Foundations_of_Analytics_Assignment_1

Bernardo Arambula

2023-08-18

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readxl)
library(readr)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ lubridate 1.9.2     ✔ tibble    3.2.1
## ✔ purrr     1.0.1     ✔ tidyr     1.3.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts
to become errors
```

Due Date: Sunday, August 20, 2023 11:59 PM

Instructions

1. There are 10 questions. Some have multiple parts.
2. All questions should be completed using R.
3. Please complete each question for full or partial credit.
4. Submit an RMD file knitted as HTML. The knitted file should show the code and the result (output) below it. No other form of submission will be accepted. You don't need to submit your RMD files.

5. Like I mentioned in the class, interpretations of your results are the key. Interpretations are at the heart of statistics. Results are meaningless with- out interpretations and insights.
6. Please comment your R code like I did on my solved examples. The reason is that we want to award you partial credit commensurate with your attempt, and we will be able to understand your logic and responses better if the sections of your work are clearly commented.
7. If you have any questions, then please post them under Discussions Homework 1 on Canvas so that the responses can benefit everyone.

Question 1

The monthly closing values of the Dow Jones Industrial Average (DJIA) for the period beginning in January 1950 are given in the CSV file Dow. According to Wikipedia, the Dow Jones Industrial Average, also referred to as the Industrial Average, the Dow Jones, the Dow 30, or simply the Dow, is one of several stock market indices created by Charles Dow. The average is named after Dow and one of his business associates, statistician Edward Jones. It is an index that shows how 30 large, publicly owned companies based in the United States have traded during a standard trading session in the stock market. It is the second oldest U.S. market index after the Dow Jones Transportation Average, which Dow also created.

The Industrial portion of the name is largely historical, as many of the modern 30 components have little or nothing to do with traditional heavy industry. The average is price-weighted, and to compensate for the effects of stock splits and other adjustments, it is currently a scaled average. The value of the Dow is not the actual average of the prices of its component stocks, but rather the sum of the component prices divided by a divisor, which changes whenever one of the component stocks has a stock split or stock dividend, so as to generate a consistent value for the index.

Along with the NASDAQ Composite, the S&P 500 Index, and the Russell 2000 Index, the Dow is among the most closely watched benchmark indices for track- ing stock market activity. Although Dow compiled the index to gauge the performance of the industrial sector within the U.S. economy, the index's per- formance continues to be influenced not only by corporate and economic reports, but also by domestic and foreign political events such as war and terrorism, as well as by natural disasters that could potentially lead to economic harm.

```
Dow <- read_csv("/Users/bernardoarambula/Documents/MSBA/BAX400 foundations/Homework1/Do
w.csv")
```

```
## Rows: 860 Columns: 2
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): Month
## num (1): Closing Values
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

a. Compute the average Dow return over the period given on the dataset.

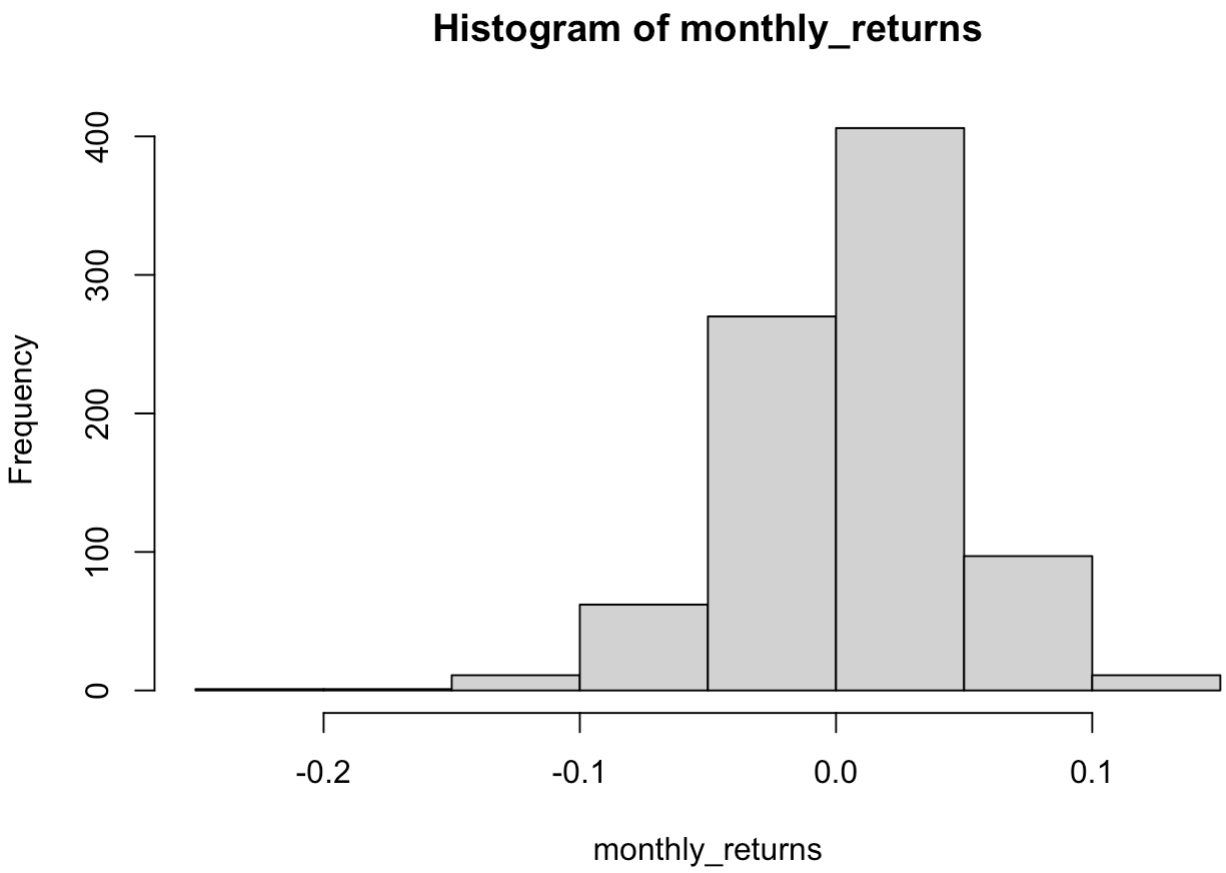
```
# Find monthly returns
monthly_returns <- (Dow$`Closing Values` - lag(Dow$`Closing Values`)) / lag(Dow$`Closing
Values`)

# Find average return, excluding NA values
average_return <- mean(monthly_returns, na.rm = TRUE)

# Print average return
average_return
```

```
## [1] 0.006890132
```

This output indicates that on average, the Dow Jones index has shown a return of approximately 0.69% per month over the specified period in this dataset (January of 1950 - August of 2021).

b. Plot a histogram and intuitively comment on whether or not the returns are normally distributed.

```
hist(monthly_returns)
```



The returns appear to have a slight left skew

c. Do the returns adhere to the Empirical Rule? Verify this using the Empirical Rule like the code shown on the solved examples.

```
Total_Sample_Size <- length(Dow$Month)
sd_monthly_returns <- sd(monthly_returns, na.rm = TRUE)

## One std dev bounds
One_SD_Range_Lower <- average_return - 1*sd_monthly_returns
One_SD_Range_Upper <- average_return + 1*sd_monthly_returns

## Proportion of observations within 1 SD (should be around 68%)
Num_Within_One_SD <- length(which(monthly_returns >= One_SD_Range_Lower & monthly_return
s <= One_SD_Range_Upper))
Percent_Within_One_SD <- 100*(Num_Within_One_SD/Total_Sample_Size)
Percent_Within_One_SD
```

```
## [1] 72.2093
```

```
## Two std dev bounds
Two_SD_Range_Lower <- average_return - 2*sd_monthly_returns
Two_SD_Range_Upper <- average_return + 2*sd_monthly_returns

## Proportion of observations within 2 SD (should be around 95%)
Num_Within_Two_SD <- length(which(monthly_returns >= Two_SD_Range_Lower & monthly_return
s <= Two_SD_Range_Upper))
Percent_Within_Two_SD <- 100*(Num_Within_Two_SD/Total_Sample_Size)
Percent_Within_Two_SD
```

```
## [1] 95.23256
```

```
## Three std dev bounds
Three_SD_Range_Lower <- average_return - 3*sd_monthly_returns
Three_SD_Range_Upper <- average_return + 3*sd_monthly_returns

## Proportion of observations within 3 SD (should be around 99.7%)
Num_Within_Three_SD <- length(which(monthly_returns >= Three_SD_Range_Lower & monthly_re
turns <= Three_SD_Range_Upper))
Percent_Within_Three_SD <- 100*(Num_Within_Three_SD/Total_Sample_Size)
Percent_Within_Three_SD
```

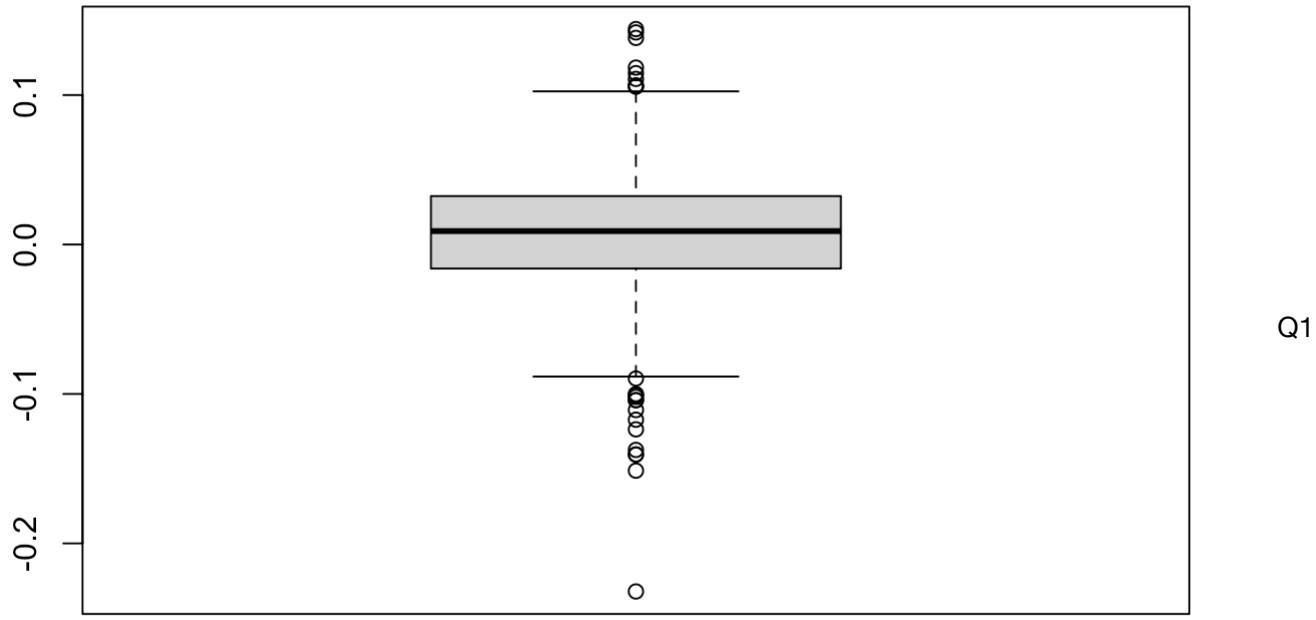
```
## [1] 98.72093
```

The calculations demonstrate that a significant majority of monthly returns fall within the expected ranges defined by the empirical rule, supporting the assumption of approximate normality in the Dow dataset and providing insights into the distribution of Dow Jones index returns over the analyzed period.

The returns do seem to adhere by the empirical rule

d. Plot a boxplot of returns and the five-number summary. Interpret the first, second, and third quartiles.

```
boxplot(monthly_returns)
```



represents the value below which 25% of the data falls. Q2 divides the dataset into two halves; 50% of the data lies below it, and 50% lies above it. Q3 is the value below which 75% of the data falls.

Q1 contains 25% of the monthly returns, Q2 contains 50% of the monthly returns and Q3 contains 75% of all monthly returns in our data

e. Identify the mild and extreme outlier returns using the IQR. Include whether each monthly return is a mild, an extreme outlier, or not an outlier. Sort the results displaying the outliers first.

```
# finding quartiles
Q1 <- quantile(monthly_returns, 0.25, na.rm = TRUE)
Q3 <- quantile(monthly_returns, 0.75, na.rm = TRUE)

# finding IQR
IQR <- Q3 - Q1

# outlier bounds
mild_lower <- Q1 - 1.5 * IQR
mild_upper <- Q3 + 1.5 * IQR
extreme_lower <- Q1 - 3 * IQR
extreme_upper <- Q3 + 3 * IQR

# Finding outliers
outliers <- ifelse(monthly_returns < mild_lower | monthly_returns > mild_upper, "Mild Ou
tlier", ifelse(monthly_returns < extreme_lower | monthly_returns > extreme_upper, "Extre
me Outlier", "Not an Outlier"))

# Creating a data frame to combine results
results <- data.frame(return = monthly_returns, outlier = outliers)

# Sorting the results with outliers first
results <- results[order(results$outlier, decreasing = FALSE), ]

head(results)
```

```
##         return      outlier
## 287 -0.1404274 Mild Outlier
## 296 -0.1041020 Mild Outlier
## 297 -0.1042029 Mild Outlier
## 301  0.1419090 Mild Outlier
## 313  0.1441442 Mild Outlier
## 340  0.1055773 Mild Outlier
```

Question 2

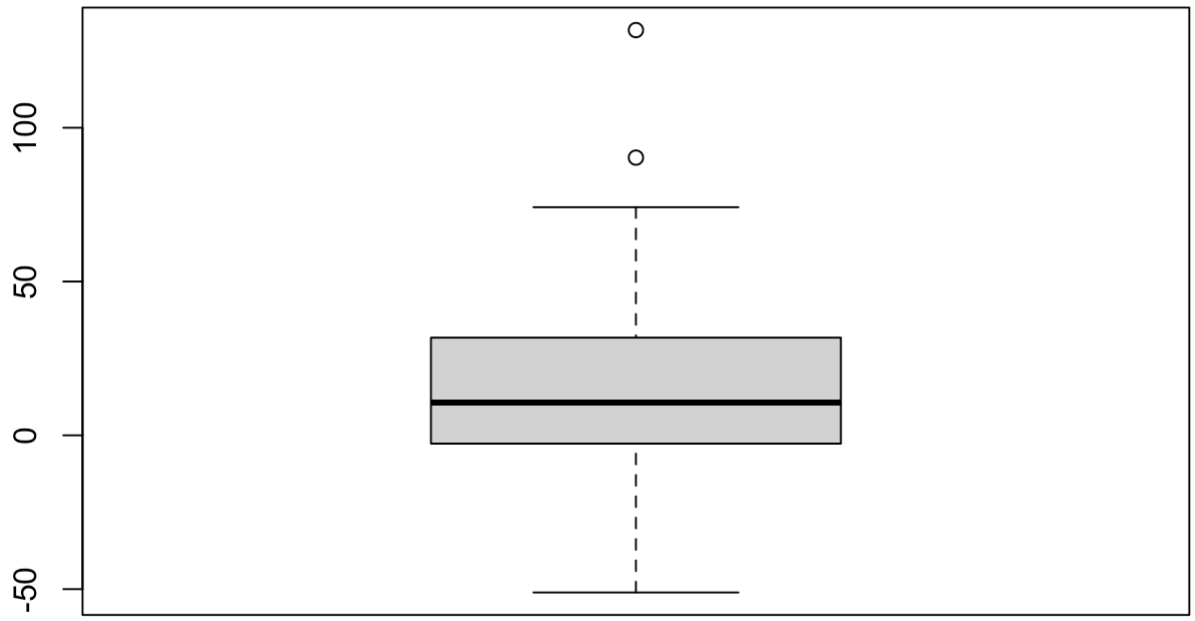
The CSV data set MutualFunds shows the annual returns (in %) for Mutual Fund 1 and Mutual Fund 2 over the past 37 years.

```
MutualFunds <- read_csv("/Users/bernardoarambula/Documents/MSBA/BAX400 foundations/Homew
ork1/MutualFunds.csv")
```

```
## Rows: 37 Columns: 3
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (3): Year, MF1, MF2
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

a. Construct a boxplot for Mutual Fund 1. Does the boxplot suggest that outliers exist?

```
boxplot(MutualFunds$MF1)
```



The boxplot sugests there are two outliers in our data

b. Use z-scores to determine if there are any outliers for Mutual Fund 1. Are your results consistent with part a? Explain why or why not.

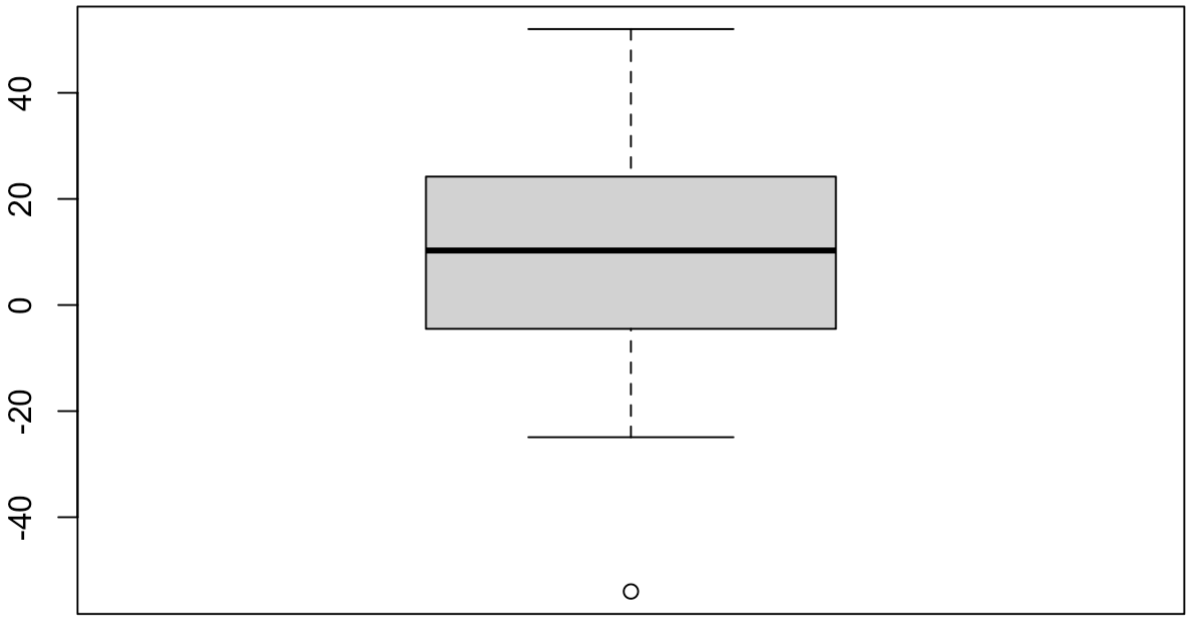
```
mean_mf1 <- mean(MutualFunds$MF1)
sd1 <- sd(MutualFunds$MF1)
z_score_mf1 <- (MutualFunds$MF1 - mean_mf1) / sd1
outliers_mf1 <- abs(z_score_mf1) > 3
sum(outliers_mf1)
```

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

My results are not consistent with part a, the results imply there is only one outlier and the boxlot implies there are two outliers. This could be because a boxplot will determine outliers based on quartile ranges and in part b we used z scores of greater than three to determine outliers

c. Construct a boxplot for Mutual Fund 2. Does the boxplot suggest that outliers exist?

```
boxplot(MutualFunds$MF2)
```



The boxplot sugests there is one outlier in our data

d. Use z-scores to determine if there are any outliers for Mutual Fund 2. Are your results consistent with part c? Explain why or why not.

```
mean_mf2 <- mean(MutualFunds$MF2)
sd2 <- sd(MutualFunds$MF2)
z_score_mf2 <- (MutualFunds$MF2 - mean_mf2) / sd2
outliers_mf2 <- abs(z_score_mf2) > 3
sum(outliers_mf2)
```

```
## [1] 0
```

Again, the results are not consistent with the previous part. My boxplot implies there is one outlier and using z scores we see no outliers this again can be explained by how we are determining outliers in both methods

Question 3

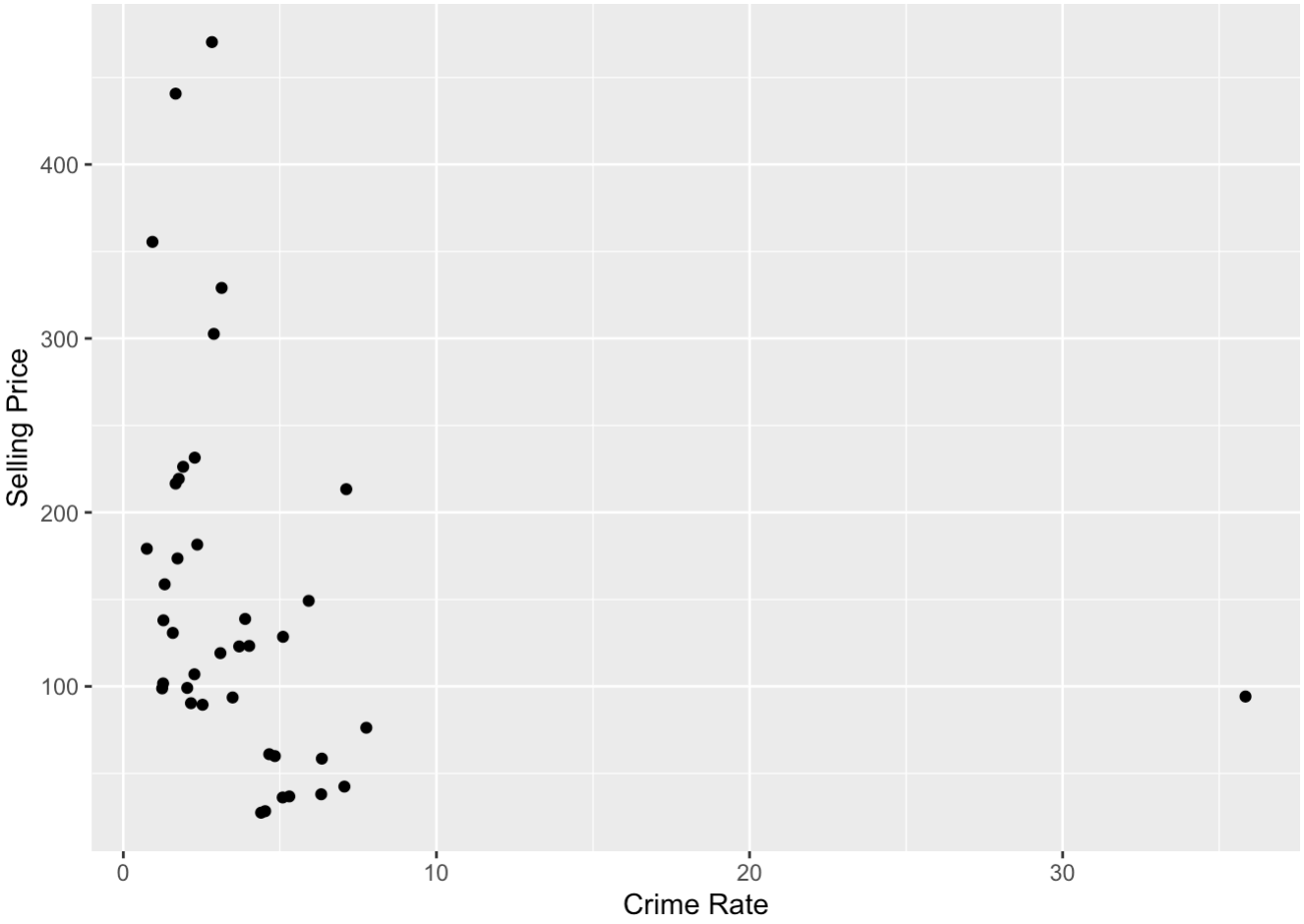
The data provided on the Excel file named MetroHomes describe housing prices near a large city. Each of the 40 data points describes a region of the metropoli- tan area. The column labeled Selling Price gives the median price for homes sold in that area during a particular year in thousands of dollars. The column labeled Crime Rate gives the number of crimes committed in that area, per 100,000 residents. Create a scatterplot of the selling price on

the crime rate. Which observation stands out from the others? Find the correlation using all of the given data. Exclude the distinct outlier and redraw the scatterplot focused on the rest of the data. Does your impression of the relationship between the crime rate and the selling price change?

```
metro_homes <- read_csv("/Users/bernardoarambula/Documents/MSBA/BAX400 foundations/Homework1/Homework 1 Question 3.csv")
```

```
## Rows: 40 Columns: 2
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## dbl (2): Crime Rate, Selling Price
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
scatter3 <- ggplot(metro_homes, aes(x = `Crime Rate`, y = `Selling Price`)) +  geom_point()
print(scatter3)
```



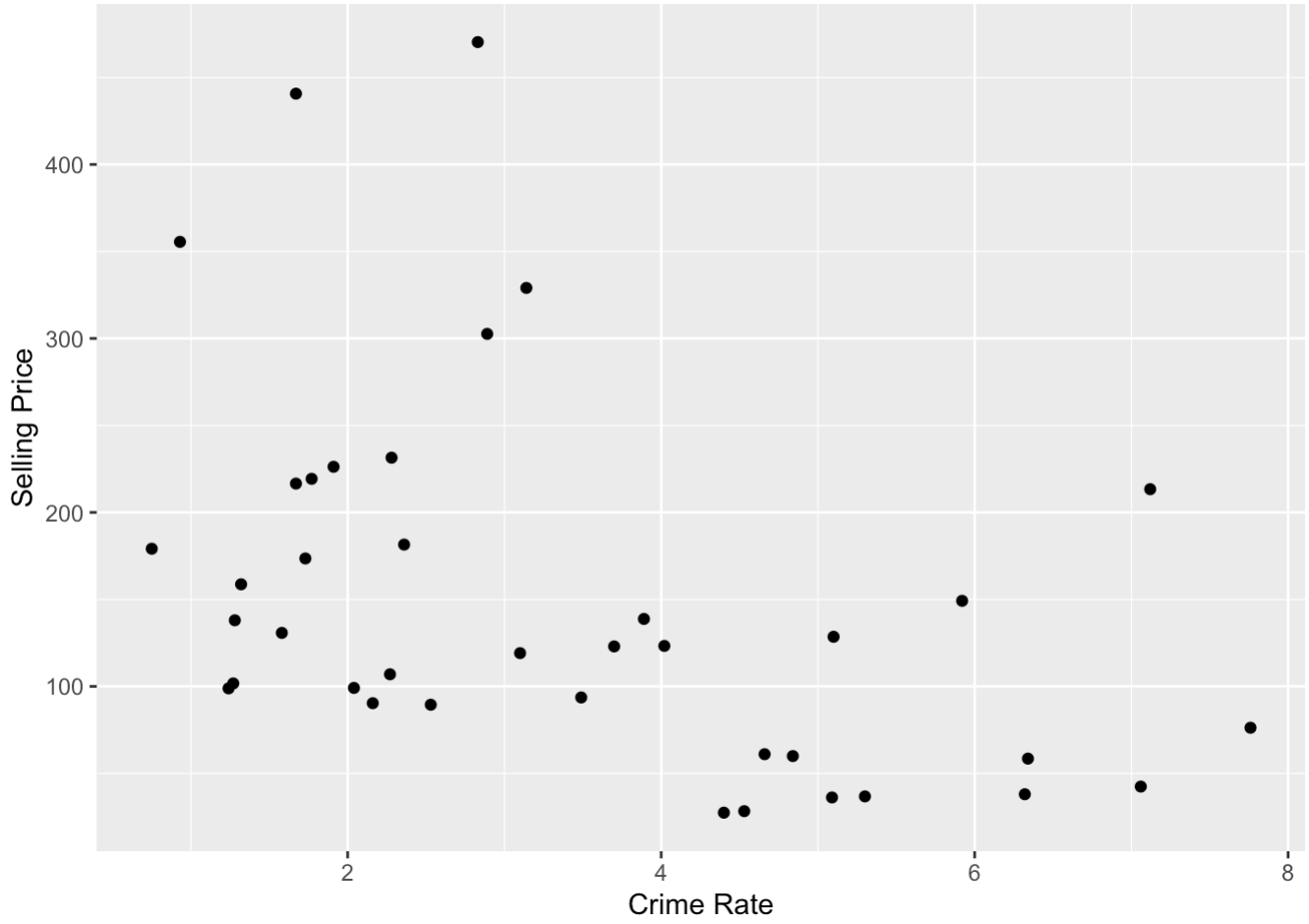
There is one outlier, the observation that has a crime rate of 35.84 seems to be seems to be unusually large. Observing the scatterplot above we see that the data point stands out from the rest.

```
qcor3 <- cor(metro_homes$`Crime Rate`, metro_homes$`Selling Price`)
sprintf("The correlation between Selling Price and Crime rate in the given data set is
%.3f.", qcor3)
```

```
## [1] "The correlation between Selling Price and Crime rate in the given data set is -
0.229."
```

**The initial correlation coefficient is -0.229. This indicates a weak negative relationship between the Crime Rate and the Selling Price, suggesting that, in general, areas with higher crime rates tend to have slightly lower median selling prices for homes.*

```
# Filtering data to exclude outlier
filtered_dataq3 <- subset(metro_homes, `Crime Rate` <= 30)
scatter3a <- ggplot(filtered_dataq3, aes(x = `Crime Rate`, y = `Selling Price`)) +  geom
_point()
print(scatter3a)
```




```
# Correlation between selling price and crime rate with outlier excluded
qcor3a <- cor(filtered_dataq3$`Crime Rate`, filtered_dataq3$`Selling Price`)
sprintf("The correlation between Selling Price and Crime rate in the given data set excl
uding the outlier is %.3f.", qcor3a)
```

```
## [1] "The correlation between Selling Price and Crime rate in the given data set exclu
ding the outlier is -0.433."
```

The new correlation coefficient is -0.433. This indicates a stronger negative relationship between the Crime Rate and the Selling Price after excluding the outlier. This suggests that, without the outlier, there is a clearer inverse relationship, where areas with higher crime rates tend to have significantly lower median selling prices for homes.

Question 4

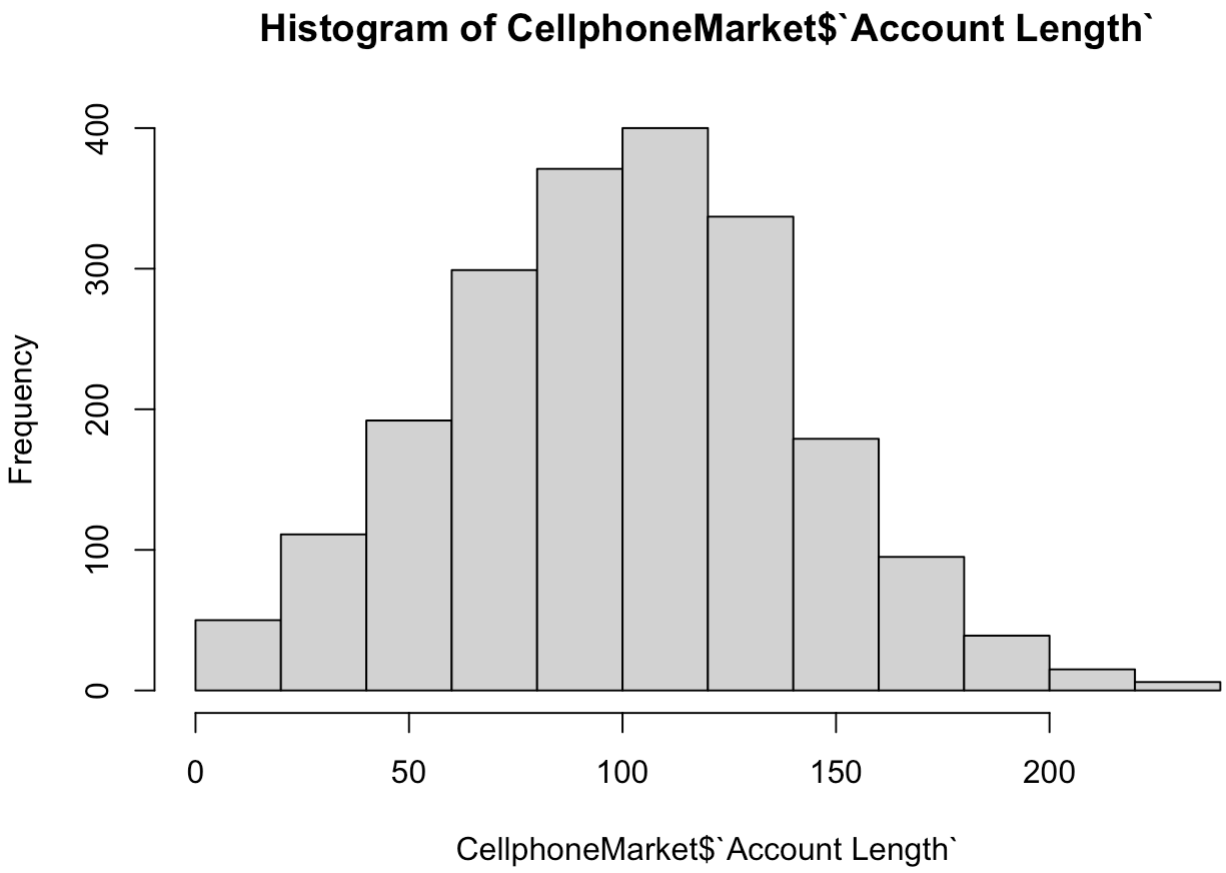
The term churn is very important to managers in the cellular phone busi- ness. Churning occurs when a customer stops using one company's service and switches to another company's service. Obviously, managers try to keep churn- ing to a minimum, not only by offering the best possible service, but by trying to identify conditions that lead to churning and taking steps to stop churning before it occurs. For example, if a company learns that customers tend to churn at the end of their two-year contract, they could offer customers an incentive to stay a month or two before the end of their two-year contract. The file Cell- phoneMarket contains data on over 2000 customers of a particular cellular phone company. Each row contains the activity of a particular customer for a given time period, and the last column indicates whether the customer churned during this time period. Use the various concepts in the EDA to learn (1) how these variables are distributed, (2) how the variables in columns B–R are related to each other, and (3) how the variables in columns B–R are related to the Churn variable in column S. Write a short explanation of your findings, including any recommendations you would make to the company to reduce churn.

```
CellphoneMarket <- read_csv("/Users/bernardoarambula/Documents/MSBA/BAX400 foundations/H
omework1/CellphoneMarket.csv")
```

```
## Rows: 2094 Columns: 19
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (3): International Plan, Voice Mail Plan, Churn?
## dbl (16): Customer, Account Length, Voice Mail Messages, Day Minutes, Day Ca...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

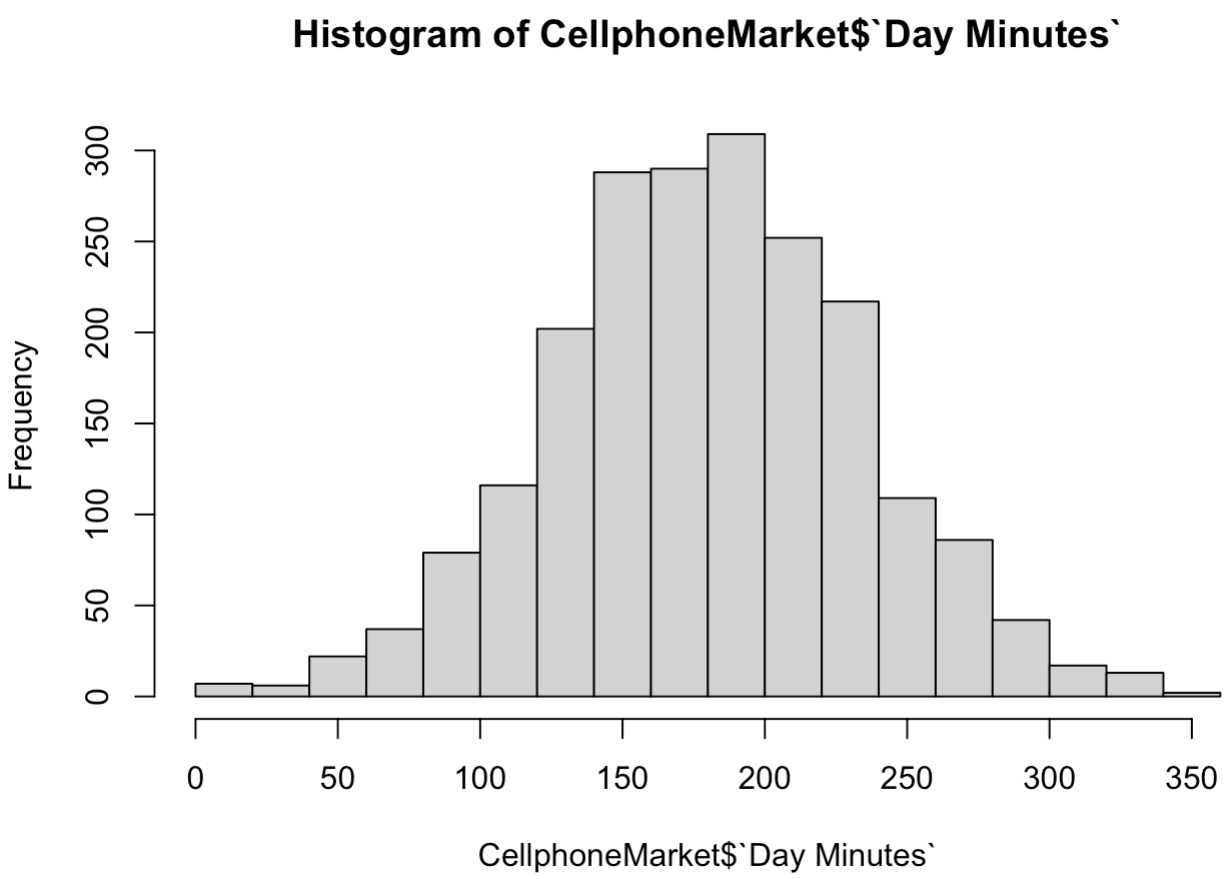
1)

```
hist(CellphoneMarket$`Account Length`)
```



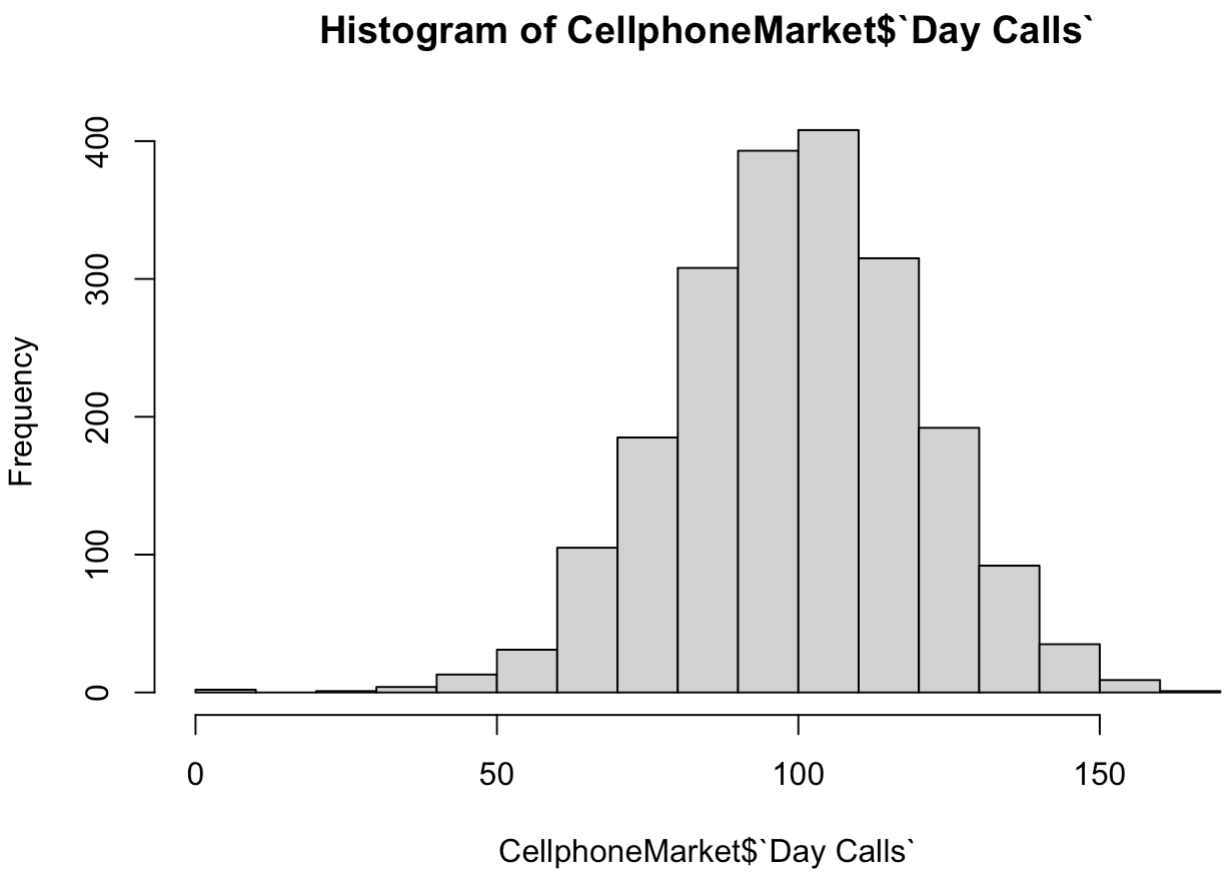
Normal dist

```
hist(CellphoneMarket$`Day Minutes`)
```



Normal Dist

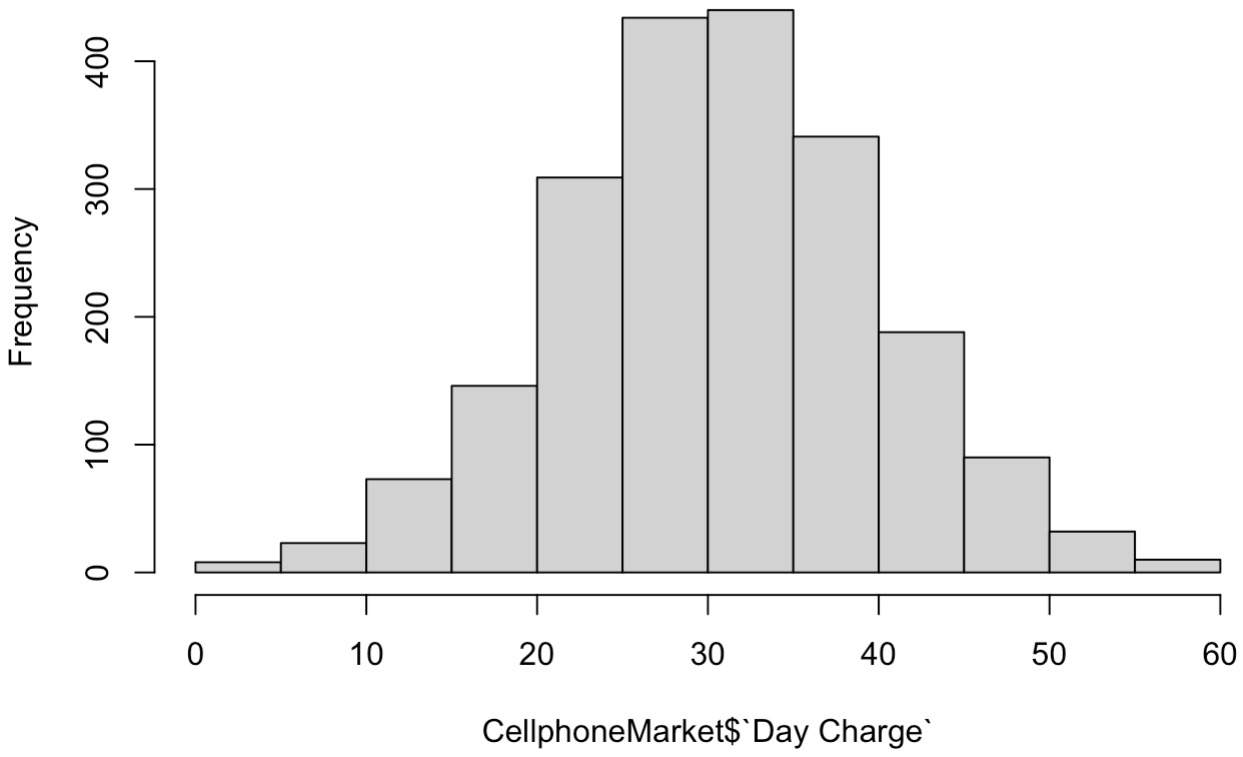
```
hist(CellphoneMarket$`Day Calls`)
```



