

Blood clot prediction

July 8, 2025

```
[3]: #install scikit-learn
!pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in
c:\users\admin\anaconda3\lib\site-packages (1.4.2)
Requirement already satisfied: numpy>=1.19.5 in
c:\users\admin\anaconda3\lib\site-packages (from scikit-learn) (1.26.4)
Requirement already satisfied: scipy>=1.6.0 in
c:\users\admin\anaconda3\lib\site-packages (from scikit-learn) (1.13.1)
Requirement already satisfied: joblib>=1.2.0 in
c:\users\admin\anaconda3\lib\site-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in
c:\users\admin\anaconda3\lib\site-packages (from scikit-learn) (2.2.0)
```

```
[71]: #import the required libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, \
    accuracy_score
from imblearn.over_sampling import SMOTE
```

```
[73]: df=pd.read_csv("blood_clot_data.csv")
df
```

```
[73]:
```

	Age	D_dimer	Platelet_Count	Blood_Pressure	Cholesterol	Smoking	\
0	69	228.155578	218113.000787	109.496820	238.778288	0	
1	32	278.655285	223450.152250	126.628519	217.087774	1	
2	89	331.090757	218842.973679	112.448305	174.150908	0	
3	78	447.535622	222226.144042	102.970293	271.012436	1	
4	38	385.765962	218130.643635	138.121941	152.254527	0	
..	
495	34	338.240975	309532.313739	131.392674	178.958606	1	
496	88	316.645221	297477.706772	132.397846	113.866283	0	
497	62	349.245126	175755.101578	114.584873	243.886108	0	
498	21	328.916864	122303.943255	148.985618	180.846502	1	

```
499    53  545.530014   296715.995562    117.007377   165.488969    1
```

```

      Diabetes  Clot_Present
0           0           0
1           0           0
2           0           0
3           1           1
4           1           0
..          ...           ...
495          0           0
496          0           0
497          0           0
498          0           1
499          0           0

```

```
[500 rows x 8 columns]
```

```
[75]: #display the first 10 rows
df.head(10)
```

```
[75]:   Age      D_dimer  Platelet_Count  Blood_Pressure  Cholesterol  Smoking \
0    69  228.155578   218113.000787    109.496820    238.778288      0
1    32  278.655285   223450.152250    126.628519    217.087774      1
2    89  331.090757   218842.973679    112.448305    174.150908      0
3    78  447.535622   222226.144042    102.970293    271.012436      1
4    38  385.765962   218130.643635    138.121941    152.254527      0
5    41  284.006147   309450.826555    141.387327    236.766167      0
6    20  298.098379   321025.212399    121.352344    240.023293      0
7    39  199.747064   221462.685313     91.134366    173.175192      1
8    70  298.148686   208382.221345    121.806333    255.698612      1
9    19  271.134136   273570.777819    135.877060    189.998139      0
```

```

      Diabetes  Clot_Present
0           0           0
1           0           0
2           0           0
3           1           1
4           1           0
5           0           0
6           1           0
7           0           0
8           1           0
9           0           0

```

```
[77]: #display the features of the dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

RangeIndex: 500 entries, 0 to 499

Data columns (total 8 columns):

#	Column	Non-Null Count	Dtype
0	Age	500 non-null	int64
1	D_dimer	500 non-null	float64
2	Platelet_Count	500 non-null	float64
3	Blood_Pressure	500 non-null	float64
4	Cholesterol	500 non-null	float64
5	Smoking	500 non-null	int64
6	Diabetes	500 non-null	int64
7	Clot_Present	500 non-null	int64

dtypes: float64(4), int64(4)

memory usage: 31.4 KB

```
[79]: #look for any null values
df.any().isnull()
```

```
[79]: Age                False
D_dimer              False
Platelet_Count       False
Blood_Pressure       False
Cholesterol          False
Smoking              False
Diabetes              False
Clot_Present         False
dtype: bool
```

```
[81]: #display the descriptive statistics
df.describe()
```

```
[81]:
```

	Age	D_dimer	Platelet_Count	Blood_Pressure	Cholesterol	\
count	500.000000	500.000000	500.000000	500.000000	500.000000	
mean	52.930000	300.011222	254931.431765	130.812426	200.811921	
std	21.009519	99.759627	49522.416610	14.924935	39.314721	
min	18.000000	30.311336	105187.231090	86.179743	79.219514	
25%	34.000000	229.653176	221457.276612	120.820958	175.336626	
50%	52.000000	298.123533	255635.170822	130.341483	200.264442	
75%	71.000000	364.028833	285683.010205	140.797155	226.767056	
max	89.000000	607.888081	378985.466883	177.896614	325.509941	

	Smoking	Diabetes	Clot_Present
count	500.000000	500.000000	500.0000
mean	0.280000	0.212000	0.1000
std	0.449449	0.409134	0.3003
min	0.000000	0.000000	0.0000
25%	0.000000	0.000000	0.0000
50%	0.000000	0.000000	0.0000

75%	1.000000	0.000000	0.0000
max	1.000000	1.000000	1.0000

```
[83]: df['Clot_Present'].value_counts()#check for balance
```

```
[83]: Clot_Present
0    450
1     50
Name: count, dtype: int64
```

```
[85]: # Features and target
X = df.drop('Clot_Present', axis=1)
y = df['Clot_Present']
```

```
[87]: X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,
↳ random_state=42)#split data
```

```
[137]: #train the model

smote = SMOTE()
X_resampled, y_resampled = smote.fit_resample(X_train, y_train)

from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier()
model.fit(X_resampled,y_resampled)
```

```
[137]: RandomForestClassifier()
```

```
[139]: #make predictions
y_pred=model.predict(X_test)
```

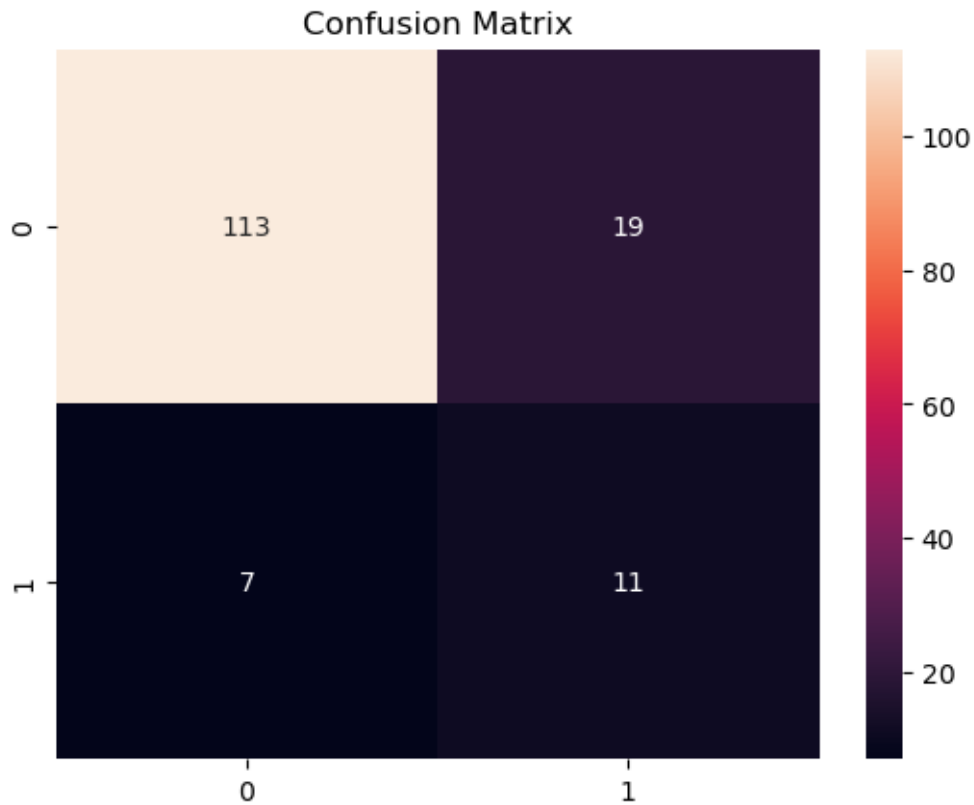
```
[141]: #evaluate model
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
print("Accuracy:",accuracy_score(y_test,y_pred))
```

```
[[113  19]
 [ 7  11]]
```

	precision	recall	f1-score	support
0	0.94	0.86	0.90	132
1	0.37	0.61	0.46	18
accuracy			0.83	150
macro avg	0.65	0.73	0.68	150
weighted avg	0.87	0.83	0.84	150

Accuracy: 0.8266666666666667

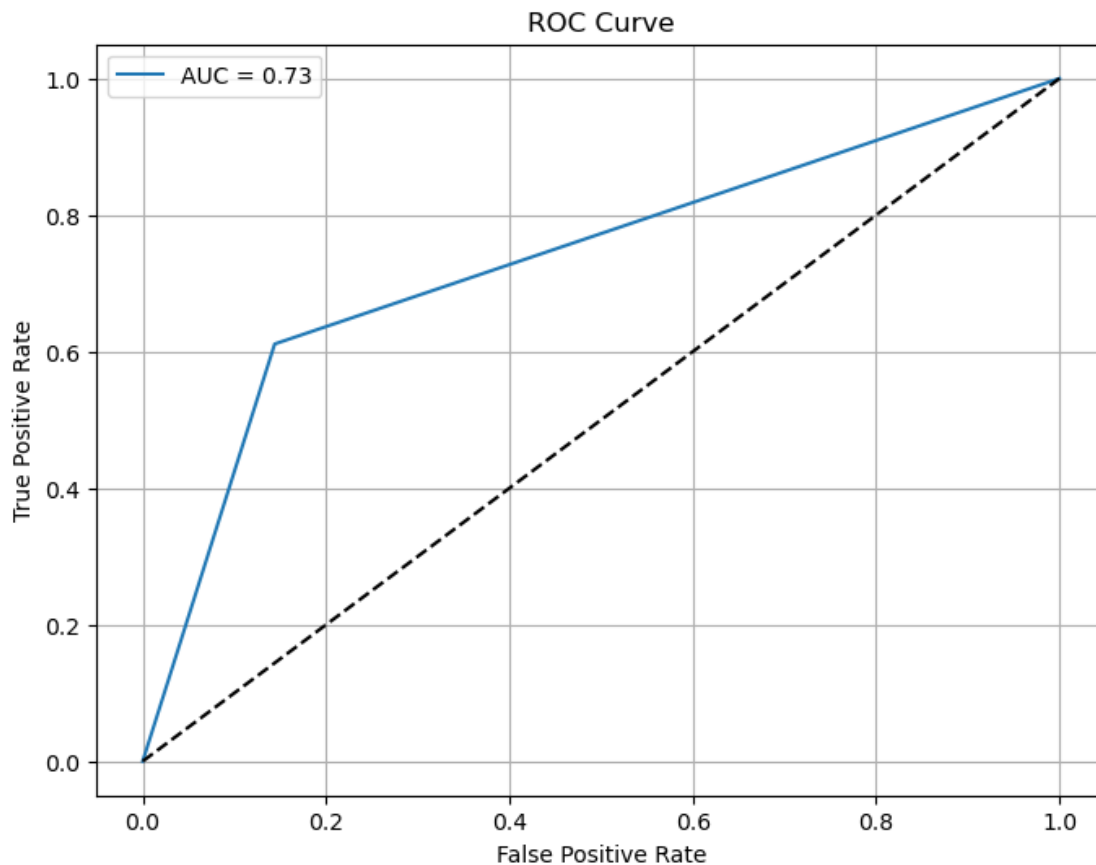
```
[143]: sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, fmt='d')
plt.title("Confusion Matrix")
plt.show()
```



```
[144]: from sklearn.metrics import roc_curve, roc_auc_score

fpr, tpr, thresholds = roc_curve(y_test, y_pred)
auc = roc_auc_score(y_test, y_pred)

plt.figure(figsize=(8,6))
plt.plot(fpr, tpr, label="AUC = {:.2f}".format(auc))
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curve")
plt.legend()
plt.grid(True)
plt.show()
```



We want to make our model more accurate so lets try and use XGBOOST

```
[149]: !pip install xgboost
```

Collecting xgboost

Downloading xgboost-3.0.2-py3-none-win_amd64.whl.metadata (2.1 kB)

Requirement already satisfied: numpy in c:\users\admin\anaconda3\lib\site-packages (from xgboost) (1.26.4)

Requirement already satisfied: scipy in c:\users\admin\anaconda3\lib\site-packages (from xgboost) (1.13.1)

Downloading xgboost-3.0.2-py3-none-win_amd64.whl (150.0 MB)

```
----- 0.0/150.0 MB ? eta -:-:--
----- 0.0/150.0 MB 1.3 MB/s eta 0:01:58
----- 0.1/150.0 MB 544.7 kB/s eta 0:04:36
----- 0.1/150.0 MB 726.2 kB/s eta 0:03:27
----- 0.1/150.0 MB 599.1 kB/s eta 0:04:11
----- 0.3/150.0 MB 1.0 MB/s eta 0:02:27
----- 0.4/150.0 MB 1.1 MB/s eta 0:02:15
----- 0.6/150.0 MB 1.5 MB/s eta 0:01:40
----- 0.6/150.0 MB 1.7 MB/s eta 0:01:30
```

```

----- 1.0/150.0 MB 2.1 MB/s eta 0:01:12
----- 1.3/150.0 MB 2.6 MB/s eta 0:00:58
----- 1.6/150.0 MB 3.0 MB/s eta 0:00:50
----- 2.2/150.0 MB 3.7 MB/s eta 0:00:40
----- 2.7/150.0 MB 4.2 MB/s eta 0:00:35
----- 3.3/150.0 MB 4.8 MB/s eta 0:00:31
----- 3.7/150.0 MB 5.0 MB/s eta 0:00:30
- ----- 4.2/150.0 MB 5.3 MB/s eta 0:00:28
- ----- 4.8/150.0 MB 5.6 MB/s eta 0:00:26
- ----- 5.3/150.0 MB 6.0 MB/s eta 0:00:25
- ----- 5.8/150.0 MB 6.3 MB/s eta 0:00:23
- ----- 6.4/150.0 MB 6.6 MB/s eta 0:00:22
- ----- 7.1/150.0 MB 7.0 MB/s eta 0:00:21
-- ----- 7.7/150.0 MB 7.1 MB/s eta 0:00:21
-- ----- 8.5/150.0 MB 7.5 MB/s eta 0:00:19
-- ----- 9.0/150.0 MB 7.6 MB/s eta 0:00:19
-- ----- 9.6/150.0 MB 7.8 MB/s eta 0:00:19
-- ----- 10.1/150.0 MB 7.9 MB/s eta 0:00:18
-- ----- 10.6/150.0 MB 10.2 MB/s eta 0:00:14
-- ----- 11.2/150.0 MB 11.3 MB/s eta 0:00:13
--- ----- 11.8/150.0 MB 11.7 MB/s eta 0:00:12
--- ----- 12.3/150.0 MB 11.7 MB/s eta 0:00:12
--- ----- 13.1/150.0 MB 11.5 MB/s eta 0:00:12
--- ----- 13.8/150.0 MB 11.9 MB/s eta 0:00:12
--- ----- 14.4/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 15.0/150.0 MB 11.5 MB/s eta 0:00:12
---- ----- 16.1/150.0 MB 12.1 MB/s eta 0:00:12
---- ----- 16.6/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 17.2/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 17.7/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 18.3/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 18.3/150.0 MB 11.9 MB/s eta 0:00:12
---- ----- 18.6/150.0 MB 11.1 MB/s eta 0:00:12
---- ----- 19.8/150.0 MB 11.7 MB/s eta 0:00:12
---- ----- 20.7/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 21.3/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 21.9/150.0 MB 11.7 MB/s eta 0:00:11
---- ----- 22.5/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 23.0/150.0 MB 12.1 MB/s eta 0:00:11
---- ----- 23.6/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 24.2/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 24.7/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 25.3/150.0 MB 12.6 MB/s eta 0:00:10
---- ----- 25.8/150.0 MB 12.1 MB/s eta 0:00:11
---- ----- 26.3/150.0 MB 11.7 MB/s eta 0:00:11
---- ----- 27.0/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 27.8/150.0 MB 11.9 MB/s eta 0:00:11
---- ----- 28.3/150.0 MB 11.7 MB/s eta 0:00:11

```

