SCN088MidCheckYourselfSU21.R

Student

2021-08-01

```
# SCN.088 - Mid Check Yourself
#
# Student name: Ezra Cohen
#
# Attribution statement: By submitting this homework on Blackboard, you
# attest that you completed this exam without assistance from any living
# person except the instructor.
#
# Type in your SUID in place of the zeros below and execute the code

suid <- 455254492


# Then select all of the code in the following block  and run it.
# Do not modify any of the code between the ======== separators
#============================================================
if (suid == 0) {cat("Please type in your SUID before running this code.")} else {
#If you didn't type in the suid it prints out statement that you need to do that first in order to continue, if y
ou have done it you can continue
cat(paste("Lyft/Uber Fare Comparison Study Number:",suid,"\n"))
# prints out Lyft/Uber Fare Comparison Study Number: 455254492
set.seed(suid)
#makes sure our random samples are perfectly replicatable
grp1id <- paste("Lyft",substr(as.hexmode(suid),1,2),sep="_"); grp2id <- paste("Uber",substr(as.hexmode(suid),1,2
),sep="_")
#grp1id is set to lyft_1b and grp2id is set to Uber_1b
ssize <- floor(runif(n=1,min=100,max=140))
#First it generates a random number between 100 and 140 then floor rounds that number down to an integer and then
 ssize is set to that value
driver <- 1:ssize
#driver is then set to all the  integers from one to ssize
base <-round(rnorm(n=ssize,mean=25,sd=5), digits=2)
#Create a vector with the length of ssize That has normal distribution a mean of 25 and standard deviation of 5 a
nd then rounds all of those numbers to have two digits and sets it to base
lyft <- round(rowMeans(cbind(base, runif(ssize,min=15,max=35))),digits=2)
#Takes the mean of each row in base after adding random values, the amount is ssize, with a minimum of 15 and a m
axmum of 35, then rounds it to have two digits afterwards and sets it to lyft
if ((suid %% 2)==1) {uber <- round(rowMeans(cbind(base, runif(ssize,min=10,max=40))),digits=2)
#If suid mod 2 (I really don't know what this does and the Google isn't helping) is one Then you continue to Take
 the mean of each row in base after adding random values, the amount is ssize, with a minimum of 10 and a maximum
 of 40, then rounds it to have two digits afterwards and sets it to uber
} else { uber <- round(rowMeans(cbind( base + rexp(n=ssize,rate=0.25), rnorm(n=ssize,mean=25,sd=5))),digits=2))
#otherwise you continue to Take the mean of each row in base after values at a rate of .25, the amount is ssize,
 and a vector with the length of ssize That has normal distribution a mean of 25 and standard deviation of 5 and
 then rounds all of those numbers to have two digits, then rounds it to have two digits afterwords and sets it to
 uber
testDF <- data.frame(driver,lyft,uber)
#Makes testDF a dataframe with driver, lyft, and Uber
names(testDF)[2]<-grp1id; names(testDF)[3]<-grp2id
#sets the name of the second column to grp1id and sets the name of the third column to grp2id
cat(paste("Sample size for this study:",ssize))
# prints Sample size for this study: and ssize
rm(base); rm(lyft); rm(uber); rm(ssize); rm(grp1id); rm(grp2id); rm(driver) }
```

```
## Lyft/Uber Fare Comparison Study Number: 455254492
## Sample size for this study: 133
```

```
#Remove the all the variables from memory that are put into parenthesis
#============================================================

str(testDF) # Your data set is called testDF; it has three variables
```

```
## 'data.frame':    133 obs. of  3 variables:
##  $ driver : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ lyft_1b: num  22.2 27 28.8 22.8 18.1 ...
##  $ Uber_1b: num  21.3 27.6 28.8 20.2 19.1 ...
```

```
summary(testDF)
```

```
##      driver         Lyft_1b          Uber_1b
##  Min.   :  1   Min.   :16.20   Min.   :16.87
##  1st Qu.: 34   1st Qu.:22.75   1st Qu.:24.22
##  Median : 67   Median :25.38   Median :26.68
##  Mean   : 67   Mean   :25.53   Mean   :27.27
##  3rd Qu.:100   3rd Qu.:28.28   3rd Qu.:30.12
##  Max.   :133   Max.   :35.98   Max.   :40.00
```

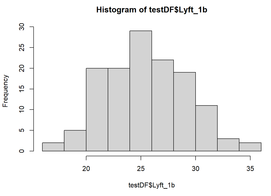
```
#library(ggplot2) I need to tag this out in order for this to knit
library(ggplot2)#I also need library this again for it to knit
#I added these two but I put them in this part because it's about the dataframe as a whole and not about completi
ng the tasks
# Add your code and comments below here. Make sure to run the code above first.

#1. Describe the fares provided by Lyft and Uber (separately) using descriptive statistics.
summary(testDF$Lyft_1b)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.20   22.75   25.38   25.53   28.28   35.98
```

