# How can I interpolate missing values in R?

Authored by

**stats writer**

July 1, 2024

Interpolation is a statistical method used to estimate missing data points within a dataset. In R, the "na.interp" function from the "impute" package can be used to interpolate missing values. This function uses linear interpolation to estimate the missing values by taking the average of the values before and after the missing data point. For example, if a dataset contains the values 1, 3, NA, 7, 9, the interpolated value for the missing data point would be (3+7)/2 = 5. The "na.interp" function can be applied to any type of dataset, including numerical, categorical, and time-series data.

## Interpolate Missing Values in R (Including Example)

## You can use the following basic syntax to interpolate missing values in a data frame column in R:

```
library(dplyr)
library(zoo)

df <- df %>%
mutate(column_name = na.approx(column_name))
```

## The following example shows how to use this syntax in practice.

### Example: Interpolate Missing Values in R

## Suppose we have the following data frame in R that shows the total sales made by a store during 15 consecutive days:

```
#create data frame
df <- data.frame(day=1:15,
```

**sales=c(3, 6, 8, 10, 14, 17, 20, NA, NA, NA, NA, 35, 39, 44, 49))**

**#view data frame**

**df**

**day sales**

**1 1 3**

**2 2 6**

**3 3 8**

**4 4 10**

**5 5 14**

**6 6 17**

**7 7 20**

**8 8 NA**

**9 9 NA**

**10 10 NA**

**11 11 NA**

**12 12 35**

**13 13 39**

**14 14 44**

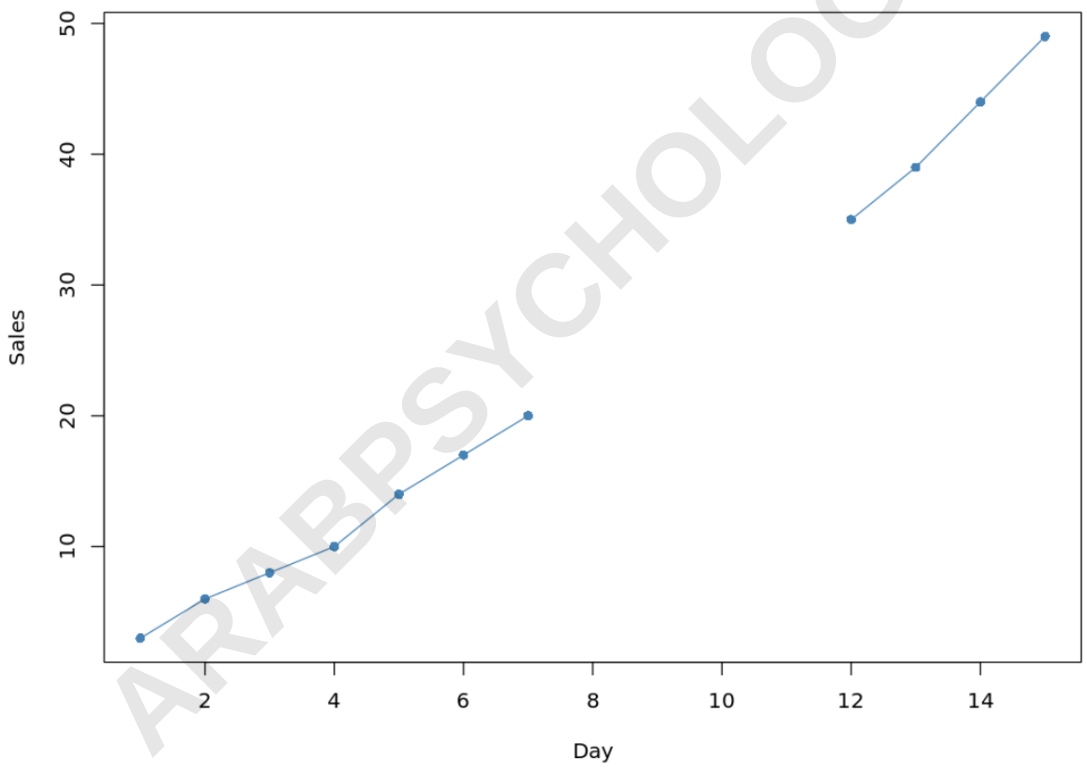
**15 15 49**

**Notice that we're missing sales numbers for four days**

**in the data frame.**

**If we create a simple line chart to visualize the sales over time, here's what it would look like:**

```
#create line chart to visualize sales
plot(df$sales, type='o', pch=16, col='steelblue', xlab='Day', ylab='Sales')
```






**To fill in the missing values, we can use the function from the zoo package along with the function from the dplyr package:**

```
library(dplyr)
library(zoo)

#interpolate missing values in 'sales' column
df <- df %>%
mutate(sales = na.approx(sales))

#view updated data frame
df

day sales
1 1 3
2 2 6
3 3 8
4 4 10
5 5 14
6 6 17
7 7 20
8 8 23
9 9 26
10 10 29
11 11 32
12 12 35
13 13 39
14 14 44
```

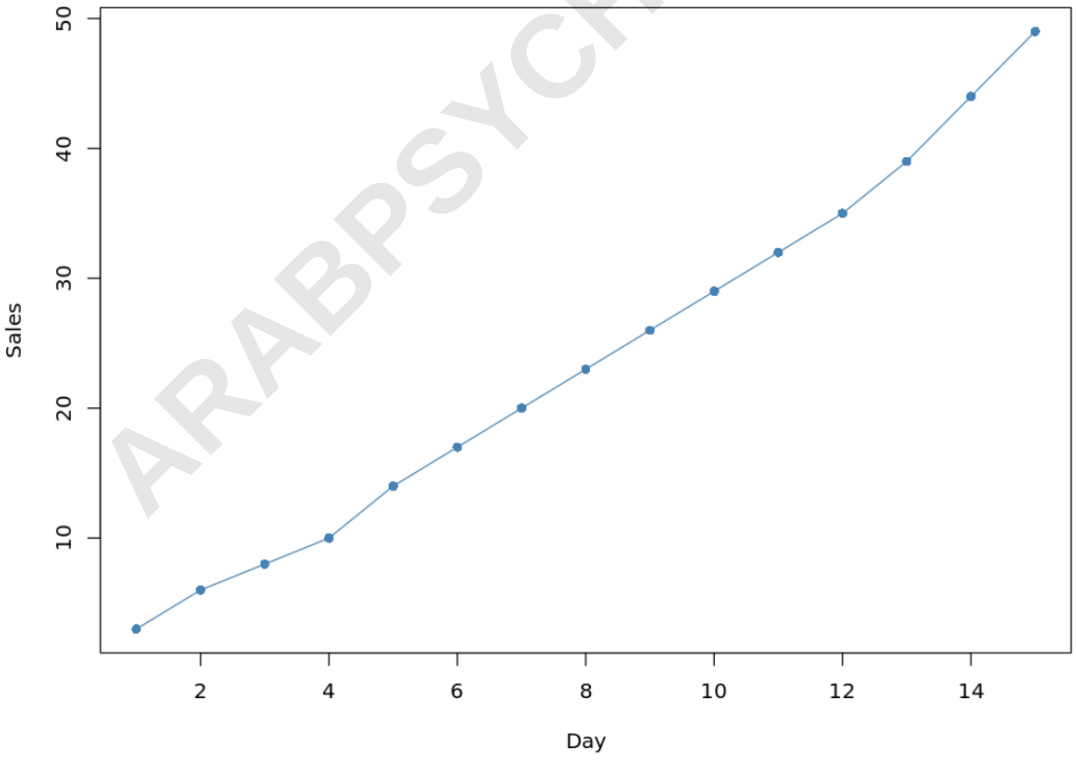
**15 15 49**

**Notice that each of the missing values has been replaced.**

**If we create another line chart to visualize the updated data frame, here's what it would look like:**

**#create line chart to visualize sales**
**plot(df$sales, type='o', pch=16, col='steelblue', xlab='Day', ylab='Sales')**






**Notice that the values chosen by the na.approx()**

**function seem to fit the trend in the data quite well.**

## Additional Resources

**The following tutorials provide additional information on how to handle missing values in R:**