Practice Session 2 Answers

Part 1 : Measures of central tendency & Measures of spread for quantitative data

1.1 Calculating the Sample Standard Deviation by Hand

Here is the formula for the sample mean:

$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$

And here is the formula for the sample standard deviation:

$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}$

Using the above data, perform the following calculations.

Complete the table for calculating the sample standard deviation.

Cost $$$	b. Deviations $(x_i - \bar{x})$	c. Deviations squared $(x_i - \bar{x})^2$
850		
900		
1400		
1200		
1050		
750		
1250		
1050		
565		
1000		
a. mean = ________		

d. Sum of squared deviations $\sum_{i=1}^{n}(x_i - \bar{x})^2$ = ______

e. Sum of squared deviations divided by n - 1: $\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$ = ______

f. Take the square root to get s: $\sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$ = ______ = s

1.2 Five-Number Summary

Using the numbers from the previous exercise, do the following:

- Find the 5-number summary (minimum, Q1, median, Q3, maximum)
- Check your work using R functions

Answers

```r
v<- c( 565, 750, 850, 900, 1000, 1050, 1050, 1200, 1250, 1400 )
v
```

```
 [1]  565  750  850  900 1000 1050 1050 1200 1250 1400
```

```r
s<- sd(v)
s
```

```
[1] 247.5889
```

```r
## you can use function fivenum()

fivenum(v)
```

```
[1]  565  850 1025 1200 1400
```

```r
## you can use function summary()

summary(v)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  565.0   862.5  1025.0  1001.5  1162.5  1400.0 
```

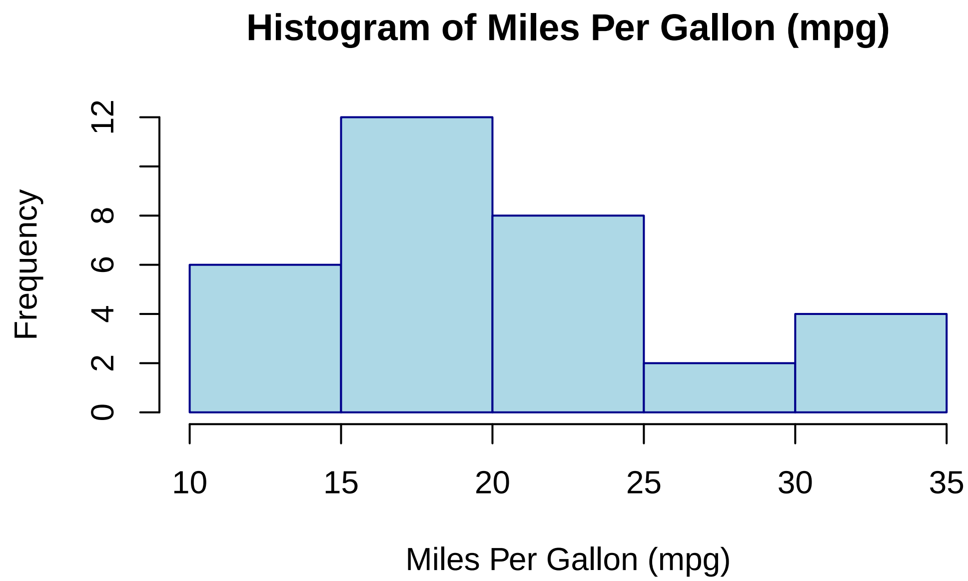
1.3 Boxplots

Consider the `mtcars` data set. This data is built into R, so you can access it directly; no downloads required! First, create a histogram of the variable `mpg`. Then create a boxplot of `mpg`.

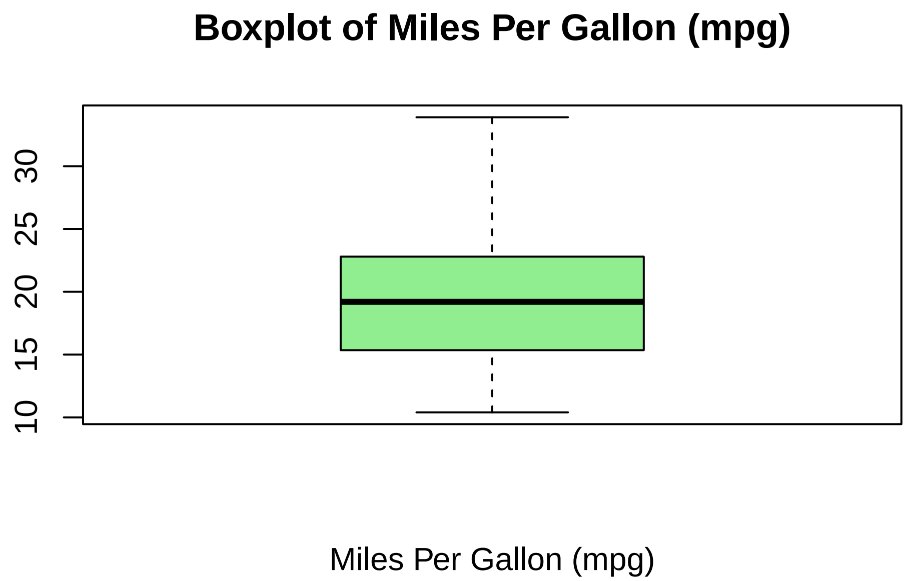
1. How do these two plots compare?
2. Create a boxplots of `mpg` per number of cylinders `cyl`.

Answer

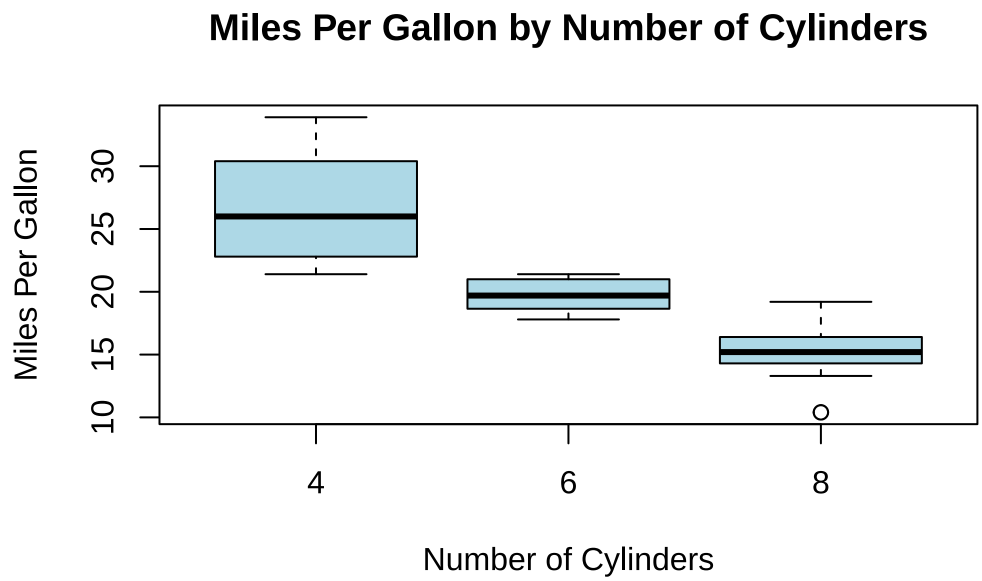
```r
# Create a histogram of 'mpg'

hist(mtcars$mpg,
     main = "Histogram of Miles Per Gallon (mpg)",
     xlab = "Miles Per Gallon (mpg)",
     ylab = "Frequency",
     col = "lightblue",
     border = "darkblue")
```




```
# Create a boxplot of 'mpg':

boxplot(mtcars$mpg,
        main = "Boxplot of Miles Per Gallon (mpg)",
        xlab = "Miles Per Gallon (mpg)",
               col = "lightgreen")
```




```
# create boxplot of mpg per cylender:

# Ensure 'cyl' is treated as a factor (categorical variable) for grouping
mtcars$cyl <- as.factor(mtcars$cyl)

# Create the boxplot
boxplot(mpg ~ cyl, data = mtcars,
        main = "Miles Per Gallon by Number of Cylinders",
        xlab = "Number of Cylinders",
        ylab = "Miles Per Gallon",
        col = "lightblue")
```



