How to Easily Impute Missing Values in Pandas with fillna()

Authored by

stats writer

December 2, 2025

Imputing missing values in pandas can be done by using the fillna() function. This is done by replacing the missing values with a value of your choice, for example, replacing the missing values with the mean of the other values. You can also use other methods such as forward or backward filling of values in order to fill in the missing values. To illustrate this, an example could be to replace all missing values in a column of a dataframe with the mean of that column. This is done by using the fillna() function and supplying the mean of the column as an argument.

You can use the following basic syntax to impute missing values in a pandas DataFrame:

df = df.interpolate()

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

Example: Interpolate Missing Values in Pandas

Suppose we have the following pandas DataFrame that shows the total sales made by a store during 15 consecutive days:

import pandas as pd
import numpy as np

#create DataFrame
df = pd.DataFrame({'day': ,
'sales': })

#view DataFrame
print(df)

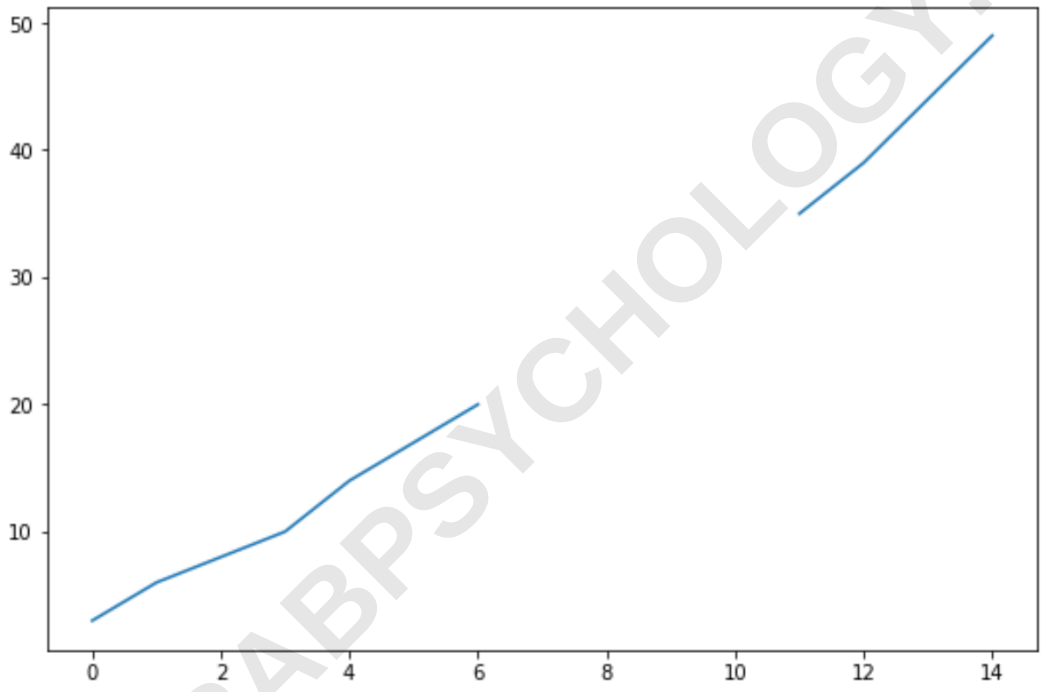
day sales
0 1 3.0
1 2 6.0
2 3 8.0
3 4 10.0
4 5 14.0
5 6 17.0
6 7 20.0
7 8 NaN
8 9 NaN
9 10 NaN
10 11 NaN

11 12 35.0
12 13 39.0
13 14 44.0
14 15 49.0

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
df.plot()



To fill in the missing values, we can use the **interpolate()** function as follows:

#interpolate missing values in 'sales' column
df = df.interpolate()

#view DataFrame
print(df)

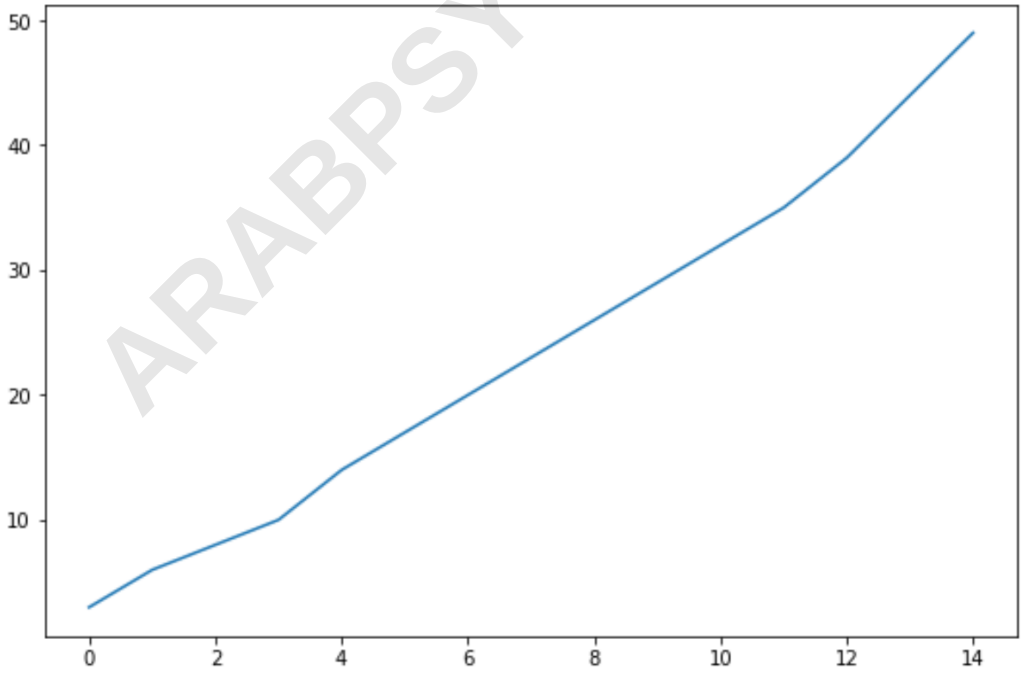
day sales
0 1 3.0
1 2 6.0

2 3 8.0
3 4 10.0
4 5 14.0
5 6 17.0
6 7 20.0
7 8 23.0
8 9 26.0
9 10 29.0
10 11 32.0
11 12 35.0
12 13 39.0
13 14 44.0
14 15 49.0

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
df.plot()



Notice that the values chosen by the **interpolate()** function seem to fit the trend in the data quite

well.

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