In [1]:

```
# Supress Warnings
import warnings
warnings.filterwarnings('ignore')

# Importing libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# visulaisation
from matplotlib.pyplot import xticks
%matplotlib inline

# Data display coustomization
#pd.set_option('display.max_rows', 50)
#pd.set_option('display.max_columns', 50)
```

In [2]:

```
merged_sample = pd.read_csv('../preprocessed_data.csv')
merged_sample.head(5)
```

Out[2]:

	Unnamed: 0	Unnamed: 0.1	ID	auth_3mth_post_acute_dia	rx_gpi2_72_pmpm_cost_6to9m_b4	atlas_pct_laccess_child15	atlas_recfacpth14	at
0	0	0.000011	0.703111	1.0	0.0	0.126735	0.073774	
1	1	0.000014	0.774533	1.0	0.0	0.128916	0.157680	
2	2	0.000017	0.946569	1.0	0.0	0.037677	0.017380	
3	3	0.000019	0.108334	1.0	0.0	0.128178	0.163629	
4	4	0.000024	0.502195	1.0	0.0	0.231772	0.158584	

5 rows × 368 columns

In [3]:

```
merged_sample_copy = merged_sample.copy()
train = merged_sample.drop(columns=['covid_vaccination'])
test = merged_sample_copy[['covid_vaccination']]
train.head()
```

Out[3]:

	Unnamed: 0	Unnamed: 0.1	ID	auth_3mth_post_acute_dia	rx_gpi2_72_pmpm_cost_6to9m_b4	atlas_pct_laccess_child15	atlas_recfacpth14	at
0	0	0.000011	0.703111	1.0	0.0	0.126735	0.073774	
1	1	0.000014	0.774533	1.0	0.0	0.128916	0.157680	
2	2	0.000017	0.946569	1.0	0.0	0.037677	0.017380	
3	3	0.000019	0.108334	1.0	0.0	0.128178	0.163629	
4	4	0.000024	0.502195	1.0	0.0	0.231772	0.158584	

5 rows × 367 columns

In [4]:

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(train, test, test_size=0.3, random_state=0)
```

In [5]:

```
y_train.value_counts()
```

Out[5]:

```
covid_vaccination
0                    118838
1                    118396
dtype: int64
```

In [6]:

```
y_test['covid_vaccination'].value_counts()
```

Out[6]:

```
1    51057
0    50615
Name: covid_vaccination, dtype: int64
```

In [7]:

```
#Import random forest model
```

```
from sklearn.ensemble import RandomForestClassifier

#Create a Gaussian Classifier
clf=RandomForestClassifier(n_estimators=100)

#Train the model using the training sets y_pred=clf.predict(X_test)
clf.fit(x_train,y_train)

# prediction on test set
preds=clf.predict(x_test)
```

In [8]:

```
import numpy
print(numpy.unique(preds))
preds
print(numpy.count_nonzero(preds == 1))
print(numpy.count_nonzero(preds == 0))
print(numpy.size)
y_test['covid_vaccination'].value_counts()
```

```
[0 1]
51056
50616
<function size at 0x000001A54102AB80>
```

Out[8]:

```
1    51057
0    50615
Name: covid_vaccination, dtype: int64
```

In [9]:

```
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
def elavutaionmetrix(x_train,y_train,y_test, preds):
    print(classification_report(y_test,preds))
    print("train accuracy:",clf.score(x_train,y_train))
    print("Test accuracy:",accuracy_score(y_test, preds))
```

In [10]:

```
elavutaionmetrix(x_train,y_train,y_test, preds)
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     50615
           1       1.00      1.00      1.00     51057

    accuracy                           1.00    101672
   macro avg       1.00      1.00      1.00    101672
weighted avg       1.00      1.00      1.00    101672

train accuracy: 1.0
Test accuracy: 0.9999901644503895
```

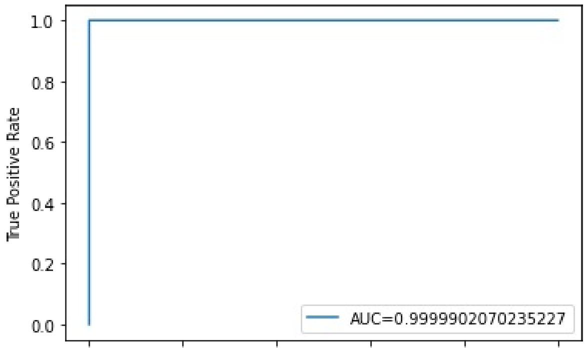
In [11]:

```
from sklearn import metrics
def printroccurve(y_test,  preds):
    fpr, tpr, _ = metrics.roc_curve(y_test,  preds)
    auc = metrics.roc_auc_score(y_test, preds)

    #create ROC curve
    plt.plot(fpr,tpr,label="AUC="+str(auc))
    plt.ylabel('True Positive Rate')
    plt.xlabel('False Positive Rate')
    plt.legend(loc=4)
    plt.show()
```

In [12]:

```
printroccurve(y_test,  preds)
```



In [14]:

```
testdataframe=pd.read_csv('preprocessed_holdout.csv',low_memory=False)
```

In [15]:

```
preds=clf.predict(testdataframe)
#merging input data with prediction
testdataframe['covid_vaccination'] = preds
```

In [16]:

```
testdataframe.head(5)
```

Out[16]:

	Unnamed: 0	Unnamed: 0.1	ID	auth_3mth_post_acute_dia	rx_gpi2_72_pmpm_cost_6to9m_b4	atlas_pct_laccess_child15	atlas_recfacpth14	at
0	0	0.000000	0.230887	1.0	0.0	0.312471	0.215479	
1	1	0.000002	0.022477	1.0	0.0	0.201069	0.123538	
2	2	0.000004	0.046047	1.0	0.0	0.196946	0.174766	
3	3	0.000006	0.510482	1.0	0.0	0.039948	0.000000	
4	4	0.000008	0.176064	1.0	0.0	0.257079	0.100361	

5 rows × 368 columns

In [17]:

```
testdataframe['covid_vaccination'].value_counts()
```

Out[17]:

```
0    355730
1    169428
Name: covid_vaccination, dtype: int64
```

In [ ]:

```
testdataframe.to_csv("randomforest_holdout.csv")
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js