

April 10, 2024

```
[2]: import numpy as np
import pandas as pd
import seaborn as sns
titanic = sns.load_dataset('titanic')
```

1. Write a Pandas program to print a concise summary of the dataset (titanic.csv).

```
[3]: titanic.head()
```

```
[3]:   survived  pclass    sex  age  sibsp  parch    fare embarked  class \
0         0        3   male  22.0     1     0   7.2500         S   Third
1         1        1  female  38.0     1     0  71.2833         C   First
2         1        3  female  26.0     0     0   7.9250         S   Third
3         1        1  female  35.0     1     0  53.1000         S   First
4         0        3   male  35.0     0     0   8.0500         S   Third

      who  adult_male deck  embark_town  alive  alone
0   man         True  NaN  Southampton    no  False
1  woman        False   C    Cherbourg   yes  False
2  woman        False  NaN  Southampton   yes   True
3  woman        False   C    Southampton   yes  False
4   man         True  NaN  Southampton    no   True
```

2. Write a Pandas program to extract the column labels, shape and data types of the dataset (titanic.csv).

```
[4]: print(titanic.columns, titanic.shape, titanic.dtypes)
```

```
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
      'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
      'alive', 'alone'],
      dtype='object') (891, 15) survived          int64
pclass          int64
sex             object
age            float64
sibsp          int64
parch          int64
fare           float64
embarked       object
```

```

class          category
who            object
adult_male     bool
deck          category
embark_town    object
alive          object
alone          bool
dtype: object

```

- Write a Pandas program to create a Pivot table with multiple indexes from the data set of titanic.csv.

```
[6]: titanic.pivot_table('age', index=['sex', 'alone'], columns='class',
    ↪aggfunc='mean')
```

```
[6]: class          First      Second      Third
sex    alone
female False  34.415094  25.545455  20.671875
       True   34.937500  33.383333  23.565789
male   False  37.466383  25.203611  18.923030
       True   44.601852  33.904762  29.184492

```

- Write a Pandas program to create a Pivot table and find survival rate by gender on various classes.

```
[7]: titanic.pivot_table('survived', index='sex', columns='class', aggfunc='mean')
```

```
[7]: class          First      Second      Third
sex
female  0.968085  0.921053  0.500000
male    0.368852  0.157407  0.135447

```

- Write a Pandas program to create a Pivot table and find survival rate by gender.

```
[8]: titanic.pivot_table('survived', index='sex', aggfunc='mean')
```

```
[8]:      survived
sex
female  0.742038
male    0.188908

```

- Write a Pandas program to create a Pivot table and find survival rate by gender, age wise of various classes.

```
[22]: titanic.pivot_table('survived', index=['sex', pd.cut(titanic['age'],
    ↪range(0,100,10))], columns='class', aggfunc='mean').fillna(0)
```

```
[22]: class          First      Second      Third
sex    age
female (0, 10]  0.000000  1.000000  0.500000

```

	(10, 20]	1.000000	1.000000	0.520000
	(20, 30]	0.952381	0.892857	0.500000
	(30, 40]	1.000000	0.941176	0.428571
	(40, 50]	0.923077	0.900000	0.000000
	(50, 60]	1.000000	0.666667	0.000000
	(60, 70]	1.000000	0.000000	1.000000
male	(0, 10]	1.000000	1.000000	0.363636
	(10, 20]	0.400000	0.100000	0.129630
	(20, 30]	0.473684	0.000000	0.144330
	(30, 40]	0.520000	0.115385	0.142857
	(40, 50]	0.375000	0.111111	0.090909
	(50, 60]	0.285714	0.000000	0.000000
	(60, 70]	0.000000	0.333333	0.000000
	(70, 80]	0.333333	0.000000	0.000000

7. Write a Pandas program to partition each of the passengers into four categories based on their age. Note: Age categories (0, 10), (10, 30), (30, 60), (60, 80)

```
[23]: titanic.pivot_table('survived', index=['sex', pd.cut(titanic['age'], [0, 10, 30, 60, 80])], columns='class', aggfunc='mean').fillna(0)
```

```
[23]: class          First    Second    Third
sex   age
female (0, 10]    0.000000    1.000000    0.500000
       (10, 30]    0.970588    0.916667    0.508772
       (30, 60]    0.979167    0.900000    0.272727
       (60, 80]    1.000000    0.000000    1.000000
male   (0, 10]    1.000000    1.000000    0.363636
       (10, 30]    0.458333    0.023256    0.139073
       (30, 60]    0.412698    0.090909    0.118421
       (60, 80]    0.083333    0.333333    0.000000
```

8. Write a Pandas program to create a Pivot table and count survival by gender, categories wise age of various classes. Note: Age categories (0, 10), (10, 30), (30, 60), (60, 80)

```
[24]: titanic.pivot_table('survived', index=['sex', pd.cut(titanic['age'], [0, 10, 30, 60, 80])], columns='class', aggfunc='sum').fillna(0)
```

```
[24]: class          First  Second  Third
sex   age
female (0, 10]         0         8      11
       (10, 30]        33        33      29
       (30, 60]        47        27       6
       (60, 80]         2         0       1
male   (0, 10]         2         9       8
       (10, 30]        11         1      21
       (30, 60]        26         4       9
       (60, 80]         1         1       0
```

9. Write a Pandas program to create a Pivot table and find survival rate by gender, age of the different categories of various classes.

```
[26]: titanic.pivot_table('survived', index=['sex', pd.cut(titanic['age'], [0, 10, 30, 60, 80])], columns='pclass', aggfunc='mean').fillna(0)
```

```
[26]: pclass          1          2          3
sex   age
female (0, 10]    0.000000    1.000000    0.500000
       (10, 30]    0.970588    0.916667    0.508772
       (30, 60]    0.979167    0.900000    0.272727
       (60, 80]    1.000000    0.000000    1.000000
male   (0, 10]    1.000000    1.000000    0.363636
       (10, 30]    0.458333    0.023256    0.139073
       (30, 60]    0.412698    0.090909    0.118421
       (60, 80]    0.083333    0.333333    0.000000
```

10. Write a Pandas program to create a Pivot table and find survival rate by gender, age of the different categories of various classes. Add the fare as a dimension of columns and partition fare column into 2 categories based on the values present in fare columns

```
[33]: titanic.pivot_table('survived', index=['sex', pd.cut(titanic['age'], [0, 10, 30, 60, 80])], columns=['class', pd.cut(titanic['fare'], [-1,0,1000])], aggfunc='mean').fillna(0)
```

```
[33]: class          First          Second   Third
fare      (-1, 0] (0, 1000] (0, 1000] (-1, 0] (0, 1000]
sex   age
female (0, 10]      0.0  0.000000  1.000000      0.0  0.500000
       (10, 30]      0.0  0.970588  0.916667      0.0  0.508772
       (30, 60]      0.0  0.979167  0.900000      0.0  0.272727
       (60, 80]      0.0  1.000000  0.000000      0.0  1.000000
male   (0, 10]      0.0  1.000000  1.000000      0.0  0.363636
       (10, 30]      0.0  0.458333  0.023256      0.5  0.134228
       (30, 60]      0.0  0.433333  0.090909      0.0  0.121622
       (60, 80]      0.0  0.083333  0.333333      0.0  0.000000
```

```
[25]:
```

```
[25]: class          First
fare      0.0000  5.0000  25.5875  25.9292  26.0000  26.2833  26.2875
sex   age
female (0, 10]      0.0      0.0      0.0      0.0      0.0      0.0      0.0
       (10, 30]      0.0      0.0      0.0      0.0      0.0      1.0      0.0
       (30, 60]      0.0      0.0      0.0      1.0      0.0      0.0      0.0
       (60, 80]      0.0      0.0      0.0      0.0      0.0      0.0      0.0
male   (0, 10]      0.0      0.0      0.0      0.0      0.0      0.0      0.0
       (10, 30]      0.0      0.0      0.0      0.0      0.0      0.0      0.0
```

	(30, 60]	0.0	0.0	0.0	0.0	0.0	0.0	1.0
	(60, 80]	0.0	0.0	0.0	0.0	0.0	0.0	0.0

class				...	Third				\
fare		26.3875	26.5500	27.7208	...	22.5250	24.1500	27.9000	29.1250
sex	age				...				
female	(0, 10]	0.0	0.000	0.0	...	0.0	0.0	0.0	0.0
	(10, 30]	0.0	0.000	0.0	...	0.0	0.0	0.0	0.0
	(30, 60]	0.0	1.000	1.0	...	0.0	0.0	0.0	0.0
	(60, 80]	0.0	0.000	0.0	...	0.0	0.0	0.0	0.0
male	(0, 10]	0.0	0.000	0.0	...	0.0	0.0	0.0	0.0
	(10, 30]	0.0	1.000	0.0	...	0.0	0.0	0.0	0.0
	(30, 60]	1.0	0.625	0.0	...	0.0	0.0	0.0	0.0
	(60, 80]	0.0	0.000	0.0	...	0.0	0.0	0.0	0.0

class									
fare		31.2750	31.3875	34.3750	39.6875	46.9000	56.4958		
sex	age								
female	(0, 10]	0.0	1.0	0.0	0.0	0.0	0.0		
	(10, 30]	0.0	0.0	0.0	0.0	0.0	0.0		
	(30, 60]	0.0	1.0	0.0	0.0	0.0	0.0		
	(60, 80]	0.0	0.0	0.0	0.0	0.0	0.0		
male	(0, 10]	0.0	0.5	0.0	0.0	0.0	0.0		
	(10, 30]	0.0	0.0	0.0	0.0	0.0	0.5		
	(30, 60]	0.0	0.0	0.0	0.0	0.0	1.0		
	(60, 80]	0.0	0.0	0.0	0.0	0.0	0.0		

[8 rows x 226 columns]