Introduction to Data Analytics 1

Pradeep Paladugula

2020-01-26

Part 1: Variables, Hypothesis, Designs

Title: Offshore outsourcing: Its advantages, disadvantages, and effect on the American economy

Abstract: The United States has trained some of the world's best computer programmers and technology experts. Despite all of this training, many businesses do not have a full understanding of information technology. As the importance of technology in the business world grows, many companies are wasting money on extensive technology projects. When problems arise, they expect that further investment will solve these issues. To prevent such problems, many companies have begun to outsource these functions in an effort to reduce costs and improve performance. The majority of these outsourced information technology and call center jobs are going to low-wage countries, such as India and China where English-speaking college graduates are being hired at substantially lower wages. The purpose of this study is to evaluate the positive and negative aspects of offshore outsourcing with a focus on the outsourcing markets in India and China, arguably the two most popular destinations for outsourcers. The cost savings associated with offshore outsourcing will be evaluated in relation to the security risks and other weakness of offshore outsourcing. In addition, an analysis of the number of jobs sent overseas versus the number of jobs created in the United States will be used to assess the effects that outsourcing is having on the American economy and job market. Finally, the value of jobs lost from the American economy will be compared to the value of jobs created. The goal of these analyses is to create a clear picture of this increasingly popular business strategy.

Answer the following questions about the abstract above:

1) What is a potential hypothesis of the researchers?

   solution: The economy fluctuation and impact that the United States of America is seeing due to the company business startegies.

2) What is one of the independent variables?

   a. What type of variable is the independent variable?

   Solution: A clear Business strategy of creating better job market. (ex: clear measurements, statergies, Pridiction)

3) What is one of the dependent variables?

a. What type of variable is the dependent variable?

Solution: the terms that leads to decrease/increase of economy of the USA, excess ammount put in to issues etc. (ex: Cost)

4) What might cause some measurement error in this experiment?

ex: When problems arise, they expect that further investment will solve these issues.

The above example would be one of the cause in this expeiment. Basically, wrong strategies,

5) What type of research design is the experiment?

a. Why?

Solution: it is a correlational research design. For example “number of jobs sent overseas versus the number of jobs created in the United States” and the cost calcualtions included in the “value of jobs lost from the American economy and jobs created”

6) How might you measure the reliability of your dependent variable?

Solution: No major fluctutation of the value of cost to the workforce/

7) Is this study ecologically valid?

solution: Yes, ecologically valid study.

8) Can this study claim cause and effect?

a. Why/why not?

9) What type of data collection did the researchers use (please note that #5 is a different question)?

Solution: It is a quantitative data.

Part 2: Use the assessment scores dataset (03_lab.csv) to answer these questions.

The provided dataset includes the following information created to match the abstract:

- Jobs: the percent of outsourced jobs for a call center.
- Cost: one calculation of the cost savings for the business.
- Cost2: a separate way to calculate cost savings for the business.
- ID: an ID number for each business.
- Where: where the jobs were outsourced to.

Calculate the following information:

1) Create a frequency table of the percent of outsourced jobs.

```
ojdata <- read.csv("03_data.csv")
ojfreq<- table(ojdata$jobs)
ojfreq
```

```
##
## 58.9029958133744 62.4527972275747 65.2031455854069 65.3856420852478
##                1                1                1                1
##  65.560820259617 65.6596393228534 65.6824465729458 66.5738121008255
##                1                1                1                1
## 66.6587244171856 67.5159168729686 67.5535434644159  68.477282691581
##                1                1                1                1
## 68.5302691640281 68.9389422299438 69.4855315177517 69.4995113394852
##                1                1                1                1
## 69.5876420782106 69.7636899789475  69.869913596915 70.0841157680136
##                1                1                1                1
## 70.3645865237917  70.463895766194 70.5688243834846  70.678602826769
##                1                1                1                1
## 70.6882491575104 70.7125731531405 70.9976021629162 71.0967687237389
##                1                1                1                1
## 71.1621711118727 71.3731018397341 71.5659120003523 71.9431944415078
##                1                1                1                1
## 72.0642509682741 72.0694050964002 72.0903185161976 72.1228735944534
##                1                1                1                1
## 72.1387784759467 72.3258764840921 72.3417827994722  72.579733916728
##                1                1                1                1
## 72.6464163450913 72.7895687833708 72.9647983985062  73.131257887151
##                1                1                1                1
## 73.4241735612223 73.4633069220981 73.5006578554975 73.6034891966211
##                1                1                1                1
## 73.7417744430295 73.7584348454866  73.759178806666 73.7934131821085
##                1                1                1                1
## 73.9082262567804 74.3122601396811 74.3348678662277 74.3461655545212
##                1                1                1                1
## 74.3508161573504 74.3735392623011 74.6113829910448 74.6338917714721
##                1                1                1                1
## 74.7077702529417 74.7372735583124   74.90975702297 74.9191894888098
##                1                1                1                1
## 74.9363839937857 75.1137444109981 75.2311604632885 75.4027435336861
##                1                1                1                1
## 75.4106145123374 75.5022139952449  75.542875599571 75.5581791962435
##                1                1                1                1
## 75.6563785438999 75.6653772800426 75.7176972621424 75.8354707813663
##                1                1                1                1
## 75.9278802156521 75.9290234219225  75.963059104548 75.9930823641479
##                1                1                1                1
## 76.0022843940167 76.0759215588339  76.155536731639 76.3229713859328
##                1                1                1                1
## 76.5113007938448 76.5620733658871 76.9071568218684 76.9263849925586
##                1                1                1                1
```

```
## 76.9966013794146 77.0261062551164 77.1251098232301 77.1577538931102
##                1                 1                 1                 1
## 77.1789212656822 77.2062451805888 77.2096173579267 77.3394573956013
##                1                 1                 1                 1
## 77.3945836371507 77.5460751827025 77.6688493342386 77.7443978183338
##                1                 1                 1                 1
## 78.2576392292226 78.3428081165541 78.3700958425642 78.8614835625907
##                1                 1                 1                 1
## 78.9526216447704 79.0077285062477 79.0742375195916 79.0954954612535
##                1                 1                 1                 1
## 79.1800558491871 79.5094410759405 80.0769304012729 80.1063067143937
##                1                 1                 1                 1
## 80.1706512463691 80.3890056862686 80.5060038849112  80.539540938016
##                1                 1                 1                 1
## 80.7404155801415 80.8594006273771 81.0799959864645 81.1295168232325
##                1                 1                 1                 1
## 81.3364498652159 81.3591173284419 81.4361244908874 81.4666569335935
##                1                 1                 1                 1
## 81.4844450774054 81.5960211443116  81.600741074996 81.6527388068295
##                1                 1                 1                 1
## 81.7545507524997 81.8302480100204 81.8907931373699 82.0770154803783
##                1                 1                 1                 1
## 82.1063739924447 82.1400995321577 82.2239619074893 82.5010500219936
##                1                 1                 1                 1
## 82.6009626420951 82.8001636767993 82.8445514055607 83.1786939319355
##                1                 1                 1                 1
##  83.300984028281 83.3297676088112 83.3348620078833 83.3382467376123
##                1                 1                 1                 1
## 83.3892124023058 83.6840598941215 83.9782747052116 84.2169095911184
##                1                 1                 1                 1
## 84.2663627727814 84.4961825574701 84.6747992907158 84.9644667609072
##                1                 1                 1                 1
## 85.1285754059075  85.219616160683 85.2648824867823 85.3004124945786
##                1                 1                 1                 1
## 85.4159498924099 85.4718580694339 85.8815090177156 86.2121883053476
##                1                 1                 1                 1
## 86.3547221533568 86.4458865107219  86.476956762778 86.4839106857072
##                1                 1                 1                 1
## 86.6106904154958 86.6811127564172 86.9237782426119 87.1143938747111
##                1                 1                 1                 1
## 87.2320745044108 87.4661697016087 87.6795250593823 87.7860744868506
##                1                 1                 1                 1
## 87.7896215565397 88.0895224022913 88.2061126971443 88.2380396184371
##                1                 1                 1                 1
##  88.287020041318 88.4420327761028 88.7428349609785 88.8823440211113
##                1                 1                 1                 1
## 88.9141899539348 89.2461764986549 89.2758334047947 89.2795991155881
##                1                 1                 1                 1
## 89.5293813873826 89.5638757382627 89.8202947445535 89.9644951795293
##                1                 1                 1                 1
```

