Cheatography

Pandas Python Cheat Sheet

by robeyjorgensen via cheatography.com/127240/cs/24745/

pd.concat()

`pd.concat([df1,df2])`	Stacks `df1` and `df2`
`pd.concat([df1,df2],keys=['1','2'])`	Creates a label (key) for each df
`pd.concat([df1,df2], join='inner')`	Default join is "inner join"
`df1.append(df2)`	Does the same thing as `pd.concat()`

Multi-indexing

`pd.MultiIndex.from_arrays([['a','a','b','b'],[1,2,1,2]])`

`pd.MultiIndex.from_tuples([('a',1),('a',2),('b',1),('b',2)])`

`pd.MultiIndex.from_product([['a','b'],[1,2]])`

`df.sort_index()`	Orders index in ascending order
`df.reset_index(name='')`	Turn the index labels into cols

All three methods of `pd.MultiIndex` give:

```
MultiIndex(
[('a', 1),('a', 2),('b', 1),
('b', 2)]
)`
```

Joins

`pd.merge(df1,df2)`

`pd.merge(df1,df2,left_on="a",right_on="b")`	Merges data using "a" and "b" as keys
`pd.merge(df1,df2,how='inner')`	Default is "outer join"
`pd.merge(df1,df2, on="a",suffixes=["_L","_R"])`	When multiple cols have same index and name, gives suffix "_L" and "_R"

Aggregation and Grouping

`df.mean(axis='columns')`	Calculates the mean across cols
`df.groupby('key').sum()`	Gives sum for each key
`df.groupby('key')['a'].sum()`	Gives sum for all rows in col "a"
`df.groupby('key')['a'].describe()`	Gives summary stats
`df.groupby('key').transform(lambda x:x-x.mean())`	Applies lambda function as aggregation function
`df.groupby('key').apply(function)`	Applies "function" as aggregation function

Pivot Tables

`df.pivot_table(index='a',columns='col',aggfunc='sum')`	Groups by index and gives the sum of column "a"
`df.pivot_table(index='a',columns='col',aggfunc='sum').plot()`	Plots values using "a" as x-axis and sum(col) as y-axis

Pivot tables do the same thing as using `groupby` with aggregation functions. The main difference is that they are a cleaner way of using multiple aggregation functions at once for a single grouping

By **robeyjorgensen**

cheatography.com/robeyjorgensen/

Published 10th October, 2020.
Last updated 10th October, 2020.