DATA ANALYTICS AND DATA VISUALIZATION WITH TABLEAU

Assignment : Day-1

PES2UG19CS315

Rahul S Bhat

7th semester

1. Check whether the dataset in gen.csv is monotonic and find correlation using the same(spearman/Pearson).

In [1]:
```
import pandas as pd
df=pd.read_csv("C:/Users/Rahul Bhat/Downloads/gen.csv")
df.head()
```

Out[1]:

	1980	Jan	22.13	23.5	5.87	mandya.xls	0.27
0	1980	Feb	24.16	22.39	6.52	mandya.xls	2.69
1	1980	Mar	26.04	24.43	7.21	mandya.xls	30.48
2	1980	Apr	27.03	36.90	7.26	mandya.xls	12.83
3	1980	May	26.60	45.12	6.88	mandya.xls	116.82
4	1980	Jun	23.09	75.47	5.29	mandya.xls	416.76

In [2]:
```
for i in df:
    print(df[i].is_monotonic)
```

```
False
False
False
False
False
False
False
```

In [3]:
```
df.corr(method='pearson')
```

Out[3]:

	1980	22.13	23.5	5.87	0.27
1980	1.000000	-0.006252	0.012344	0.008500	0.012212
22.13	-0.006252	1.000000	-0.266426	0.646530	-0.141930
23.5	0.012344	-0.266426	1.000000	-0.675761	0.687188
5.87	0.008500	0.646530	-0.675761	1.000000	-0.526892
0.27	0.012212	-0.141930	0.687188	-0.526892	1.000000

2.use the explorer tool in weka and justify MCC Kappa statastic ROC curve values for dataset using any algorithm.

Classifier output

```
               Class
Test mode:     10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Time taken to build model: 0.09 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         199               69.5804 %
Incorrectly Classified Instances        87               30.4196 %
Kappa statistic                          0.1736
Mean absolute error                      0.3727
Root mean squared error                  0.4613
Relative absolute error                 89.0857 %
Root relative squared error            100.9171 %
Total Number of Instances              286

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.871     0.718     0.742      0.871    0.801      0.184    0.634     0.798     no-recurrence-events
                 0.282     0.129     0.480      0.282    0.356      0.184    0.634     0.409     recurrence-events
Weighted Avg.    0.696     0.543     0.664      0.696    0.669      0.184    0.634     0.682

=== Confusion Matrix ===

   a   b   <-- classified as
 175  26 |   a = no-recurrence-events
  61  24 |   b = recurrence-events
```

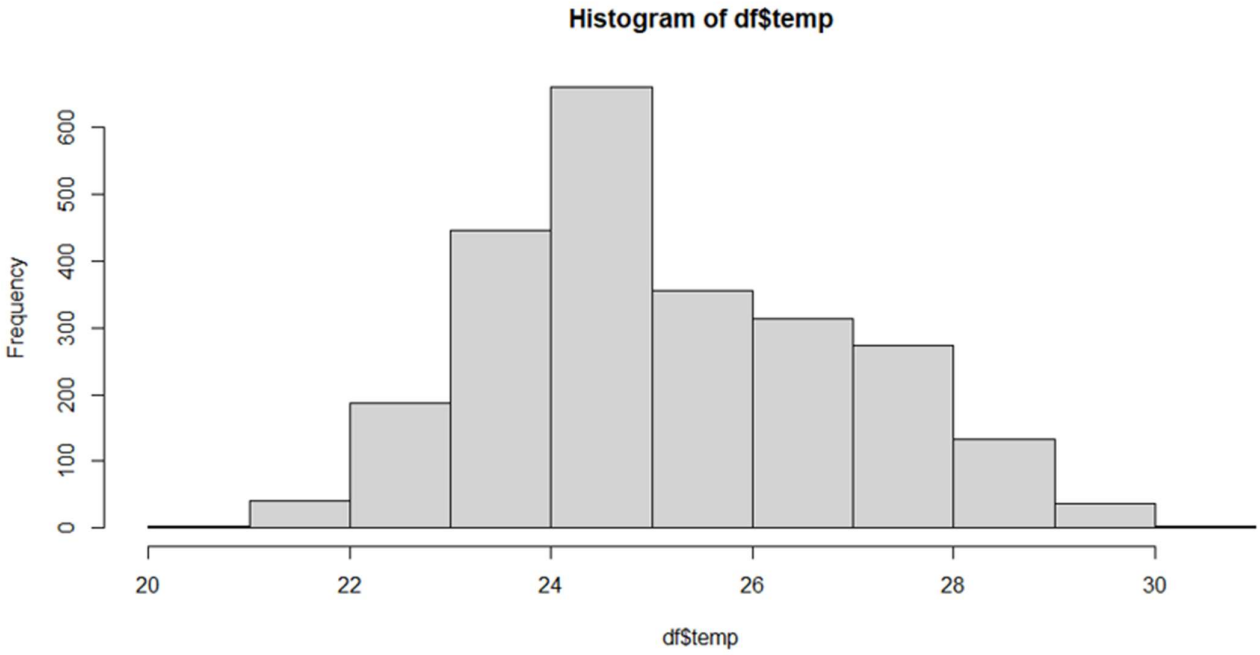
3.calculate mean,median and mode for col rainfall ,temp,VP,PET in R

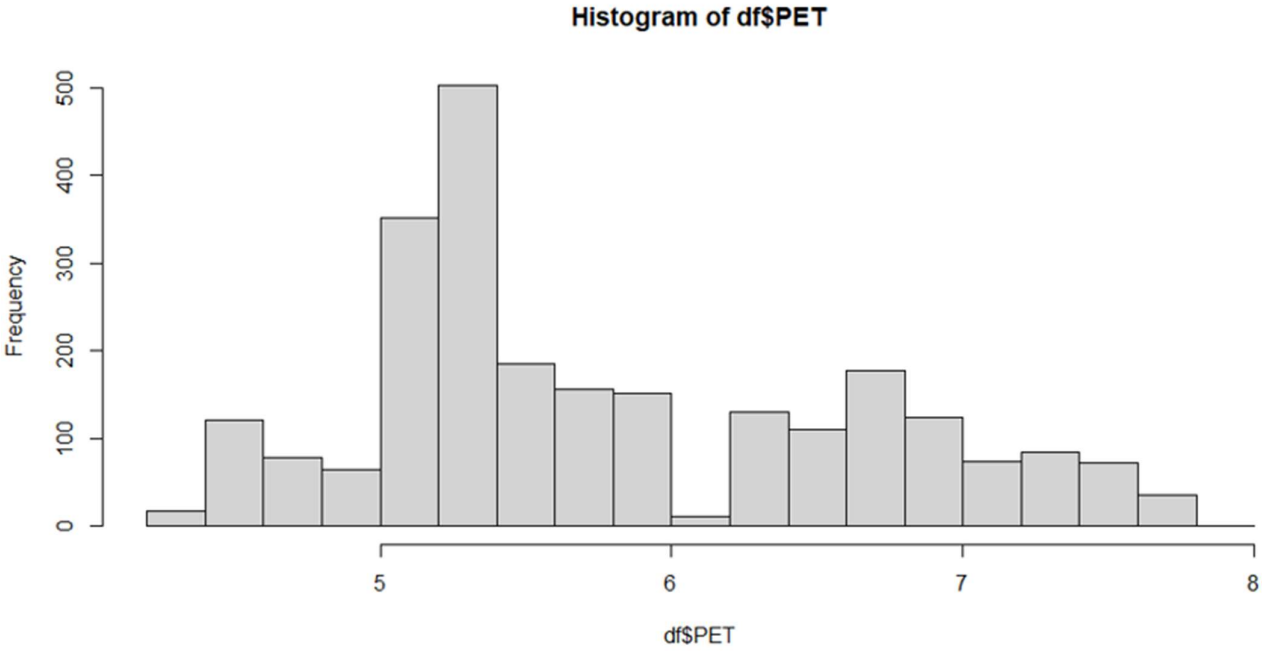
```
df=read.csv("C:/Users/Rahul Bhat/Downloads/gen1.csv")
summary(df)

find_mode <- function(x) {
  u <- unique(x)
  tab <- tabulate(match(x, u))
  u[tab == max(tab)]
}

find_mode(df$temp)
find_mode(df$vp)
find_mode(df$PET)
find_mode(df$rainfall)
```

