How do I fix the ValueError “Index contains duplicate entries, cannot reshape”?

Authored by

stats writer

July 1, 2024

The ValueError "Index contains duplicate entries, cannot reshape" occurs when attempting to reshape an array or dataframe that contains duplicate values in the index. This error can be resolved by first identifying and removing any duplicate entries in the index, and then attempting to reshape the data again. Alternatively, using functions such as "drop_duplicates" or "reset_index" can also help resolve this error. It is important to ensure that the index is unique before reshaping the data in order to avoid this error.

Fix: ValueError: Index contains duplicate entries, cannot reshape

One error you may encounter when using pandas is:

ValueError: Index contains duplicate entries, cannot reshape

This error usually occurs when you attempt to reshape a pandas DataFrames by using the pivot() function, but there are multiple values in the resulting DataFrame that share the same index values.

The following example shows how to fix this error in practice.

How to Reproduce the Error

Suppose we have the following pandas DataFrame:

import pandas as pd

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

#view DataFrame
df

team position points
0 A G 5
1 A G 7
2 A F 7
3 A F 9
4 B G 4
5 B G 9
6 B F 9
7 B F 12
```


Now suppose we attempt to pivot the DataFrame, using team as the rows and position as the columns:

```
#attempt to reshape DataFrame
df.pivot(index='team', columns='position',
values='points')
```

ValueError: Index contains duplicate entries, cannot reshape

We receive an error because there are multiple rows in the DataFrame that share the same values for team and position.

Thus, when we attempt to reshape the DataFrame, pandas doesn't know which points value to display in each cell in the resulting DataFrame.

How to Fix the Error

To fix this error, we can use the pivot_table() function with a specific aggfunc argument to aggregate the data values in a certain way.

For example, we can use pivot_table() to create a new DataFrame that uses team as the rows, position as the columns, and the sum of the points values in the cells of the DataFrame:

```
df.pivot_table(index='team', columns='position', values='points', aggfunc='sum')
```

position F G

team
A 16 12
B 21 13

Notice that we don't receive an error this time.

The values in the DataFrame show the sum of points for each combination of team and position.

```
df.pivot_table(index='team', columns='position', values='points', aggfunc='mean')
```

position F G
team
A 8.0 6.0
B 10.5 6.5

By using the aggfunc argument within the pivot_table() function, we're able to avoid any errors.

Note: You can find the complete documentation for the pivot_table() function .

Additional Resources

The following tutorials explain how to fix other common

errors in Python: