Breast Cancer Tumor Detection - XGBoost

Importig the basic libraries

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

from sklearn.model_selection import train_test_split
```

Load Dataset from Local directory

```
from google.colab import files
uploaded = files.upload()
```

dataset.csv

- **dataset.csv**(application/vnd.ms-excel) - 19635 bytes, last modified: 5/9/2020 - 100% done

```
Saving dataset.csv to dataset (3).csv
```

Importing the Dataset

```
dataset = pd.read_csv('dataset.csv')
print(dataset.shape)
print(dataset.head(5))
```

```
(683, 11)
   Sample code number  Clump Thickness  ...  Mitoses  Class
0             1000025                5  ...        1      2
1             1002945                5  ...        1      2
2             1015425                3  ...        1      2
3             1016277                6  ...        1      2
4             1017023                4  ...        1      2

[5 rows x 11 columns]
```

Segregating Dataset

```
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

Splitting Dataset into Train & Test

```
X_train, X_test, y_train, y_test = train_test_split(X, y,test_size = 0.2, random_state = 0)
```

Training with XGBoost

```
from xgboost import XGBClassifier
model = XGBClassifier()
model.fit(X_train, y_train)
```

```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=1, gamma=0,
              learning_rate=0.1, max_delta_step=0, max_depth=3,
              min_child_weight=1, missing=None, n_estimators=100, n_jobs=1,
              nthread=None, objective='binary:logistic', random_state=0,
              reg_alpha=0, reg_lambda=1, scale_pos_weight=1, seed=None,
              silent=None, subsample=1, verbosity=1)
```

Forming Confusion Matrix

```
from sklearn.metrics import confusion_matrix, accuracy_score
y_pred = model.predict(X_test)
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)
```

```
[[84  3]
 [ 0 50]]
0.9781021897810219
```

K-Fold Cross Validation

```
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = model, X = X_train, y = y_train, cv = 10)
print("Accuracy: {:.2f} %".format(accuracies.mean()*100))
```

```
Accuracy: 96.53 %
```