Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2023-24

Class:	TE	**Semester:**	VI
Course Code:	CSL601	**Course Name:**	Data Analytics and Visualization

Name of Student:	Parth Manoj Raut
Roll No.:	44
Experiment No.:	4
Title of the Experiment:	Time Series Analysis in Python/R.
Date of Performance:	
Date of Submission:	

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Performance	5	
Understanding	5	
Journal work and timely submission	10	
Total	20	

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Performance	4-5	2-3	1
Understanding	4-5	2-3	1
Journal work and timely submission	8-10	5-8	1-4

Checked by

Name of Faculty : Ms Bhavika Gharat

Signature :

Date :

Experiment No- 4

Aim :- Implement Time Series Analysis for rainfall in R Programming.

Objective:- To understand the use of time series models for prediction.

Description:-

- Time series analysis is a specific way of analyzing a sequence of data points collected over an interval of time. In time series analysis, analysts record data points at consistent intervals over a set period of time rather than just recording the data points randomly.

The basic syntax for ts() function in time series analysis is –

- timeseries.object.name <- ts(data, start, end, frequency)

Following is the description of the parameters used –

- data is a vector or matrix containing the values used in the time series.
- start specifies the start time for the first observation in time series.
- end specifies the end time for the last observation in time series.
- frequency specifies the number of observations per unit time.

- Except the parameter "data" all other parameters are optional.

Different Time Intervals

The value of the frequency parameter in the ts() function decides the time intervals at which the data points are measured. A value of 12 indicates that the time series is for 12 months. Other values and its meaning is as below –

- frequency = 12 pegs the data points for every month of a year.
- frequency = 4 pegs the data points for every quarter of a year.
- frequency = 6 pegs the data points for every 10 minutes of an hour.
- frequency = 24*6 pegs the data points for every 10 minutes of a day.

Program :-

Consider the annual rainfall details at a place starting from January 2012. We create an R time series object for a period of 12 months and plot it.

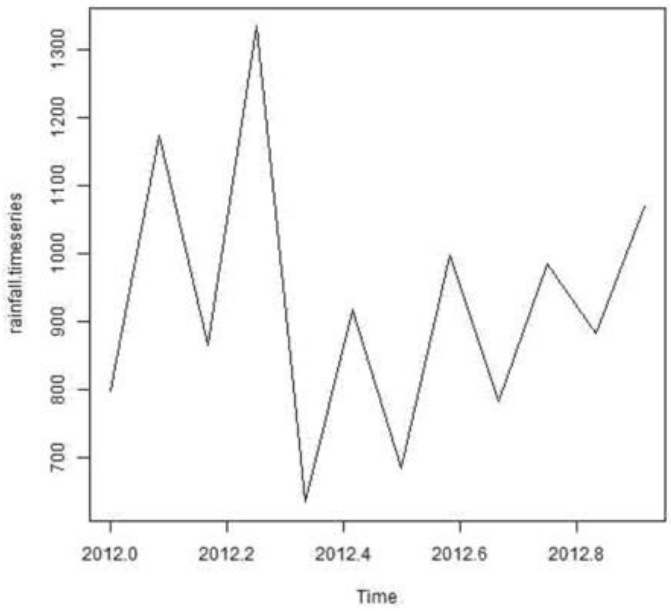
```
# Get the data points in the form of a R vector.
rainfall <- c(799, 1174.8,865.1, 1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8, 1071)
# Convert it to a time series object.
rainfall.timeseries <- ts(rainfall ,start = c(2012, 1),frequency = 12)
# Print the timeseries data.
print(rainfall. timeseri es)
# Give the chart file a name.
png(file = "rainfall.png")

# Plot a graph of the time series.
plot(rainfall.timeseries)


# Save the file.
dev.off()
```

Output:

```
> rainfall <- c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)
> rainfall.timeseries <- ts(rainfall,start = c(2012,1),frequency = 12)
> print(rainfall.timeseries)
       Jan    Feb    Mar    Apr    May    Jun    Jul    Aug    Sep    Oct    Nov    Dec
2012  799.0 1174.8  865.1 1334.6  635.4  918.5  685.5  998.6  784.2  985.0  882.8 1071.0
> png(file = "rainfall.png")
> plot(rainfall.timeseries)
```



Multiple Time Series

We can plot multiple time series in one chart by combining both the series into a matrix.