Left

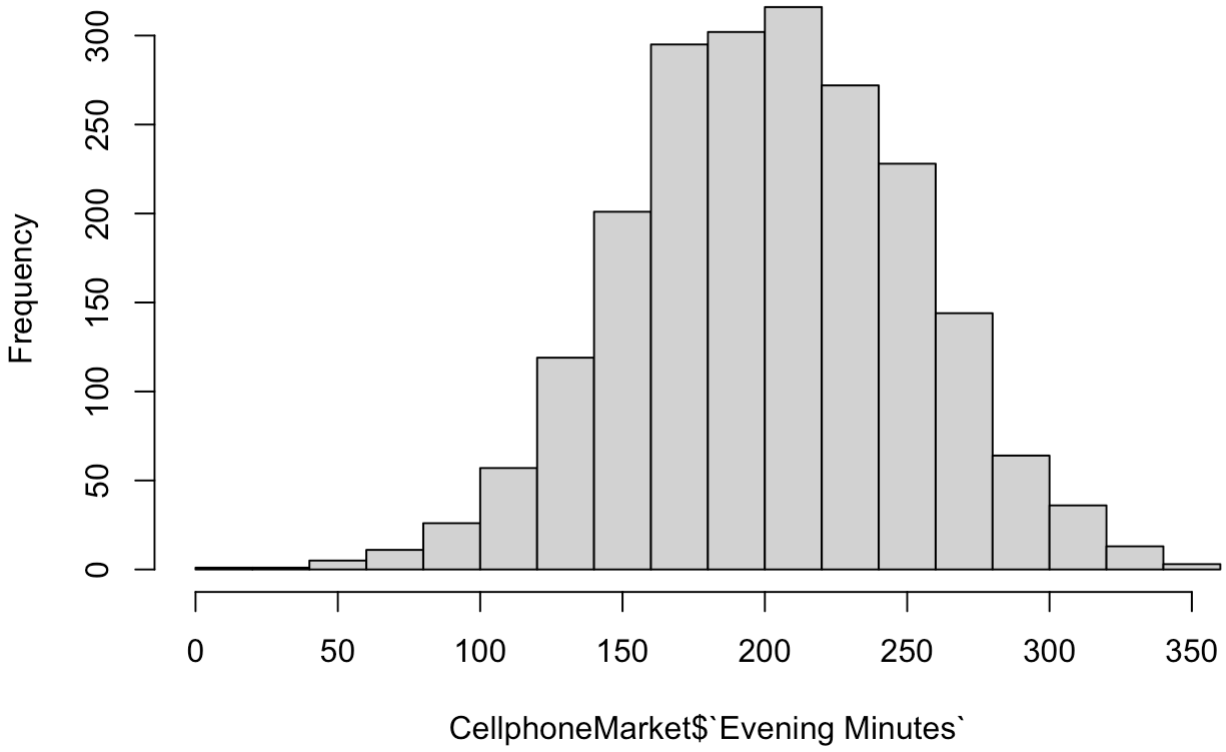
Skew

```
hist(CellphoneMarket$`Day Charge`)
```



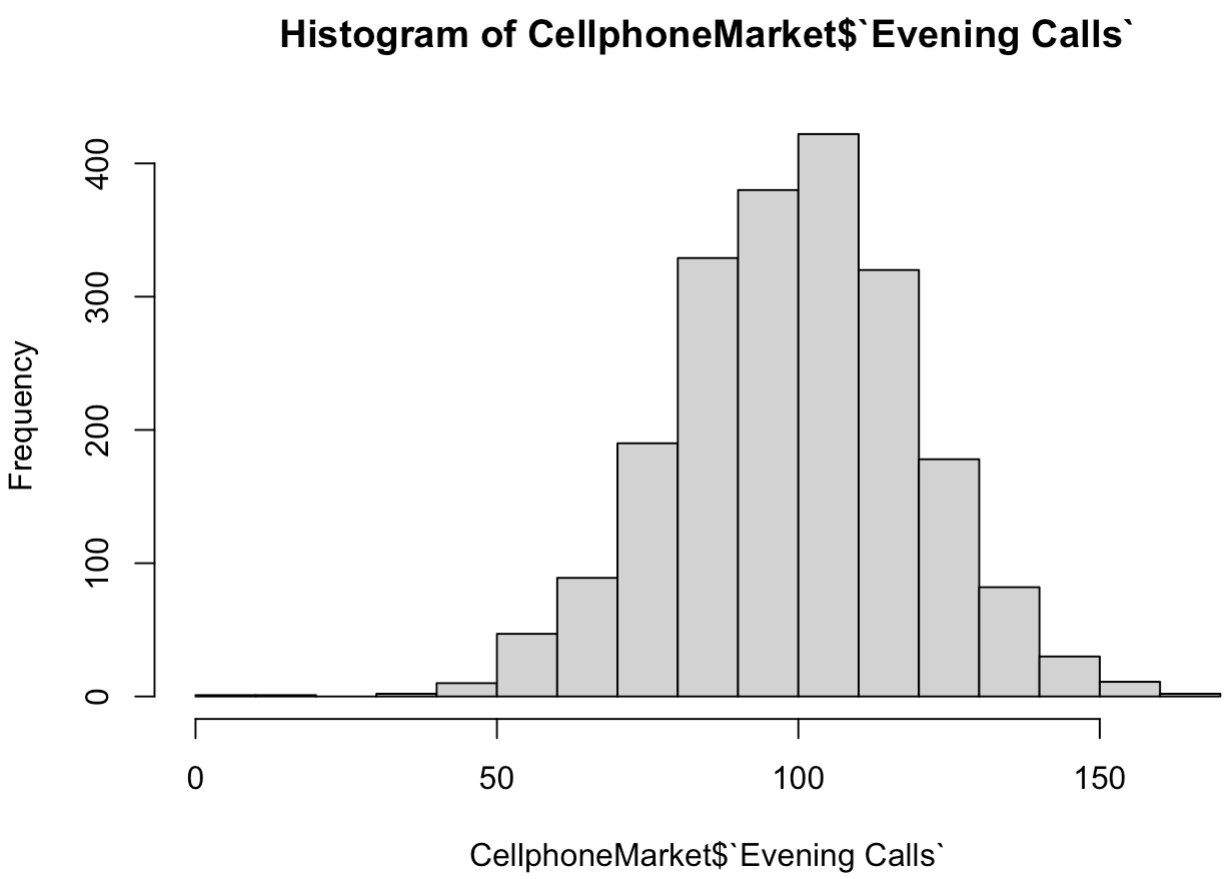
Normal Dist

```
hist(CellphoneMarket$`Evening Minutes`)
```



Normal Dist

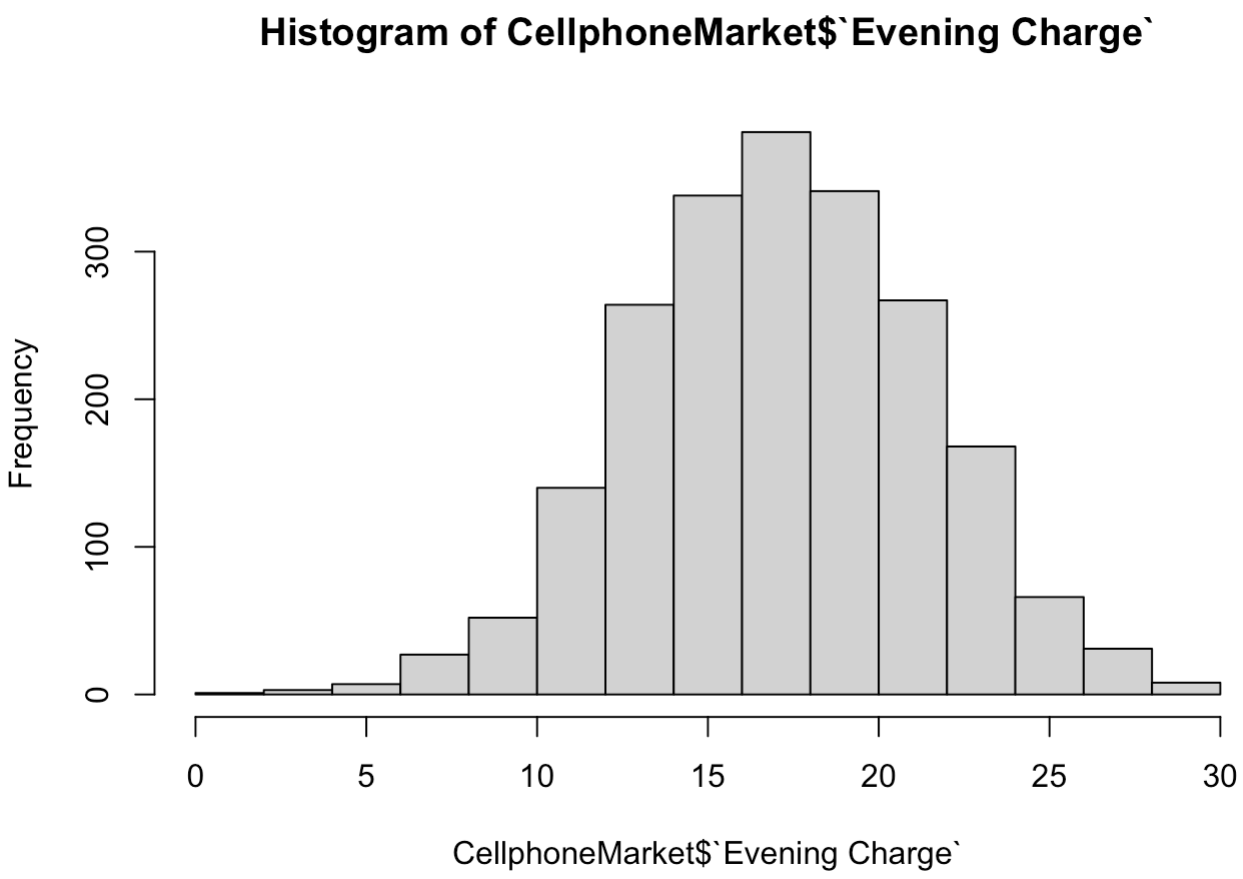
```
hist(CellphoneMarket$`Evening Calls`)
```



Left

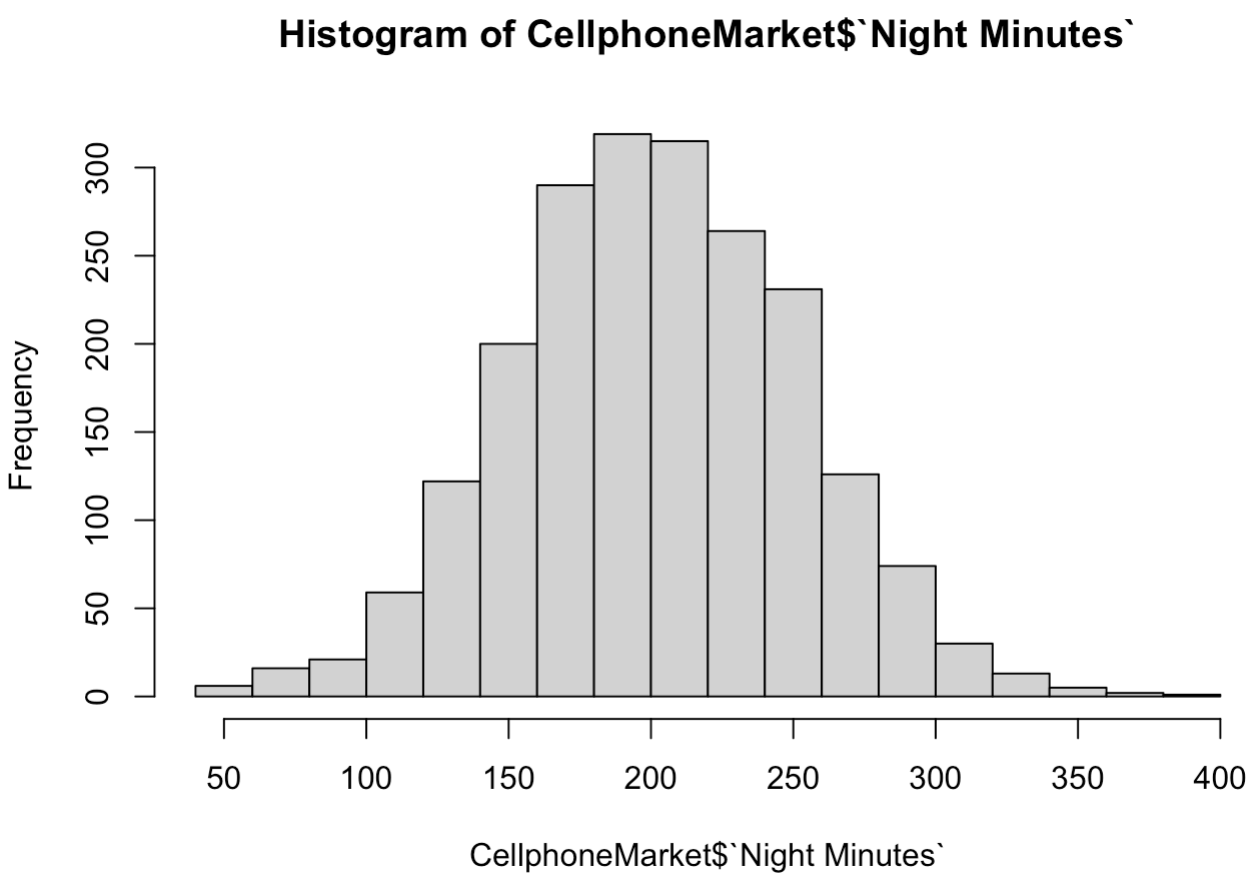
Skew

```
hist(CellphoneMarket$`Evening Charge`)
```



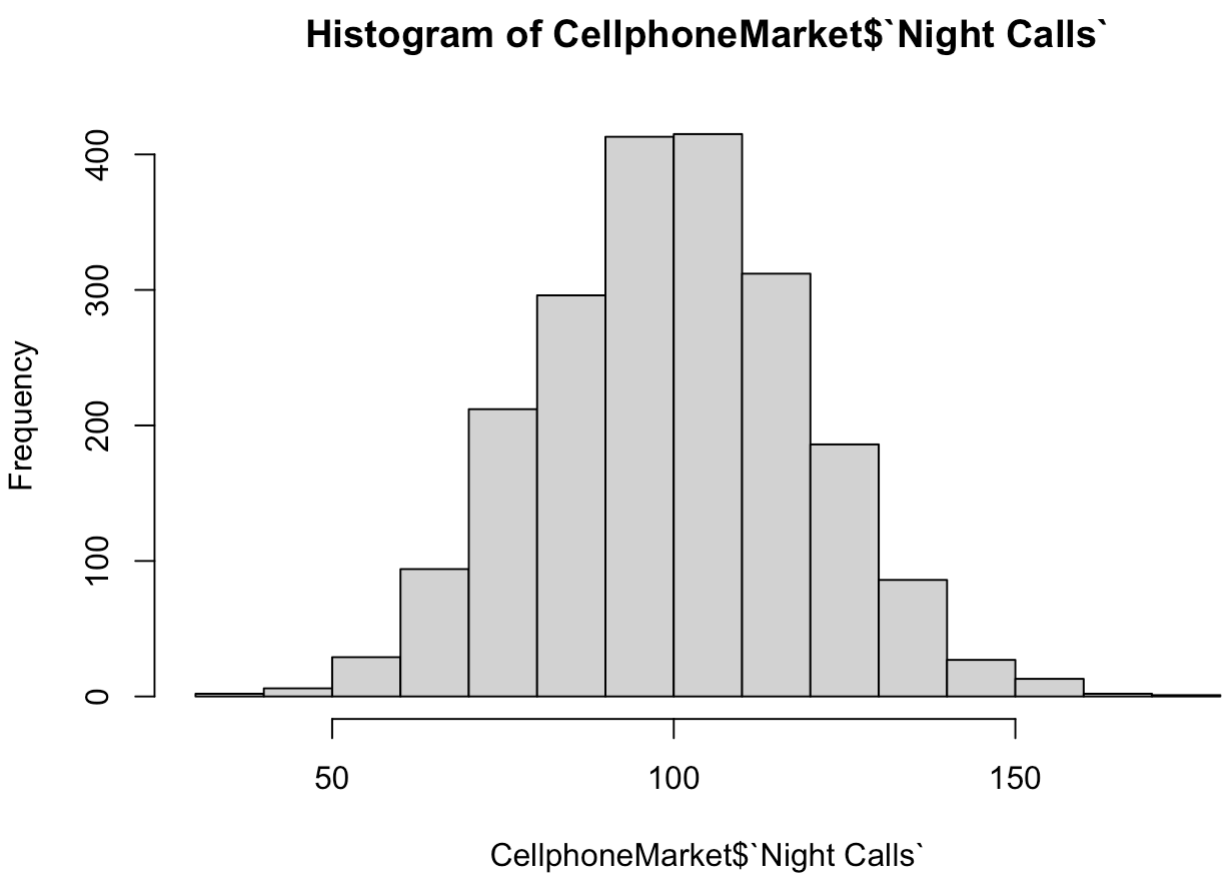
Normal Dist

```
hist(CellphoneMarket$`Night Minutes`)
```



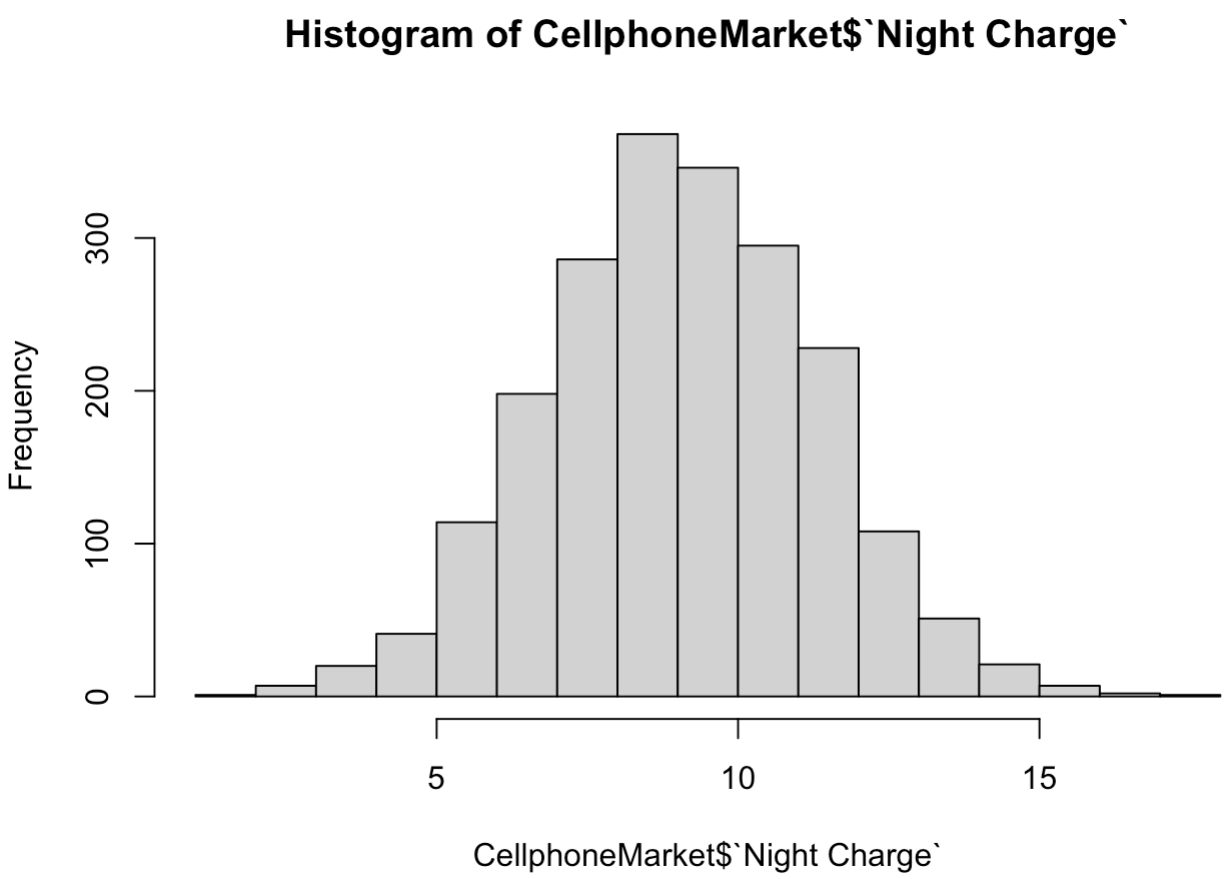
Normal Dist

```
hist(CellphoneMarket$`Night Calls`)
```



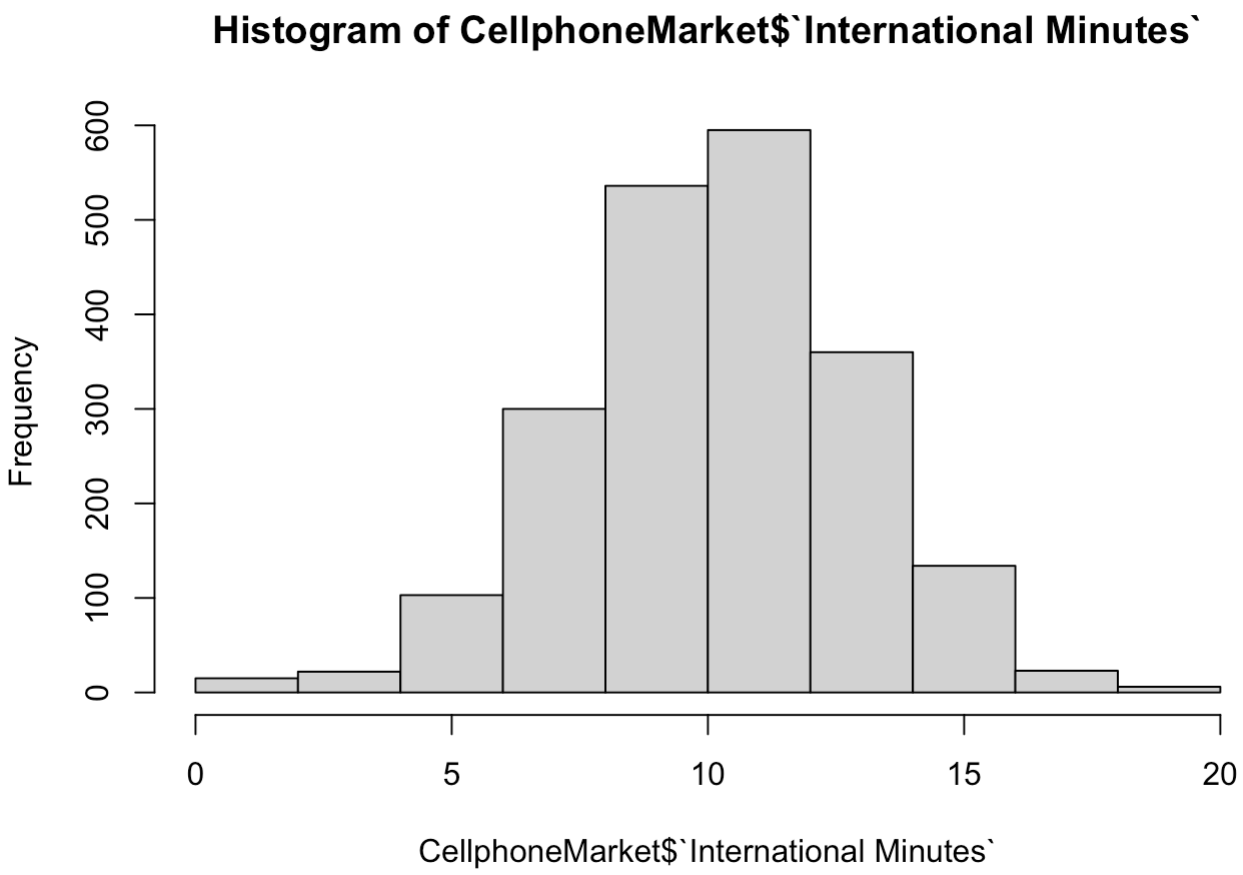
Normal Dist

```
hist(CellphoneMarket$`Night Charge`)
```



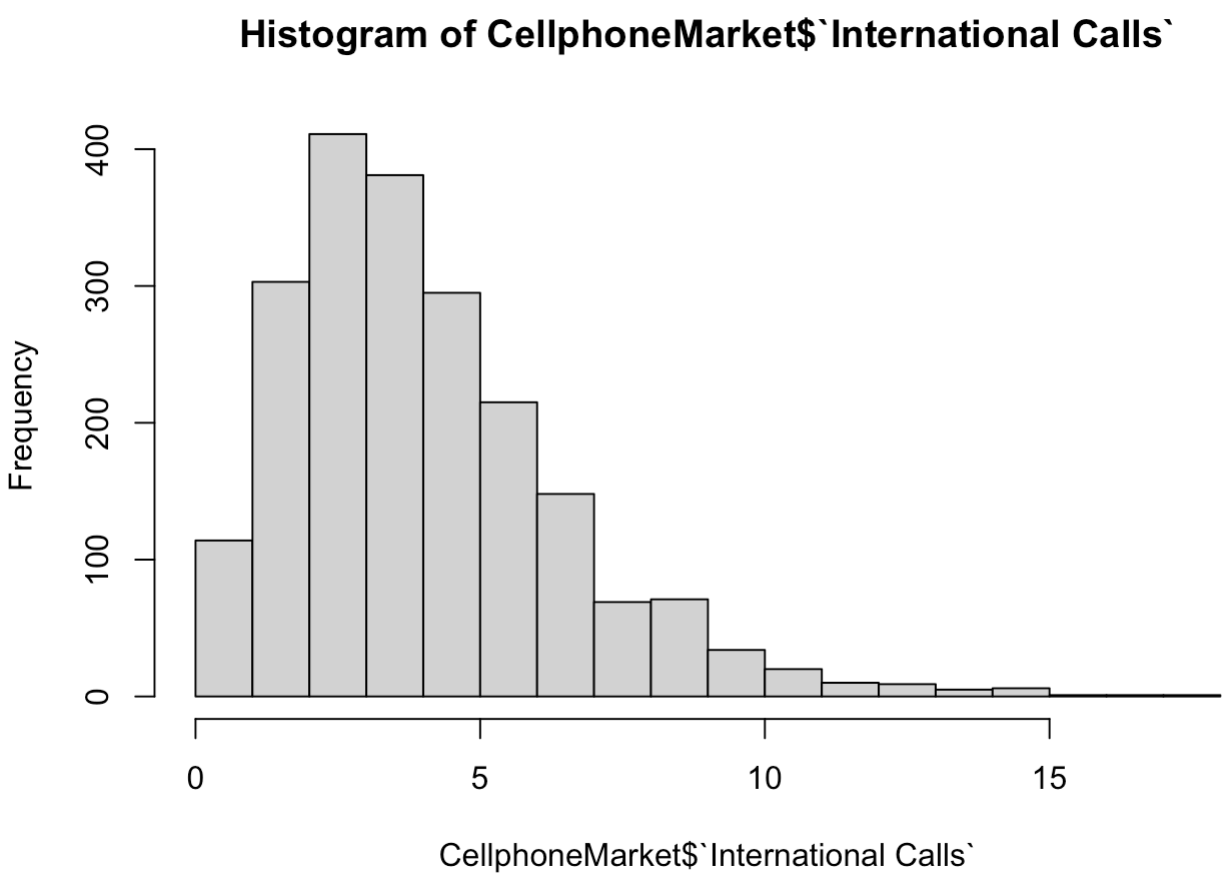
Normal Dist

```
hist(CellphoneMarket$`International Minutes`)
```



Normal Dist

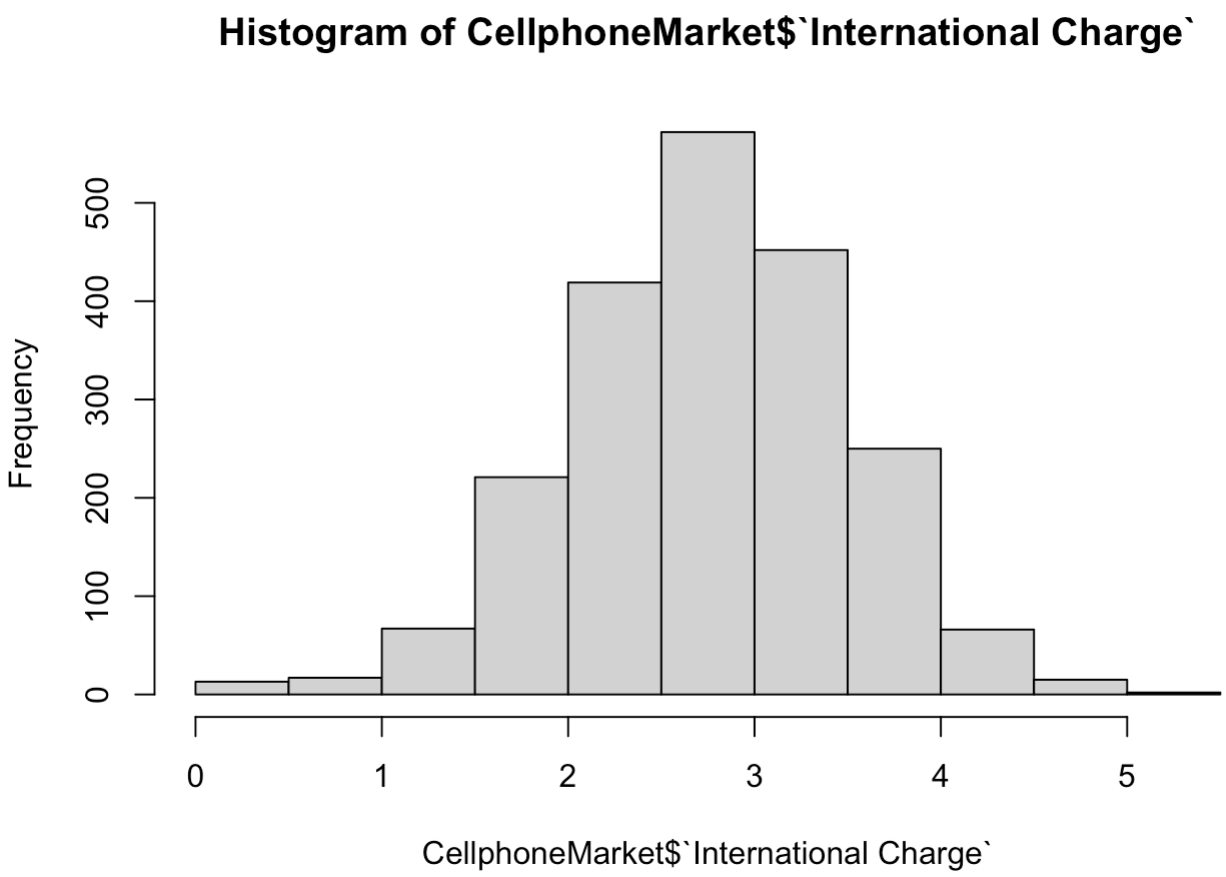
```
hist(CellphoneMarket$`International Calls`)
```



Right

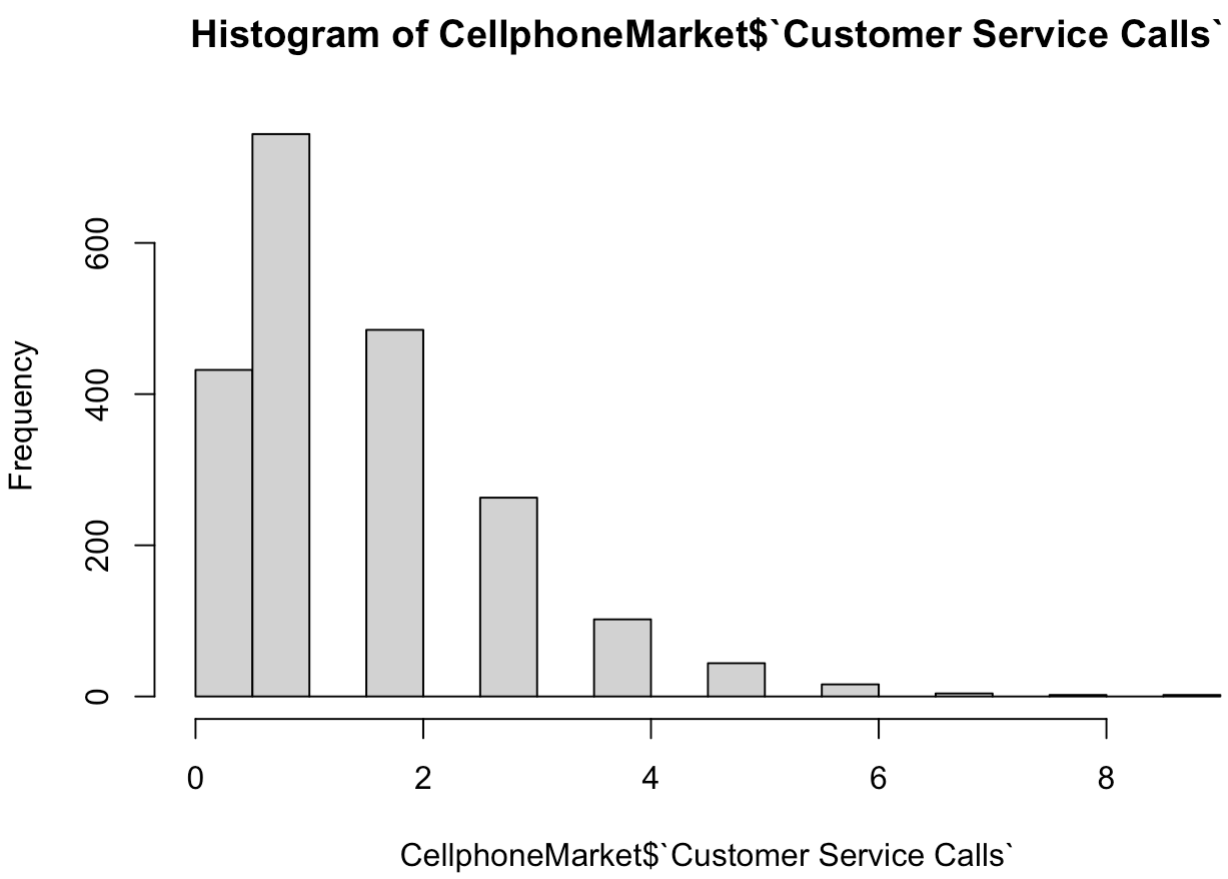
Skew

```
hist(CellphoneMarket$`International Charge`)
```



Normal Dist

```
hist(CellphoneMarket$`Customer Service Calls`)
```



Right

Skew

2)

```
# Mutate yes no columns to numeric
CellphoneMarket <- CellphoneMarket %>%
  mutate(`International Plan` = ifelse(`International Plan` == "Yes", 1, 0),)

CellphoneMarket <- CellphoneMarket %>%
  mutate(`Voice Mail Plan` = ifelse(`Voice Mail Plan` == "Yes", 1, 0))

# Select columns B to R
data_b_to_r <- CellphoneMarket %>% select('Account Length':'Customer Service Calls')

# Correlation matrix
cor_matrix <- cor(data_b_to_r)
```

```
## Warning in cor(data_b_to_r): the standard deviation is zero
```

```
print(cor_matrix)
```

```
##                         Account Length International Plan Voice Mail Plan
## Account Length             1.000000000                NA              NA
## International Plan                  NA                 1              NA
## Voice Mail Plan                     NA                NA               1
## Voice Mail Messages       -0.003285039                NA              NA
## Day Minutes                0.027294716                NA              NA
## Day Calls                  0.049966197                NA              NA
## Day Charge                 0.027293631                NA              NA
## Evening Minutes            0.013731075                NA              NA
## Evening Calls              0.027972841                NA              NA
## Evening Charge             0.013753808                NA              NA
## Night Minutes             -0.015299570                NA              NA
## Night Calls               -0.024372352                NA              NA
## Night Charge              -0.015291400                NA              NA
## International Minutes     -0.020481745                NA              NA
## International Calls        0.008254763                NA              NA
## International Charge      -0.020555500                NA              NA
## Customer Service Calls    -0.021252408                NA              NA
##                         Voice Mail Messages  Day Minutes    Day Calls
## Account Length               -0.003285039  0.027294716  0.049966197
## International Plan                     NA           NA           NA
## Voice Mail Plan                        NA           NA           NA
## Voice Mail Messages             1.000000000  0.020187375 -0.001714622
## Day Minutes                     0.020187375  1.000000000 -0.019506537
## Day Calls                      -0.001714622 -0.019506537  1.000000000
## Day Charge                      0.020190854  0.999999952 -0.019501031
## Evening Minutes                 0.017134923 -0.001067150  0.004223405
## Evening Calls                   0.009124203  0.023979285  0.013117225
## Evening Charge                  0.017141094 -0.001084264  0.004215994
## Night Minutes                   0.004910758  0.020876909  0.001657302
## Night Calls                     0.007471198  0.017903747 -0.021283335
## Night Charge                    0.004877521  0.020831673  0.001643368
## International Minutes           0.008756495  0.002915123  0.032224343
## International Calls             0.009975834  0.023665661  0.001420175
## International Charge            0.008835347  0.002986995  0.032267960
## Customer Service Calls         -0.006105574 -0.021887645 -0.015085772
##                          Day Charge Evening Minutes Evening Calls
## Account Length           0.027293631     0.013731075   0.027972841
## International Plan                NA              NA            NA
## Voice Mail Plan                   NA              NA            NA
## Voice Mail Messages      0.020190854     0.017134923   0.009124203
## Day Minutes              0.999999952    -0.001067150   0.023979285
## Day Calls               -0.019501031     0.004223405   0.013117225
## Day Charge               1.000000000    -0.001060409   0.023977571
## Evening Minutes         -0.001060409     1.000000000  -0.011699751
## Evening Calls            0.023977571    -0.011699751   1.000000000
## Evening Charge          -0.001077523     0.999999770  -0.011684099
## Night Minutes            0.020871488    -0.020370817  -0.001629504
## Night Calls              0.017911035     0.006592421  -0.005736107
## Night Charge             0.020826233    -0.020388216  -0.001606969
## International Minutes    0.002916771    -0.016936230   0.044643571
## International Calls      0.023662430    -0.011394623   0.010217684
```

```
## International Charge    0.002988620    -0.017003731   0.044652186
## Customer Service Calls -0.021895330    -0.020754050   0.007680288
##                        Evening Charge Night Minutes  Night Calls Night Charge
## Account Length            0.013753808  -0.015299570 -0.024372352 -0.015291400
## International Plan                 NA            NA           NA           NA
## Voice Mail Plan                    NA            NA           NA           NA
## Voice Mail Messages       0.017141094   0.004910758  0.007471198  0.004877521
## Day Minutes              -0.001084264   0.020876909  0.017903747  0.020831673
## Day Calls                 0.004215994   0.001657302 -0.021283335  0.001643368
## Day Charge               -0.001077523   0.020871488  0.017911035  0.020826233
## Evening Minutes           0.999999770  -0.020370817  0.006592421 -0.020388216
## Evening Calls            -0.011684099  -0.001629504 -0.005736107 -0.001606969
## Evening Charge            1.000000000  -0.020374606  0.006582550 -0.020392012
## Night Minutes            -0.020374606   1.000000000  0.013092155  0.999999198
## Night Calls               0.006582550   0.013092155  1.000000000  0.013085577
## Night Charge             -0.020392012   0.999999198  0.013085577  1.000000000
## International Minutes    -0.016939313  -0.001721787 -0.014620528 -0.001701377
## International Calls      -0.011407957  -0.014868368  0.003784129 -0.014850146
## International Charge     -0.017006804  -0.001737888 -0.014599793 -0.001717505
## Customer Service Calls   -0.020768818  -0.041359074 -0.016132223 -0.041350228
##                        International Minutes International Calls
## Account Length                -0.020481745         0.008254763
## International Plan                      NA                  NA
## Voice Mail Plan                         NA                  NA
## Voice Mail Messages            0.008756495         0.009975834
## Day Minutes                    0.002915123         0.023665661
## Day Calls                      0.032224343         0.001420175
## Day Charge                     0.002916771         0.023662430
## Evening Minutes               -0.016936230        -0.011394623
## Evening Calls                  0.044643571         0.010217684
## Evening Charge                -0.016939313        -0.011407957
## Night Minutes                 -0.001721787        -0.014868368
## Night Calls                   -0.014620528         0.003784129
## Night Charge                  -0.001701377        -0.014850146
## International Minutes          1.000000000         0.025508199
## International Calls            0.025508199         1.000000000
## International Charge           0.999992709         0.025635528
## Customer Service Calls         0.004979737        -0.017527201
##                        International Charge Customer Service Calls
## Account Length                -0.020555500           -0.021252408
## International Plan                      NA                     NA
## Voice Mail Plan                         NA                     NA
## Voice Mail Messages            0.008835347           -0.006105574
## Day Minutes                    0.002986995           -0.021887645
## Day Calls                      0.032267960           -0.015085772
## Day Charge                     0.002988620           -0.021895330
## Evening Minutes               -0.017003731           -0.020754050
## Evening Calls                  0.044652186            0.007680288
## Evening Charge                -0.017006804           -0.020768818
## Night Minutes                 -0.001737888           -0.041359074
## Night Calls                   -0.014599793           -0.016132223
## Night Charge                  -0.001717505           -0.041350228
```

```
## International Minutes           0.999992709           0.004979737
## International Calls             0.025635528          -0.017527201
## International Charge            1.000000000           0.004891190
## Customer Service Calls          0.004891190           1.000000000
```

3)

```
# Convert Churn? to numeric (1 for Yes, 0 for No)
CellphoneMarket <- CellphoneMarket %>%
  mutate(Churn = ifelse(`Churn?` == "Yes", 1, 0))

# Select columns B to R and the Churn? column
data_b_to_r <- CellphoneMarket %>%
  select(`Account Length`:`Customer Service Calls`, Churn)

# Calculate point-biserial correlation for continuous variables
continuous_vars <- data_b_to_r %>%
  select(-Churn) %>%
  select_if(is.numeric)

cor_results <- data.frame(variable = colnames(continuous_vars),
                          correlation = sapply(continuous_vars, function(x) cor(x, data_
b_to_r$Churn)))
```

```
## Warning in cor(x, data_b_to_r$Churn): the standard deviation is zero

## Warning in cor(x, data_b_to_r$Churn): the standard deviation is zero
```

```
print(cor_results)
```

```
##                                    variable  correlation
## Account Length                Account Length  0.012276200
## International Plan         International Plan           NA
## Voice Mail Plan               Voice Mail Plan           NA
## Voice Mail Messages       Voice Mail Messages -0.091783211
## Day Minutes                       Day Minutes  0.237603938
## Day Calls                           Day Calls  0.027559244
## Day Charge                         Day Charge  0.237601831
## Evening Minutes               Evening Minutes  0.084111384
## Evening Calls                   Evening Calls  0.009236285
## Evening Charge                 Evening Charge  0.084088839
## Night Minutes                   Night Minutes  0.026950582
## Night Calls                       Night Calls  0.005877139
## Night Charge                     Night Charge  0.026951943
## International Minutes   International Minutes  0.046841208
## International Calls       International Calls -0.054976601
## International Charge     International Charge  0.046907863
## Customer Service Calls Customer Service Calls  0.202931934
```

Question 5

In the book Foundations of Financial Management (7th ed.), Stanley B. Block and Geoffrey A. Hirt discuss a semiconductor firm that is considering two choices: (1) expanding the production of semiconductors for sale to end users or (2) entering the highly competitive home computer market. The cost of both projects is $60 million, but the net present value of the cash flows from sales and the risks are different. Figure 1 below gives a tree diagram of the project choices. The tree diagram gives a probability distribution of expected sales for each project. It also gives the present value of cash flows from sales and the net present value (NPV = present value of cash flow from sales minus initial cost) corresponding to each sales alternative. Note that figures in parentheses denote losses.

a. Find the expected net present value of expanding semiconductor business project.

```
# Define the probabilities and NPVs
q5probsexpand <- c(0.5, 0.25, 0.25)
q5npvexpand <- c(40, 15, -20)

# Calculate the expected NPV (ENPV)
q5enpvexpand <- sum(q5probsexpand * q5npvexpand)

# Print the result
sprintf("The expected NPV of expanding the semiconductor business is %.3f.", q5enpvexpan
d)
```

```
## [1] "The expected NPV of expanding the semiconductor business is 18.750."
```

b. Determine variance and standard deviation of the net present value

```
# Define probabilities and NPVs
q5probsexpand <- c(0.5, 0.25, 0.25)
q5npvexpand <- c(40, 15, -20)

# Calculate the expected NPV (ENPV)
q5enpvexpand <- sum(q5probsexpand * q5npvexpand)

# Calculate variance of NPV
q5varianceexpand <- sum((q5npvexpand - q5enpvexpand)^2 * q5probsexpand)

# Calculate standard deviation of NPV
q5sdexpand <- sqrt(q5varianceexpand)

# Print the results
sprintf("The variance of expanding the semiconductor business is %.3f.", q5varianceexpan
d)
```

```
## [1] "The variance of expanding the semiconductor business is 604.688."
```

```
sprintf("The standard deviation of expanding the semiconductor business is %.3f.", q5sde
xpand)
```

```
## [1] "The standard deviation of expanding the semiconductor business is 24.590."
```

c. Find the expected net present value of entering home computer market.

```
q5probshome <- c(.2, .5, .3)
q5npvhome <- c(140, 15, -35)

q5enpvhome <- sum(q5probshome * q5npvhome)

sprintf("The expected NPV of entering the home computer market is %.3f.", q5enpvhome)
```

```
## [1] "The expected NPV of entering the home computer market is 25.000."
```

d. Determine variance and standard deviation of the net present value

```
# Calculate variance of NPV for entering home computer market
q5variancehome <- sum((q5npvhome - q5enpvhome)^2 * q5probshome)

# Calculate standard deviation of NPV for entering home computer market
q5sdhome <- sqrt(q5variancehome)

# Print the results
sprintf("The variance of entering the home computer market is %.3f.", q5variancehome)
```

```
## [1] "The variance of entering the home computer market is 3775.000."
```

```
sprintf("The standard deviation of entering the home computer market is %.3f.", q5sdhom
e)
```

```
## [1] "The standard deviation of entering the home computer market is 61.441."
```

e. Which project has the higher expected net present value?

Entering the home computer market has the higher expected net present value of $25 million.

f. Which project carries the least risk? Explain.

It is less risky to expand because the NPV is lower between the two

g. Calculate the relative variation for each project choice. Compare them to see which project carries the least risk. Is your response consistent with the part f) above?

```
q5cvexpand <- q5sdexpand / q5enpvexpand
sprintf("The coeffecient of variance for expanding the semiconducter business is %.3f.",
q5cvexpand <- q5sdexpand / q5enpvexpand
)
```

```
## [1] "The coeffecient of variance for expanding the semiconducter business is 1.311."
```

```
q5cvhome <- q5sdhome / q5enpvhome
sprintf("The coeffecient of variance for entering the home computer market is %.3f.",q5c
vhome <- q5sdhome / q5enpvhome)
```

```
## [1] "The coeffecient of variance for entering the home computer market is 2.458."
```

Question 6

An office machine costs $7,500 to replace unless a mysterious, hard-to-find prob- lem can be found and fixed. Repair calls from any service technician cost $500 each, and you're willing to spend up to $2,000 to get this machine fixed. You estimate that a repair technician has a 27% chance of fixing it.

a. Create a probability model for the number of visits needed to fix the machine or exhaust your budget of $2,000.

```
# Define the probability of fixing the machine
p_fix <- 0.27
p_not_fix <- 1 - p_fix

# Number of visits (1, 2, 3, 4, 5) where 5 represents the scenario of exhausting the bud
get without fixing the machine
visits <- 1:5

# Calculate probabilities
probabilities <- c((p_not_fix^0) * p_fix,  # P(X=1)
                   (p_not_fix^1) * p_fix,  # P(X=2)
                   (p_not_fix^2) * p_fix,  # P(X=3)
                   (p_not_fix^3) * p_fix,  # P(X=4)
                   (p_not_fix^4))          # P(X=5, exhausting the budget without fixin
g)

# Create a data frame to represent the probability model
probability_model <- data.frame(Visits = visits, Probability = probabilities)

# Print the probability model
print(probability_model)
```

```
##   Visits Probability
## 1      1   0.2700000
## 2      2   0.1971000
## 3      3   0.1438830
## 4      4   0.1050346
## 5      5   0.2839824
```

b. Find the expected number of service technicians that will be called in.

```
# Calculate the expected number of visits
expected_visits <- sum(probability_model$Visits * probability_model$Probability)

# Print the expected number of visits
sprintf("The expected number of service technicians that will be called in is %.3f.", ex
pected_visits)
```

```
## [1] "The expected number of service technicians that will be called in is 2.936."
```

c. Find the expected amount spent on this machine. (You must spend $7,500 for a new one if you do not find a vendor that repairs yours.)

```
# Define cost of repair and replacement
repair_cost <- 500
replacement_cost <- 7500
max_visits <- 4

# Expected cost of repairs
expected_repair_cost <- expected_visits * repair_cost

# Probability of needing a replacement
prob_replacement <- probabilities[max_visits + 1]

# Expected cost of replacement
expected_replacement_cost <- prob_replacement * replacement_cost

# Total expected amount spent
expected_amount_spent <- expected_repair_cost + expected_replacement_cost

# Print the results
sprintf("The expected amount spent on this machine is $%.2f.", expected_amount_spent)
```

```
## [1] "The expected amount spent on this machine is $3597.82."
```

Question 7

A manager at 24/7 Fitness Center is strategic about contacting open house attendees. With her strategy, she believes that 40% of the attendees she contacts will purchase a club membership. Suppose she contacts 20 open house attendees.

a. What is the probability that exactly 10 of the attendees will pur- chase a club membership?

```
prob_10 <- dbinom(10, size = 20, prob = 0.40)

sprintf("The probability of exactly 10 will purchase a club membership is %.3f.", prob_1
0)
```

```
## [1] "The probability of exactly 10 will purchase a club membership is 0.117."
```

b. What is the probability that no more than 10 of the attendees will purchase a club membership?

```
prob_nm10 <- pbinom(10, size = 20, prob = 0.40, lower.tail = TRUE)

sprintf("The probability of no more than 10 purchasing a club membership is %.3f.", prob
_nm10)
```

```
## [1] "The probability of no more than 10 purchasing a club membership is 0.872."
```

c. What is the probability that at least 15 of the attendees will purchase a club membership?

```
prob_al10 <- pbinom(14, size = 20, prob = 0.40, lower.tail = FALSE)

sprintf("The probability of at least 15 purchasing a club membership is %.3f.", prob_al1
0)
```

```
## [1] "The probability of at least 15 purchasing a club membership is 0.002."
```

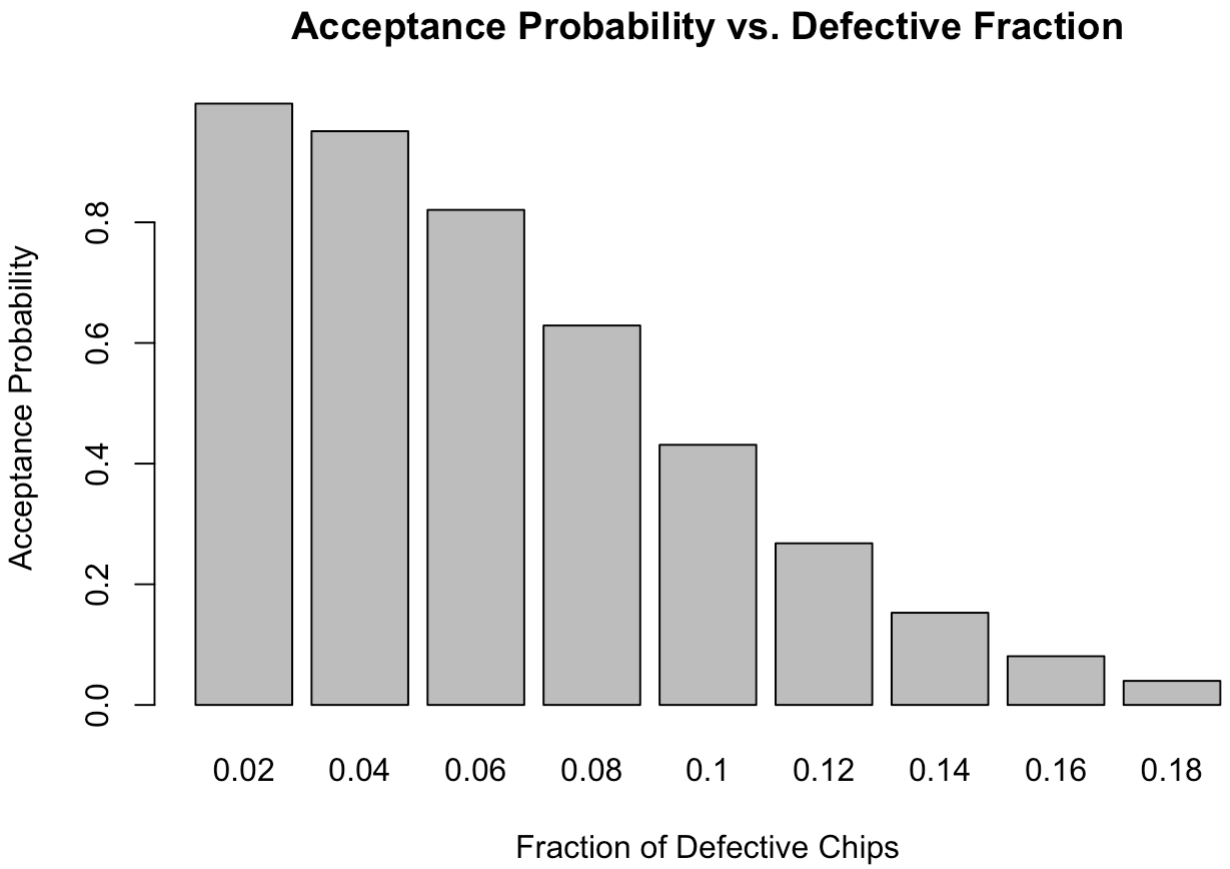
Question 8

Sampling is a very common practice in quality control. Sampling plans are cre- ated to control the quality of manufactured items that are ready to be shipped. To illustrate the use of a sampling plan, suppose that a chip manufacturing company produces and ships electronic computer chips in lots, each lot consist- ing of 1000 chips. This company's sampling plan specifies that quality control personnel should randomly sample 50 chips from each lot and accept the lot for shipping if the number of defective chips is four or fewer. The lot will be rejected if the number of defective chips is five or more. Find the probability of accepting a lot as a function of the actual fraction of defective chips. In particular, let the actual fraction of defective chips in a given lot equal any of 0.02, 0.04, 0.06, 0.08, 0.10, 0.12, 0.14, 0.16, 0.18. Then compute the lot acceptance probability for each of these lot defective fractions. Create a bar plot showing how the acceptance probability varies with the frac- tion defective. Comment on what you observe. A revised sampling plan call for accepting a given lot if the number of defective chips found in the random sample of 50 chips is five or fewer.Repeat the above exercise and summarize any notable differences between the two bar plots.

```
probs <- c(0.02, 0.04, 0.06, 0.08, 0.10, 0.12, 0.14, 0.16, 0.18)
acceptance <- pbinom(4,50,probs)

AcceptProbs <- data.frame(probs,acceptance)
AcceptProbs
```

```
##   probs acceptance
## 1  0.02 0.99679026
## 2  0.04 0.95102853
## 3  0.06 0.82059605
## 4  0.08 0.62895014
## 5  0.10 0.43119841
## 6  0.12 0.26795385
## 7  0.14 0.15281289
## 8  0.16 0.08078129
## 9  0.18 0.03988459
```

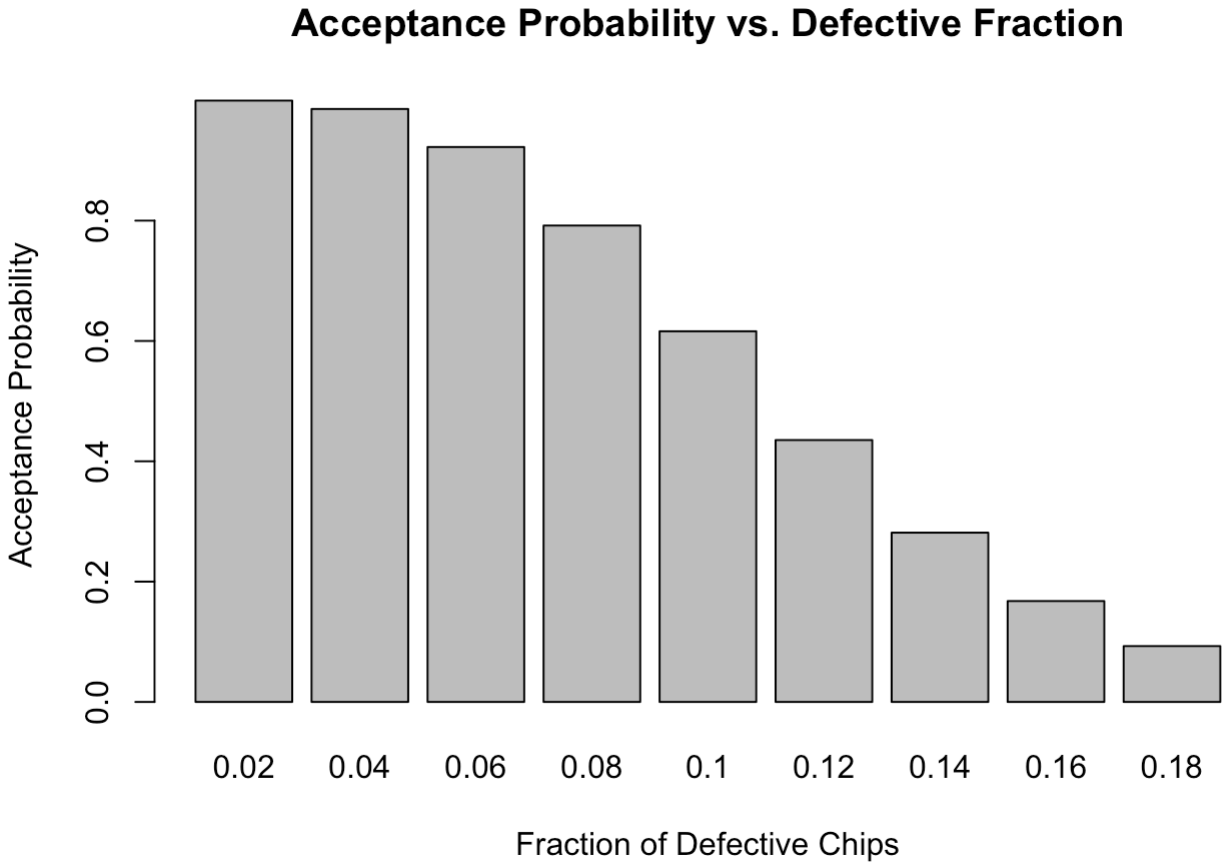
```
barplot(AcceptProbs$acceptance, names.arg = AcceptProbs$probs, main = "Acceptance Probab
ility vs. Defective Fraction", xlab = "Fraction of Defective Chips", ylab = "Acceptance
Probability")
```




```
acceptance2 <- pbinom(5,50,probs)
acceptprobs2 <- data.frame(probs, acceptance2)
acceptprobs2
```

```
##   probs acceptance2
## 1  0.02  0.99952178
## 2  0.04  0.98558960
## 3  0.06  0.92235940
## 4  0.08  0.79187371
## 5  0.10  0.61612301
## 6  0.12  0.43533565
## 7  0.14  0.28138662
## 8  0.16  0.16772728
## 9  0.18  0.09285919
```

```
barplot(acceptprobs2$acceptance, names.arg = acceptprobs2$probs, main = "Acceptance Prob
ability vs. Defective Fraction", xlab = "Fraction of Defective Chips", ylab = "Acceptanc
e Probability")
```