```

----- 28.8/150.0 MB 12.8 MB/s eta 0:00:10
----- 29.4/150.0 MB 12.6 MB/s eta 0:00:10
----- 29.9/150.0 MB 12.1 MB/s eta 0:00:10
----- 30.4/150.0 MB 11.9 MB/s eta 0:00:11
----- 30.8/150.0 MB 11.9 MB/s eta 0:00:11
----- 31.6/150.0 MB 11.9 MB/s eta 0:00:10
----- 32.1/150.0 MB 11.9 MB/s eta 0:00:10
----- 32.8/150.0 MB 11.7 MB/s eta 0:00:11
----- 33.2/150.0 MB 11.7 MB/s eta 0:00:10
----- 33.8/150.0 MB 11.7 MB/s eta 0:00:10
----- 34.3/150.0 MB 11.7 MB/s eta 0:00:10
----- 34.9/150.0 MB 11.9 MB/s eta 0:00:10
----- 35.4/150.0 MB 11.7 MB/s eta 0:00:10
----- 35.9/150.0 MB 11.7 MB/s eta 0:00:10
----- 36.5/150.0 MB 11.7 MB/s eta 0:00:10
----- 37.1/150.0 MB 11.7 MB/s eta 0:00:10
----- 37.6/150.0 MB 11.7 MB/s eta 0:00:10
----- 38.2/150.0 MB 11.7 MB/s eta 0:00:10
----- 38.8/150.0 MB 11.7 MB/s eta 0:00:10
----- 39.3/150.0 MB 11.5 MB/s eta 0:00:10
----- 40.0/150.0 MB 11.7 MB/s eta 0:00:10
----- 40.6/150.0 MB 11.7 MB/s eta 0:00:10
----- 41.1/150.0 MB 11.9 MB/s eta 0:00:10
----- 41.7/150.0 MB 11.7 MB/s eta 0:00:10
----- 42.3/150.0 MB 11.7 MB/s eta 0:00:10
----- 42.9/150.0 MB 11.5 MB/s eta 0:00:10
----- 43.6/150.0 MB 11.5 MB/s eta 0:00:10
----- 44.2/150.0 MB 11.5 MB/s eta 0:00:10
----- 44.7/150.0 MB 11.5 MB/s eta 0:00:10
----- 45.2/150.0 MB 11.7 MB/s eta 0:00:09
----- 45.8/150.0 MB 11.5 MB/s eta 0:00:10
----- 46.4/150.0 MB 11.5 MB/s eta 0:00:10
----- 47.1/150.0 MB 11.5 MB/s eta 0:00:09
----- 47.7/150.0 MB 11.5 MB/s eta 0:00:09
----- 48.5/150.0 MB 11.7 MB/s eta 0:00:09
----- 48.9/150.0 MB 11.5 MB/s eta 0:00:09
----- 48.9/150.0 MB 11.5 MB/s eta 0:00:09
----- 50.0/150.0 MB 11.3 MB/s eta 0:00:09
----- 50.5/150.0 MB 11.3 MB/s eta 0:00:09
----- 50.9/150.0 MB 11.1 MB/s eta 0:00:09
----- 51.3/150.0 MB 11.1 MB/s eta 0:00:09
----- 51.5/150.0 MB 10.9 MB/s eta 0:00:10
----- 51.5/150.0 MB 10.9 MB/s eta 0:00:10
----- 52.0/150.0 MB 10.2 MB/s eta 0:00:10
----- 52.6/150.0 MB 10.2 MB/s eta 0:00:10
----- 52.9/150.0 MB 9.9 MB/s eta 0:00:10
----- 53.2/150.0 MB 9.9 MB/s eta 0:00:10
----- 53.5/150.0 MB 9.6 MB/s eta 0:00:11

```



```

----- 53.8/150.0 MB 9.5 MB/s eta 0:00:11
----- 54.1/150.0 MB 9.2 MB/s eta 0:00:11
----- 54.5/150.0 MB 9.1 MB/s eta 0:00:11
----- 54.7/150.0 MB 8.8 MB/s eta 0:00:11
----- 55.2/150.0 MB 8.7 MB/s eta 0:00:11
----- 55.5/150.0 MB 8.6 MB/s eta 0:00:11
----- 56.3/150.0 MB 8.8 MB/s eta 0:00:11
----- 56.8/150.0 MB 8.7 MB/s eta 0:00:11
----- 57.3/150.0 MB 8.6 MB/s eta 0:00:11
----- 58.1/150.0 MB 8.7 MB/s eta 0:00:11
----- 58.7/150.0 MB 8.6 MB/s eta 0:00:11
----- 59.1/150.0 MB 8.6 MB/s eta 0:00:11
----- 59.8/150.0 MB 9.0 MB/s eta 0:00:11
----- 60.3/150.0 MB 8.7 MB/s eta 0:00:11
----- 60.8/150.0 MB 8.7 MB/s eta 0:00:11
----- 61.3/150.0 MB 8.8 MB/s eta 0:00:11
----- 61.8/150.0 MB 9.6 MB/s eta 0:00:10
----- 62.9/150.0 MB 9.8 MB/s eta 0:00:09
----- 63.5/150.0 MB 10.2 MB/s eta 0:00:09
----- 64.0/150.0 MB 10.6 MB/s eta 0:00:09
----- 64.6/150.0 MB 10.9 MB/s eta 0:00:08
----- 65.4/150.0 MB 11.5 MB/s eta 0:00:08
----- 66.0/150.0 MB 11.9 MB/s eta 0:00:08
----- 66.5/150.0 MB 11.9 MB/s eta 0:00:08
----- 67.1/150.0 MB 11.9 MB/s eta 0:00:07
----- 67.7/150.0 MB 11.7 MB/s eta 0:00:08
----- 68.5/150.0 MB 11.9 MB/s eta 0:00:07
----- 69.0/150.0 MB 11.9 MB/s eta 0:00:07
----- 69.5/150.0 MB 11.9 MB/s eta 0:00:07
----- 70.1/150.0 MB 11.7 MB/s eta 0:00:07
----- 70.6/150.0 MB 11.7 MB/s eta 0:00:07
----- 71.2/150.0 MB 11.9 MB/s eta 0:00:07
----- 71.8/150.0 MB 11.9 MB/s eta 0:00:07
----- 72.3/150.0 MB 12.1 MB/s eta 0:00:07
----- 72.8/150.0 MB 11.9 MB/s eta 0:00:07
----- 73.4/150.0 MB 11.9 MB/s eta 0:00:07
----- 73.9/150.0 MB 11.7 MB/s eta 0:00:07
----- 74.5/150.0 MB 11.7 MB/s eta 0:00:07
----- 75.0/150.0 MB 11.9 MB/s eta 0:00:07
----- 75.6/150.0 MB 11.7 MB/s eta 0:00:07
----- 76.1/150.0 MB 11.7 MB/s eta 0:00:07
----- 76.8/150.0 MB 11.9 MB/s eta 0:00:07
----- 77.3/150.0 MB 11.9 MB/s eta 0:00:07
----- 77.8/150.0 MB 11.9 MB/s eta 0:00:07
----- 78.4/150.0 MB 11.9 MB/s eta 0:00:07
----- 79.0/150.0 MB 11.9 MB/s eta 0:00:06
----- 79.5/150.0 MB 11.9 MB/s eta 0:00:06
----- 80.0/150.0 MB 11.9 MB/s eta 0:00:06

```

-----	-----	80.6/150.0	MB	11.9	MB/s	eta	0:00:06
-----	-----	81.7/150.0	MB	11.9	MB/s	eta	0:00:06
-----	-----	81.9/150.0	MB	11.5	MB/s	eta	0:00:06
-----	-----	81.9/150.0	MB	11.5	MB/s	eta	0:00:06
-----	-----	82.4/150.0	MB	10.7	MB/s	eta	0:00:07
-----	-----	83.2/150.0	MB	10.6	MB/s	eta	0:00:07
-----	-----	83.7/150.0	MB	10.6	MB/s	eta	0:00:07
-----	-----	84.4/150.0	MB	10.7	MB/s	eta	0:00:07
-----	-----	85.1/150.0	MB	10.7	MB/s	eta	0:00:07
-----	-----	85.8/150.0	MB	10.9	MB/s	eta	0:00:06
-----	-----	86.4/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	86.9/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	87.5/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	88.0/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	88.6/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	89.1/150.0	MB	10.9	MB/s	eta	0:00:06
-----	-----	89.7/150.0	MB	10.9	MB/s	eta	0:00:06
-----	-----	90.2/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	90.7/150.0	MB	10.7	MB/s	eta	0:00:06
-----	-----	91.3/150.0	MB	10.9	MB/s	eta	0:00:06
-----	-----	91.9/150.0	MB	10.9	MB/s	eta	0:00:06
-----	-----	92.4/150.0	MB	12.4	MB/s	eta	0:00:05
-----	-----	92.9/150.0	MB	12.1	MB/s	eta	0:00:05
-----	-----	93.5/150.0	MB	12.1	MB/s	eta	0:00:05
-----	-----	94.0/150.0	MB	12.1	MB/s	eta	0:00:05
-----	-----	94.5/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	95.1/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	95.7/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	96.3/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	96.8/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	97.4/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	98.1/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	98.6/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	99.2/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	99.6/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	100.1/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	100.7/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	101.2/150.0	MB	11.7	MB/s	eta	0:00:05
-----	-----	101.8/150.0	MB	11.9	MB/s	eta	0:00:05
-----	-----	102.4/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	102.9/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	103.5/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	104.0/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	104.6/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	105.2/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	105.8/150.0	MB	11.9	MB/s	eta	0:00:04
-----	-----	106.4/150.0	MB	11.7	MB/s	eta	0:00:04
-----	-----	106.8/150.0	MB	11.7	MB/s	eta	0:00:04

```

----- 107.5/150.0 MB 11.7 MB/s eta 0:00:04
----- 107.9/150.0 MB 11.7 MB/s eta 0:00:04
----- 108.5/150.0 MB 11.7 MB/s eta 0:00:04
----- 109.4/150.0 MB 11.7 MB/s eta 0:00:04
----- 109.9/150.0 MB 11.9 MB/s eta 0:00:04
----- 110.6/150.0 MB 11.7 MB/s eta 0:00:04
----- 111.1/150.0 MB 11.7 MB/s eta 0:00:04
----- 111.6/150.0 MB 11.7 MB/s eta 0:00:04
----- 112.2/150.0 MB 11.7 MB/s eta 0:00:04
----- 112.7/150.0 MB 11.7 MB/s eta 0:00:04
----- 113.3/150.0 MB 11.7 MB/s eta 0:00:04
----- 113.7/150.0 MB 11.9 MB/s eta 0:00:04
----- 114.6/150.0 MB 11.7 MB/s eta 0:00:04
----- 115.1/150.0 MB 11.7 MB/s eta 0:00:03
----- 115.7/150.0 MB 11.7 MB/s eta 0:00:03
----- 116.2/150.0 MB 12.1 MB/s eta 0:00:03
----- 116.7/150.0 MB 11.9 MB/s eta 0:00:03
----- 117.3/150.0 MB 11.9 MB/s eta 0:00:03
----- 117.8/150.0 MB 11.9 MB/s eta 0:00:03
----- 118.3/150.0 MB 11.9 MB/s eta 0:00:03
----- 118.9/150.0 MB 12.1 MB/s eta 0:00:03
----- 119.6/150.0 MB 11.7 MB/s eta 0:00:03
----- 120.1/150.0 MB 11.7 MB/s eta 0:00:03
----- 120.6/150.0 MB 11.9 MB/s eta 0:00:03
----- 121.2/150.0 MB 11.7 MB/s eta 0:00:03
----- 121.8/150.0 MB 11.9 MB/s eta 0:00:03
----- 122.3/150.0 MB 11.9 MB/s eta 0:00:03
----- 122.8/150.0 MB 11.9 MB/s eta 0:00:03
----- 123.4/150.0 MB 11.9 MB/s eta 0:00:03
----- 124.0/150.0 MB 12.1 MB/s eta 0:00:03
----- 124.6/150.0 MB 11.7 MB/s eta 0:00:03
----- 125.6/150.0 MB 11.9 MB/s eta 0:00:03
----- 126.2/150.0 MB 11.9 MB/s eta 0:00:03
----- 126.8/150.0 MB 11.9 MB/s eta 0:00:02
----- 127.3/150.0 MB 11.9 MB/s eta 0:00:02
----- 127.8/150.0 MB 11.9 MB/s eta 0:00:02
----- 128.3/150.0 MB 11.7 MB/s eta 0:00:02
----- 128.8/150.0 MB 11.7 MB/s eta 0:00:02
----- 129.4/150.0 MB 11.7 MB/s eta 0:00:02
----- 129.7/150.0 MB 11.5 MB/s eta 0:00:02
----- 130.2/150.0 MB 11.5 MB/s eta 0:00:02
----- 130.7/150.0 MB 11.5 MB/s eta 0:00:02
----- 130.9/150.0 MB 11.1 MB/s eta 0:00:02
----- 131.2/150.0 MB 11.1 MB/s eta 0:00:02
----- 131.4/150.0 MB 10.6 MB/s eta 0:00:02
----- 131.7/150.0 MB 10.2 MB/s eta 0:00:02
----- 132.2/150.0 MB 10.2 MB/s eta 0:00:02
----- 132.5/150.0 MB 9.9 MB/s eta 0:00:02

```