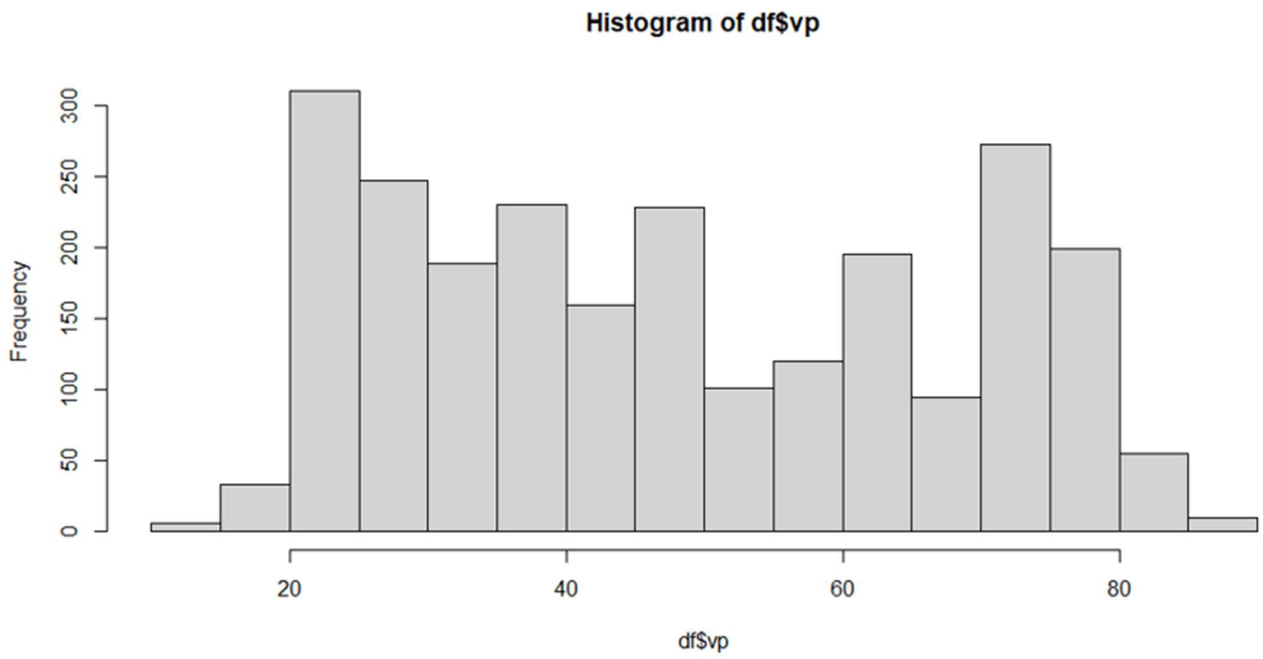
```
> df=read.csv("C:/Users/Rahul Bhat/Downloads/gen1.csv")
> summary(df)
      temp             vp             PET            rainfall      
 Min.   :20.70   Min.   :10.98   Min.   :4.310   Min.   :   0.00  
 1st Qu.:23.89   1st Qu.:30.69   1st Qu.:5.200   1st Qu.:   8.41  
 Median :24.80   Median :46.01   Median :5.460   Median :  78.12  
 Mean   :25.15   Mean   :48.51   Mean   :5.793   Mean   : 149.56  
 3rd Qu.:26.51   3rd Qu.:66.91   3rd Qu.:6.500   3rd Qu.: 192.52  
 Max.   :30.22   Max.   :87.23   Max.   :7.820   Max.   :1227.59  
                                                 NA's   :2        
> find_mode <- function(x) {
+   u <- unique(x)
+   tab <- tabulate(match(x, u))
+   u[tab == max(tab)]
+ }
> find_mode(df$temp)
[1] 24.51 24.26
> find_mode(df$vp)
[1] 80
> find_mode(df$PET)
[1] 5.22
> find_mode(df$rainfall)
[1] 0
```

