

How can I find the difference between two rows in Pandas?

Authored by
stats writer

April 28, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I find the difference between two rows in Pandas?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=140441>

To find the difference between two rows in Pandas, you can use the ".diff()" function which calculates the difference between two consecutive rows in a dataframe. This function can be applied to a specific column or the entire dataframe. The result will be a new column with the calculated differences. Additionally, you can also use other functions such as ".subtract()" or simply subtract the two rows manually using basic arithmetic operations. This allows for easy comparison and analysis of data within a Pandas dataframe.

Pandas: Find the Difference Between Two Rows

You can use the DataFrame.diff() function to find the difference between two rows in a pandas DataFrame.

This function uses the following syntax:

```
DataFrame.diff( periods=1, axis=0)
```

where:

periods: The number of previous rows for calculating the difference.
axis: Find difference over rows (0) or columns (1).

The following examples show how to use this function in practice.

Example 1: Find Difference Between Each Previous Row

Suppose we have the following pandas DataFrame:

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'period': ,
'sales': ,
'returns': })
```

```
#view DataFrame
df
```

```
period sales returns
0 1 12 2
1 2 14 2
2 3 15 3
3 4 15 3
4 5 18 5
5 6 20 4
6 7 19 4
7 8 24 6
```

The following code shows how to find the difference between every current row in a DataFrame and the previous row:

```
#add new column to represent sales differences
```

between each row

```
df = df.diff()
```

```
#view DataFrame
```

```
df
```

```
period sales returns sales_diff
```

```
0 1 12 2 NaN
```

```
1 2 14 2 2.0
```

```
2 3 15 3 1.0
```

```
3 4 15 3 0.0
```

```
4 5 18 5 3.0
```

```
5 6 20 4 2.0
```

```
6 7 19 4 -1.0
```

```
7 8 24 6 5.0
```

Note that we can also find the difference between several rows prior. For example, the following code shows how to find the difference between each current row and the row that occurred three rows earlier:

```
#add new column to represent sales differences  
between current row and 3 rows earlier
```

```
df = df.diff(periods=3)
```

#view DataFrame

```
df
```

```
period sales returns sales_diff
```

```
0 1 12 2 NaN
```

```
1 2 14 2 NaN
```

```
2 3 15 3 NaN
```

```
3 4 15 3 3.0
```

```
4 5 18 5 4.0
```

```
5 6 20 4 5.0
```

```
6 7 19 4 4.0
```

```
7 8 24 6 6.0
```

Example 2: Find Difference Based on Condition

We can also filter the DataFrame to show rows where the difference between the current row and the previous row is less than or greater than some value.

For example, the following code returns only the rows where the value in the current row is less than the value in the previous row:

```
import pandas as pd
```

#create DataFrame

```
df = pd.DataFrame({'period': ,  
'sales': ,  
'returns': })
```

```
#find difference between each current row and the  
previous row
```

```
df = df.diff()
```

```
#filter for rows where difference is less than zero
```

```
df = df[df < 0]
```

```
#view DataFrame
```

```
df
```

```
period sales returns sales_diff
```

```
3 4 13 3 -2.0
```

```
6 7 19 4 -1.0
```