Breast Cancer Classification

Asma Ahmed Al-Thagafi

Table of contents:

01

Project overview:

Recently, The healthcare sector is poised for a radical transformation led by artificial intelligence and machine learning techniques and is powered by an abundance of data sources.

In this project:

I am going to use machine learning techniques to help in the early detection of breast cancer which will increase chances of treatment.

02

Data Preparation:

I used the UCI Machine Learning Repository for breast cancer dataset.

Importing and cleaning data:

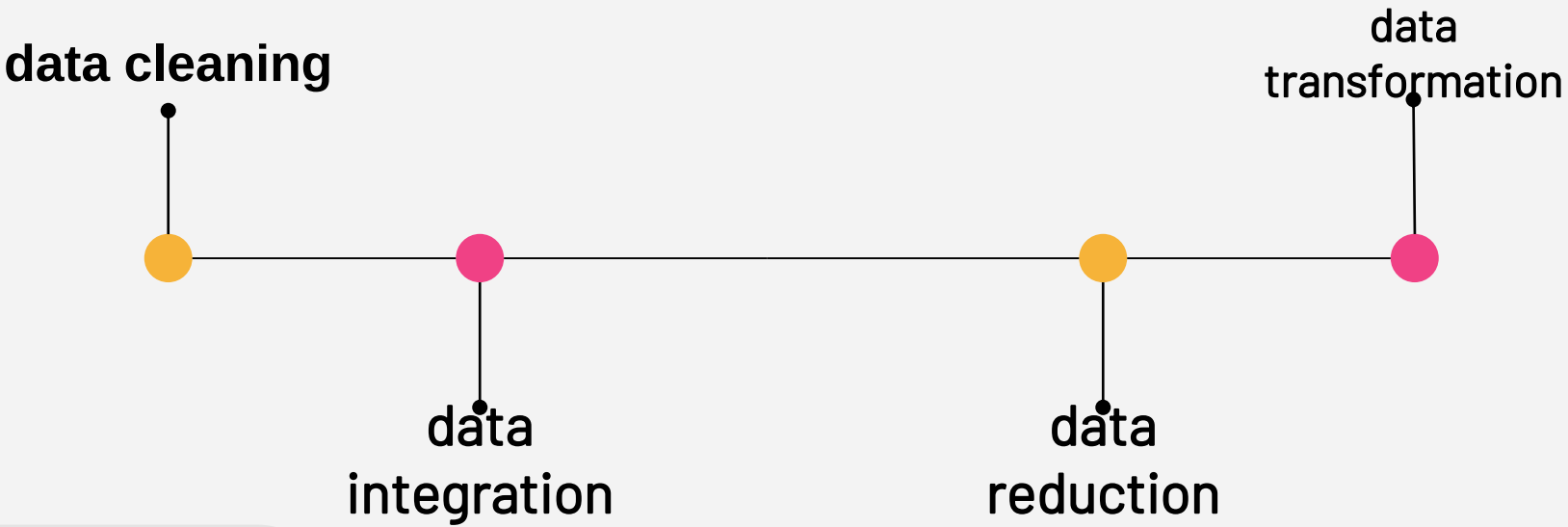
In [81]:
```
df = pd.read_csv('breast-cancer-wisconsin.txt')
df.info()
df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15855 entries, 0 to 15854
Data columns (total 12 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   Index                        15855 non-null  int64 
 1   ID                           15855 non-null  int64 
 2   Clump Thickness              15855 non-null  int64 
 3   Uniformity of Cell Size      15827 non-null  object
 4   Uniformity of Cell Shape     15827 non-null  object
 5   Marginal Adhesion            15827 non-null  object
 6   Single Epithelial Cell Size  15827 non-null  object
 7   Bare Nuclei                  15827 non-null  object
 8   Bland Chromatin              15827 non-null  object
 9   Normal Nucleoli              15827 non-null  object
 10  Mitoses                      15827 non-null  object
 11  Class                        15827 non-null  object
dtypes: int64(3), object(9)
memory usage: 1.5+ MB
```

Out[81]:

	Index	ID	Clump Thickness	Uniformity of Cell Size	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	Class
0	0	1241035	7	8	3	7	4	5	7	8	2	4
1	1	1107684	6	10	5	5	4	10	6	10	1	4
2	2	691628	8	6	4	10	10	1	3	5	1	4
3	3	1226612	7	5	6	3	3	8	7	4	1	4
4	4	1142706	5	10	10	10	6	10	6	5	2	4

Pre-prossising Steps:

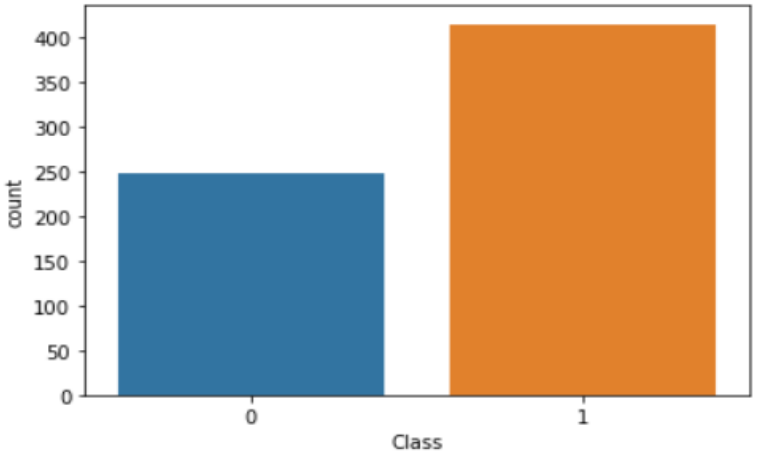


Visualization of the data

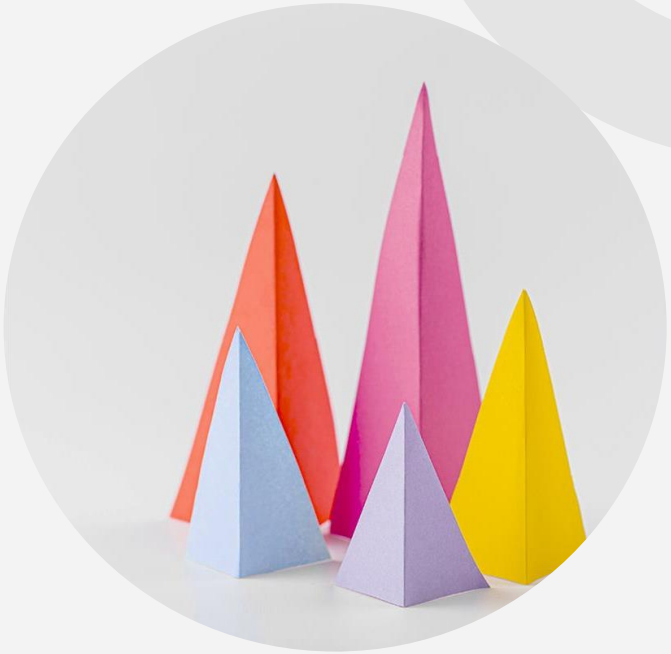
VISUALIZING THE DATA:

```
In [22]: sns.countplot(df['Class'], label = "Count")
```

```
Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x16a3e2eaaf0>
```



Out[27]: <seaborn.axisgrid.PairGrid at 0x16a43453400>



03

Machine Learning Models

To build a ML model which classifies breast cancer:

1- Splitting the dataset to training data=80% and testing data=20%.

2- Train the data by used:

Logistic Regression,

k-nearest neighbors algorithm,

Random Forest.

Logistic Regression:

In [74]:
```
#Using Logistic Regression
from sklearn.linear_model import LogisticRegression
log = LogisticRegression(random_state = 0)
log.fit(X_train, y_train)
print('Logistic Regression Training Accuracy:', log.score(X_train, y_train))
#Check precision, recall, f1-score
print(classification_report(y_test, log.predict(X_test)))
```

```
Logistic Regression Training Accuracy: 0.9583333333333334
              precision    recall  f1-score   support

           0       0.90      0.96      0.93        46
           1       0.98      0.94      0.96        87

    accuracy                           0.95       133
   macro avg       0.94      0.95      0.94       133
weighted avg       0.95      0.95      0.95       133
```

k-nearest neighbors algorithm:

In [79]:
```
#Using KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train, y_train)
print('KNeighborsClassifier Training Accuracy:', knn.score(X_train, y_train))
#Check precision, recall, f1-score
print(classification_report(y_test, knn.predict(X_test)))
```

```
KNeighborsClassifier Training Accuracy: 0.9678030303030303
              precision    recall  f1-score   support

           0       0.98      0.97      0.97        59
           1       0.97      0.99      0.98        74

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

Random Forest:

In [72]:
```
#Using Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score
forest = RandomForestClassifier(n_estimators = 10, criterion = 'entropy', random_state = 0)
forest.fit(X_train, y_train)
print('Random Forest Classifier Training Accuracy:', forest.score(X_train, y_train))

#Check precision, recall, f1-score
print(classification_report(y_test, forest.predict(X_test)))
```

```
Random Forest Classifier Training Accuracy: 0.9924242424242424
              precision    recall  f1-score   support

           0       0.88      0.96      0.92        46
           1       0.98      0.93      0.95        87

    accuracy                           0.94       133
   macro avg       0.93      0.94      0.93       133
weighted avg       0.94      0.94      0.94       133
```

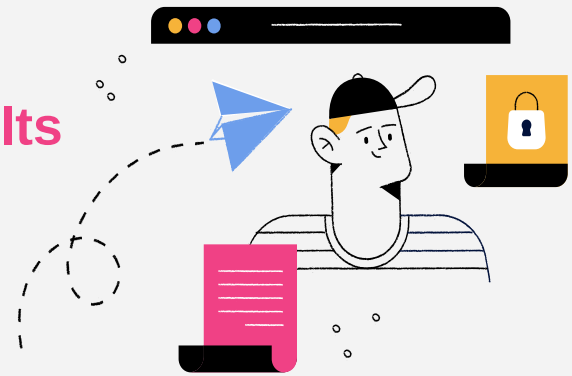
04

Results:

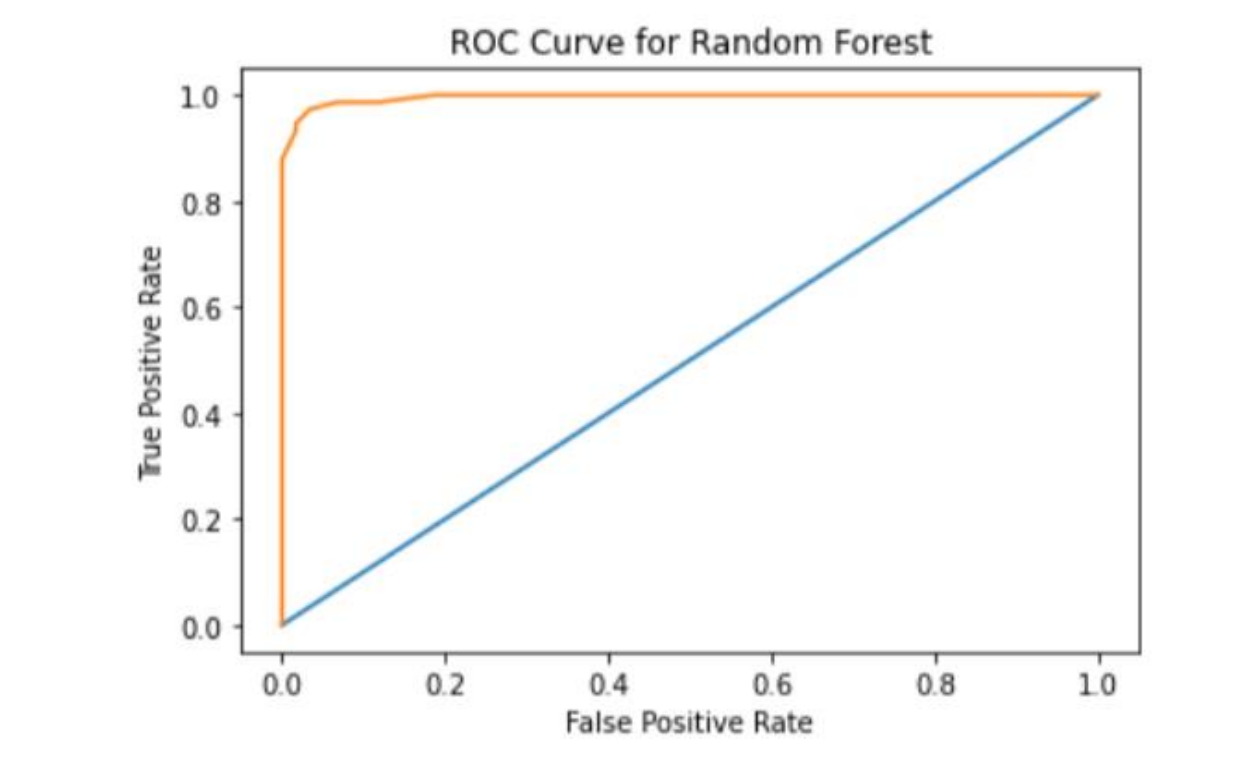
After applying the different classification models, I got the accuracies with different models below:

1. Logistic Regression 0.95%
2. Nearest Neighbors 0.96%
3. Random Forest Classification 0.99%

Random Forest Classification gave the best results for this dataset



ROC Curve


ROC Curve for Random Forest
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

Resources:

1. https://github.com/Al-asma/T5
2. https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+%28original%29
3. https://medium.com/swlh/breast-cancer-classification-using-python-e83719e5f97d

Thank you for your attention

—Any Quistion?