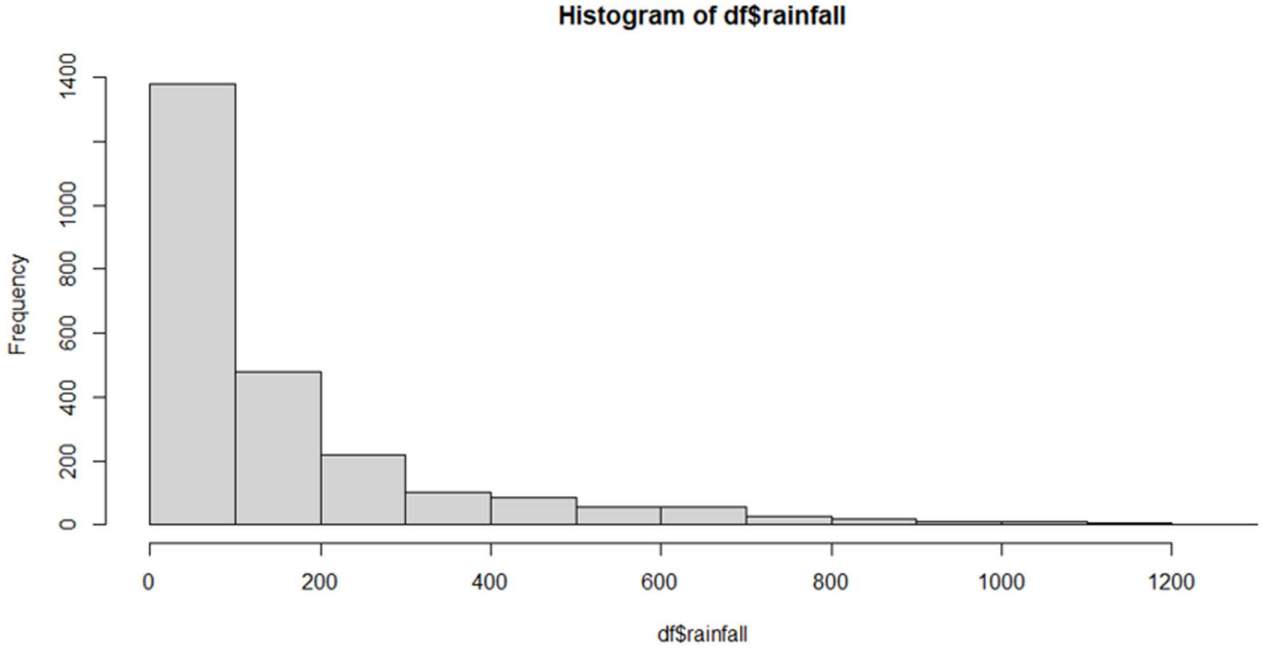
4.plot histogram for temp,Vp,PET in R

```
hist(df$temp)
hist(df$vp)
hist(df$PET)
hist(df$rainfall)
```






Histogram of df$vp
Frequency
0
50
100
150
200
250
300
20
40
60
80
df$vp


Histogram of df$rainfall
Frequency
0
200
400
600
800
1000
1400
0
200
400
600
800
1000
1200
df$rainfall