In [9]:

```
import pandas as pd
df = pd.read_csv('SmokingDataSet.csv')
df.head()
```

Out[9]:

	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type
0	Male	67.0	0	1	Yes	Private	Urban
1	Male	80.0	0	1	Yes	Private	Rural
2	Female	49.0	0	0	Yes	Private	Urban
3	Female	79.0	1	0	Yes	Self-employed	Rural
4	Male	81.0	0	0	Yes	Private	Urban

In [10]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4981 entries, 0 to 4980
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   gender             4981 non-null   object 
 1   age                4981 non-null   float64
 2   hypertension       4981 non-null   int64  
 3   heart_disease      4981 non-null   int64  
 4   ever_married       4981 non-null   object 
 5   work_type          4981 non-null   object 
 6   Residence_type     4981 non-null   object 
 7   avg_glucose_level  4981 non-null   float64
 8   bmi                4981 non-null   float64
 9   smoking_status     4981 non-null   object 
 10  stroke             4981 non-null   int64  
dtypes: float64(3), int64(3), object(5)
memory usage: 428.2+ KB
```

In [11]:

```
objectList = list(df.select_dtypes(include='object'))
objectList
```

Out[11]:

```
['gender', 'ever_married', 'work_type', 'Residence_type', 'smoking_status
']
```

In [12]:

```
from sklearn import preprocessing
for i in objectList:
    Encoder = preprocessing.LabelEncoder()
    df[i]= Encoder.fit_transform(df[i])
```

In [13]:

```
df.isnull().sum()
```

```
Out[13]: gender               0
         age                  0
         hypertension         0
         heart_disease        0
         ever_married         0
         work_type            0
         Residence_type       0
         avg_glucose_level    0
         bmi                  0
         smoking_status       0
         stroke               0
         dtype: int64
```

```
In [14]: x = df.drop(columns=['stroke'],axis=1)
         y = df['stroke']
```

```
In [15]: from imblearn.over_sampling import RandomOverSampler
         over_sampler = RandomOverSampler(sampling_strategy='minority')
         x,y = over_sampler.fit_resample(x,y)
```

```
In [16]: from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20,
```

```
In [17]: from sklearn import svm
         model_svm = svm.SVC()
         model_svm.fit(x_train,y_train)
         y_pred = model_svm.predict(x_test)
```

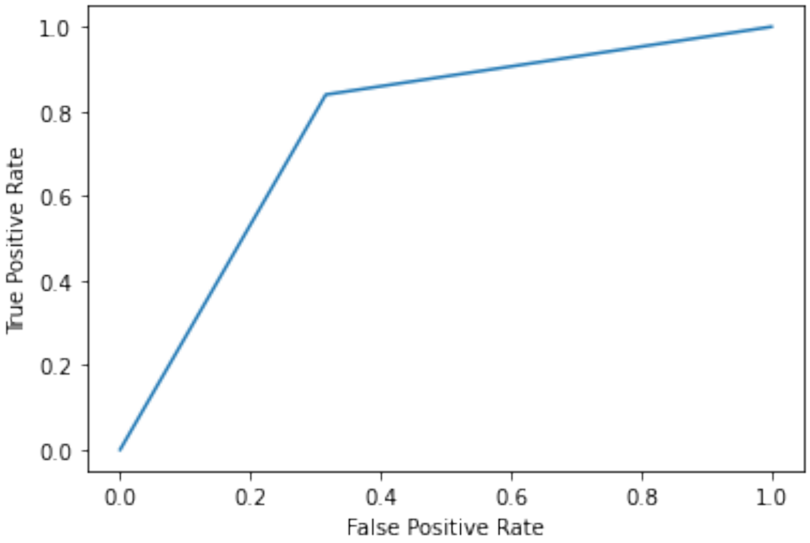
```
In [18]: from sklearn.metrics import confusion_matrix
         cm_log = confusion_matrix(y_test,y_pred)
         cm_log
```

```
Out[18]: array([[648, 299],
                [152, 795]])
```

```
In [19]: from sklearn.metrics import roc_auc_score, roc_curve
         import matplotlib.pyplot as plt

         def plot_roc_curve(y_test,y_pred):
             fpr, tpr, thresholds = roc_curve(y_test,y_pred)
             plt.plot(fpr, tpr)
             plt.xlabel('False Positive Rate')
             plt.ylabel('True Positive Rate')
         plot_roc_curve(y_test,y_pred)
         print(f'model(SVM) AUC score: {roc_auc_score(y_test, y_pred)}')
```

```
model(SVM) AUC score: 0.7618796198521647
```


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate