

How can you calculate the mean by group in Pandas, and what are some examples of doing so?

Authored by
stats writer

June 27, 2024

RECOMMENDED CITATION

stats writer (2024). *How can you calculate the mean by group in Pandas, and what are some examples of doing so?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=154545>

Pandas is a popular Python library used for data analysis and manipulation. One of its useful functions is the ability to calculate the mean by group. This means that the mean or average value of a specific attribute can be calculated for each group or category within the dataset.

To calculate the mean by group in Pandas, the data must first be grouped by the desired attribute using the `groupby()` function. Then, the `mean()` function can be applied to the grouped data to calculate the mean for each group. This method allows for efficient and accurate analysis of data within different groups.

For example, a company may have sales data for different regions and want to calculate the average sales for each region. By using the `groupby()` and `mean()` functions in Pandas, the company can easily determine the average sales for each region without having to manually calculate it for each region separately.

Another example could be a study on the average income for different age groups. The data can be grouped by age groups and then the `mean()` function can be used to calculate the average income for each group, providing valuable insights into income distribution within different age groups.

In summary, the ability to calculate the mean by group in Pandas allows for efficient and accurate analysis of data within different categories or groups, providing valuable information and insights for various data analysis tasks.

Calculate the Mean by Group in Pandas (With Examples)

You can use the following methods to calculate the mean value by group in pandas:

Method 1: Calculate Mean of One Column Grouped by One Column

`df.groupby().mean()`

Method 2: Calculate Mean of Multiple Columns Grouped by One Column

```
df.groupby().mean()
```

Method 3: Calculate Mean of One Column Grouped by Multiple Columns

```
df.groupby().mean()
```

The following examples show how to use each method in practice with the following pandas DataFrame:

```
import pandas as pd

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

#view DataFrame
print(df)

team position points assists
```

0 A G 30 4
1 A F 22 3
2 A F 19 7
3 A G 14 7
4 B F 14 12
5 B F 11 15
6 B G 20 8
7 B G 28 4

Example 1: Calculate Mean of One Column Grouped by One Column

The following code shows how to calculate the mean value of the points column, grouped by the team column:

```
#calculate mean of points grouped by team  
df.groupby('team').mean()
```

team

A 21.25

B 18.25

Name: points, dtype: float64

From the output we can see:

The mean points value for team A is 21.25. The mean points value for team B is 18.25.

Example 2: Calculate Mean of Multiple Columns Grouped by One Column

The following code shows how to calculate the mean value of the points column and the mean value of the assists column, grouped by the team column:

```
#calculate mean of points and mean of assists grouped  
by team  
df.groupby('team').mean()
```

```
points assists
```

```
team
```

```
A 21.25 5.25
```

```
B 18.25 9.75
```

Example 3: Calculate Mean of One Column Grouped by Multiple Columns

The following code shows how to calculate the mean value of the points column, grouped by the team and position columns:

```
#calculate mean of points, grouped by team and
```

position

df.groupby().mean()

team position

A F 20.5

G 22.0

B F 12.5

G 24.0

Name: points, dtype: float64

From the output we can see:

The mean points value for players on team A and position F is 20.5. The mean points value for players on team A and position G is 22. The mean points value for players on team B and position F is 12.5. The mean points value for players on team B and position G is 24.

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