1.4 Quantitative data : histograms and outliers

Generate histograms for each of the following data sets. Use the `$` command to access the individual data sets. For each histogram, add the mean to the plot using `abline()`. Do you see any potential outliers? Also calculate the five-number summary for each using R.

```
set.seed(999)
s2_data = data.frame(
  dat1 = -rchisq(1000, df = 1),
  dat2 = rchisq(1000, df = 1),
  dat3 = runif(1000),
  dat4 = rnorm(1000),
  dat5 = sample(c(rnorm(1000, mean = 2), rnorm(1000, mean = 10)), size = 1000)
)
```

Answers:

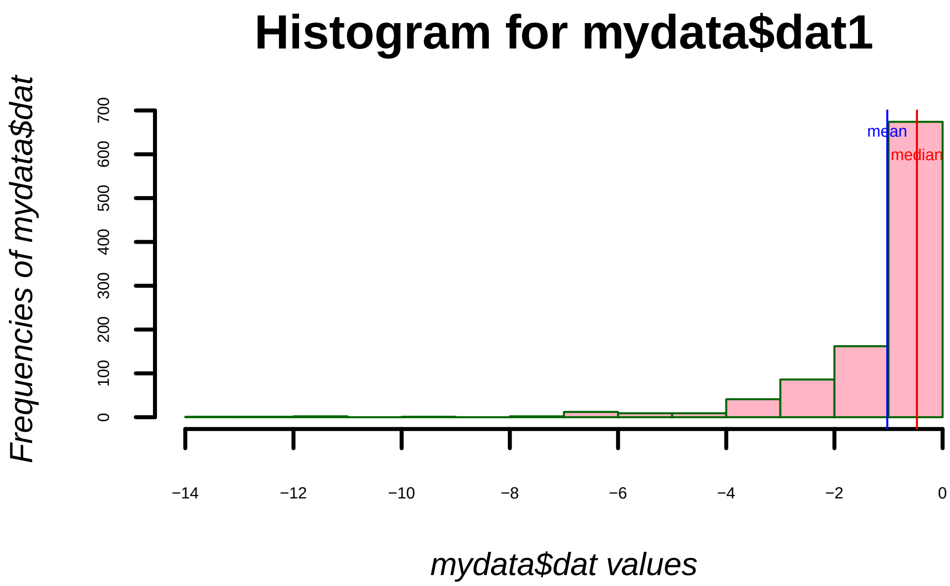
```
mean1 <- mean (s2_data$ dat1 )
mean1
```

```
[1] -1.022716
```

```
median1 <-  median (s2_data$ dat1 )
median1
```

```
[1] -0.4746566
```

```
hist1<- hist( s2_data$ dat1,

     col = "pink1",                                # bins color
     main = "Histogram for mydata$dat1",           # title
     xlab = "mydata$dat values",                   # x-axis label
     ylab = "Frequencies of mydata$dat",           # y-axis label
     border = "darkgreen",                         # bins border's color
     lwd = 2,                                      # border thickness
     cex.main = 1.5,                               # title size
     cex.lab = 1,                                  # axis labels size
     cex.axis = 0.5,                               # tick labels size
     font.main = 2,                                # bold title
     font.lab = 3)                                 # italic axis labels


abline(v= mean1, col="blue" )
abline(v= median1, col="red2" )

text(x = median1, y = 600,  cex= 0.5, labels = "median",
     adj = 0.5, col = "red") # Add label slightly above the line

text(x = mean1, y = 650,  cex= 0.5, labels = "mean",
     adj = 0.5, col = "blue") # Add label slightly above the line
```



1.5 Percentiles

Compute the 25th, 50th, and 75th percentile for the 5 data sets in the `s2_data` data.frame. Which has the smallest median? Which as the largest?

