

How can I use Pandas to find the maximum value in each row of a data frame?

Authored by
stats writer

June 24, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use Pandas to find the maximum value in each row of a data frame?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151497>

Pandas is a powerful Python library used for data analysis and manipulation. One of its useful features is the ability to find the maximum value in each row of a data frame. This can be achieved by using the "max" function along with the "axis=1" parameter, which specifies that the operation should be performed on each row. This will return a series containing the maximum value for each row in the data frame. This method can be particularly useful in identifying outliers or finding the highest value in a particular category within the data set.

Pandas: Find the Max Value in Each Row

You can use the following basic syntax to find the max value in each row of a pandas DataFrame:

```
df = df.max(axis=1)
```

This particular syntax creates a new column called max that contains the max value in each row of the DataFrame.

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

Example: Find the Max Value in Each Row in Pandas

Suppose we have the following pandas DataFrame:

```
import pandas as pd
```

```
import numpy as np
```

```
#create DataFrame
```

```
df = pd.DataFrame({'points': ,  
'rebounds': ,  
'assists': })
```

```
#view DataFrame  
print(df)
```

```
points rebounds assists  
0 4.0 NaN 10  
1 NaN 3.0 9  
2 10.0 9.0 4  
3 2.0 7.0 4  
4 15.0 6.0 3  
5 NaN 8.0 7  
6 7.0 14.0 10  
7 22.0 10.0 11
```

We can use the following syntax to create a new column called max that contains the max value in each row:

```
#create new column that contains max value of each  
row  
df = df.max(axis=1)
```

```
#view updated DataFrame
```

```
print(df)
```

```
points rebounds assists max
```

```
0 4.0 NaN 10 10.0
```

```
1 NaN 3.0 9 9.0
```

```
2 10.0 9.0 4 10.0
```

```
3 2.0 7.0 4 7.0
```

```
4 15.0 6.0 3 15.0
```

```
5 NaN 8.0 7 8.0
```

```
6 7.0 14.0 10 14.0
```

```
7 22.0 10.0 11 22.0
```

The new column called max contains the max value in each row.

For example, we can see:

The max value in the first row is 10. The max value in the second row is 9. The max value in the third row is 10.

And so on.

Also notice that the max() function automatically ignores NaN values when determining the max value in each row.

You can also find the max value in each row for only specific columns.

For example, you can use the following syntax to find the max value in each row and only consider the points and rebounds columns:

```
#add new column that contains max value of each row  
for points and rebounds columns
```

```
df = df].max(axis=1)
```

```
#view updated DataFrame
```

```
print(df)
```

```
points rebounds assists max
```

```
0 4.0 NaN 10 4.0
```

```
1 NaN 3.0 9 3.0
```

```
2 10.0 9.0 4 10.0
```

```
3 2.0 7.0 4 7.0
```

```
4 15.0 6.0 3 15.0
```

```
5 NaN 8.0 7 8.0
```

```
6 7.0 14.0 10 14.0
```

```
7 22.0 10.0 11 22.0
```

Note: You can find the complete documentation for the

pandas max() function .

The following tutorials explain how to perform other common tasks in pandas:

ARABPSYCHOLOGY.COM