```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
data = pd.read_csv("/content/CyberSecurityData.csv")
data.head()
```

	id	having_IP_Address	URL_Length	Shortining_Service	having_At_Symbol	double_s
0	1	-1	1	1	1	
1	2	1	1	1	1	
2	3	1	0	1	1	
3	4	1	0	1	1	
4	5	1	0	-1	1	

```
data.drop(["id"],axis=1,inplace=True)
data.columns
```

```
Index(['having_IP_Address', 'URL_Length', 'Shortining_Service',
       'having_At_Symbol', 'double_slash_redirecting', 'Prefix_Suffix',
       'having_Sub_Domain', 'SSLfinal_State', 'Domain_registeration_length',
       'Favicon', 'port', 'HTTPS_token', 'Request_URL', 'URL_of_Anchor',
       'Links_in_tags', 'SFH', 'Submitting_to_email', 'Abnormal_URL',
       'Redirect', 'on_mouseover', 'RightClick', 'popUpWidnow', 'Iframe',
       'age_of_domain', 'DNSRecord', 'web_traffic', 'Page_Rank',
       'Google_Index', 'Links_pointing_to_page', 'Statistical_report',
       'Result'],
      dtype='object')
```

Double-click (or enter) to edit

```
data.shape
```

```
(11055, 31)
```

```
data.isnull().values.any()
```

```
False
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
```

```
x=data.drop('URL_Length',axis=1)
y=data['URL_Length']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  99.19719583898689
Test set accuracy :  95.2962460425147
              precision    recall  f1-score   support

          -1       0.97      0.97      0.97      1800
           0       0.67      0.67      0.67        30
           1       0.88      0.89      0.88       381

    accuracy                           0.95      2211
   macro avg       0.84      0.84      0.84      2211
weighted avg       0.95      0.95      0.95      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_IP_Address',axis=1)
y=data['having_IP_Address']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.69832654907282
Test set accuracy :  86.2053369516056
              precision    recall  f1-score   support

          -1       0.79      0.81      0.80       762
           1       0.90      0.89      0.89      1449

    accuracy                           0.86      2211
   macro avg       0.85      0.85      0.85      2211
weighted avg       0.86      0.86      0.86      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_At_Symbol',axis=1)
```

```
y=data['having_At_Symbol']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  98.28132066938036
     Test set accuracy :  92.9895974672094
                  precision    recall  f1-score   support

              -1       0.76      0.79      0.78       342
               1       0.96      0.96      0.96      1869

        accuracy                           0.93      2211
       macro avg       0.86      0.87      0.87      2211
    weighted avg       0.93      0.93      0.93      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('double_slash_redirecting',axis=1)
y=data['double_slash_redirecting']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.90954319312529
     Test set accuracy :  98.73360470375395
                  precision    recall  f1-score   support

              -1       0.94      0.96      0.95       278
               1       0.99      0.99      0.99      1933

        accuracy                           0.99      2211
       macro avg       0.97      0.98      0.97      2211
    weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_Sub_Domain',axis=1)
y=data['having_Sub_Domain']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
```

```
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  91.06739032112166
Test set accuracy :  75.98371777476255
              precision    recall  f1-score   support

          -1       0.69      0.76      0.72       650
           0       0.77      0.74      0.75       712
           1       0.81      0.78      0.80       849

    accuracy                           0.76      2211
   macro avg       0.76      0.76      0.76      2211
weighted avg       0.76      0.76      0.76      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Prefix_Suffix',axis=1)
y=data['Prefix_Suffix']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  97.78380823156942
Test set accuracy :  95.43193125282677
              precision    recall  f1-score   support

          -1       0.97      0.98      0.97      1930
           1       0.84      0.79      0.82       281

    accuracy                           0.95      2211
   macro avg       0.90      0.89      0.89      2211
weighted avg       0.95      0.95      0.95      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('HTTPS_token',axis=1)
y=data['HTTPS_token']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
```

```
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  99.6268656716418
Test set accuracy :  97.33152419719585
              precision    recall  f1-score   support

          -1       0.92      0.92      0.92       368
           1       0.98      0.98      0.98      1843

    accuracy                           0.97      2211
   macro avg       0.95      0.95      0.95      2211
weighted avg       0.97      0.97      0.97      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Links_in_tags',axis=1)
y=data['Links_in_tags']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  89.52962460425147
Test set accuracy :  75.62189054726367
              precision    recall  f1-score   support

          -1       0.75      0.79      0.77       785
           0       0.75      0.76      0.76       869
           1       0.78      0.71      0.74       557

    accuracy                           0.76      2211
   macro avg       0.76      0.75      0.75      2211
weighted avg       0.76      0.76      0.76      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('popUpWidnow',axis=1)
y=data['popUpWidnow']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
```

```
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.97738579828132
     Test set accuracy :  99.41203075531433
                  precision    recall  f1-score   support

              -1       0.98      0.99      0.99       439
               1       1.00      0.99      1.00      1772

        accuracy                           0.99      2211
       macro avg       0.99      0.99      0.99      2211
    weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Redirect',axis=1)
y=data['Redirect']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  98.85798281320669
     Test set accuracy :  93.93939393939394
                  precision    recall  f1-score   support

               0       0.96      0.97      0.97      1951
               1       0.77      0.69      0.73       260

        accuracy                           0.94      2211
       macro avg       0.87      0.83      0.85      2211
    weighted avg       0.94      0.94      0.94      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('on_mouseover',axis=1)
y=data['on_mouseover']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
```

```
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.86431478968792
     Test set accuracy :  98.41700587969244
                  precision    recall  f1-score   support

              -1       0.93      0.94      0.93       260
               1       0.99      0.99      0.99      1951

        accuracy                           0.98      2211
       macro avg       0.96      0.96      0.96      2211
    weighted avg       0.98      0.98      0.98      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Abnormal_URL',axis=1)
y=data['Abnormal_URL']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.79647218453188
     Test set accuracy :  98.50746268656717
                  precision    recall  f1-score   support

              -1       0.95      0.95      0.95       327
               1       0.99      0.99      0.99      1884

        accuracy                           0.99      2211
       macro avg       0.97      0.97      0.97      2211
    weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_IP_Address',axis=1)
y=data['having_IP_Address']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.69832654907282
Test set accuracy :  86.2053369516056
              precision    recall  f1-score   support

          -1       0.79      0.81      0.80       762
           1       0.90      0.89      0.89      1449

    accuracy                           0.86      2211
   macro avg       0.85      0.85      0.85      2211
weighted avg       0.86      0.86      0.86      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Shortining_Service',axis=1)
y=data['Shortining_Service']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  99.81908638625056
Test set accuracy :  98.95974672094076
              precision    recall  f1-score   support

          -1       0.95      0.98      0.96       287
           1       1.00      0.99      0.99      1924

    accuracy                           0.99      2211
   macro avg       0.97      0.98      0.98      2211
weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('SSLfinal_State',axis=1)
y=data['SSLfinal_State']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.56264133876074
```

```
Test set accuracy :  86.38625056535504
              precision    recall  f1-score   support

          -1       0.81      0.81      0.81       710
           0       0.72      0.74      0.73       247
           1       0.92      0.92      0.92      1254

    accuracy                           0.86      2211
   macro avg       0.82      0.82      0.82      2211
weighted avg       0.86      0.86      0.86      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Domain_registeration_length',axis=1)
y=data['Domain_registeration_length']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.23473541383989
Test set accuracy :  88.78335594753504
              precision    recall  f1-score   support

          -1       0.91      0.92      0.91      1436
           1       0.85      0.82      0.84       775

    accuracy                           0.89      2211
   macro avg       0.88      0.87      0.88      2211
weighted avg       0.89      0.89      0.89      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Favicon',axis=1)
y=data['Favicon']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  100.0
Test set accuracy :  99.27634554500227
```

```
                   precision    recall  f1-score   support

               -1       0.99      0.97      0.98       426
                1       0.99      1.00      1.00      1785

         accuracy                           0.99      2211
        macro avg       0.99      0.98      0.99      2211
     weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('port',axis=1)
y=data['port']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.96607869742198
     Test set accuracy :  99.09543193125283
                   precision    recall  f1-score   support

               -1       0.97      0.96      0.97       295
                1       0.99      1.00      0.99      1916

         accuracy                           0.99      2211
        macro avg       0.98      0.98      0.98      2211
     weighted avg       0.99      0.99      0.99      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Request_URL',axis=1)
y=data['Request_URL']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  95.61284486657621
     Test set accuracy :  88.15015829941203
                   precision    recall  f1-score   support
```

```
                -1       0.86      0.87      0.86       950
                 1       0.90      0.89      0.90      1261

          accuracy                           0.88      2211
         macro avg       0.88      0.88      0.88      2211
      weighted avg       0.88      0.88      0.88      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('URL_of_Anchor',axis=1)
y=data['URL_of_Anchor']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  94.4708276797829
     Test set accuracy :  83.5820895522388
                  precision    recall  f1-score   support

               -1       0.86      0.86      0.86       694
                0       0.84      0.85      0.85      1043
                1       0.78      0.76      0.77       474

         accuracy                           0.84      2211
        macro avg       0.83      0.83      0.83      2211
     weighted avg       0.84      0.84      0.84      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('SFH',axis=1)
y=data['SFH']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  96.24604251469923
     Test set accuracy :  87.51696065128901
                  precision    recall  f1-score   support

               -1       0.92      0.94      0.93      1712
```

```
                0       0.53      0.54      0.54       151
                1       0.78      0.71      0.75       348

         accuracy                           0.88      2211
        macro avg       0.75      0.73      0.74      2211
     weighted avg       0.87      0.88      0.87      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Statistical_report',axis=1)
y=data['Statistical_report']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  98.29262777023972
     Test set accuracy :  92.49208502939847
                  precision    recall  f1-score   support

               -1       0.70      0.79      0.74       300
                1       0.97      0.95      0.96      1911

         accuracy                           0.92      2211
        macro avg       0.83      0.87      0.85      2211
     weighted avg       0.93      0.92      0.93      2211
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Result',axis=1)
y=data['Result']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
dt=DecisionTreeClassifier(random_state=0)
dt.fit(x_train,y_train)
y_predict=dt.predict(x_test)
print("train set accuracy: ",100*dt.score(x_train,y_train))
print("Test set accuracy : ",100*dt.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  99.02758932609679
     Test set accuracy :  96.29127091813659
                  precision    recall  f1-score   support

               -1       0.97      0.95      0.96      1014
                1       0.96      0.97      0.97      1197
```

accuracy			0.96	2211
macro avg	0.96	0.96	0.96	2211
weighted avg	0.96	0.96	0.96	2211