Answers:

```
percentiles <- quantile(s2_data$dat1 , c ( 0.25, 0.5,  0.75))
percentiles
```

```
       25%        50%        75% 
-1.3871368 -0.4746566 -0.1150274 
```

“

1.6 Normal Distribution and +/- 2 Standard Deviations

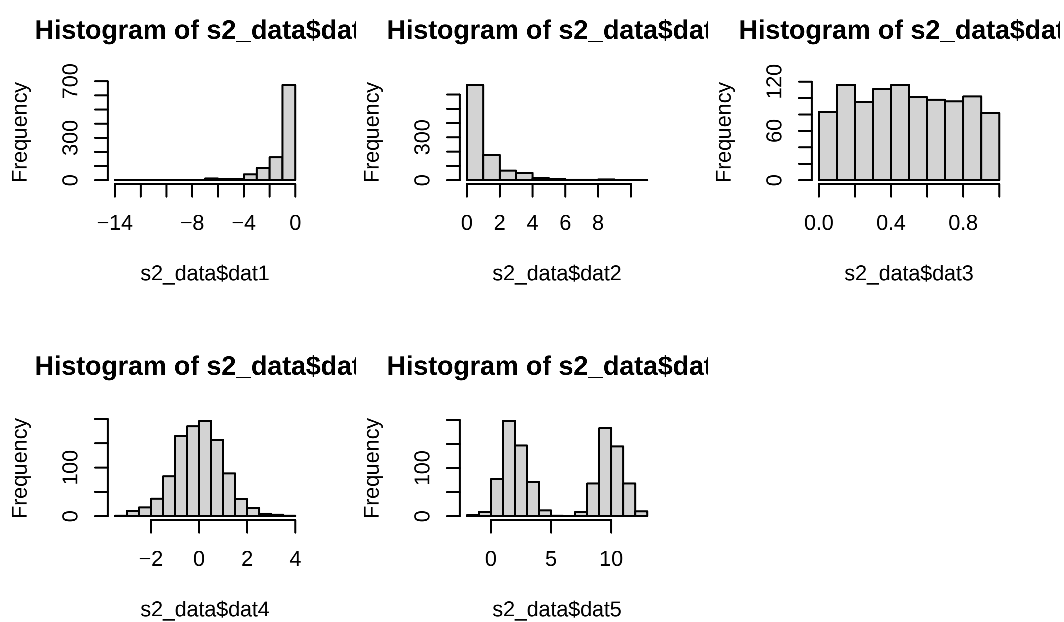
The normal distribution (also known as the "bell-curve") occurs very frequently in mathematics, statistics, and the natural and social sciences. Which of the 5 data sets in the `s2_data` data.frame appears to be normally distributed?

Using this data set, find the mean and standard deviation, then calculate 1 standard deviations above, and 1 standard deviations below the mean. What percentiles do these values correspond to?

Note: approximately, there is **68%** of data between **1 standard deviations above**, and **1 standard deviations below** the mean, in the normal distribution.

Answers:

```
par(mfrow=c(2,3))

hist1<- hist( s2_data$ dat1)
  hist2<- hist( s2_data$ dat2)
  hist3<- hist(s2_data$ dat3)
  hist4<- hist( s2_data$ dat4)
  hist5<- hist( s2_data$ dat5)
```



The data `s2_data$dat4` has normal distribution. So, we expect that 68% of data between 1 standard deviations above, and 1 standard deviations below the mean, in the normal distribution.

```r
## method 1 : using the formula
mean4 <- mean (s2_data$ dat4 )
mean4
```

```
[1] -0.01332436
```

```r
sd4 <- sd (s2_data$ dat4 )
sd4
```

```
[1] 1.003102
```

```r
endpoints_1sd_below_above_mean <- c( mean4- sd4, mean4+sd4)
endpoints_1sd_below_above_mean
```

```
[1] -1.0164261  0.9897774
```

```r
## method 2 : using the percentiles 16% and 84 % to represent the end points of middle 68% o

q<- quantile(s2_data$dat4, c(0.16, 0.84) )
q
```

```
       16%        84%
-0.9581738  0.9486344
```

1.7 Z-Scores

Read the following description on Z-scores, then answer the question below.

5.1 Standardizing with *z*-Scores

Expressing a distance from the mean in standard deviations *standardizes* the performances. To **standardize** a value, we subtract the mean and then divide this difference by the standard deviation:

$$z = \frac{y - \bar{y}}{s}$$

NOTATION ALERT

We always use the letter *z* to denote values that have been standardized with the mean and standard deviation.

