How can I apply a function to a Pandas groupby object?

Authored by

stats writer

July 1, 2024

Applying a function to a Pandas groupby object allows for the manipulation and transformation of data within specific groupings. This process involves using the groupby() function to group data by a chosen variable, followed by applying a desired function to the grouped data. This allows for efficient and streamlined data analysis, as the function is only applied to the relevant subsets of data. Additionally, this method is useful for performing calculations and aggregations on grouped data, providing valuable insights and summaries of the data.

Apply Function to Pandas Groupby

You can use the following basic syntax to use the groupby() and apply() functions together in a pandas DataFrame:

df.groupby('var1').apply(lambda x: some function)

The following examples show how to use this syntax in practice with the following pandas DataFrame:

import pandas as pd

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

#view DataFrame
print(df)

team points_for points_against
0 A 18 14
1 A 22 21
2 A 19 19
3 B 14 14
4 B 11 12
5 B 20 20
6 B 28 21

Example 1: Use groupby() and apply() to Find Relative Frequencies

The following code shows how to use the groupby() and apply() functions to find the relative frequencies of each team name in the pandas DataFrame:

```
#find relative frequency of each team name in DataFrame
df.groupby('team').apply(lambda x: x.count() / df.shape)
```

team
A 0.428571
B 0.571429
dtype: float64

From the output we can see that team A occurs in

42.85% of all rows and team B occurs in 57.14% of all rows.

Example 2: Use groupby() and apply() to Find Max Values

The following code shows how to use the groupby() and apply() functions to find the max "points_for" values for each team:

```
#find max "points_for" values for each team
df.groupby('team').apply(lambda x: x.max())
```

```
team
A 22
B 28
dtype: int64
```

From the output we can see that the max points scored by team A is 22 and the max points scored by team B is 28.

Example 3: Use groupby() and apply() to Perform Custom Calculation

The following code shows how to use the groupby() and apply() functions to find the mean difference between "points_for" and "points_against" for each team:

#find max "points_for" values for each team
df.groupby('team').apply(lambda x: (x - x).mean())

team
A 1.666667
B 1.500000
dtype: float64

From the output we can see that the mean difference between "points for" and "points against" is 1.67 for team A and 1.50 for team B.

Additional Resources