

```
In [1]: # Standard data science imports
import numpy as np
import pandas as pd
from pandas import Series, DataFrame

# Visualization Libraries
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

# Scikit-Learn
import sklearn
from sklearn import datasets
from sklearn import preprocessing
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.metrics import classification_report
```

```
In [2]: # Warning Code
import warnings
warnings.filterwarnings('ignore')
```

```
In [3]: churn_df = pd.read_csv('churn_clean.csv', index_col=0)
```

```
In [4]: # Examine dataset
churn_df.columns
```

```
Out[4]: Index(['Customer_id', 'Interaction', 'UID', 'City', 'State', 'County', 'Zip',
       'Lat', 'Lng', 'Population', 'Area', 'TimeZone', 'Job', 'Children',
       'Age', 'Income', 'Marital', 'Gender', 'Churn', 'Outage_sec_perweek',
       'Email', 'Contacts', 'Yearly_equip_failure', 'Techie', 'Contract',
       'Port_modem', 'Tablet', 'InternetService', 'Phone', 'Multiple',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection', 'TechSupport',
       'StreamingTV', 'StreamingMovies', 'PaperlessBilling', 'PaymentMethod',
       'Tenure', 'MonthlyCharge', 'Bandwidth_GB_Year', 'Item1', 'Item2',
       'Item3', 'Item4', 'Item5', 'Item6', 'Item7', 'Item8'],
      dtype='object')
```

```
In [5]: churn_df.shape
```

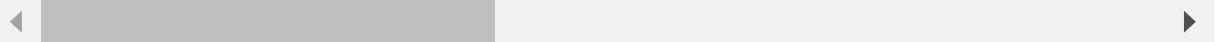
```
Out[5]: (10000, 49)
```

In [6]: `churn_df.head()`

Out[6]:

	Customer_id	Interaction	UID	City	State	Co
CaseOrder						
1	K409198	aa90260b-4141-4a24-8e36-b04ce1f4f77b	e885b299883d4f9fb18e39c75155d990	Point Baker	AK	Pri V
2	S120509	fb76459f-c047-4a9d-8af9-e0f7d4ac2524	f2de8bef964785f41a2959829830fb8a	West Branch	MI	Og
3	K191035	344d114c-3736-4be5-98f7-c72c281e2d35	f1784cfa9f6d92ae816197eb175d3c71	Yamhill	OR	Y
4	D90850	abfa2b40-2d43-4994-b15a-989b8c79e311	dc8a365077241bb5cd5ccd305136b05e	Del Mar	CA	
5	K662701	68a861fd-0d20-4e51-a587-8a90407ee574	aabb64a116e83fdc4befc1fbab1663f9	Needville	TX	

5 rows × 49 columns



```
In [7]: churn_df.info
```

```
Out[7]: <bound method DataFrame.info of
Interaction \
CaseOrder
1          K409198  aa90260b-4141-4a24-8e36-b04ce1f4f77b
2          S120509  fb76459f-c047-4a9d-8af9-e0f7d4ac2524
3          K191035  344d114c-3736-4be5-98f7-c72c281e2d35
4          D90850  abfa2b40-2d43-4994-b15a-989b8c79e311
5          K662701  68a861fd-0d20-4e51-a587-8a90407ee574
...
9996        M324793  45deb5a2-ae04-4518-bf0b-c82db8dbe4a4
9997        D861732  6e96b921-0c09-4993-bbda-a1ac6411061a
9998        I243405  e8307ddf-9a01-4fff-bc59-4742e03fd24f
9999        I641617  3775ccfc-0052-4107-81ae-9657f81ecdf3
10000       T38070  9de5fb6e-bd33-4995-aec8-f01d0172a499
                                         Customer_id
                                         UID           City State \
CaseOrder
1          e885b299883d4f9fb18e39c75155d990  Point Baker   AK
2          f2de8bef964785f41a2959829830fb8a  West Branch   MI
3          f1784cfa9f6d92ae816197eb175d3c71  Yamhill      OR
4          dc8a365077241bb5cd5ccd305136b05e  Del Mar      CA
5          aabb64a116e83fdc4befc1fbab1663f9  Needville   TX
...
9996        9499fb4de537af195d16d046b79fd20a  Mount Holly   VT
9997        c09a841117fa81b5c8e19afec2760104  Clarksville   TN
9998        9c41f212d1e04dca84445019bbc9b41c  Mobeetie     TX
9999        3e1f269b40c235a1038863ecf6b7a0df  Carrollton   GA
10000       0ea683a03a3cd544aefe8388aab16176  Clarkesville  GA
                                         County    Zip      Lat      Lng Population ...
\
CaseOrder
1          Prince of Wales-Hyder  99927  56.25100 -133.37571      38 ...
2                      Ogemaw    48661  44.32893 -84.24080  10446 ...
3                      Yamhill   97148  45.35589 -123.24657   3735 ...
4                      San Diego  92014  32.96687 -117.24798  13863 ...
5                      Fort Bend  77461  29.38012 -95.80673  11352 ...
...
9996        Rutland      5758  43.43391 -72.78734      640 ...
9997        Montgomery  37042  36.56907 -87.41694  77168 ...
9998        Wheeler     79061  35.52039 -100.44180      406 ...
9999        Carroll     30117  33.58016 -85.13241  35575 ...
10000       Habersham    30523  34.70783 -83.53648  12230 ...
                                         MonthlyCharge Bandwidth_GB_Year Item1  Item2  Item3  Item4  Item5 \
CaseOrder
1          172.455519        904.536110    5      5      5      3      4
2          242.632554        800.982766    3      4      3      3      4
3          159.947583        2054.706961   4      4      2      4      4
4          119.956840        2164.579412   4      4      4      2      5
5          149.948316        271.493436   4      4      4      3      4
...
9996        159.979400        6511.252601   3      2      3      3      4
9997        207.481100        5695.951810   4      5      5      4      4
9998        169.974100        4159.305799   4      4      4      4      4
9999        252.624000        6468.456752   4      4      6      4      3
10000       217.484000        5857.586167   2      2      3      3      3
```

CaseOrder	Item6	Item7	Item8
1	4	3	4
2	3	4	4
3	3	3	3
4	4	3	3
5	4	4	5
...
9996	3	2	3
9997	5	2	5
9998	4	4	5
9999	3	5	4
10000	3	4	1

[10000 rows x 49 columns]>

In [8]: `churn_df.describe()`

Out[8]:

	Zip	Lat	Lng	Population	Children	Age
count	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	49153.319600	38.757567	-90.782536	9756.562400	2.0877	53.078400
std	27532.196108	5.437389	15.156142	14432.698671	2.1472	20.698882
min	601.000000	17.966120	-171.688150	0.000000	0.0000	18.000000
25%	26292.500000	35.341828	-97.082813	738.000000	0.0000	35.000000
50%	48869.500000	39.395800	-87.918800	2910.500000	1.0000	53.000000
75%	71866.500000	42.106908	-80.088745	13168.000000	3.0000	71.000000
max	99929.000000	70.640660	-65.667850	111850.000000	10.0000	89.000000

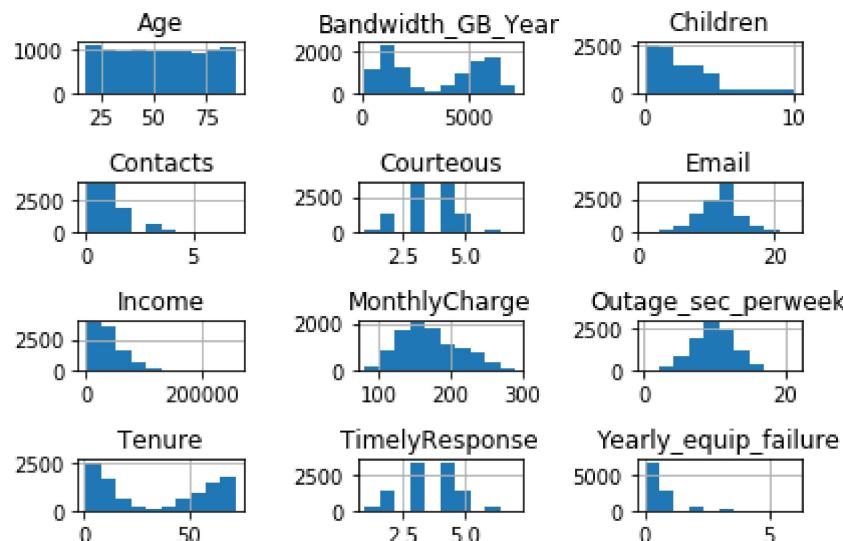
8 rows × 22 columns

```
In [9]: churn_df.dtypes
```

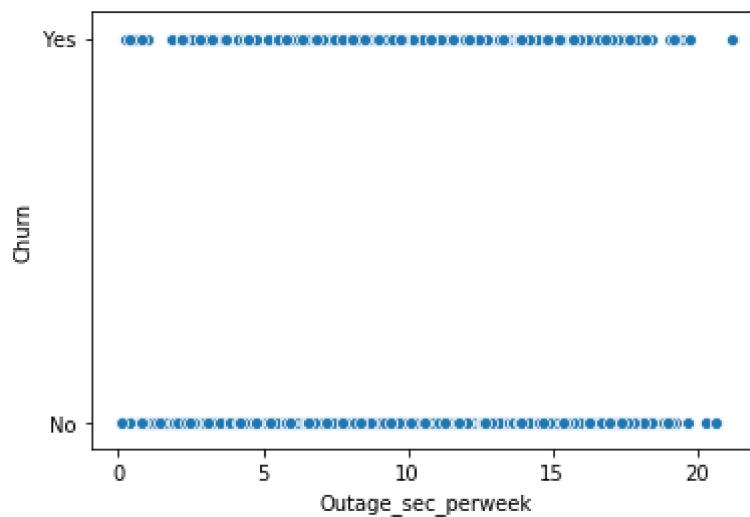
```
Out[9]: Customer_id          object
Interaction          object
UID                  object
City                 object
State                object
County               object
Zip                  int64
Lat                  float64
Lng                  float64
Population           int64
Area                 object
TimeZone             object
Job                 object
Children             int64
Age                  int64
Income               float64
Marital              object
Gender               object
Churn                object
Outage_sec_perweek  float64
Email                int64
Contacts             int64
Yearly_equip_failure int64
Techie               object
Contract             object
Port_modem            object
Tablet               object
InternetService       object
Phone                object
Multiple              object
OnlineSecurity        object
OnlineBackup           object
DeviceProtection      object
TechSupport            object
StreamingTV           object
StreamingMovies        object
PaperlessBilling       object
PaymentMethod          object
Tenure                float64
MonthlyCharge         float64
Bandwidth_GB_Year     float64
Item1                int64
Item2                int64
Item3                int64
Item4                int64
Item5                int64
Item6                int64
Item7                int64
Item8                int64
dtype: object
```

```
In [10]: # Rename survey items
churn_df.rename(columns = {'Item1':'TimelyResponse',
                           'Item2':'Fixes',
                           'Item3':'Replacements',
                           'Item4':'Reliability',
                           'Item5':'Options',
                           'Item6':'Respectful',
                           'Item7':'Courteous',
                           'Item8':'Listening'},
                           inplace=True)
```

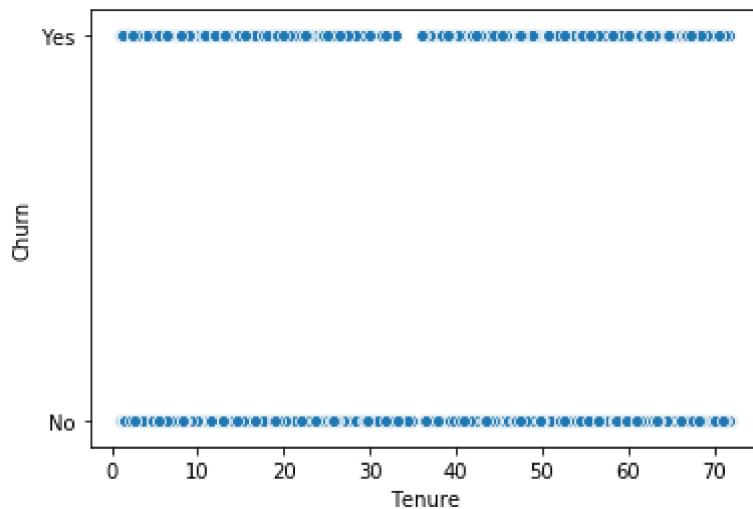
```
In [11]: # Create histograms
churn_df[['Children', 'Age', 'Income', 'Outage_sec_perweek', 'Email',
          'Contacts', 'Yearly_equip_failure', 'Tenure', 'MonthlyCharge',
          'Bandwidth_GB_Year', 'TimelyResponse', 'Courteous']].hist()
plt.savefig('churn_pyplot.jpg')
plt.tight_layout()
```



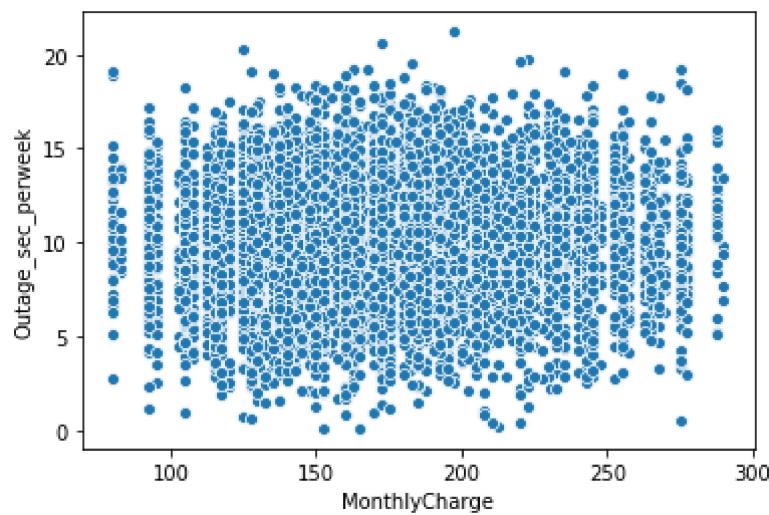
```
In [12]: # Create scatterplot Outage_sec_perweek
sns.scatterplot(x=churn_df['Outage_sec_perweek'], y=churn_df['Churn'])
plt.show()
```



```
In [13]: # Create scatterplot Tenure  
sns.scatterplot(x=churn_df['Tenure'], y=churn_df['Churn'])  
plt.show()
```



```
In [14]: # Create scatterplot MonthlyCharge to Outage_sec_perweek  
sns.scatterplot(x=churn_df['MonthlyCharge'], y=churn_df['Outage_sec_perweek'])  
plt.show()
```



```
In [17]: # Scatter matrix of nuerical variables
churn_numeric = churn_df[['Children', 'Age', 'Income', 'Outage_sec_perweek',
                           'Email', 'Contacts', 'Yearly_equip_failure', 'Tenure',
                           'MonthlyCharge', 'Bandwidth_GB_Year', 'Replacements',
                           'Reliability', 'Options', 'Respectful', 'Courteous',
                           'Listening']]
pd.plotting.scatter_matrix(churn_numeric, figsize = [15, 15])
```

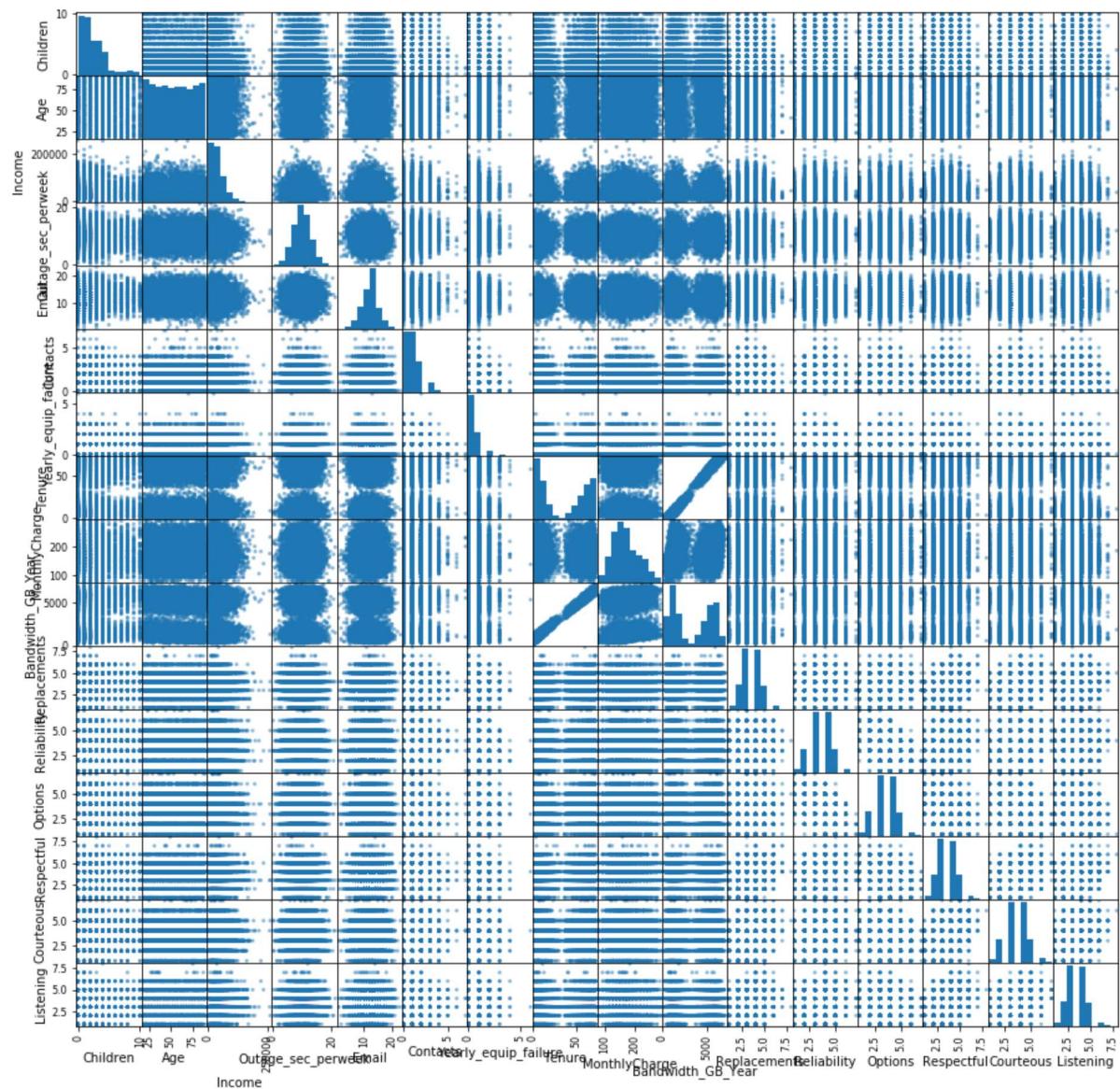
```
Out[17]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EECBF648>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEE7D7C8>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEE96948>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEEABE08>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE742308>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE7567C8>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE76E048>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE784C48>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE789748>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE7A7408>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE7E7088>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE808BC8>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE83A7C8>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE8753C8>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE8AF348>,
   <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE8E8448
],
 [
   [<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE91F508>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE95EE48>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE990748>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EE9C9848>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEA01988>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEA3BA88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEA73B08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEABC48>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEAE4D48>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEB1EE48>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEB55F88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191E99A8E88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEE34EC8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EEB999C8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF1F3AC8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF228BC8
],
 [
   [<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF262C88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF299D88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF2D2E88>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF309FC8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF348148>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF3801C8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF3B92C8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF3F1408>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF42C508>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF463608>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF49C748>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF4D57C8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF50D8C8>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF544A08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF57EB08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF5B9C08
],
 [
   [<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF5F1D08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF628E08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF663F08>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF69E088>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF6D8188>,
    <matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF710248>,
```

```
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF74A388>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF77F488>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF7BB588>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF7F9E88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF82D748>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF865848>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF89D948>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF8D7A88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF90EB88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF947C48
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF97FD88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF9B9E88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EF9F0F88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFA2E108>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFA68208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFAA1288>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFAD83C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFB124C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFB4A5C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFB82708>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFBBB788>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFBF6888>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFC2B9C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFC65AC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFC9FBC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFCD8CC8
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFD0EDC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFD48EC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFD7FFC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFDDBF148>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFDF7208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFE31308>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFE68448>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFEA1548>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFEDEE48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFF14708>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFF4C808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFF83908>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFFBD488>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191EFFF7B48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F002DC08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0066D48
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F009EE48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F00D6F48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F01140C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F014F208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0187248>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F01C0348>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F01F7488>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0232588>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0270E88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F02A2788>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F02DB888>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0315988>,
```