The values are called **standardized values**, and are commonly denoted with the letter *z*. Usually we just call them ***z*-scores**.

z-scores measure the distance of a value from the mean in standard deviations. A *z*-score of 2 says that a data value is two standard deviations above the mean. It doesn't matter whether the original variable was measured in fathoms, dollars, or carats; those units don't apply to *z*-scores. Data values below the mean have negative *z*-scores, so a *z*-score of −1.6 means that the data value was 1.6 standard deviations below the mean. Of course, regardless of the direction, the farther a data value is from the mean, the more unusual it is, so a *z*-score of −1.3 is more extraordinary than a *z*-score of 1.2.

15. Temperatures A town's January high temperatures average 2°C with a standard deviation of 6°, while in July the mean high temperature is 24° and the standard deviation is 5°. In which month is it more unusual to have a day with a high temperature of 13°? Explain.

```
## Januay z-score_jan

jan_z_score <- (13-2)/ 6
jan_z_score
```

```
[1] 1.833333
```

```
## July z-score

jul_z_score <- (13-24)/ 5
jul_z_score
```

```
[1] -2.2
```

Part 2 : The Relationship Between Two Quantitative Variables/ Correlation and Regression

You might use the formula of the correlation between two quantitative variables:

$r_{xy} = \frac{1}{(n-1)s_z s_y} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$

Remember that the fitted regression line is defined by the equation:

- $\hat{y} = a + bx$, or
- $\widehat{\text{Response}} = a + b \cdot \text{(Explanatory)}$
- Residuals = observed - predicted = y - $\hat{y}$

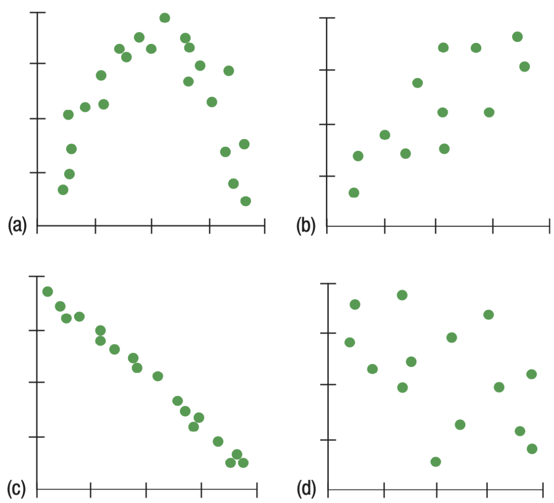
Where:

- Response: is the response variable or the dependent variable
- Explanatory: is the independent variable
- a: is the y-intercept
- b: is the slope of the regression line

You may use the following R functions: `plot()`, `lm()`, `cor()`, `abline()`. And you might need to download Lock5Data using `library(Lock5Data)`.

2.1 Describe scatterplots

Here are several scatterplots. The calculated correlations are 0.006, - 0.977, - 0.487, and 0.777. Match each scatter plot with the appropriate correlation coeffecient.



Answers:

a. 0.006

b. 0.777

c. • 0.977

d. • 0.487

2.2 Create scatterplots in R

Load the data `FloridaLakes` from library(Lock5Data).

1. Describe the type of each of the variables `pH` ,`Calcium`, and `Alkalinity`.
2. Create a three scatter plots for each pair of the variables: `pH` vs `Calcium` ,`pH` vs `Alkalinity` , and `Calcium` vs `Alkalinity` . Add the main title to each plot.
3. What is the correlation coefficient between `pH` and `Calcium`. Is it positive or negative?
4. What do these coefficients mean in the context of this data ?
5. Try to calculate the correlation coefficient between pH and Calcium without using `R` function.

