da-lab-4

March 18, 2023

[8]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

[34]:
```
per = pd.read_csv('/content/drive/MyDrive/person.csv', na_values = ['??','???'])
per.dropna(inplace = True)
per.reset_index(drop = True, inplace = True)
per.head(5)
```

[34]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	London	31.0	Male	Single	50000
4	Los Angeles	57.0	Female	Divorced	40000

[35]:
```
p = pd.read_csv('/content/drive/MyDrive/person.csv', na_values = ['??','???'])
p.head(5)
```

[35]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	Berlin	NaN	Female	Married	120000
4	London	31.0	Male	Single	50000

[10]:
```
per.shape
```

[10]: (18, 5)

[11]:
```
per = per.set_index('Gender')
per.head(5)
```

[11]:

Gender	City	Age	Marital Status	Income
Male	New York	32.0	Single	55000
Female	Toronto	45.0	Married	75000

```
Male           Paris  28.0         Single   45000
Male          London  31.0         Single   50000
Female  Los Angeles   57.0       Divorced   40000
```

[12]:
```
#per.isnull().any()
per.isnull().sum()
#per.isnull().sum().sum()
```

[12]:
```
City              0
Age               0
Marital Status    0
Income            0
dtype: int64
```

[13]:
```
#drop missing value
per1 = per.dropna()
```

[14]:
```
per1.shape
```

[14]:
```
(18, 4)
```

[15]:
```
per2 = per.dropna(how = 'all')
```

[16]:
```
per3 = per.dropna(axis = 1)
```

0.1 Detecting missing values/ invalid values

[19]:
```
#To find unique values of column
per['City'].unique()
```

[19]:
```
array(['New York', 'Toronto', 'Paris', 'London', 'Los Angeles ', 'Tokyo',
       'Chicago', 'Vancouver', 'Munich'], dtype=object)
```

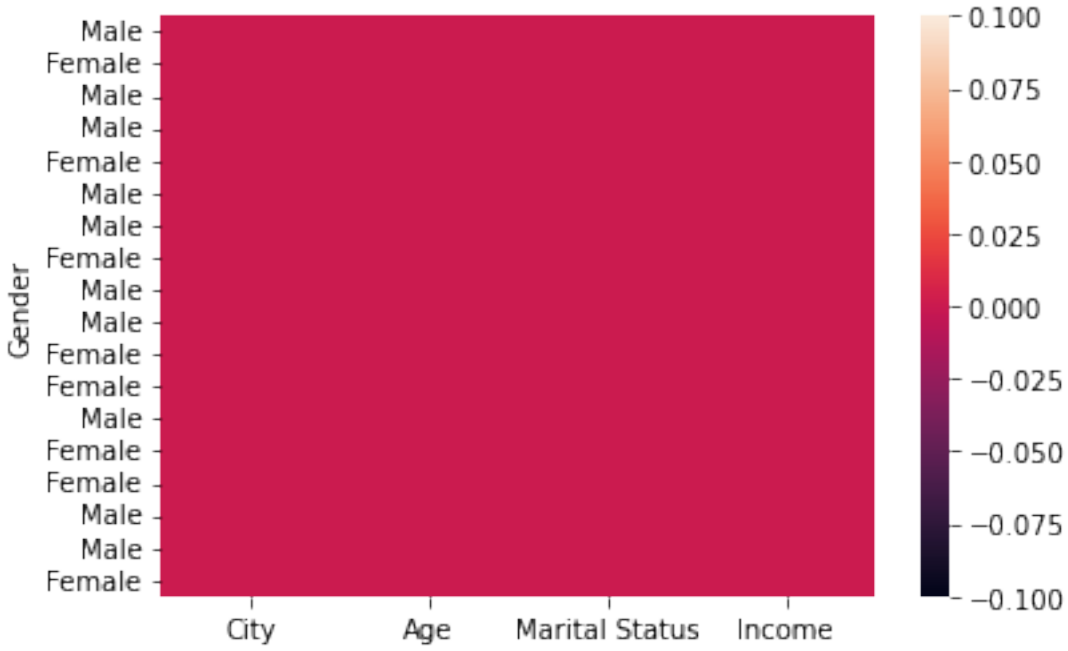
[20]:
```
#Find frequency of unique calues of column
per['City'].value_counts()
```

[20]:
```
London          4
Paris           3
Tokyo           3
New York        2
Munich          2
Toronto         1
Los Angeles     1
Chicago         1
Vancouver       1
Name: City, dtype: int64
```

[21]:
```
#convert any string to NaN values
per['City'].replace('??', np.nan, inplace = True)
```

[22]:
```
sns.heatmap(per.isnull())
```

[22]:
```
<AxesSubplot:ylabel='Gender'>
```



[23]:
```
missing = per[per.isnull().any(axis = 1)]
missing.head()
```

[23]:
```
Empty DataFrame
Columns: [City, Age, Marital Status, Income]
Index: []
```

[25]:
```
#Changing datatypes
per4 = per.astype({'Income':float})
per.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 18 entries, Male to Female
Data columns (total 4 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   City            18 non-null     object
 1   Age             18 non-null     float64
 2   Marital Status  18 non-null     object
```

```
 3   Income          18 non-null     int64
dtypes: float64(1), int64(1), object(2)
memory usage: 720.0+ bytes
```

0.2 Filling missing values

[36]:
```
per5 = p.fillna(value = 0)
per5.head()
```

[36]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	Berlin	0.0	Female	Married	120000
4	London	31.0	Male	Single	50000

[37]:
```
per6 = p.fillna(method = 'pad')
per6.head()
```

[37]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	Berlin	28.0	Female	Married	120000
4	London	31.0	Male	Single	50000

[38]:
```
per7 = p.fillna(method = 'bfill')
per7.head()
```

[38]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	Berlin	31.0	Female	Married	120000
4	London	31.0	Male	Single	50000

[39]:
```
per8 = p.interpolate(method = 'pad')
per8
```

[39]:

	City	Age	Gender	Marital Status	Income
0	New York	32.0	Male	Single	55000
1	Toronto	45.0	Female	Married	75000
2	Paris	28.0	Male	Single	45000
3	Berlin	28.0	Female	Married	120000
4	London	31.0	Male	Single	50000
5	Los Angeles	57.0	Female	Divorced	40000
6	Tokyo	39.0	Male	Married	95000

```
7        Vancouver  39.0  Female        Married   90000
8           London  35.0    Male         Single   60000
9            Paris  29.0  Female         Single   35000
10         Chicago  46.0    Male        Married   80000
11           Tokyo  46.0  Female         Single   65000
12          London  48.0    Male        Married   10000
13       Vancouver  50.0  Female       Divorced   70000
14  San Francisco   50.0    Male         Single   40000
15           Paris  41.0  Female        Married   85000
16          Munich  36.0    Male         Single   95000
17           Tokyo  26.0  Female         Single   55000
18       Vancouver  26.0    Male        Married   90000
19        New York  53.0  Female        Married  150000
20          London  47.0    Male        Married  100000
21           Paris  47.0  Female         Single   50000
22          Munich  42.0    Male        Married   80000
23           Tokyo  37.0  Female         Single   75000
24  San Francisco   37.0    Male        Married  120000
```

[ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```