How can I calculate a rolling average in R using an example?

Authored by

stats writer

June 27, 2024

Calculating a rolling average in R is a method used to smooth out data by taking the average of a specified number of consecutive data points. This can be useful in identifying trends and patterns in data over time. To calculate a rolling average in R, you can use the "rollmean" function from the "zoo" package. For example, if we have a dataset of daily sales for a month, we can use the "rollmean" function to calculate the 7-day rolling average. This will take the average of the current day and the past 6 days, and continue this calculation for the entire month. This will result in a new dataset with smoothed out values, making it easier to analyze and interpret the data.

Calculate a Rolling Average in R (With Example)

In time series analysis, a rolling average represents the average value of a certain number of previous periods.

The easiest way to calculate a rolling average in R is to use the rollmean() function from the zoo package:

```
library(dplyr)library(zoo)

#calculate 3-day rolling average
df %>%
mutate(rolling_avg = rollmean(values, k=3, fill=NA, align='right'))
```

This particular example calculates a 3-day rolling average for the column titled values.

The following example shows how to use this function in practice.

Example: Calculate Rolling Average in R

Suppose we have the following data frame in R that shows the sales of some product during 10 consecutive days:

```
#create data frame
df <- data.frame(day=c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10),
sales=c(25, 20, 14, 16, 27, 20, 12, 15, 14, 19))

#view data frame
df

day sales
1 1 25
2 2 20
3 3 14
4 4 16
5 5 27
6 6 20
7 7 12
8 8 15
9 9 14
10 10 19
```

We can use the following syntax to create a new column called avg_sales3 that displays the rolling average value of sales for the previous 3 days in each row of the data frame:

```
library(dplyr)
library(zoo)

#calculate 3-day rolling average of sales
df %>%
mutate(avg_sales3 = rollmean(sales, k=3, fill=NA, align='right'))
```

```
day sales avg_sales3
1 1 25 NA
2 2 20 NA
3 3 14 19.66667
4 4 16 16.66667
5 5 27 19.00000
6 6 20 21.00000
7 7 12 19.66667
8 8 15 15.66667
9 9 14 13.66667
10 10 19 16.00000
```

Note: The value for k in the rollmean() function controls the number of previous periods used to calculate the rolling average.

The avg_sales3 column shows the rolling average value of sales for the previous 3 periods.

For example, the first value of 19.66667 is calculated as:

3-Day Moving Average = (25 + 20 + 14) / 3 = 19.66667

You can also calculate several rolling averages at once by using multiple rollmean() functions within the mutate() function.

For example, the following code shows how to calculate the 3-day and 4-day moving average of sales:

```
library(dplyr)
library(zoo)

#calculate 3-day and 4-day rolling average of sales
df %>%
mutate(avg_sales3 = rollmean(sales, k=3, fill=NA, align='right'),
avg_sales4 = rollmean(sales, k=4, fill=NA, align='right'))
```

day sales avg_sales3 avg_sales4

1 1 25 NA NA

2 2 20 NA NA

3 3 14 19.66667 NA

4 4 16 16.66667 18.75

5 5 27 19.00000 19.25

6 6 20 21.00000 19.25

7 7 12 19.66667 18.75

8 8 15 15.66667 18.50

9 9 14 13.66667 15.25

10 10 19 16.00000 15.00

How to Plot Multiple Columns in R

How to Average Across Columns in R

How to Calculate the Mean by Group in R