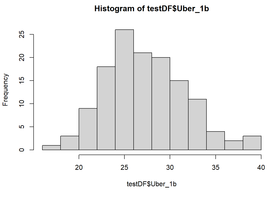
```
#For lyft the Prices range from  $16.20 to  $35.98  half of them being lower than $25.53 and  half being higher,
 further 75% are below around $28.28 and 75% are above $22.75
summary(testDF$Uber_1b)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.87   24.22   26.68   27.27   30.12   40.00
```

```
#For uber the Prices range from $16.87 to $40 half of them being lower than $26.68 and half being higher further,
 75% are below $30.12 and 75% are above $24.22
#2. Describe the shape of the distribution for Lyft fares and for Uber fares.
hist(testDF$Lyft_1b)
```




```
#The histogram shows normal distribution with a bell-shaped curve and a center of around 25, most of be prices th
at were charged were between $20 and $33, the rest are mostly outliers
hist(testDF$Uber_1b)
```




```
#The histogram shows normal distribution with a bell-shaped curve and a center of around 25, most of be prices th
at were charged were between $20 and $34, the rest are mostly outliers
#3. On average, which company is more expensive, Lyft or Uber? By how much?
mean(testDF$Lyft_1b)
```

```
## [1] 25.53211
```

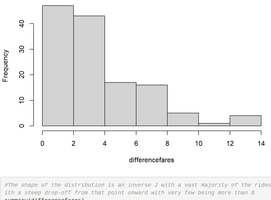
```
mean(testDF$Uber_1b)
```

```
## [1] 27.2694
```

```
abs(mean(testDF$Uber_1b)-mean(testDF$Lyft_1b))
```

```
## [1] 1.737293
```

```
#on avarage Uber is more expensive by about $1.73
#4. Create a new variable that represents the difference in fares for each trip. Describe the shape of the distri
bution for the new variable.
differencefares<-abs(testDF$Lyft_1b-testDF$Uber_1b)
hist(differencefares)
```

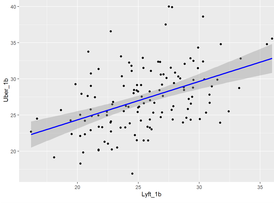



```
#The shape of the distribution is an inverse J with a vast majority of the rides being within $4 of one another w
ith a steep drop-off from that point onward with very few being more than 8
summary(differencefares)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.060   1.610   2.950   3.662   4.820  13.820
```

```
#This was not part of the task but I put this line of code in to prove a point, the third quartile meaning 75% is
 below that is only around $5 of difference
#5. Does an X-Y scatterplot of the Lyft and Uber fares show an obvious pattern/shape?
rides_scatterplot<-ggplot(testDF,aes(x=Lyft_1b,y=Uber_1b)) + geom_point() + geom_smooth(method = "lm",color = "bl
ue")
rides_scatterplot
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
#Yes, it has an obvious upward trend (line is just there to help with the visualization)

#Looking at this data it is almost undeniable that they are coordinating their prices. Looking at the summary for
 both of them it is quite easy to tell that the range and placement of median and quartiles are very similar for b
oth companies, and looking at the the histograms shows an almost identical distribution of prices with the majority o
f prices being around the same area. When looking at the differences of their average prices there is only a $1.7
3 difference between them, and when looking at the differences of their prices on a driver by driver basis we can
 see that a vast majority of the time the price is under $5 difference between the two. Finally when we look at th
e scatterplot there is an obvious connection between the two with an upward trend of the graph showing that as th
e price for one goes up the price for the other also does.

lmrides<-lm(formula = Lyft_1b~Uber_1b,testDF)
summary(lmrides)
```

```
##
## Call:
## lm(formula = Lyft_1b ~ Uber_1b, data = testDF)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.7433 -2.3618  0.0098  2.2304  7.5771
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 14.84922    1.84261   8.059 4.19e-13 ***
## Uber_1b      0.39175    0.06669   5.874 3.32e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.416 on 131 degrees of freedom
## Multiple R-squared:  0.2085,    Adjusted R-squared:  0.2024
## F-statistic:  34.5 on 1 and 131 DF,  p-value: 3.315e-08
```

```
#this was just to prove a point and wasn't specified in instructions, looking at the p value you can see 3 '*'s in
dicating extreme significance also the standard error is very low, the r squared value is low but there is no set
 r squared value that is good or bad
```