How can I create a new column in Pandas using multiple if else conditions?

Authored by

stats writer

June 26, 2024

To create a new column in Pandas using multiple if else conditions, one can use the .apply() function and pass in a lambda function that contains the desired conditions. This will allow for the creation of a new column based on the specified conditions for each row of data. Alternatively, one can use the .loc function to specify the rows and columns to be updated based on the conditions, and assign the desired values to the new column. This allows for a more direct approach to creating the new column. Both methods require a deep understanding of Pandas syntax and data manipulation techniques.

Pandas: Create New Column Using Multiple If Else Conditions

You can use the following syntax to create a new column in a pandas DataFrame using multiple if else conditions:

```
#define conditions
conditions = == 'A') & (df < 20),
(df == 'A') & (df >= 20),
(df == 'B') & (df < 20),
(df == 'B') & (df >= 20)
]

#define results
results =

#create new column based on conditions in column1 and column2
df = np.select(conditions, results)
```

This particular example creates a column called new_column whose values are based on the values in column1 and column2 in the DataFrame.

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

Example: Create New Column Using Multiple If Else Conditions in Pandas

Suppose we have the following pandas DataFrame that contains information about various basketball players:

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'team': ,
'points': })

#view DataFrame
print(df)

team points
0 A 15
1 A 18
2 A 22
```

3 A 24

4 B 12

5 B 17

6 B 20

7 B 28

Now suppose we would like to create a new column called class that classifies each player into one of the following four groups:

Bad_A if team is A and points < 20Good_A if team is A and points ≥ 20Bad_B if team is B and points < 20Good_B if team is B and points ≥ 20

We can use the following syntax to do so:

```
import numpy as np

#define conditions
conditions = == 'A') & (df < 20),
(df == 'A') & (df >= 20),
(df == 'B') & (df < 20),
(df == 'B') & (df >= 20)
]
```

```
#define results
results =

#create new column based on conditions in column1 and column2
df = np.select(conditions, results)

#view updated DataFrame
print(df)

team points class
0 A 15 Bad_A
1 A 18 Bad_A
2 A 22 Good_A
3 A 24 Good_A
4 B 12 Bad_B
5 B 17 Bad_B
6 B 20 Good_B
7 B 28 Good_B
```

The new column called class displays the classification of each player based on the values in the team and points columns.

Note: You can find the complete documentation for the

NumPy select() function .

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