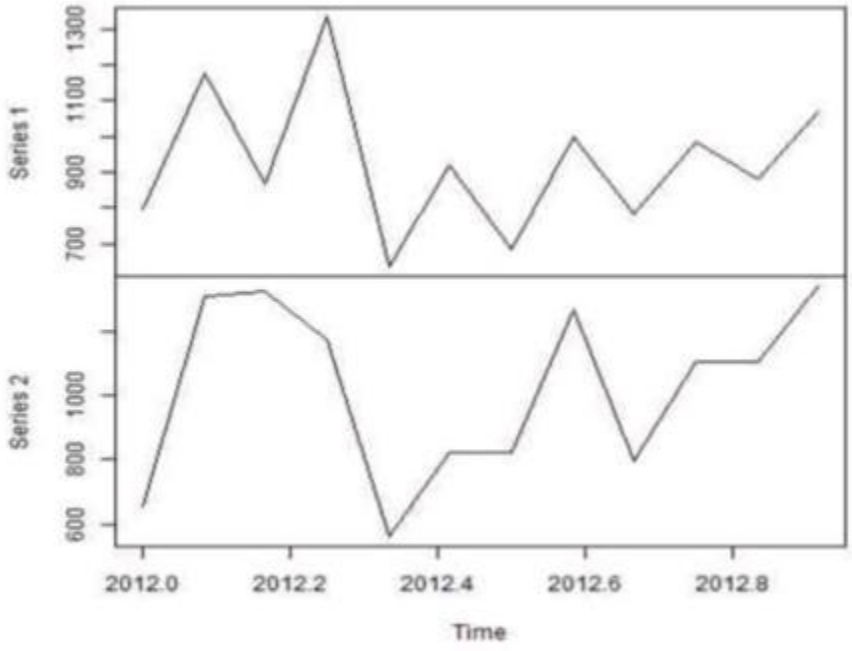
```
# Get the data points in form of a R vector.
rainfall I <- c(799, 1174.8,865.1, 1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8, 1071)
rainfall2 <-
c(655, 1306.9, 1323.4, 1172.2,562.2,824,822.4, 1265.5,799.6, 1105.6, 1106.7, 1337.8)

# Convert them to a matrix.
combined.rainfall <- matrix(c(rainfall l,rainfall2),nrow = 12)
print (combined.rainfall)
# Convert it to a time series object.
rainfall.timeseries <- ts(combined.rainfall ,start = c(2012, 1),frequency = 12)
# Print the timeseries data. print(rainfall.
timeseri es)
# Give the chart file a name.
png(file = "rainfall_combined. png")
# Plot a graph of the time series.
plot(rainfall.timeseries , main = "Multiple Time Series")

# Save the file. dev.off()
```

Output:

```
> rainfall1 <- c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)
> rainfall2 <-c(655,1306.9,1323.4,1172.2,562.2,824,822.4,1265.5,799.6,1105.6,1106.7,1337.8)
> combined.rainfall <- matrix(c(rainfall1,rainfall2),nrow = 12)
> print (combined.rainfall)
       [,1]   [,2]
 [1,]  799.0  655.0
 [2,] 1174.8 1306.9
 [3,]  865.1 1323.4
 [4,] 1334.6 1172.2
 [5,]  635.4  562.2
 [6,]  918.5  824.0
 [7,]  685.5  822.4
 [8,]  998.6 1265.5
 [9,]  784.2  799.6
[10,]  985.0 1105.6
[11,]  882.8 1106.7
[12,] 1071.0 1337.8
> rainfall.timeseries <- ts(combined.rainfall,start = c(2012,1),frequency = 12)
> print(rainfall.timeseries)
         Series 1 Series 2
Jan 2012    799.0    655.0
Feb 2012   1174.8   1306.9
Mar 2012    865.1   1323.4
Apr 2012   1334.6   1172.2
May 2012    635.4    562.2
Jun 2012    918.5    824.0
Jul 2012    685.5    822.4
Aug 2012    998.6   1265.5
Sep 2012    784.2    799.6
Oct 2012    985.0   1105.6
Nov 2012    882.8   1106.7
Dec 2012   1071.0   1337.8
> png(file = "rainfall_combined.png")
> plot(rainfall.timeseries, main = "Multiple Time Series")
> dev.off()
```



Conclusion :-

1. An orderly set of data arranged in accordance with their time of occurrence is called

Time Series

2.The graph of time series is called

Time Series Plot

3.Use of Matrix()-

Matrix() function in R is used to create matrices, which are essentially rectangular arrays of numbers, symbols, or even expressions. These matrices can be used for various data analysis tasks, but they're not specifically designed for time series data