.0001
 1 - 5      -23.95 0.620 491 -38.616 <.0001
 1 - 4      -12.93 0.617 491 -20.967 <.0001
 1 - 3       -8.57 0.627 491 -13.676 <.0001
 5 - 4       11.02 0.566 491  19.475 <.0001
 5 - 3       15.38 0.574 491  26.776 <.0001
 4 - 3        4.36 0.573 491   7.618 <.0001

P value adjustment: tukey method for comparing a family of 5 estimates

> adjust_MLR_s_c <- lm(rides$Price~rides$Source+rides$Company+rides$Distance +
rides$Temperature+rides$Product.Level)
> Anova(adjust_MLR_s_c, type = 3)
Anova Table (Type III tests)

Response: rides$Price
                    Sum Sq  Df  F value Pr(>F)
(Intercept)             37   1   2.3233 0.1281
rides$Source           240  11   1.3564 0.1904
rides$Company           26   1   1.6141 0.2045
rides$Distance        3951   1 245.3896 <2e-16 ***
rides$Temperature       24   1   1.4853 0.2235
rides$Product.Level  34310   4 532.7014 <2e-16 ***
Residuals             7713 479
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary(adjust_MLR_s_c)

Call:
lm(formula = rides$Price ~ rides$Source + rides$Company + rides$Distance +
    rides$Temperature + rides$Product.Level)

Residuals:
    Min      1Q  Median      3Q     Max
-10.088  -2.027  -0.485   1.337  48.032

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                     2.06953    1.35774   1.524  0.12811
rides$SourceBoston University   1.41360    0.86260   1.639  0.10192
```

```
rides$SourceWest End          -0.67477    0.94178  -0.716  0.47404
rides$SourceNorth Station     -0.16548    0.83777  -0.198  0.84351
rides$SourceSouth Station      0.34466    0.92748   0.372  0.71035
rides$SourceNorth End          0.29140    0.89243   0.327  0.74417
rides$SourceFenway             0.91559    0.93718   0.977  0.32908
rides$SourceBeacon Hill       -1.14647    0.85195  -1.346  0.17903
rides$SourceFinancial District  1.03815    0.87213   1.190  0.23450
rides$SourceBack Bay           0.61166    0.90892   0.673  0.50130
rides$SourceHaymarket Square   0.60555    0.96790   0.626  0.53186
rides$SourceTheatre District   0.34411    0.89572   0.384  0.70102
rides$CompanyLyft             -0.47873    0.37682  -1.270  0.20454
rides$Distance                 2.83131    0.18074  15.665  < 2e-16 ***
rides$Temperature              0.03292    0.02702   1.219  0.22354
rides$Product.Level1          -1.89690    0.60462  -3.137  0.00181 **
rides$Product.Level5          22.11006    0.55914  39.543  < 2e-16 ***
rides$Product.Level4          10.92929    0.55684  19.627  < 2e-16 ***
rides$Product.Level3           6.60332    0.55906  11.812  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.013 on 479 degrees of freedom
Multiple R-squared:  0.8472,	Adjusted R-squared:  0.8414 
F-statistic: 147.5 on 18 and 479 DF,  p-value: < 2.2e-16

> conf_dist <- c(2.83131 - (1.6479*0.18074) , 2.8131 + (1.6479*0.18074)); 
conf_dist
[1] 2.533469 3.110941
> prod_level_means <- emmeans(adjust_MLR_s_c, specs = "Product.Level" , contr = 
"pairwise")
> prod_level_means
$emmeans
 Product.Level emmean    SE  df lower.CL upper.CL
 2               9.57 0.374 479     8.84    10.31
 1               7.68 0.474 479     6.74     8.61
 5              31.68 0.409 479    30.88    32.49
 4              20.50 0.402 479    19.71    21.29
 3              16.18 0.414 479    15.36    16.99

Results are averaged over the levels of: Source, Company 
Confidence level used: 0.95 

$contrasts
 contrast estimate    SE  df t.ratio p.value
 2 - 1        1.90 0.605 479   3.137 0.0155
 2 - 5      -22.11 0.559 479 -39.543 <.0001
 2 - 4      -10.93 0.557 479 -19.627 <.0001
 2 - 3       -6.60 0.559 479 -11.812 <.0001
 1 - 5      -24.01 0.623 479 -38.529 <.0001
 1 - 4      -12.83 0.622 479 -20.633 <.0001
 1 - 3       -8.50 0.627 479 -13.560 <.0001
```

```
 5 - 4       11.18 0.575 479  19.445 <.0001
 5 - 3       15.51 0.582 479  26.662 <.0001
 4 - 3        4.33 0.577 479   7.498 <.0001

Results are averaged over the levels of: Source, Company
P value adjustment: tukey method for comparing a family of 5 estimates
```

6. State Your Conclusion (no more than 100 words)

State the conclusion so that a none-statistician can understand.

Key take-aways from this statistical analysis:

- For a one-mile increase in the distance travelled, we are 95% confident the price will increase between $2.53 and $3.11.
- The product used has a significant effect on price. We are 95% confident that upgrading from UberPool to UberX or Lyft Shared to Lyft results in a $1.90 price increase; upgrading from UberX to UberXL or Lyft to LyftXL results in a $6.60 price increase; upgrading from UberXL to UberBlack or LyftXL to LuxBlack results in a $4.33 price increase; and upgrading from UberBlack to UberBlackSUV or LuxBlack to LuxBlackXL results in a $11.18 price increase.
- We cannot say with 95% confidence the outside temperature, the ridesharing company used, and the pickup location have a significant effect on the price of the ride.

Solution Submission

1. **Fill up this word file and upload it.**
2. **Upload your data set. This is the data set after cleaning (a small CSV file)**
3. **Upload your R file as a file with name "mini-project-solution.R"**

Grading will be done based on

1. **Originality of selected data set and data analysis approach**
2. **Data Preparation set and cleanup**
3. **General Correctness of data analysis**
4. **Quality of your R code and output results**
5. **Correct final conclusion**