DATA ANALYTICS – 4027

LAB-10

Name: Hari Krishna P

Reg No: 19BCE7675

DATE: 13/11/2021

Contents:

- **Outlier detection**
- **Ggstatplot (Dataframes: mtcars,swiss,longley)**
- **Regression Model**

Submitted to:

Prof . Hari Seetha

Outlier Detection:
Refer

1. https://statsandr.com/blog/outliers-detection-in-r/

2. https://www.journaldev.com/47986/outlier-analysis-in-r

Ex-10

1. install the package ggstatsplot.

```
package 'ggstatsplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\HARIKRISHNA\AppData\Local\Temp\RtmpWcEjCe\downloaded_packages
> |
```

2. Load the package

```
> library(ggstatsplot)
You can cite this package as:
     Patil, I. (2021). Visualizations with statistical details: The 'ggstatsplot' approach.
     Journal of Open Source Software, 6(61), 3167, doi:10.21105/joss.03167
> |
```

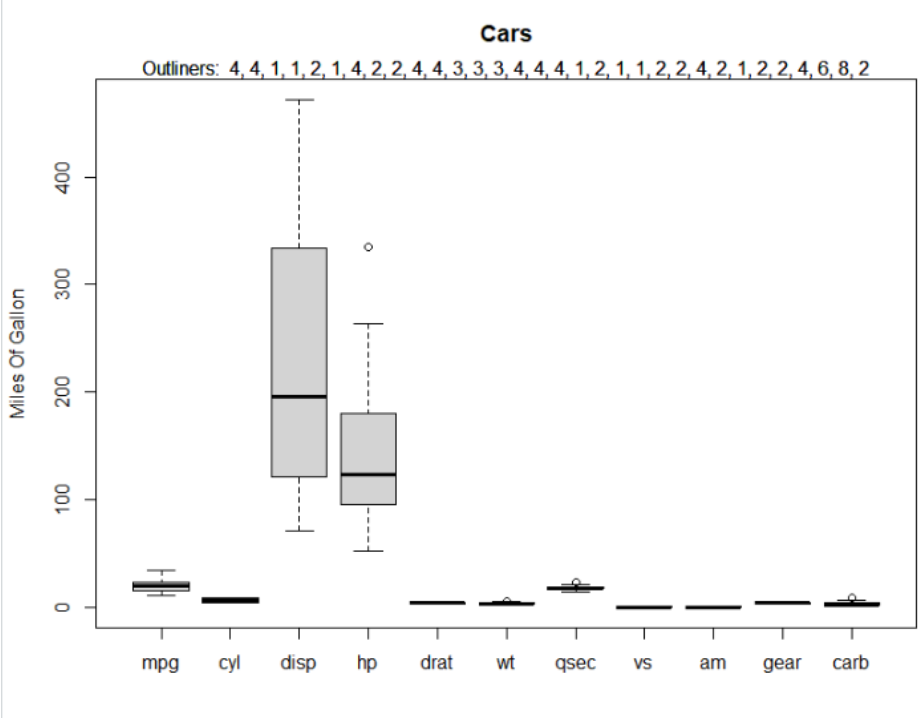
3. Load the dataset

attach(mtcars)

4. Create a boxplot of the dataset, outliers are shown as two distinct points

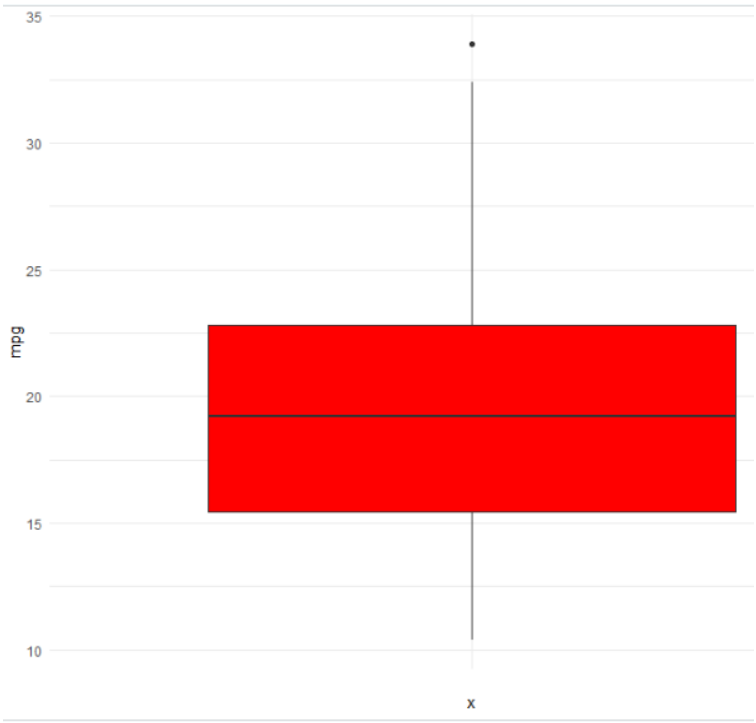
```
boxplot(mtcars,
        ylab = "Miles Of Gallon",
```

```
      main = "Cars"
)
mtext(paste("Outliners: ",paste(carb,collapse = ", ")))
```



5. Create a boxplot that labels the outliers

```
library(ggplot2)
 ggplot(mtcars)+
  + aes(x = "", y =mpg)+
  + geom_boxplot(fill = "RED")+
  + theme_minimal()
```



6. Use the quantile() function to find the 25th and the 75th percentile of the dataset, and the IQR() function which gives the difference of the 75th and 25th percentiles.Find the cut-off ranges beyond which all data points are outliers.

```
> outliner <- which(mtcars$mpg < lower|mtcars$mpg>upper)
> outliner
 [1]  7  8 14 15 16 17 18 19 20 23 24 26 27 28 31
```

7. Save the outliers in a vector

```
> outliner <- which(mtcars$mpg < lower|mtcars$mpg>upper)
> outliner
 [1]  7  8 14 15 16 17 18 19 20 23 24 26 27 28 31
```

8. Remove outliers

```
> boxplot(mtcars, plot = FALSE)$out
[1] 335.000   5.424   5.345  22.900   8.000
>
```

9. Show the boxplot without outliers

```
> boxplot(mtcars,plot = FALSE)
$stats
      [,1] [,2]   [,3] [,4]  [,5]   [,6]   [,7] [,8] [,9] [,10] [,11]
[1,] 10.40    4  71.10   52 2.760 1.5130 14.500    0    0     3     1
[2,] 15.35    4 120.65   96 3.080 2.5425 16.885    0    0     3     2
[3,] 19.20    6 196.30  123 3.695 3.3250 17.710    0    0     4     2
[4,] 22.80    8 334.00  180 3.920 3.6500 18.900    1    1     4     4
[5,] 33.90    8 472.00  264 4.930 5.2500 20.220    1    1     5     6

$n
 [1] 32 32 32 32 32 32 32 32 32 32 32

$conf
         [,1]     [,2]     [,3]     [,4]     [,5]     [,6]    [,7]       [,8]       [,9]    [,10]    [,11]
[1,] 17.11916 4.882771 136.7098  99.5382 3.460382 3.015667 17.1472 -0.2793072 -0.2793072 3.720693 1.441386
[2,] 21.28084 7.117229 255.8902 146.4618 3.929618 3.634333 18.2728  0.2793072  0.2793072 4.279307 2.558614

$out
[1] 335.000   5.424   5.345  22.900   8.000

$group
[1]  4  6  6  7 11

$names
 [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear" "carb"
```