Answers:

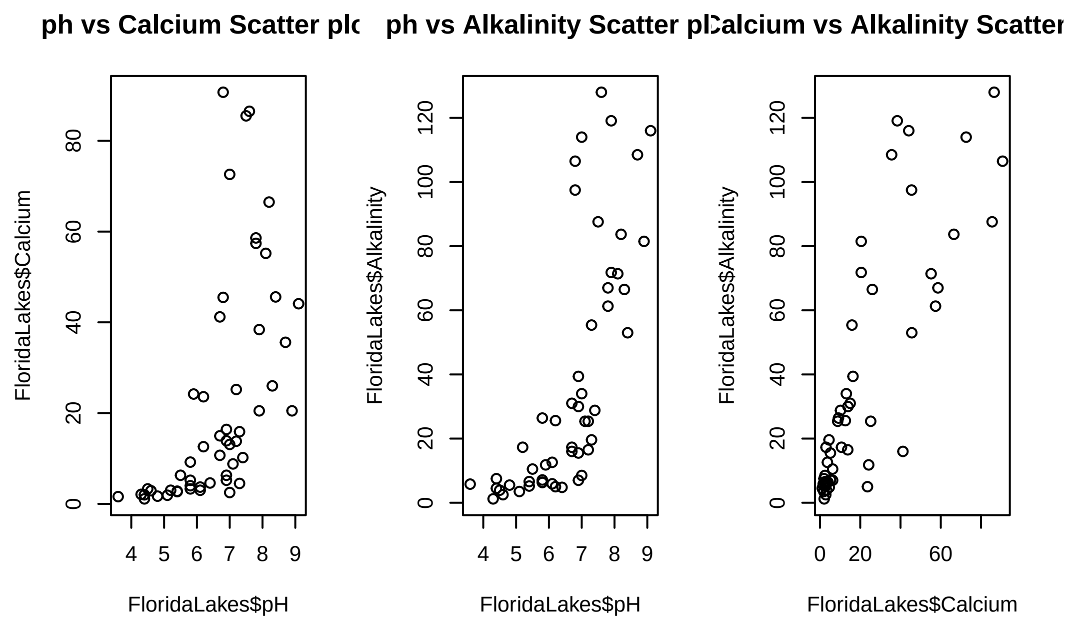
```
# download the data and load it into R
library(Lock5Data)
data(FloridaLakes)

# Describe the structure of the variables of FloridaLake
#str(FloridaLakes)
```

1. Describe the type of each of the variables `pH` and `Calcium`, `Alkalinity`.

The variables `pH` and `Calcium`, `Alkalinity` are quantitative variables (continuous)

2.Create a scatter plots for each of the pair of variables (add the main title of the plot).

```
# Create scatter pH vs  Calcium
par(mfrow= c(1,3))

plot(FloridaLakes$pH,
     FloridaLakes$Calcium,
     main = "ph vs Calcium Scatter plot")

# Create scatter plot pH vs  Alkalinity

plot(FloridaLakes$pH,
     FloridaLakes$Alkalinity ,
     main = "ph vs Alkalinity Scatter plot")

# Create scatter plot Calcium vs  Alkalinity

plot(FloridaLakes$Calcium, FloridaLakes$Alkalinity,
    main = "Calcium vs Alkalinity Scatter plot")
```



3. What is the correlation coefficient between `pH` and `Calcium`. Is it positive or negative.

```
# correlation

cor_pH_Cal <- cor(FloridaLakes$pH, FloridaLakes$Calcium)

cor_pH_Alk <- cor(FloridaLakes$pH, FloridaLakes$Alkalinity)

cor_Cal_Alk <- cor(FloridaLakes$Calcium, FloridaLakes$Alkalinity)

cor_pH_Cal
```

```
[1] 0.5771327
```

```
cor_pH_Alk
```

```
[1] 0.7191657
```

```
cor_Cal_Alk
```

```
[1] 0.8326042
```

4. What do these coefficients mean in the context of this data.

A correlation coefficient of 0.577 between `pH`and `Calcium` means that there is a linear positive association of moderate strength of 0.577.

5.calculate the correlation coefficient between `pH` and `Calcium` without using `R` function.

```
## Method 2 to calculate the correlation using the formula

n<- length(FloridaLakes$pH)

X<-FloridaLakes$pH

Y<- FloridaLakes$Calcium


cor_pH_Cal<- sum((X - mean(X))*(Y-mean(Y)))*(1/(n - 1))*1/sd(X)*1/sd(Y)

cor_pH_Cal
```

```
[1] 0.5771327
```