Jeff_Smolinski___5.2.3

October 11, 2020

Jeff Smolinski 5.2.3 10.11.2020

[63]:
```
p_titanic = titanic_df.
 ↪pivot_table(index='sex',columns='class',values='survived')
p_titanic.head()
```

[63]:
```
class      First    Second     Third
sex
female  0.968085  0.921053  0.500000
male    0.368852  0.157407  0.135447
```

Wow. Good to be female.

[41]:
```
import pandas as pd

import numpy as np
data = pd.Series(np.random.randn(9),␣
 ↪index=[['a','a','a','b','b','c','c','d','d'],[1,2,3,1,3,1,2,2,3]])
data
```

[41]:
```
a  1   -0.213564
   2   -0.532472
   3    0.428397
b  1    0.831080
   3    0.071384
c  1   -0.399650
   2    0.286835
d  2   -0.392845
   3   -0.235736
dtype: float64
```

[4]:
```
print(data['a'])
```

```
1    0.487038
2    1.122319
3    0.524621
dtype: float64
```

[6]:
```
print(data[:,3])
```

```
a    0.524621
b   -0.430330
d   -0.083057
dtype: float64
```

[9]:
```
data.unstack()
#data.stack()
```

[9]:
```
          1         2         3
a  0.487038  1.122319  0.524621
b  0.258974       NaN -0.430330
c -0.611461  1.645155       NaN
d       NaN  2.280249 -0.083057
```

[12]:
```
frame=pd.DataFrame(np.arange(12).
 ↪reshape((4,3)),index=[['a','a','b','b'],[1,2,1,2]],
                   columns=[['Ohio','Ohio','Colorado'],['Green','Red','Green']])
frame
```

[12]:
```
     Ohio     Colorado
    Green Red    Green
a 1     0   1        2
  2     3   4        5
b 1     6   7        8
  2     9  10       11
```

[14]:
```
frame['Colorado']
```

[14]:
```
     Green
a 1      2
  2      5
b 1      8
  2     11
```

[24]:
```
frame=pd.DataFrame({'a': np.random.randint(0,high=10,size=7),
                    'b': range(7,0,-1),
                    'c': ['one','one','one','two','two','two','two'],
                    'd':[0,1,2,0,1,2,3]})
frame
```

[24]:
```
   a  b    c  d
0  9  7  one  0
1  2  6  one  1
2  2  5  one  2
3  0  4  two  0
4  9  3  two  1
5  0  2  two  2
```

```
6  7  1  two  3
```

[27]:
```
frame.set_index(['a','b']).sort_index().reset_index()
```

[27]:
```
   a  b    c  d
0  0  2  two  2
1  0  4  two  0
2  2  5  one  2
3  2  6  one  1
4  7  1  two  3
5  9  3  two  1
6  9  7  one  0
```

[32]:
```
#frame=pd.DataFrame(np.arange(12).reshape((4,3)),
#index=[['a','a','b','b'],[1,2,1,2]],
#columns=[['Ohio','Ohio','Colorado'],['Green','Red','Green']])
#frame
#frame.sum(level='Red')
# has no label
```

[39]:
```
df1 = pd.DataFrame({'lkey':['b','b','a','c','a','a','b'],'data1':range(7)})
print(df1)
df2 = pd.DataFrame({'rkey':['a','b','d'],'data2':range(3)})
print(df2)
df_merge = pd.merge(df1,df2,left_on='lkey', right_on='rkey')
df_merge
```

```
  lkey  data1
0    b      0
1    b      1
2    a      2
3    c      3
4    a      4
5    a      5
6    b      6
  rkey  data2
0    a      0
1    b      1
2    d      2
```

[39]:
```
   lkey  data1 rkey  data2
0    b      0    b      1
1    b      1    b      1
2    b      6    b      1
3    a      2    a      0
4    a      4    a      0
5    a      5    a      0
```

[40]:
```
df1 = pd.DataFrame({'lkey':['b','b','a','c','a','a','b'],'data1':range(7)})
print(df1)
df2 = pd.DataFrame({'rkey':['a','b','d'],'data2':range(3)})
print(df2)
df_merge = pd.merge(df1,df2,left_on='lkey', right_on='rkey', how='outer')
df_merge
```

```
  lkey  data1
0    b      0
1    b      1
2    a      2
3    c      3
4    a      4
5    a      5
6    b      6
  rkey  data2
0    a      0
1    b      1
2    d      2
```

[40]:

	lkey	data1	rkey	data2
0	b	0.0	b	1.0
1	b	1.0	b	1.0
2	b	6.0	b	1.0
3	a	2.0	a	0.0
4	a	4.0	a	0.0
5	a	5.0	a	0.0
6	c	3.0	NaN	NaN
7	NaN	NaN	d	2.0

[45]:
```
import seaborn
titanic_df = seaborn.load_dataset('titanic')
titanic_df.head(4)
```

[45]:

	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	

	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

[46]:
```
p_titanic = titanic_df.drop_duplicates('age').
 ↪pivot(index='age',columns='class',values='fare')
p_titanic.head()
```

[46]:
```
class  First  Second    Third
age
NaN      NaN     NaN   8.4583
0.42     NaN     NaN   8.5167
0.67     NaN    14.5      NaN
0.75     NaN     NaN  19.2583
0.83     NaN    29.0      NaN
```

[49]:
```
p_titanic = titanic_df.pivot_table(index='age',columns='class',values='fare')
p_titanic.head()
```

[49]:
```
class   First  Second    Third
age
0.42      NaN     NaN   8.5167
0.67      NaN  14.500      NaN
0.75      NaN     NaN  19.2583
0.83      NaN  23.875      NaN
0.92   151.55     NaN      NaN
```

[53]:
```
print('AGGREGATE BY MEAN/MIN...')
p_titanic = titanic_df.pivot_table(index='age',columns='class',values='fare',␣
 ↪aggfunc=np.min)
p_titanic.head()
```

```
AGGREGATE BY MEAN/MIN...
```

[53]:
```
class   First  Second    Third
age
0.42      NaN     NaN   8.5167
0.67      NaN   14.50      NaN
0.75      NaN     NaN  19.2583
0.83      NaN   18.75      NaN
0.92   151.55     NaN      NaN
```

[57]:
```
titanic_copy = p_titanic.copy()
titanic_copy.columns = titanic_copy.columns.astype(str)
melted_titanic_df = pd.melt(
    titanic_copy.reset_index(),
    id_vars='age',
    var_name='class_renamed',
    value_vars=['First','Second','Third'],
    value_name='fare'
)
```

```
print(len(titanic_copy))
melted_titanic_df.sample(5)
```

```
88
```

[57]:

	age	class_renamed	fare
226	37.0	Third	7.925
112	19.0	Second	10.500
119	24.0	Second	10.500
170	66.0	Second	10.500
262	74.0	Third	7.775

[60]:
```
titanic_df.head(4)
```

[60]:

	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	

	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