Practical 3 -Data Wrangling

November 19, 2019

[24]:
```
import pandas
one=pandas.DataFrame({
  'emp_id':[101,102,103,104,105,106],
  'dept_name':['CSE','IT','Mechanical','Civil','BioTechnology','ECE'],
  'aisle':[1,2,3,4,5,6]})
first=pandas.DataFrame(one)
```

[25]:
```
two=pandas.DataFrame({
  'emp_id':[107,108,109,110,111,112],
  'dept_name':['BCA','BSC','MCA','MSc','Mtech','BBA'],
  'aisle':[7,8,9,10,11,12]})
second=pandas.DataFrame(two)
```

[26]:
```
pandas.isnull(first).any()
```

[26]:
```
emp_id       False
dept_name    False
aisle        False
dtype: bool
```

[27]:
```
first.head()
```

[27]:

	emp_id	dept_name	aisle
0	101	CSE	1
1	102	IT	2
2	103	Mechanical	3
3	104	Civil	4
4	105	BioTechnology	5

[28]:
```
first.dropna()
```

[28]:

	emp_id	dept_name	aisle
0	101	CSE	1
1	102	IT	2
2	103	Mechanical	3
3	104	Civil	4
4	105	BioTechnology	5
5	106	ECE	6

[29]:
```
first.groupby('aisle').dept_name.value_counts()
```

[29]:
```
aisle  dept_name
1      CSE              1
2      IT               1
3      Mechanical       1
4      Civil            1
5      BioTechnology    1
6      ECE              1
Name: dept_name, dtype: int64
```

[30]:
```
two.aisle.unique()
```

[30]:
```
array([ 7,  8,  9, 10, 11, 12], dtype=int64)
```

[31]:
```
len(two)
```

[31]:
```
6
```

[32]:
```
two[two['aisle']>3]
```

[32]:
```
   emp_id dept_name  aisle
0     107       BCA      7
1     108       BSC      8
2     109       MCA      9
3     110       MSc     10
4     111     Mtech     11
5     112       BBA     12
```

[33]:
```
two[two['aisle']==2]
```

[33]:
```
Empty DataFrame
Columns: [emp_id, dept_name, aisle]
Index: []
```

[34]:
```
two[(two['aisle']==7) & (two['dept_name']=='BCA')]
```

[34]:
```
   emp_id dept_name  aisle
0     107       BCA      7
```

[36]:
```
(two[(two['aisle']==7) & (two['dept_name']=='BCA')]).describe()
```

[36]:
```
       emp_id  aisle
count     1.0    1.0
mean    107.0    7.0
std       NaN    NaN
min     107.0    7.0
25%     107.0    7.0
50%     107.0    7.0
75%     107.0    7.0
max     107.0    7.0
```

[37]:
```
two[two.aisle.isin([2,7,9])]
```

[37]:
```
   emp_id dept_name  aisle
0     107       BCA      7
```

```
2     109       MCA      9
```

[38]:
```
(two.pivot(index='emp_id',columns='aisle',values='dept_name')).tail()
```

[38]:
```
aisle    7    8    9    10     11   12
emp_id
108     NaN  BSC  NaN  NaN    NaN  NaN
109     NaN  NaN  MCA  NaN    NaN  NaN
110     NaN  NaN  NaN  MSc    NaN  NaN
111     NaN  NaN  NaN  NaN  Mtech  NaN
112     NaN  NaN  NaN  NaN    NaN  BBA
```

[39]:
```
(two.pivot(index='emp_id',columns='aisle',values='dept_name')).tail(3)
```

[39]:
```
aisle    7    8    9    10     11   12
emp_id
110     NaN  NaN  NaN  MSc    NaN  NaN
111     NaN  NaN  NaN  NaN  Mtech  NaN
112     NaN  NaN  NaN  NaN    NaN  BBA
```

[40]:
```
(two.pivot(index='emp_id',columns='aisle',values='dept_name')).tail(3).shift(1)
```

[40]:
```
aisle    7    8    9    10     11   12
emp_id
110     NaN  NaN  NaN  NaN    NaN  NaN
111     NaN  NaN  NaN  MSc    NaN  NaN
112     NaN  NaN  NaN  NaN  Mtech  NaN
```

[41]:
```
(two.melt(id_vars=['emp_id'],value_name='dept_name')).tail()
```

[41]:
```
    emp_id variable dept_name
7      108    aisle         8
8      109    aisle         9
9      110    aisle        10
10     111    aisle        11
11     112    aisle        12
```

[45]:
```
melt1=two.melt(id_vars=['emp_id'],value_name='dept_name')
melt2=two.melt(id_vars=['emp_id'],value_name='aisle')
pandas.merge(melt1,melt2,on=['emp_id']).tail(1)
```

[45]:
```
    emp_id variable_x dept_name variable_y aisle
23     112      aisle        12      aisle    12
```

[46]:
```
from functools import reduce
base=two[['aisle','dept_name','emp_id']]
feature=[base]+[melt1,melt2]
abt=reduce(lambda left,right: pandas.
 ↪merge(left,right,on=['emp_id']),[melt1,melt2])
abt.tail(1)
```

[46]:
```
    emp_id variable_x dept_name variable_y aisle
23     112      aisle        12      aisle    12
```

[47]:
```
pandas.concat([one,two])
```

[47]:

	emp_id	dept_name	aisle
0	101	CSE	1
1	102	IT	2
2	103	Mechanical	3
3	104	Civil	4
4	105	BioTechnology	5
5	106	ECE	6
0	107	BCA	7
1	108	BSC	8
2	109	MCA	9
3	110	MSc	10
4	111	Mtech	11
5	112	BBA	12

[48]:
```
res=pandas.concat([one,two])
res.to_excel('res.xlsx')
```

[50]:
```
import numpy
df=pandas.DataFrame(numpy.random.randn(7,3),
  index=pandas.date_range('1/1/2000',periods=7),
  columns=['A','B','C'])
```

[51]:
```
print(df)
```

```
                   A         B         C
2000-01-01 -0.339726  0.902747 -0.307331
2000-01-02  0.393683 -0.406915 -0.202124
2000-01-03 -0.009233 -0.882355 -0.553303
2000-01-04 -0.218265 -0.906099  0.517039
2000-01-05  1.084773 -0.695704 -1.015009
2000-01-06 -0.874640  1.023624 -0.166914
2000-01-07 -1.026444 -2.320872  0.065103
```

[54]:
```
print(df.rolling(window=3,min_periods=1))
Rolling[window=3,min_periods=1,center=False,axis=0]
```

```
  File "<ipython-input-54-17e1e045f994>", line 2
    rolling[window=3,min_periods=1,center=False,axis=0]
                    ^
SyntaxError: invalid syntax
```

[ ]:
```

```