```

----- 132.8/150.0 MB 9.8 MB/s eta 0:00:02
----- 133.3/150.0 MB 9.5 MB/s eta 0:00:02
----- 134.1/150.0 MB 9.6 MB/s eta 0:00:02
----- 134.7/150.0 MB 9.6 MB/s eta 0:00:02
----- 135.4/150.0 MB 9.6 MB/s eta 0:00:02
----- 136.0/150.0 MB 9.6 MB/s eta 0:00:02
----- 136.6/150.0 MB 9.5 MB/s eta 0:00:02
----- 137.2/150.0 MB 9.6 MB/s eta 0:00:02
----- 137.8/150.0 MB 9.6 MB/s eta 0:00:02
----- 138.4/150.0 MB 9.6 MB/s eta 0:00:02
----- 138.8/150.0 MB 9.6 MB/s eta 0:00:02
----- 139.4/150.0 MB 9.6 MB/s eta 0:00:02
----- 139.9/150.0 MB 9.8 MB/s eta 0:00:02
----- 140.5/150.0 MB 9.9 MB/s eta 0:00:01
----- 141.1/150.0 MB 10.2 MB/s eta 0:00:01
----- 141.8/150.0 MB 11.1 MB/s eta 0:00:01
----- 142.5/150.0 MB 11.3 MB/s eta 0:00:01
----- 143.1/150.0 MB 11.9 MB/s eta 0:00:01
----- 143.6/150.0 MB 12.1 MB/s eta 0:00:01
----- 144.2/150.0 MB 11.7 MB/s eta 0:00:01
----- 144.9/150.0 MB 11.9 MB/s eta 0:00:01
----- 145.5/150.0 MB 11.9 MB/s eta 0:00:01
----- 146.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 146.6/150.0 MB 11.7 MB/s eta 0:00:01
----- 147.1/150.0 MB 11.7 MB/s eta 0:00:01
----- 147.6/150.0 MB 11.9 MB/s eta 0:00:01
----- 148.2/150.0 MB 11.9 MB/s eta 0:00:01
----- 148.7/150.0 MB 11.9 MB/s eta 0:00:01
----- 149.3/150.0 MB 11.9 MB/s eta 0:00:01
----- 149.8/150.0 MB 11.7 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 11.9 MB/s eta 0:00:01
----- 150.0/150.0 MB 8.2 MB/s eta 0:00:00

```

Installing collected packages: xgboost
 Successfully installed xgboost-3.0.2

```

[202]: #import and train the model
from xgboost import XGBClassifier

xgb_model = XGBClassifier( eval_metric='logloss', random_state=42)
xgb_model.fit(X_train, y_train) # we will use imbalanced data because the
    ↪ SMOTE-balance data has high False Positive abd False negative

```

```
[202]: XGBClassifier(base_score=None, booster=None, callbacks=None,
                    colsample_bylevel=None, colsample_bynode=None,
                    colsample_bytree=None, device=None, early_stopping_rounds=None,
                    enable_categorical=False, eval_metric='logloss',
                    feature_types=None, feature_weights=None, gamma=None,
                    grow_policy=None, importance_type=None,
                    interaction_constraints=None, learning_rate=None, max_bin=None,
                    max_cat_threshold=None, max_cat_to_onehot=None,
                    max_delta_step=None, max_depth=None, max_leaves=None,
                    min_child_weight=None, missing=nan, monotone_constraints=None,
                    multi_strategy=None, n_estimators=None, n_jobs=None,
                    num_parallel_tree=None, ...)
```

```
[178]: #evaluating our model
y_pred_xgb = xgb_model.predict(X_test)
y_prob_xgb = xgb_model.predict_proba(X_test)[:, 1]
```

```
[180]: from sklearn.metrics import classification_report, confusion_matrix,
        accuracy_score

print(confusion_matrix(y_test, y_pred_xgb))
print(classification_report(y_test, y_pred_xgb))
print("Accuracy:", accuracy_score(y_test, y_pred_xgb))
```

```
[[132  0]
 [ 8 10]]

              precision    recall  f1-score   support

         0       0.94        1.00        0.97        132
         1       1.00        0.56        0.71         18

   accuracy                   0.95        150
  macro avg       0.97        0.78        0.84        150
weighted avg       0.95        0.95        0.94        150
```

Accuracy: 0.9466666666666667

```
[196]: y_pred_thresh = (y_prob_xgb > 0.4).astype(int)
print(confusion_matrix(y_test, y_pred_thresh))
print(classification_report(y_test, y_pred_thresh))
```

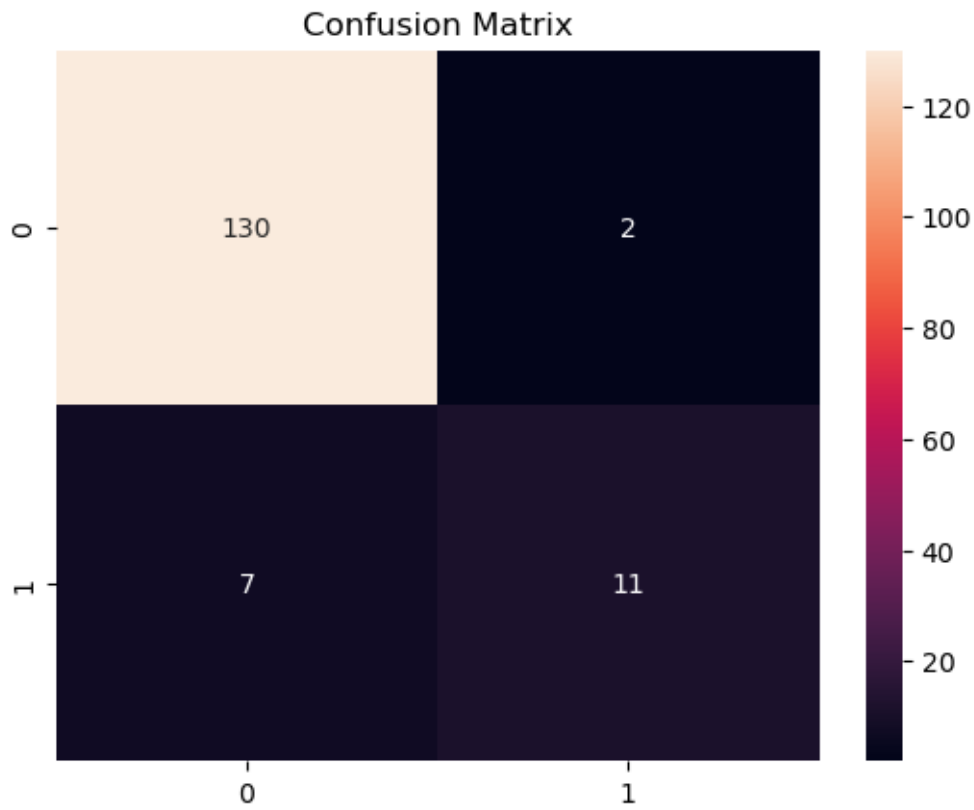
```
[[130  2]
 [ 7 11]]