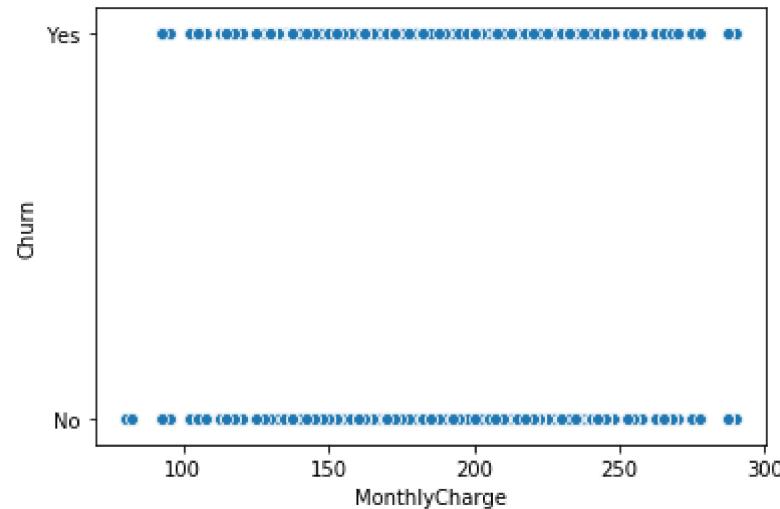
```
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F034BAC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0384BC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F03BEC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F03F5DC8
>,
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F042DEC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0467FC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F04A5148>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F04DE1C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F05162C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0550408>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0587508>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F05C1608>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F05FA6C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0634808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0668908>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F06A2A08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F06DBB48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0714C48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F074BCC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F0786E08
>,
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F178FF08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F17CC048>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1805188>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F183F208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1875308>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F18AE448>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F18E7548>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1927E48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1958748>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1992848>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F19CA948>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1A04A88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1A3BB88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1A76C48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1AACD88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1AE4D88
>,
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1B1DE88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1B57F88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1B96108>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1BCD188>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1C07288>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1C3E3C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1C764C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1CAE5C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1CE96C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1D1F7C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1D588C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1D94A08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1DCEB08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1E02BC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1E3BD08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1E74E08
>,
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1EAAF08>,
```

```
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1EEB088>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1F24108>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1F5B208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1F94308>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F1FC448>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F20085C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F203D708>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2077808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F20B0988>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F20E9AC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F211EC08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2158D88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2190E48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F21C7F88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2207148
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2243288>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F227A3C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F22B2548>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F22EA608>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2325748>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F235A8C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2395A08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F23D4E48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2405C48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F243CDC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2474F08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F24B5088>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F24ED208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F25242C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F255F408>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2596588
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F25D06C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2608808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2643908>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2679A48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F26B2B88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F26EAD08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2723E48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2758F48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2799108>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F27D1248>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F280A388>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F28424C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F287D5C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F28B4708>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F28ED848>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F29259C8
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2960B08>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2996C48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F29CFD88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2A08EC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2A49048>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2A7F1C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2AB7308>,
```

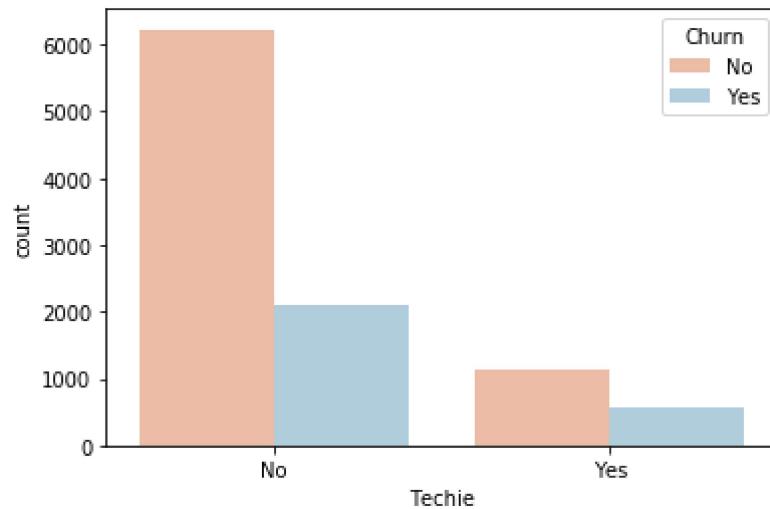
```
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2AEF408>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2B28588>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2B606C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2B9B808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2BD2988>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2C0AA48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2C41B88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2C7BC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2CB2E48
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2CEBF88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2D2C0C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2D65248>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2D9C388>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2DD54C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2E0E648>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2E46708>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2E7D848>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2EB89C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2EEE808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2F27C48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2F5ED48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2F99E88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F2FCFFC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F3012188>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F30492C8
>],
[<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F3082408>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F30B8548>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F30F36C8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F312C808>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F3163948>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F319CAC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F31D5B88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F320CCC8>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F3245E48>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F327BF88>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F32BC108>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F32F2208>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F332E348>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F3367488>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F33A1608>,
<matplotlib.axes._subplots.AxesSubplot object at 0x00000191F33D7748
>],
dtype=object)
```



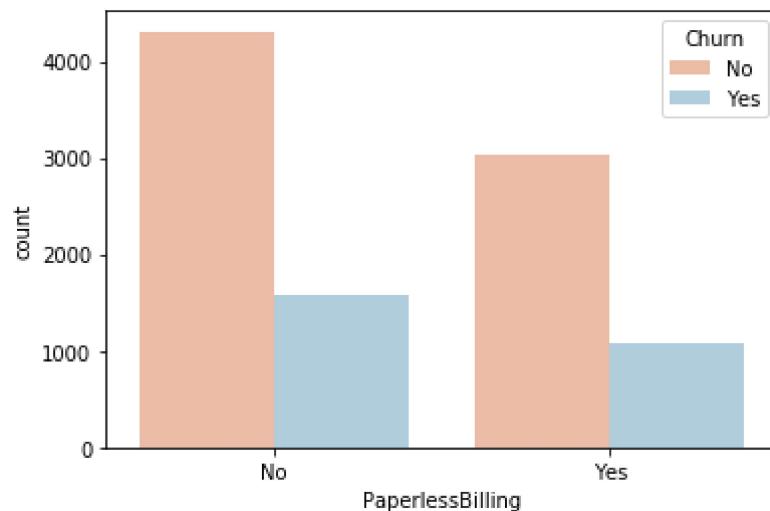
```
In [19]: # Scatterplot MonthlyCharge to churn
sns.scatterplot(x = churn_df['MonthlyCharge'], y = churn_df['Churn'])
plt.show()
```