There aren't any major changes when we revise the sample plan to reject lots with 5 defective chips from 4 defective chips. Both barplots have a right skew and and the only difference is the probability of acceptance slightly increases when we choose to reject lots with 5 or less instead of 4 or less.

Question 9

A leading pizza vendor has a contract to supply pizza at all home baseball games in Sacramento. Before each game begins, a constant challenge is to determine how many pizzas to make available at the games. Ken Binlard, a business an- alyst, has determined that his fixed cost of providing pizzas is $1,000. Ken believes that this cost

should be equally allocated between two types of pizzas. Ken will supply only two types of pizzas: plain cheese and veggie-and-cheese combo. It costs Ken $4.50 to produce a plain cheese pizza and $5.00 to pro- duce a veggie-and-cheese pizza. The selling price for both pizzas at the game is $9.00. Left over pizzas will have no value and will be donated to the homeless. From experience, Ken has arrived at the following demand distributions for the two types of pizza at home games:

a. For both plain cheese and veggie-and-cheese combo, determine the profit (or loss) associated with producing at different possible demand levels. For instance, determine the profit if 200 plain cheese pizzas are produced and 200 are demanded. What is the profit if 200 plain cheese pizzas are produced but 300 were demanded, and so on? Summarize your results in a two-way data table using R. A two-way data table is one in which rows correspond to one variable (say, demand) and columns correspond to another variable (say, production). The body of the table contains the data. You will create two such tables – one for plain cheese and another for veggie-and-cheese pizza.

```
# Define the parameters
plain_cheese_demand <- c(200, 300, 400, 500, 600, 700, 800, 900)
plain_cheese_prob <- c(.1, .15, .15, .20, .20, .10, .05, .05)
veggie_cheese_demand <- c(300, 400, 500, 600, 700, 800)
veggie_cheese_prob <- c(.1, .2, .25, .25, .15, .05)
fixed_cost <- 1000
cost_cheese <- 4.50
cost_veggie_cheese <- 5
Price_pizza <- 9

# Calculate profit function
calc_profit <- function(demand, production, cost) {
  revenue <- pmin(demand, production) * Price_pizza
  cost_total <- production * cost + fixed_cost / 2
  profit <- revenue - cost_total
  return(profit)
}

# Create data table for plain cheese
plain_cheese_production <- seq(200, 900, by=100)
plain_cheese_profit_table <- outer(plain_cheese_demand, plain_cheese_production, Vectori
ze(function(d, p) calc_profit(d, p, cost_cheese)))

# Create data table for veggie-and-cheese
veggie_cheese_production <- seq(300, 800, by=100)
veggie_cheese_profit_table <- outer(veggie_cheese_demand, veggie_cheese_production, Vect
orize(function(d, p) calc_profit(d, p, cost_veggie_cheese)))

# Print the results
plain_cheese_profit_table
```

```
##      [,1] [,2] [,3] [,4]  [,5]  [,6]  [,7]  [,8]
## [1,]  400  -50 -500 -950 -1400 -1850 -2300 -2750
## [2,]  400  850  400  -50  -500  -950 -1400 -1850
## [3,]  400  850 1300  850   400   -50  -500  -950
## [4,]  400  850 1300 1750  1300   850   400   -50
## [5,]  400  850 1300 1750  2200  1750  1300   850
## [6,]  400  850 1300 1750  2200  2650  2200  1750
## [7,]  400  850 1300 1750  2200  2650  3100  2650
## [8,]  400  850 1300 1750  2200  2650  3100  3550
```

```
veggie_cheese_profit_table
```

```
##      [,1] [,2] [,3] [,4]  [,5]  [,6]
## [1,]  700  200 -300 -800 -1300 -1800
## [2,]  700 1100  600  100  -400  -900
## [3,]  700 1100 1500 1000   500     0
## [4,]  700 1100 1500 1900  1400   900
## [5,]  700 1100 1500 1900  2300  1800
## [6,]  700 1100 1500 1900  2300  2700
```

```
# Optionally convert to data frames for better visualization
plain_cheese_df <- as.data.frame(plain_cheese_profit_table)
colnames(plain_cheese_df) <- paste("Prod", plain_cheese_production, sep="_")
rownames(plain_cheese_df) <- paste("Demand", plain_cheese_demand, sep="_")
print("Plain Cheese Pizza Profit Table:")
```

```
## [1] "Plain Cheese Pizza Profit Table:"
```

```
print(plain_cheese_df)
```

```
##            Prod_200 Prod_300 Prod_400 Prod_500 Prod_600 Prod_700 Prod_800
## Demand_200      400      -50     -500     -950    -1400    -1850    -2300
## Demand_300      400      850      400      -50     -500     -950    -1400
## Demand_400      400      850     1300      850      400      -50     -500
## Demand_500      400      850     1300     1750     1300      850      400
## Demand_600      400      850     1300     1750     2200     1750     1300
## Demand_700      400      850     1300     1750     2200     2650     2200
## Demand_800      400      850     1300     1750     2200     2650     3100
## Demand_900      400      850     1300     1750     2200     2650     3100
##            Prod_900
## Demand_200    -2750
## Demand_300    -1850
## Demand_400     -950
## Demand_500      -50
## Demand_600      850
## Demand_700     1750
## Demand_800     2650
## Demand_900     3550
```

```
veggie_cheese_df <- as.data.frame(veggie_cheese_profit_table)
colnames(veggie_cheese_df) <- paste("Prod", veggie_cheese_production, sep="_")
rownames(veggie_cheese_df) <- paste("Demand", veggie_cheese_demand, sep="_")
print("Veggie-and-Cheese Pizza Profit Table:")
```

```
## [1] "Veggie-and-Cheese Pizza Profit Table:"
```

```
print(veggie_cheese_df)
```

```
##            Prod_300 Prod_400 Prod_500 Prod_600 Prod_700 Prod_800
## Demand_300      700      200     -300     -800    -1300    -1800
## Demand_400      700     1100      600      100     -400     -900
## Demand_500      700     1100     1500     1000      500        0
## Demand_600      700     1100     1500     1900     1400      900
## Demand_700      700     1100     1500     1900     2300     1800
## Demand_800      700     1100     1500     1900     2300     2700
```

b. Compute the expected profit associated with each possible production level (assuming Ken will only produce at one of the possible demand levels) for each type of pizza. Hint: This would be a vector of expected val- ues. You will need two such vectors of expected values– one for plain cheese and another for veggie-and-cheese pizza

```r
# Define the parameters
plain_cheese_demand <- c(200, 300, 400, 500, 600, 700, 800, 900)
plain_cheese_prob <- c(.1, .15, .15, .20, .20, .10, .05, .05)
veggie_cheese_demand <- c(300, 400, 500, 600, 700, 800)
veggie_cheese_prob <- c(.1, .2, .25, .25, .15, .05)
fixed_cost <- 1000
cost_cheese <- 4.50
cost_veggie_cheese <- 5
Price_pizza <- 9

# Calculate profit function
calc_profit <- function(demand, production, cost) {
  revenue <- pmin(demand, production) * Price_pizza
  cost_total <- production * cost + fixed_cost / 2
  profit <- revenue - cost_total
  return(profit)
}

# Create production levels
plain_cheese_production <- seq(200, 900, by=100)
veggie_cheese_production <- seq(300, 800, by=100)

# Calculate expected profit for plain cheese
plain_cheese_profit_table <- outer(plain_cheese_demand, plain_cheese_production, Vectori
ze(function(d, p) calc_profit(d, p, cost_cheese)))
expected_profit_cheese <- colSums(plain_cheese_profit_table * plain_cheese_prob)

# Calculate expected profit for veggie-and-cheese
veggie_cheese_profit_table <- outer(veggie_cheese_demand, veggie_cheese_production, Vect
orize(function(d, p) calc_profit(d, p, cost_veggie_cheese)))
expected_profit_veggie_cheese <- colSums(veggie_cheese_profit_table * veggie_cheese_pro
b)

# Print the expected profits
expected_profit_cheese
```

```
## [1]  400  760  985 1075  985  715  355  -50
```

```r
expected_profit_veggie_cheese
```

```
## [1]  700 1010 1140 1045  725  270
```

```r
# Optionally create data frames for better visualization
expected_profit_cheese_df <- data.frame(Production=plain_cheese_production, ExpectedProf
it=expected_profit_cheese)
expected_profit_veggie_cheese_df <- data.frame(Production=veggie_cheese_production, Expe
ctedProfit=expected_profit_veggie_cheese)

print("Expected Profit for Plain Cheese Pizza:")
```

```
## [1] "Expected Profit for Plain Cheese Pizza:"
```

```
print(expected_profit_cheese_df)
```

```
##   Production ExpectedProfit
## 1        200            400
## 2        300            760
## 3        400            985
## 4        500           1075
## 5        600            985
## 6        700            715
## 7        800            355
## 8        900            -50
```

```
print("Expected Profit for Veggie-and-Cheese Pizza:")
```

```
## [1] "Expected Profit for Veggie-and-Cheese Pizza:"
```

```
print(expected_profit_veggie_cheese_df)
```

```
##   Production ExpectedProfit
## 1        300            700
## 2        400           1010
## 3        500           1140
## 4        600           1045
## 5        700            725
## 6        800            270
```

c. If Ken wants to maximize the expected profit from pizza sales at the game, then how many of each type of pizza should he produce? Hint: The answer to this question is based on the results of part 2 above. **Ken should produce 500 units of both types of pizza**

Question 10

The government of country B pegged the value of their currency, called pesos, to the currency of government A, called dollars, at 16 pesos per dollar. Because interest rates in country B were higher than those in the country A, many investors (including banks) bought bonds in country B to earn higher returns than were available in the country A. The benefits of the higher interest rates, however, masked the possibility that the peso would be allowed to float and lose substantial value compared to the dollar. Suppose you are an investor and believe that the probability of the exchange rate for the next year remains at 16 pesos per dollar is 0.6, but that the rate could soar to 32 per dollar with probability 0.4.

a. Suppose you are a resident of country A. Consider two invest- ments: Deposit $6,000 today in a savings account in country A that pays 12% annual interest or deposit $6,000 in an account in country B that pays 24% interest. Please note that the latter requires currency conversion. Which choice has a higher expected value in one year?

```r
principal <- 6000
intA <- 0.12
intB <- 0.24
price1 <- 16
price2 <- 32
prob1 <- 0.6
prob2 <- 0.4

# Investment return in country A
expected_return_a <- principal * (1 + intA)

# Investment return in country B
final_amount_b <- principal * (1 + intB)

# Convert back to dollars considering possible exchange rates
return_b1 <- final_amount_b / price1
return_b2 <- final_amount_b / price2

# Expected value of the investment in country B
expected_value_b <- (return_b1 * prob1) + (return_b2 * prob2)

# Compare expected values
expected_return_a
```

```
## [1] 6720
```

```r
expected_value_b
```

```
## [1] 372
```

Given this calculation, the investment in country A has a significantly higher expected value compared to the investment in country B. Therefore, depositing $6,000 in a savings account in country A would be the better choice.

b. Now suppose you are a resident of country B with 96,000 pesos to invest. You can convert these pesos to dollars, deposit in an account in country A and collect 12% interest, and then convert them back at the end of the year, or you can get 24% from a local investment. Compare the expected value in pesos of each of these investments. Which looks better?

```
# Initial amount in pesos
partb_principal <- 96000

# Exchange rates and interest rates
intA <- 0.12
intB <- 0.24
price1 <- 16
price2 <- 32
prob1 <- 0.6
prob2 <- 0.4

# Investment return in country B (local investment)
partb_expected_return_b <- partb_principal * (1 + intB)

# Investment return in country A (foreign investment)
# Convert pesos to dollars first
dollars_invested <- partb_principal / price1

# Apply interest rate in country A
dollars_after_interest <- dollars_invested * (1 + intA)

# Convert back to pesos considering possible exchange rates
pesos_returned1 <- dollars_after_interest * price1
pesos_returned2 <- dollars_after_interest * price2

# Expected value of the investment in country A
expected_value_a <- (pesos_returned1 * prob1) + (pesos_returned2 * prob2)

# Print the results
partb_expected_return_b
```

```
## [1] 119040
```

```
expected_value_a
```

```
## [1] 150528
```

Given this calculation, the investment in country A (converting pesos to dollars, earning 12% interest, and then converting back) has a higher expected value compared to the investment in country B. Therefore, depositing 96,000 pesos in a foreign account in country A would be the better choice.

c. Explain the difference in strategies that obtain the higher expected value.

the strategy of investing in country A with initial conversion to dollars and subsequent conversion back to pesos leverages potential exchange rate gains, which can outweigh the higher interest rate offered in country B when considering the risks associated with exchange rate fluctuations. This approach aims to optimize the expected value by balancing interest rate differentials and exchange rate movements, leading to a higher expected value compared to the straightforward investment in country B's higher interest rate environment.