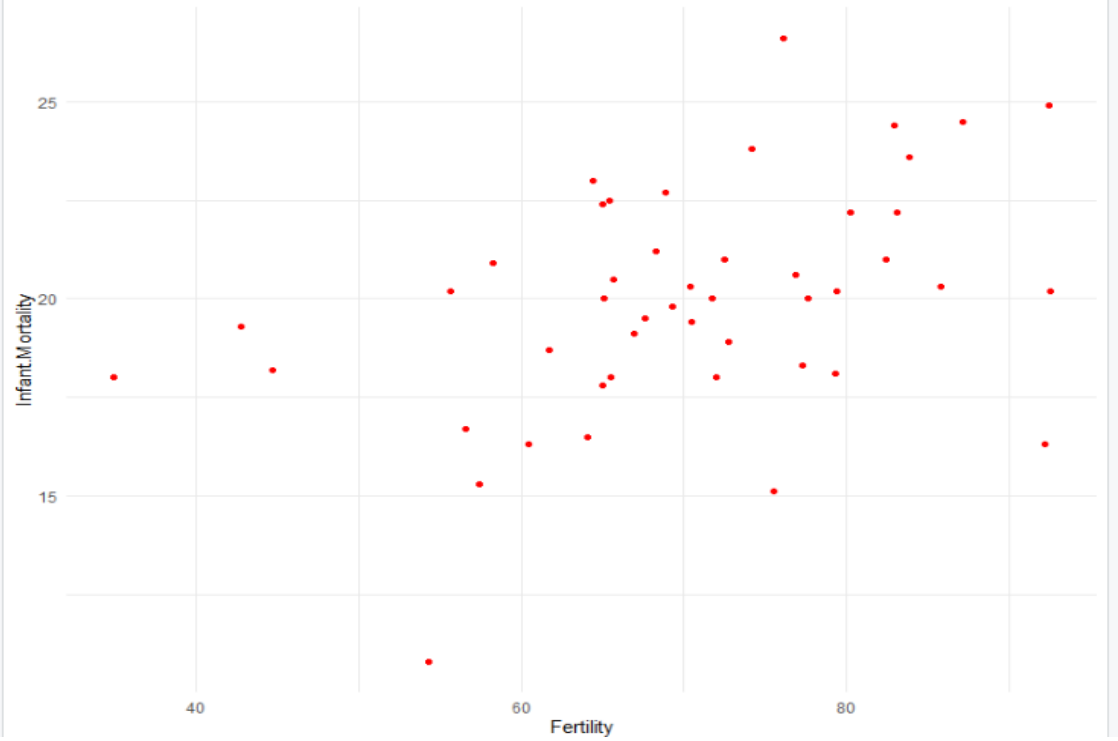
Correlation Analysis in R
Refer

1. https://statsandr.com/blog/correlation-coefficient-and-correlation-test-in-r/

    1. Load Swiss data
       attach(swiss)

    2. Creating a scatter plot using g Fertility on X-axis and Infant_Mortality on Y-axis and also check whetherthey have linear relationship

```
ggplot(swiss) +
    aes(x = Fertility, y =Infant.Mortality ) +
    geom_point(colour = "RED") +
    theme_minimal()
```

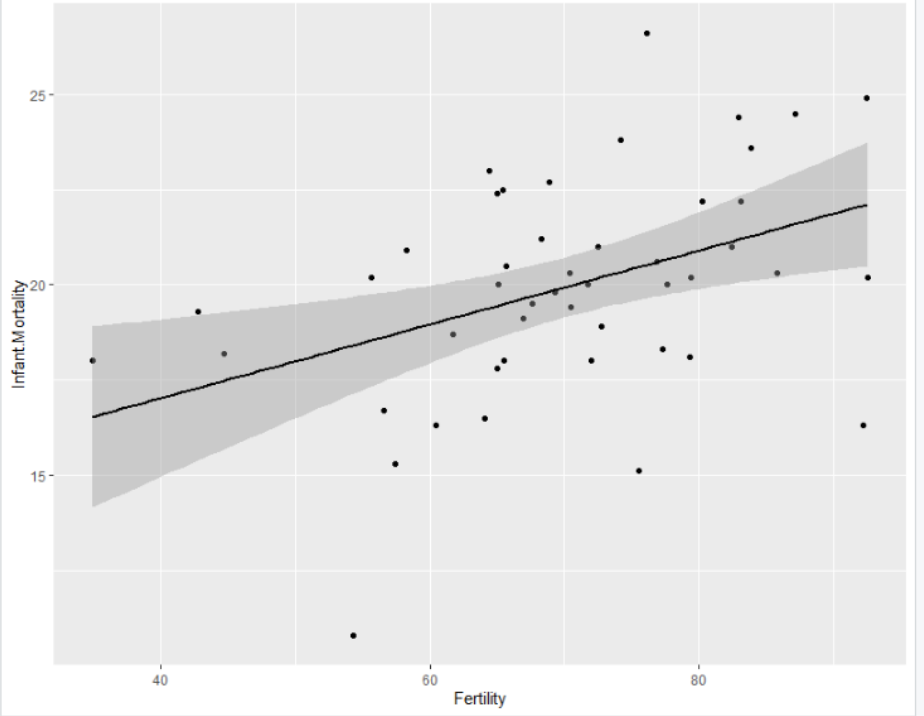


3. Test for normality using Shapiro Test
   shapiro.test(swiss$Infant.Mortality)

```
        Shapiro-Wilk normality test

data:  swiss$Infant.Mortality
W = 0.97762, p-value = 0.4978
```

