In [1]:
```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:
```
Diabetes=pd.read_csv("C:/Users/Pratik Temkar/Documents/ASMITA/MINIPROJECT/diabetes.csv.csv")
Diabetes.head()
```

Out[2]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	O
0	6	148	72	35	0	33.6	0.627	50	
1	1	85	66	29	0	26.6	0.351	31	
2	8	183	64	0	0	23.3	0.672	32	
3	1	89	66	23	94	28.1	0.167	21	
4	0	137	40	35	168	43.1	2.288	33	

In [3]:
```
inputData=Diabetes.iloc[:,:8]
outputData=Diabetes.iloc[:,8]
```

In [4]:
```
from sklearn.linear_model import LogisticRegression
logit1=LogisticRegression()
logit1.fit(inputData,outputData)
```

Out[4]:
```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l2', random_state=None, solver='warn',
          tol=0.0001, verbose=0, warm_start=False)
```

In [5]:
```
logit1.score(inputData,outputData)
```

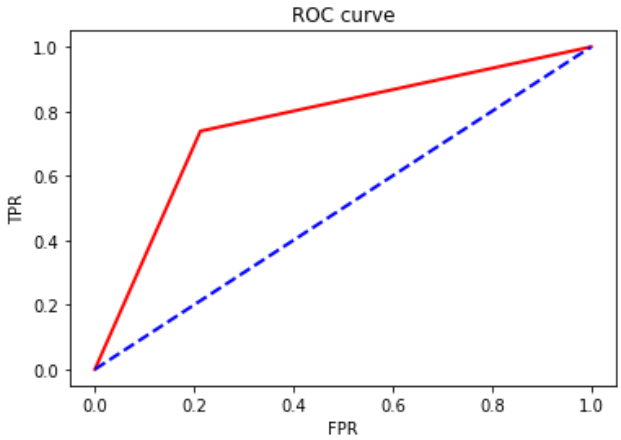
Out[5]:
```
0.7747395833333334
```

In [6]:
```
###Confusion matrix with sklearn
from sklearn.metrics import confusion_matrix, roc_curve, roc_auc_score
confusion_matrix(logit1.predict(inputData),outputData)
```

Out[6]:
```
array([[448, 121],
       [ 52, 147]], dtype=int64)
```

In [7]:
```
##Computing false and true positive rates
fpr, tpr,_=roc_curve(logit1.predict(inputData),outputData,drop_intermedi
ate=False)

import matplotlib.pyplot as plt
plt.figure()
##Adding the ROC
plt.plot(fpr, tpr, color='red',
 lw=2, label='ROC curve')
##Random FPR and TPR
plt.plot([0, 1], [0, 1], color='blue', lw=2, linestyle='--')
##Title and label
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC curve')
plt.show()
```



In [8]:
```
roc_auc_score(logit1.predict(inputData),outputData)
```

Out[8]:
```
0.763019844388904
```