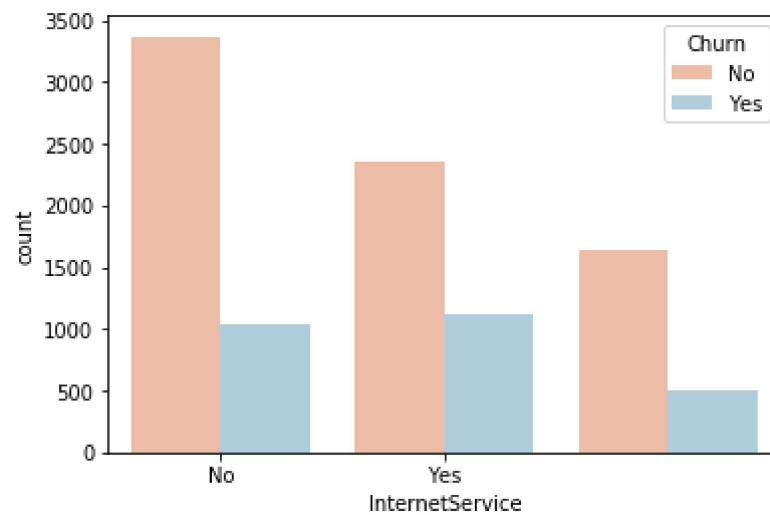
```
In [21]: # Binary dataset
plt.figure()
sns.countplot(x='Techie', hue='Churn', data=churn_df, palette='RdBu')
plt.xticks([0,1], ['No', 'Yes'])
plt.show()
```



```
In [22]: plt.figure()
sns.countplot(x='PaperlessBilling', hue='Churn', data=churn_df, palette='RdBu')
plt.xticks([0,1], ['No', 'Yes'])
plt.show()
```

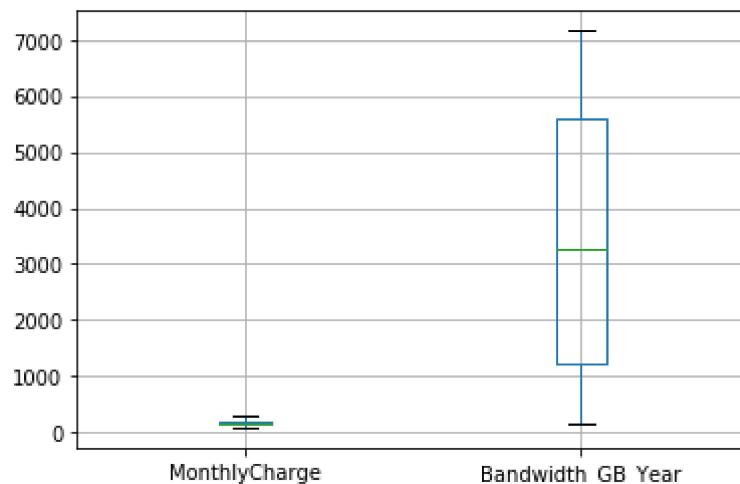


```
In [23]: plt.figure()
sns.countplot(x='InternetService', hue='Churn', data=churn_df, palette='RdBu')
plt.xticks([0,1], ['No', 'Yes'])
plt.show()
```

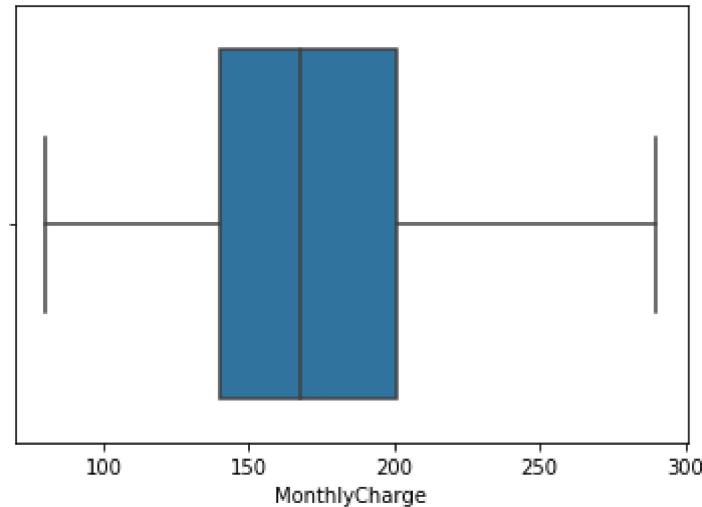


```
In [24]: # Boxplots MonthlyCharge and Bandwidth_GB_Year
churn_df.boxplot(column=['MonthlyCharge', 'Bandwidth_GB_Year'])
```

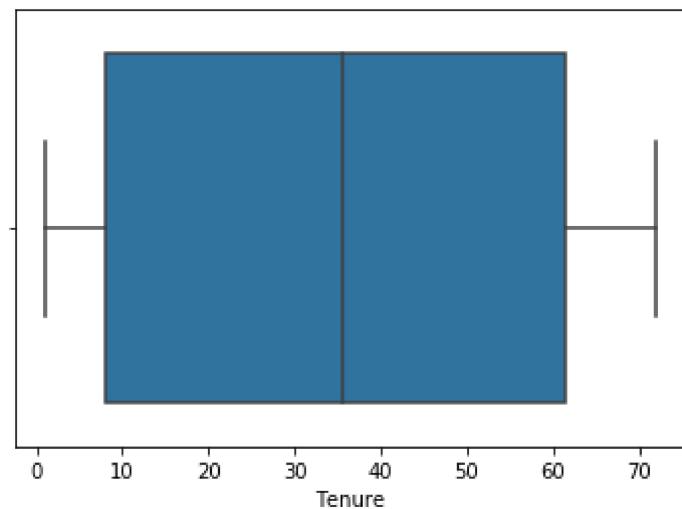
```
Out[24]: <matplotlib.axes._subplots.AxesSubplot at 0x191ffd50088>
```



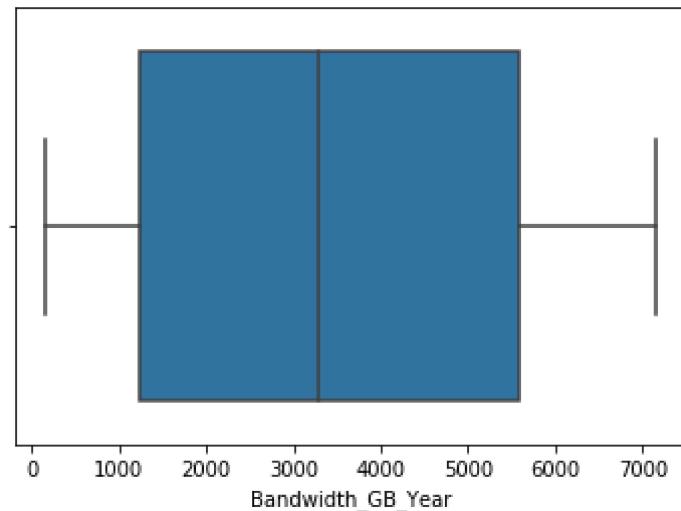
```
In [25]: sns.boxplot('MonthlyCharge', data = churn_df)  
plt.show()
```



```
In [26]: sns.boxplot('Tenure', data = churn_df)  
plt.show()
```



```
In [27]: sns.boxplot('Bandwidth_GB_Year', data = churn_df)  
plt.show()
```



```
In [29]: # Find null values  
data_null = churn_df.isnull().sum()  
print(data_null)
```

```
Customer_id          0  
Interaction         0  
UID                 0  
City                0  
State               0  
County              0  
Zip                 0  
Lat                 0  
Lng                 0  
Population          0  
Area                0  
TimeZone            0  
Job                 0  
Children            0  
Age                 0  
Income              0  
Marital             0  
Gender              0  
Churn               0  
Outage_sec_perweek 0  
Email               0  
Contacts            0  
Yearly_equip_failure 0  
Techie              0  
Contract            0  
Port_modem          0  
Tablet              0  
InternetService     0  
Phone               0  
Multiple            0  
OnlineSecurity      0  
OnlineBackup         0  
DeviceProtection    0  
TechSupport          0  
StreamingTV         0  
StreamingMovies     0  
PaperlessBilling    0  
PaymentMethod       0  
Tenure              0  
MonthlyCharge       0  
Bandwidth_GB_Year   0  
TimelyResponse      0  
Fixes               0  
Replacements        0  
Reliability         0  
Options              0  
Respectful          0  
Courteous            0  
Listening            0  
dtype: int64
```

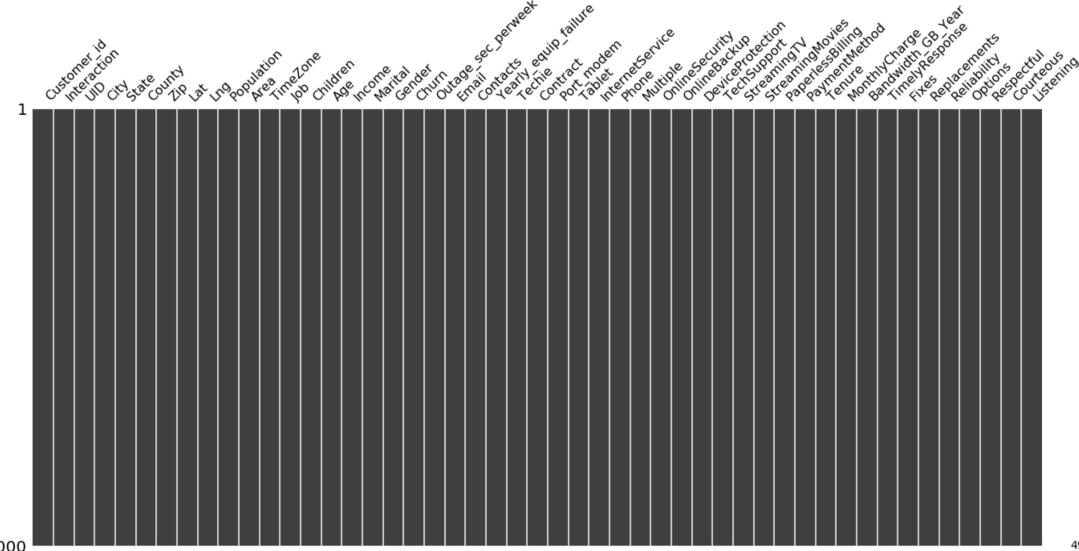
```
In [32]: # Look for missing data
```

```
# Install library
!pip install missingno

# Importing Libraries
import missingno as msno

# Visualize missing values
msno.matrix(churn_df);
```

Requirement already satisfied: missingno in c:\users\coope\anaconda3\lib\site-packages (0.5.0)
Requirement already satisfied: matplotlib in c:\users\coope\anaconda3\lib\site-packages (from missingno) (3.1.1)
Requirement already satisfied: seaborn in c:\users\coope\anaconda3\lib\site-packages (from missingno) (0.9.0)
Requirement already satisfied: numpy in c:\users\coope\anaconda3\lib\site-packages (from missingno) (1.16.5)
Requirement already satisfied: scipy in c:\users\coope\anaconda3\lib\site-packages (from missingno) (1.3.1)
Requirement already satisfied: cycler>=0.10 in c:\users\coope\anaconda3\lib\site-packages (from matplotlib->missingno) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\coope\anaconda3\lib\site-packages (from matplotlib->missingno) (1.1.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users\coope\anaconda3\lib\site-packages (from matplotlib->missingno) (2.4.2)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\coope\anaconda3\lib\site-packages (from matplotlib->missingno) (2.8.0)
Requirement already satisfied: pandas>=0.15.2 in c:\users\coope\anaconda3\lib\site-packages (from seaborn->missingno) (0.25.1)
Requirement already satisfied: six in c:\users\coope\anaconda3\lib\site-packages (from cycler>=0.10->matplotlib->missingno) (1.12.0)
Requirement already satisfied: setuptools in c:\users\coope\anaconda3\lib\site-packages (from kiwisolver>=1.0.1->matplotlib->missingno) (41.4.0)
Requirement already satisfied: pytz>=2017.2 in c:\users\coope\anaconda3\lib\site-packages (from pandas>=0.15.2->seaborn->missingno) (2019.3)



```
In [33]: # Dummy variables for qualitative binary data
churn_df['DummyGender'] = [1 if v == 'Male' else 0 for v in churn_df['Gender']]
churn_df['DummyChurn'] = [1 if v == 'Yes' else 0 for v in churn_df['Churn']] # ## If the customer left (churned) they get a '1'
churn_df['DummyTechie'] = [1 if v == 'Yes' else 0 for v in churn_df['Techie']]
churn_df['DummyContract'] = [1 if v == 'Two Year' else 0 for v in churn_df['Contract']]
churn_df['DummyPort_modem'] = [1 if v == 'Yes' else 0 for v in churn_df['Port_modem']]
churn_df['DummyTablet'] = [1 if v == 'Yes' else 0 for v in churn_df['Tablet']]
churn_df['DummyInternetService'] = [1 if v == 'Fiber Optic' else 0 for v in churn_df['InternetService']]
churn_df['DummyPhone'] = [1 if v == 'Yes' else 0 for v in churn_df['Phone']]
churn_df['DummyMultiple'] = [1 if v == 'Yes' else 0 for v in churn_df['Multiple']]
churn_df['DummyOnlineSecurity'] = [1 if v == 'Yes' else 0 for v in churn_df['OnlineSecurity']]
churn_df['DummyOnlineBackup'] = [1 if v == 'Yes' else 0 for v in churn_df['OnlineBackup']]
churn_df['DummyDeviceProtection'] = [1 if v == 'Yes' else 0 for v in churn_df['DeviceProtection']]
churn_df['DummyTechSupport'] = [1 if v == 'Yes' else 0 for v in churn_df['TechSupport']]
churn_df['DummyStreamingTV'] = [1 if v == 'Yes' else 0 for v in churn_df['StreamingTV']]
churn_df['StreamingMovies'] = [1 if v == 'Yes' else 0 for v in churn_df['StreamingMovies']]
churn_df['DummyPaperlessBilling'] = [1 if v == 'Yes' else 0 for v in churn_df['PaperlessBilling']]
```

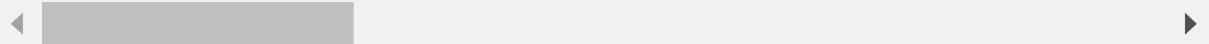
```
In [34]: # Drop original data for dummy data
churn_df = churn_df.drop(columns=['Gender', 'Churn', 'Techie', 'Contract', 'Port_modem', 'Tablet',
 'InternetService', 'Phone', 'Multiple', 'OnlineSecurity',
 'OnlineBackup', 'DeviceProtection', 'TechSupport',
 'StreamingTV', 'StreamingMovies', 'PaperlessBilling'])
```

In [35]: `churn_df.head()`

Out[35]:

	Customer_id	Interaction	UID	City	State	Co
CaseOrder						
1	K409198	aa90260b-4141-4a24-8e36-b04ce1f4f77b	e885b299883d4f9fb18e39c75155d990	Point Baker	AK	Pri V
2	S120509	fb76459f-c047-4a9d-8af9-e0f7d4ac2524	f2de8bef964785f41a2959829830fb8a	West Branch	MI	Og
3	K191035	344d114c-3736-4be5-98f7-c72c281e2d35	f1784cfa9f6d92ae816197eb175d3c71	Yamhill	OR	Y
4	D90850	abfa2b40-2d43-4994-b15a-989b8c79e311	dc8a365077241bb5cd5ccd305136b05e	Del Mar	CA	
5	K662701	68a861fd-0d20-4e51-a587-8a90407ee574	aabb64a116e83fdc4befc1fbab1663f9	Needville	TX	

5 rows × 48 columns



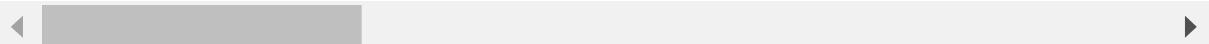
In [37]: `# Drop less meaningful data`

```
churn_df = churn_df.drop(columns=['Customer_id', 'Interaction', 'UID',
 'City', 'State', 'County', 'Zip', 'Lat', 'Lng',
 'Area', 'TimeZone', 'Job', 'Marital', 'PaymentMethod'])
churn_df.head()
```

Out[37]:

	Population	Children	Age	Income	Outage_sec_perweek	Email	Contacts	Yearly_e
CaseOrder								
1	38	0	68	28561.99	7.978323	10	0	
2	10446	1	27	21704.77	11.699080	12	0	
3	3735	4	50	9609.57	10.752800	9	0	
4	13863	1	48	18925.23	14.913540	15	2	
5	11352	0	83	40074.19	8.147417	16	2	

5 rows × 34 columns



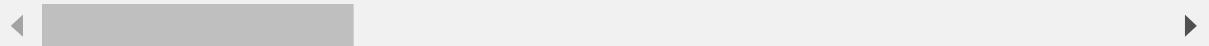
```
In [39]: # Move DummyChurn to end
churn_df = churn_df[['Children', 'Age', 'Income', 'Outage_sec_perweek', 'Email',
'Contacts', 'Yearly_equip_failure', 'Tenure', 'MonthlyCharge', 'Bandwidth_GB_Year',
'TimelyResponse', 'Fixes', 'Replacements', 'Reliability', 'Options', 'Respectful', 'Courteous', 'Listening',
'DummyGender', 'DummyTechie', 'DummyContract', 'DummyPort_modem', 'DummyTablet', 'DummyInternetService', 'DummyPhone',
'DummyMultiple', 'DummyOnlineSecurity', 'DummyOnlineBackup', 'DummyDeviceProtection', 'DummyTechSupport', 'DummyStreamingTV',
'DummyPaperlessBilling', 'DummyChurn']]
```

```
In [40]: churn_df.head()
```

Out[40]:

	CaseOrder	Children	Age	Income	Outage_sec_perweek	Email	Contacts	Yearly_equip_failure
1	0	68	28561.99		7.978323	10	0	1
2	1	27	21704.77		11.699080	12	0	1
3	4	50	9609.57		10.752800	9	0	1
4	1	48	18925.23		14.913540	15	2	0
5	0	83	40074.19		8.147417	16	2	1

5 rows × 33 columns



```
In [41]: # List features
features = (list(churn_df.columns[:-1]))
print('Features for analysis include: \n', features)
```

Features for analysis include:

```
['Children', 'Age', 'Income', 'Outage_sec_perweek', 'Email', 'Contacts', 'Yearly_equip_failure', 'Tenure', 'MonthlyCharge', 'Bandwidth_GB_Year', 'TimelyResponse', 'Fixes', 'Replacements', 'Reliability', 'Options', 'Respectful', 'Courteous', 'Listening', 'DummyGender', 'DummyTechie', 'DummyContract', 'DummyPort_modem', 'DummyTablet', 'DummyInternetService', 'DummyPhone', 'DummyMultiple', 'DummyOnlineSecurity', 'DummyOnlineBackup', 'DummyDeviceProtection', 'DummyTechSupport', 'DummyStreamingTV', 'DummyPaperlessBilling']
```

```
In [42]: #Extract cleaned dataset
churn_df.to_csv('churn_prepared.csv')
```

```
In [43]: # Re-read dataset
churn_df = pd.read_csv('churn_prepared.csv')

# Set predictor variables
X = churn_df.drop('DummyChurn', axis=1).values
y = churn_df['DummyChurn'].values
```

```
In [44]: # Import sklearn packages
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import cross_val_score, train_test_split
```

```
In [45]: # Set seed
SEED = 1
# Create training and test sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = SEED)
```

```
In [46]: # KNN model
knn = KNeighborsClassifier(n_neighbors = 7)

# Fit data
knn.fit(X_train, y_train)

# Predict outcomes from test set
y_pred = knn.predict(X_test)
```

```
In [47]: # Print initial score of KNN model
print('Initial accuracy score KNN model: ', accuracy_score(y_test, y_pred))
```

Initial accuracy score KNN model: 0.7145

```
In [48]: # Compute classification metrics
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.78	0.83	0.81	1442
1	0.49	0.40	0.44	558
accuracy			0.71	2000
macro avg	0.63	0.62	0.62	2000
weighted avg	0.70	0.71	0.71	2000

```
In [49]: # Create pipeline object
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.metrics import accuracy_score
```

```
In [50]: # Set steps
steps = [('scaler', StandardScaler()), ('knn', KNeighborsClassifier())]
```

```
In [51]: # Instantiate pipeline
pipeline = Pipeline(steps)
```

```
In [52]: # Split dataframe
X_train_scaled, X_test_scaled, y_train_scaled, y_test_scaled = train_test_split(X, y, test_size = 0.2, random_state = SEED)
```

```
In [53]: # Scale dateframe with pipeline
knn_scaled = pipeline.fit(X_train_scaled, y_train_scaled)
```

```
In [54]: # Predict from scaled dataframe
y_pred_scaled = pipeline.predict(X_test_scaled)
```

```
In [55]: print('New accuracy score of scaled KNN model: {:.3f}'.format(accuracy_score(y_test_scaled, y_pred_scaled)))
```

New accuracy score of scaled KNN model: 0.790

```
In [56]: # After scaling
print(classification_report(y_test_scaled, y_pred_scaled))
```

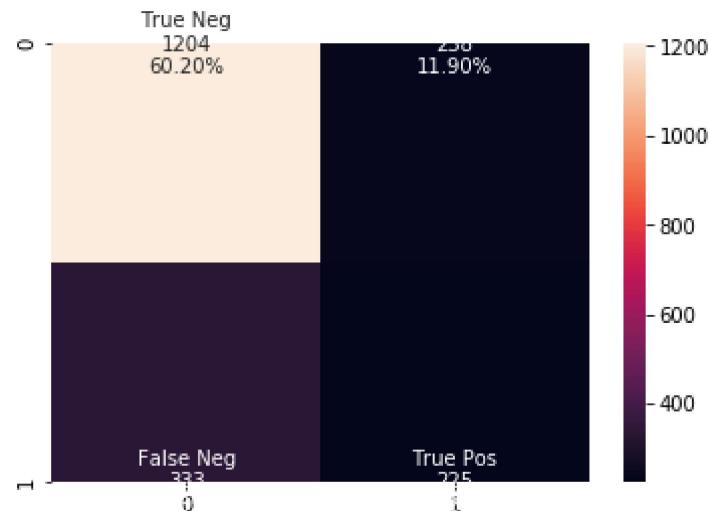
	precision	recall	f1-score	support
0	0.84	0.88	0.86	1442
1	0.64	0.56	0.60	558
accuracy			0.79	2000
macro avg	0.74	0.72	0.73	2000
weighted avg	0.78	0.79	0.79	2000

```
In [57]: # Import confusion_matrix
from sklearn.metrics import confusion_matrix
cf_matrix = confusion_matrix(y_test, y_pred)
print(cf_matrix)
```

```
[[1204 238]
 [ 333 225]]
```

```
In [58]: # Create a visually more intuitive confusion matrix
group_names = ['True Neg', 'False Pos', 'False Neg', 'True Pos']
group_counts = ["{0:0.0f}".format(value) for value in
    cf_matrix.flatten()]
group_percentages = ["{0:.2%}".format(value) for value in
    cf_matrix.flatten()/np.sum(cf_matrix)]
labels = [f"\n{v1}\n{v2}\n{v3}" for v1, v2, v3 in
    zip(group_names,group_counts,group_percentages)]
labels = np.asarray(labels).reshape(2,2)
sns.heatmap(cf_matrix, annot=labels, fmt='')
```

Out[58]: <matplotlib.axes._subplots.AxesSubplot at 0x19182ca6c88>



```
In [59]: # Import GridSearchCV for cross validation of model
from sklearn.model_selection import GridSearchCV
```

```
In [60]: # Set up parameters grid
param_grid = {'n_neighbors': np.arange(1, 50)}
```

```
In [61]: # Re-intantiate KNN for cross validation
knn = KNeighborsClassifier()
```

```
In [62]: # Instantiate GridSearch cross validation
knn_cv = GridSearchCV(knn, param_grid, cv=5)
```

```
In [63]: # Fit model
knn_cv.fit(X_train, y_train)
```

```
Out[63]: GridSearchCV(cv=5, error_score='raise-deprecating',
                      estimator=KNeighborsClassifier(algorithm='auto', leaf_size=30,
                                                     metric='minkowski',
                                                     metric_params=None, n_jobs=None,
                                                     n_neighbors=5, p=2,
                                                     weights='uniform'),
                      iid='warn', n_jobs=None,
                      param_grid={'n_neighbors': array([ 1,  2,  3,  4,  5,  6,  7,
                           8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
                           18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
                           35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49])},
                      pre_dispatch='2*n_jobs', refit=True, return_train_score=False,
                      scoring=None, verbose=0)
```

```
In [64]: print('Best parameters for this KNN model: {}'.format(knn_cv.best_params_))
```

Best parameters for this KNN model: {'n_neighbors': 6}

```
In [65]: # Best score
print('Best score for this KNN model: {:.3f}'.format(knn_cv.best_score_))
```

Best score for this KNN model: 0.735

```
In [66]: # Import ROC AUC metrics
from sklearn.metrics import roc_auc_score
```

```
In [67]: # Fit into data
knn_cv.fit(X, y)
```

```
Out[67]: GridSearchCV(cv=5, error_score='raise-deprecating',
                      estimator=KNeighborsClassifier(algorithm='auto', leaf_size=30,
                                                     metric='minkowski',
                                                     metric_params=None, n_jobs=None,
                                                     n_neighbors=5, p=2,
                                                     weights='uniform'),
                      iid='warn', n_jobs=None,
                      param_grid={'n_neighbors': array([ 1,  2,  3,  4,  5,  6,  7,
                           8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
                           18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
                           35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49])},
                      pre_dispatch='2*n_jobs', refit=True, return_train_score=False,
                      scoring=None, verbose=0)
```

```
In [68]: # Compute predicted probabilities
y_pred_prob = knn_cv.predict_proba(X_test)[:,1]
```

```
In [69]: # AUC score
print("The Area under curve (AUC) on validation dataset is: {:.4f}".format(roc_auc_score(y_test, y_pred_prob)))
```

The Area under curve (AUC) on validation dataset is: 0.7959

```
In [70]: # Compute cross-validated AUC scores  
cv_auc = cross_val_score(knn_cv, X, y, cv=5, scoring='roc_auc')
```

```
In [71]: print("AUC scores computed using 5-fold cross-validation: {}".format(cv_auc))
```

```
AUC scores computed using 5-fold cross-validation: [0.68120909 0.17406045 0.9  
6370684 0.96560711 0.58834745]
```

```
In [ ]:
```