In [5]:

```
# Import the libraries
import numpy as np
import os,sys
from sklearn.preprocessing import MinMaxScaler
import pandas as pd
import sklearn
import xgboost
from xgboost import XGBClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

In [7]:

```
# Datasets
parkinsons = pd.read_csv('parkinsons.data')
```

In [9]:

```
# Head of data
parkinsons.head()
```

Out[9]:

	name	MDVP:Fo(Hz)	MDVP:Fhi(Hz)	MDVP:Flo(Hz)	MDVP:Jitter(%)	MDVP:Jitter(Abs)	MDVP:RAP	MDVP:PPQ	Jitter:DDP
0	phon_R01_S01_1	119.992	157.302	74.997	0.00784	0.00007	0.00370	0.00554	0.01109
1	phon_R01_S01_2	122.400	148.650	113.819	0.00968	0.00008	0.00465	0.00696	0.01394
2	phon_R01_S01_3	116.682	131.111	111.555	0.01050	0.00009	0.00544	0.00781	0.01633
3	phon_R01_S01_4	116.676	137.871	111.366	0.00997	0.00009	0.00502	0.00698	0.01505
4	phon_R01_S01_5	116.014	141.781	110.655	0.01284	0.00011	0.00655	0.00908	0.01966

5 rows × 24 columns

In [17]:

```
# Features and labels
Features = parkinsons.loc[:,parkinsons.columns != 'status'].values[:,1:]
labels = parkinsons.loc[:,'status'].values
```

In [20]:

```
# Scale the feature between -1 and 1
scaler = MinMaxScaler((-1,1))
X = scaler.fit_transform(Features)
y = labels
```

```
C:\Users\nasim\Anaconda3\New folder\lib\site-packages\sklearn\utils\validation.py:475:
DataConversionWarning: Data with input dtype object was converted to float64 by MinMaxScaler.
  warnings.warn(msg, DataConversionWarning)
```

In [21]:

```
# Train the datasets
X_train,X_test,Y_train,Y_test = train_test_split(X,y,test_size = 0.2, random_state = 7)
```

In [24]:

```
# Train the Model
```

```
model = XGBClassifier()
model.fit(X_train,Y_train)
```

Out[24]:

```
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)
```

In [28]:

```
# Calculate the accuracy Score
y_pred = model.predict(X_test)
print(accuracy_score(Y_test, y_pred)*100)
```

```
94.87179487179486
```

```
C:\Users\nasim\Anaconda3\New folder\lib\site-packages\sklearn\preprocessing\label.py:151:
DeprecationWarning: The truth value of an empty array is ambiguous. Returning False, but in future
this will result in an error. Use `array.size > 0` to check that an array is not empty.
  if diff:
```