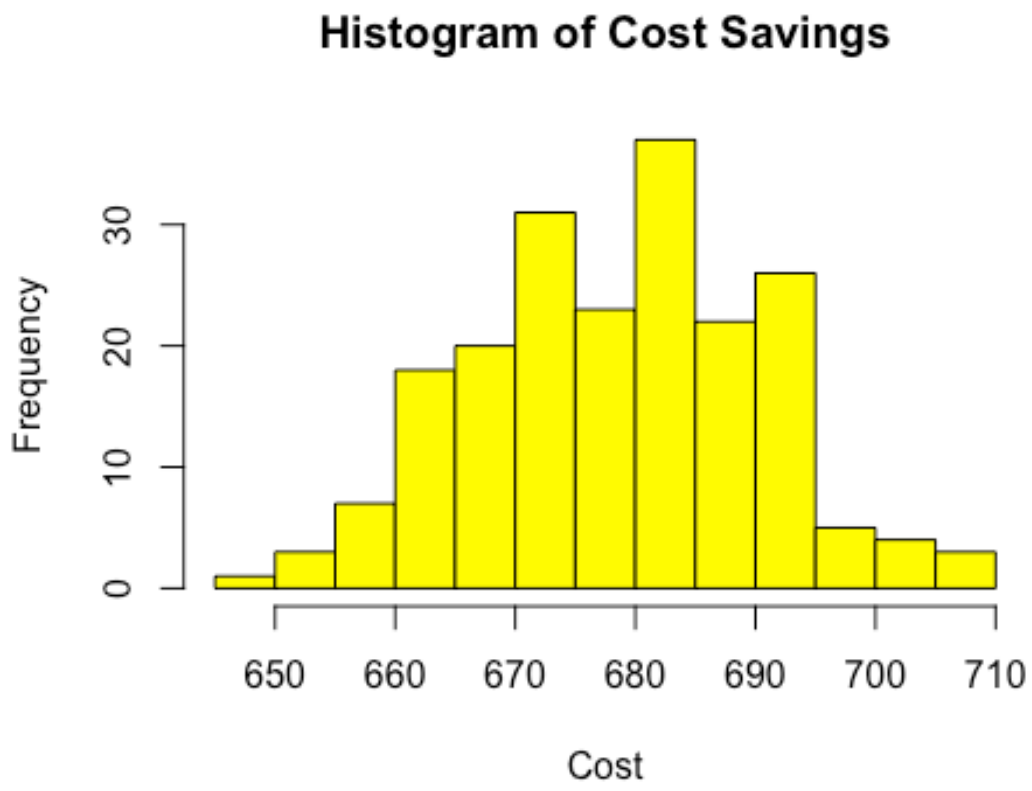
```
## 91.0175359337366  91.447253650563  92.569285969437  92.933285347916
##                1                1                1                1
## 93.2687684124082 93.7386682283128 93.8133956375988 93.9583158096184
##                1                1                1                1
## 94.6707613799703 97.2715399751448 97.3785749059487 97.8409722396027
##                1                1                1                1
```

2) Create histograms of the two types of cost savings. You will want to add the breaks argument to the hist() function. This argument adds more bars to the histogram, which makes it easier to answer the following questions:

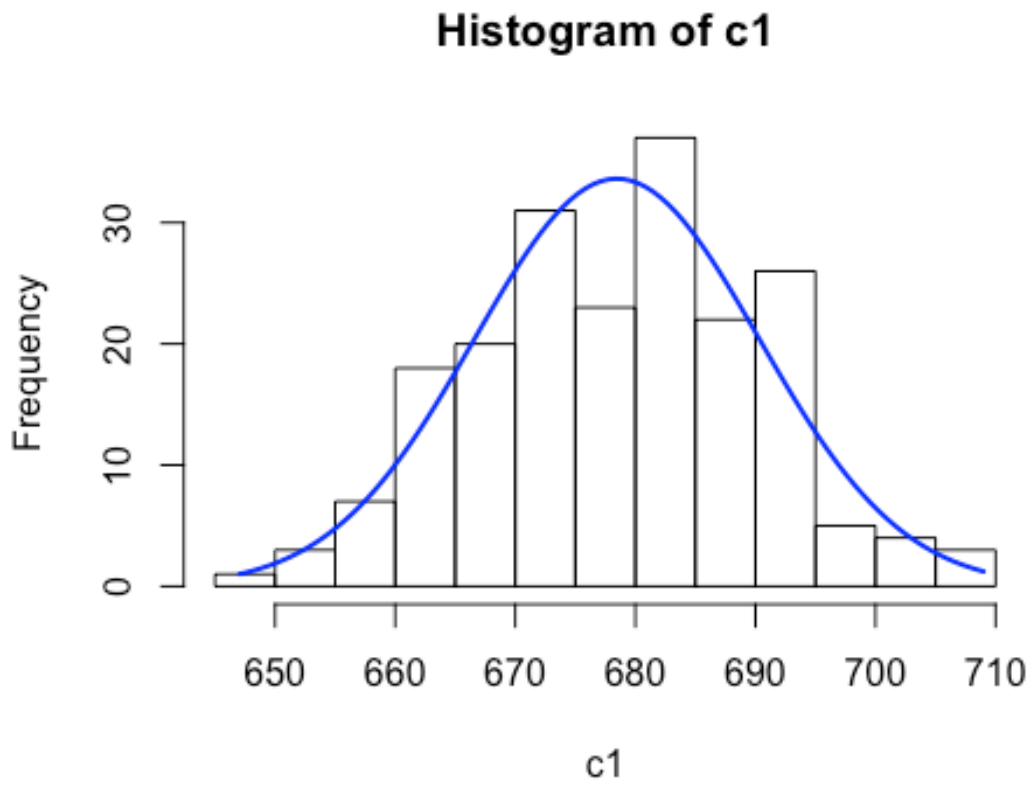
```
hist(dataset$column, breaks = 15)
```

15 is a great number to pick, but it can be any number. For this assignment, try 15 to see a medium number of bars.

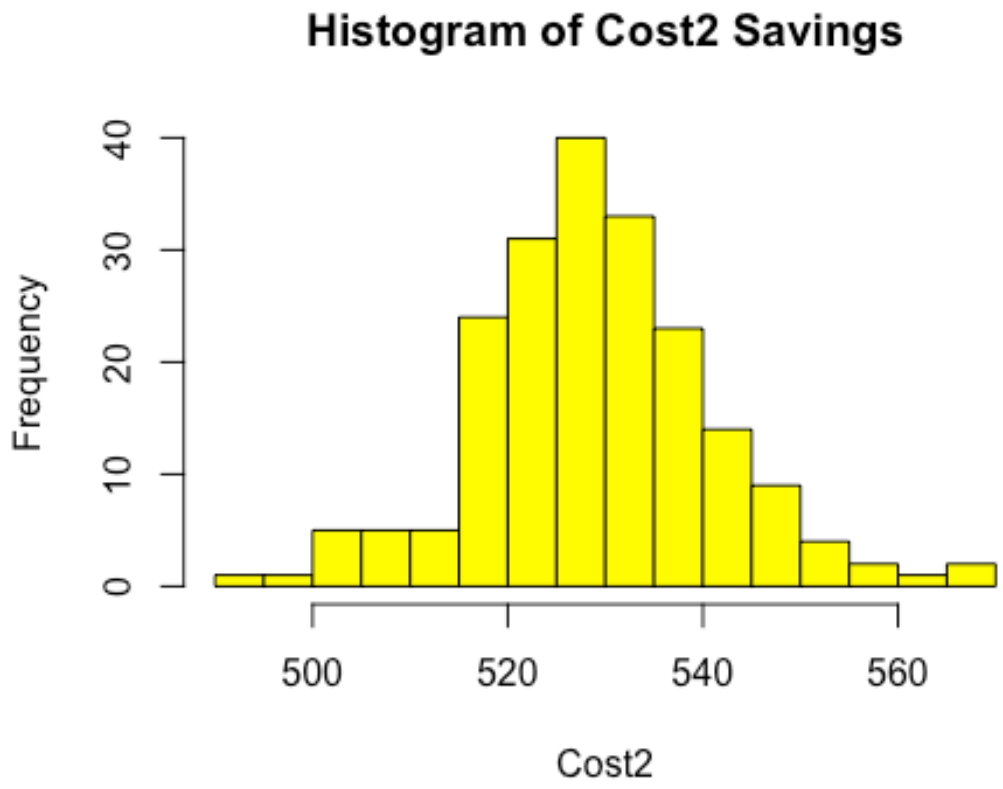
```
normlineplot = function(x, histgraph) {
     m <- mean(x)
     sd<- sqrt(var(x))
     xaxis<- seq(min(c1), max(c1), length.out = 100)
     yaxis<- dnorm( xaxis, mean = m, sd = sd)
     yaxis<- yaxis * diff(histgraph$mids[1:2]) * length(c1)
     plot(histgraph)
     lines(xaxis, yaxis, col = "blue", lwd = 2)
}

c1<- ojdata$cost
hc1<- hist(c1, breaks = 15, main = ("Histogram of Cost Savings"), xlab =
"Cost", col = "yellow")
```



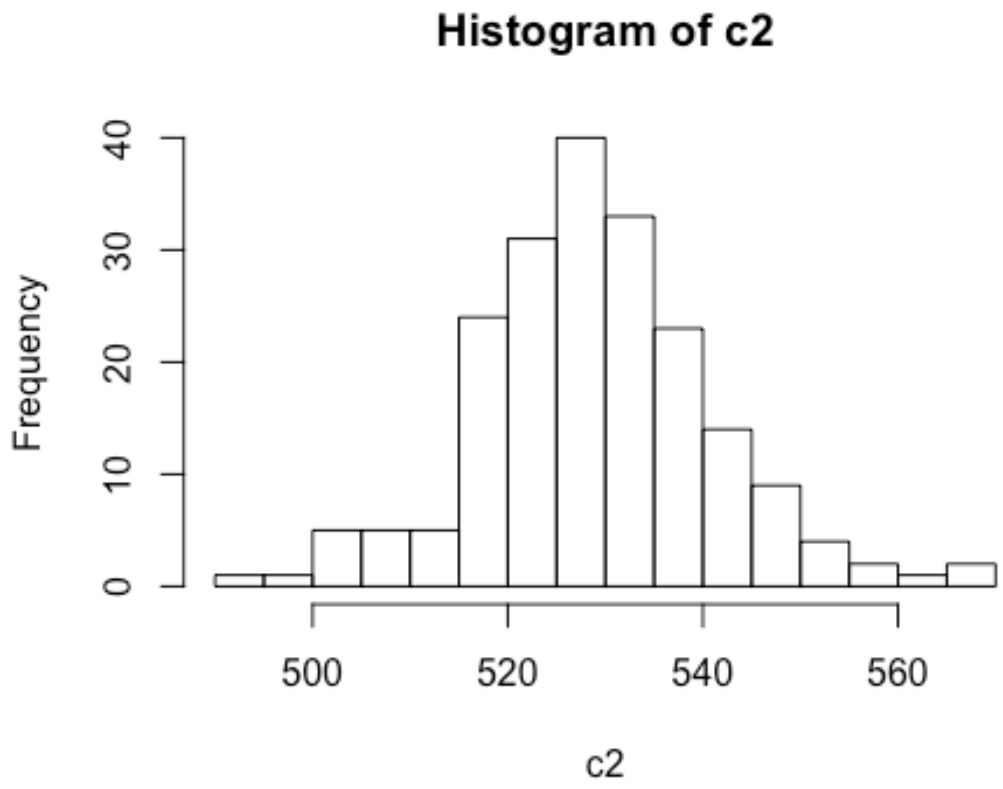

```
normlineplot(c1, hc1)
```




```
c2<- ojdata$cost2
hc2<- hist(c2, breaks = 15, main = ("Histogram of Cost2 Savings"), xlab =
"Cost2", col = "yellow")
```



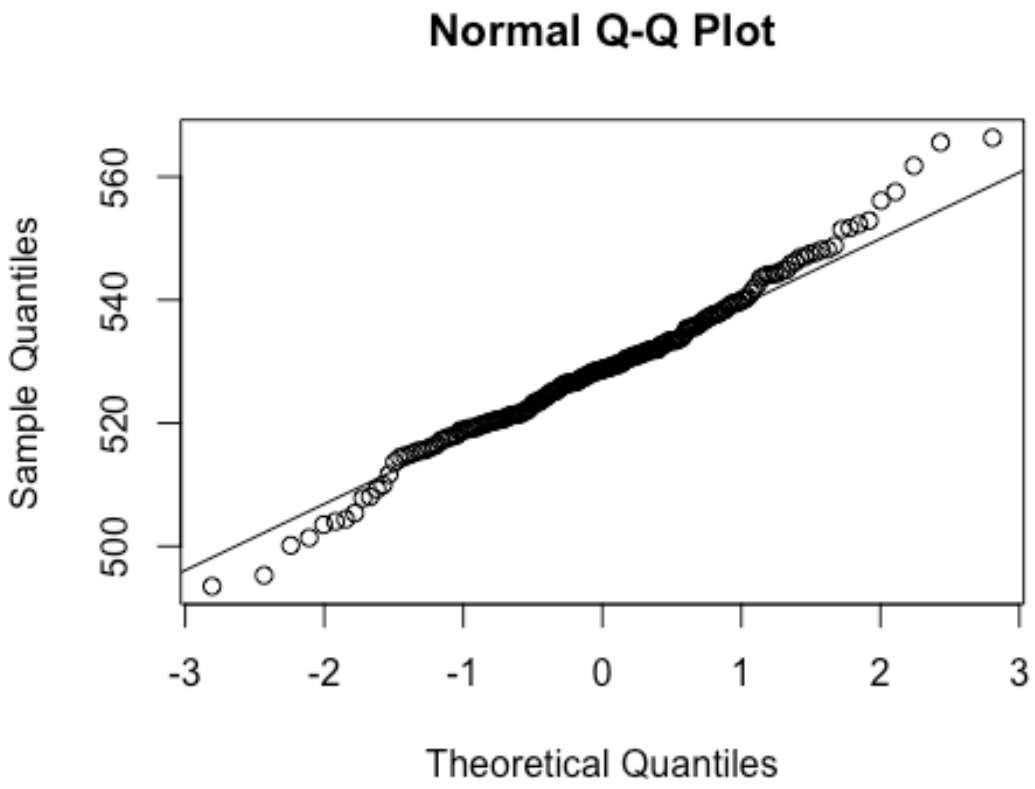

```
normlineplot(c2, hc2) #Failed to produce the nomral line histogram grap.
Still working on it.
```



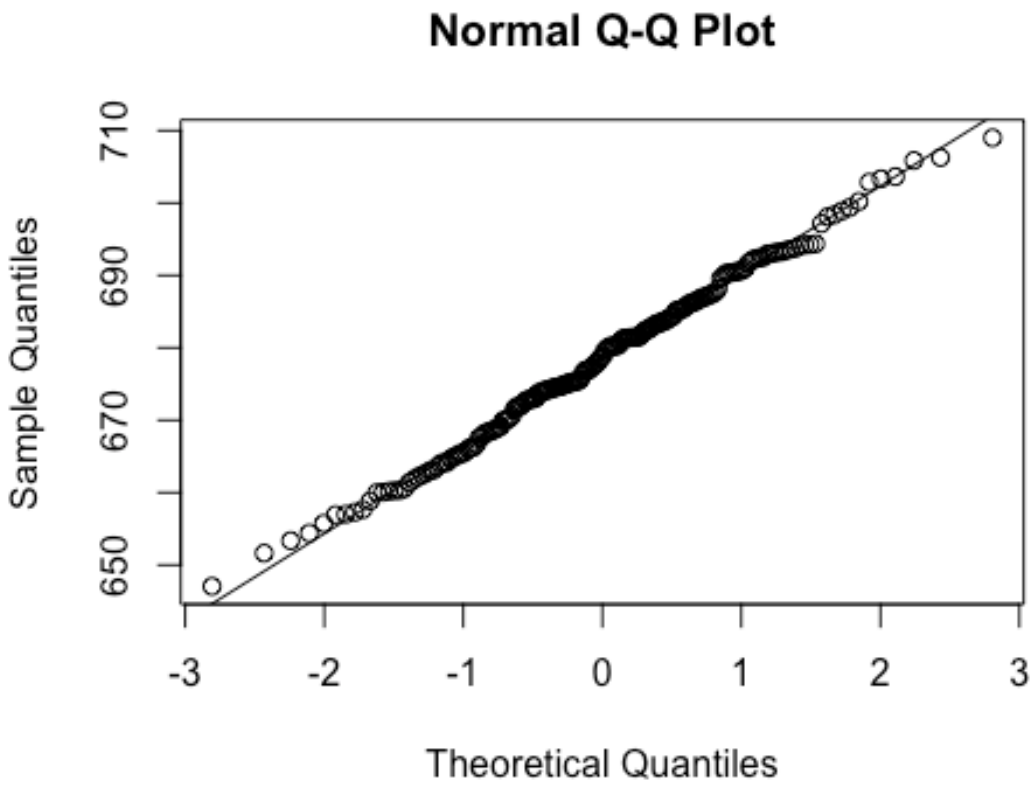
3) Examine these histograms to answer the following questions:

   a. Which cost savings appears the most normal? #Cost savings apprears to be noral

```
qqnorm(c2);qqline(c2)
```




```
qqnorm(c1);qqline(c1)
```




```
#reference: https://stackoverflow.com/questions/28734985/r-qqplot-
argument-y-is-missing-error/40624387

#Solution: As most of the Cost2 data thickly falls on the normality line,
and also from the prvious question pictures we can conclude from seeing
he bell curve that the Cost2 data is said to be more normal.
```

b. Which cost savings data is multimodal?

```
install.packages("LaplacesDemon")

## Error in contrib.url(repos, "source"): trying to use CRAN without
setting a mirror

is.multimodal(c1)

## Error in is.multimodal(c1): could not find function "is.multimodal"

is.multimodal(c2)

## Error in is.multimodal(c2): could not find function "is.multimodal"
```

```
    #multimodal: whether data has the multiple modes

    #NO COST DATA ARE MULTIMODAL
```

c. Which cost savings data looks the most skewed (and in which direction positive or negative)?

```
    library(e1071)
    skewness(c1) #ojdata$cost is most skewed as it is a negative skewed
    (towards left)

    ## [1] -0.001619656

    skewness(c2) #ojdata$cost positive skewed (towards right)

    ## [1] 0.1573273
```

d. Which cost savings data looks the most kurtotic?

4) Calculate the z-scores for each cost savings, so they are all on the same scale.

```
#reference: https://stats.seandolinar.com/calculating-z-scores-with-r/
#reference: https://www.r-bloggers.com/r-tutorial-series-centering-variables-
and-generating-z-scores-with-the-scale-function/
# the means and standard deviations of these measurements should all be
completely different. In order to get the distributions standardized, the
measurements can be changed into z-scores.
#Z-scores are a stand-in for the actual measurement, and they represent the
distance of a value from the mean measured in standard deviations. So a z-
score of 2.0 means the measurement is 2 standard deviations away from the
mean.

zscoredata = function(x, value) {
    m<- mean(x)
    varx= function(x){sqrt(var(x)*(length(x)-1)/length(x))}
    zs = function(x, y, z){x-y/z}
    zsx<- zs(value, m, varx(x))
    return(zsx)
}

length(c2)

## [1] 200

zscoredata(c2, 200)  #Still working onfiguring this out.

## [1] 156.1874

zsc1<- scale(c1, center = TRUE, scale = TRUE)
zsc2<- scale(c2, center = TRUE, scale = TRUE)
```

6) How many of the cost saving scores were more extreme than 95% of the data (i.e., number of z-scores at a $p < .05$)?

```
summary(zsc1<.05)

##      V1
##  Mode :logical
##  FALSE:99
##  TRUE :101

summary(zsc2<.05)

##      V1
##  Mode :logical
##  FALSE:90
##  TRUE :110

a.  Cost Savings 1: 101

c.  Cost Savings 2: 110
```

7) Which business had:

   a. the highest cost savings?

```
which.max(c1)

## [1] 100

which.max(c2)

## [1] 97

# Highest Cost Saving 1: S100
# Highest Cost Saving 2: S97
```

Highest Cost Saving 1: S100 Highest Cost Saving 2: S97

   b. the the lowest cost savings?

```
which.min(c1)

## [1] 190

which.min(c2)

## [1] 92
```

lowest Cost Saving 1: S190 lowest Cost Saving 2: S92

   c. Use both cost savings columns and find the ID number of the business with the lowest and highest z-score.

```
c1[which.max(scale(c1 ,center=TRUE,scale=TRUE))] #: S100 of cost1 is the
highest Z-Score

## [1] 708.9968
```

```r
c2[which.min(scale(c2 ,center=TRUE,scale=TRUE))] #: S92 of Cost2 is the
lowest Z-Score
```

```
## [1] 493.5102
```