4. Find the correlation between Fertility and Infant_Mortality
   And test for significance using Pearson,Kendall,Spearman's correlation methods

```
> cor(swiss$Fertility,swiss$Infant.Mortality)
[1] 0.416556
> dataset1<- cor.test(swiss$Fertility,swiss$Infant.Mortality,method = "pearson")
> dataset1

        Pearson's product-moment correlation

data:  swiss$Fertility and swiss$Infant.Mortality
t = 3.0737, df = 45, p-value = 0.003585
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.1469699 0.6285366
sample estimates:
      cor
0.416556
```



Regression in R:

Refer: https://www.learnbymarketing.com/tutorials/linear-regression-in-r/
https://www.tutorialspoint.com/r/r_linear_regression.htm

1.Construct a Regression Model using Longley dataset and perform analysis using various regression methods and comment on the observations.

```
> attach(longley)
> View(longley)
> x1 <- lm(Employed~.,longley)
> x1

Call:
lm(formula = Employed ~ ., data = longley)

Coefficients:
 (Intercept)  GNP.deflator           GNP    Unemployed  Armed.Forces    Population          Year
  -3.482e+03     1.506e-02    -3.582e-02    -2.020e-02    -1.033e-02    -5.110e-02     1.829e+00
```

```
> summary(x1)

Call:
lm(formula = Employed ~ ., data = longley)

Residuals:
     Min       1Q   Median       3Q      Max
-0.41011 -0.15767 -0.02816  0.10155  0.45539

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.482e+03  8.904e+02  -3.911 0.003560 **
GNP.deflator  1.506e-02  8.492e-02   0.177 0.863141
GNP          -3.582e-02  3.349e-02  -1.070 0.312681
Unemployed   -2.020e-02  4.884e-03  -4.136 0.002535 **
Armed.Forces -1.033e-02  2.143e-03  -4.822 0.000944 ***
Population   -5.110e-02  2.261e-01  -0.226 0.826212
Year          1.829e+00  4.555e-01   4.016 0.003037 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3049 on 9 degrees of freedom
Multiple R-squared:  0.9955,    Adjusted R-squared:  0.9925
F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10


> prediction<-predict(x1, longley)
> prediction
    1947     1948     1949     1950     1951     1952     1953     1954     1955     1956     1957     1958     1959     1960     1961     1962
60.05566 61.21601 60.12471 61.59711 62.91129 63.88831 65.15305 63.77418 66.00470 67.40161 68.18627 66.55206 68.81055 69.64967 68.98907 70.75776
> |


> x3<-mean((longley$Employed - prediction)^2)
> print(x3)
[1] 0.0522765
> |
```

```
> step(x1)
Start:  AIC=-33.22
Employed ~ GNP.deflator + GNP + Unemployed + Armed.Forces + Popula
    Year

               Df Sum of Sq     RSS     AIC
- GNP.deflator  1   0.00292 0.83935 -35.163
- Population    1   0.00475 0.84117 -35.129
- GNP           1   0.10631 0.94273 -33.305
<none>                     0.83642 -33.219
- Year          1   1.49881 2.33524 -18.792
- Unemployed    1   1.59014 2.42656 -18.178
- Armed.Forces  1   2.16091 2.99733 -14.798

Step:  AIC=-35.16
Employed ~ GNP + Unemployed + Armed.Forces + Population + Year

               Df Sum of Sq    RSS     AIC
- Population    1   0.01933 0.8587 -36.799
<none>                     0.8393 -35.163
- GNP           1   0.14637 0.9857 -34.592
- Year          1   1.52725 2.3666 -20.578
- Unemployed    1   2.18989 3.0292 -16.628
- Armed.Forces  1   2.39752 3.2369 -15.568

Step:  AIC=-36.8
Employed ~ GNP + Unemployed + Armed.Forces + Year

               Df Sum of Sq    RSS     AIC
<none>                     0.8587 -36.799
- GNP           1    0.4647 1.3234 -31.879
- Year          1    1.8980 2.7567 -20.137
- Armed.Forces  1    2.3806 3.2393 -17.556
- Unemployed    1    4.0491 4.9077 -10.908

Call:
lm(formula = Employed ~ GNP + Unemployed + Armed.Forces + Year,
```