              precision    recall  f1-score   support

         0       0.95        0.98        0.97        132
         1       0.85        0.61        0.71         18
```

accuracy			0.94	150
macro avg	0.90	0.80	0.84	150
weighted avg	0.94	0.94	0.94	150

```
[198]: sns.heatmap(confusion_matrix(y_test, y_pred_thresh), annot=True, fmt='d')
plt.title("Confusion Matrix")
plt.show()
```

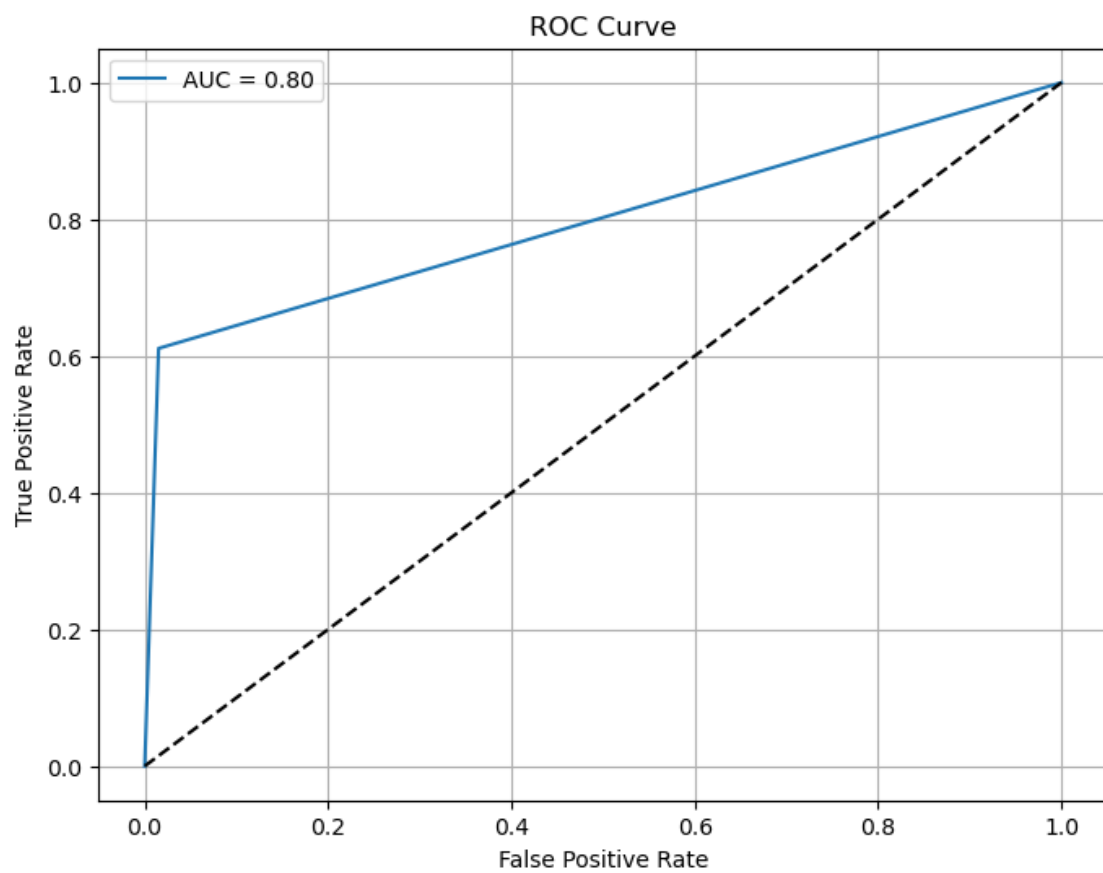


```
[200]: from sklearn.metrics import roc_curve, roc_auc_score

fpr, tpr, thresholds = roc_curve(y_test, y_pred_thresh)
auc = roc_auc_score(y_test, y_pred_thresh)

plt.figure(figsize=(8,6))
plt.plot(fpr, tpr, label="AUC = {:.2f}".format(auc))
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel("False Positive Rate")
plt.ylabel("True Positive Rate")
plt.title("ROC Curve")
plt.legend()
```

```
plt.grid(True)  
plt.show()
```



[]: