In [1]:
```
#importing libraries
import pandas as pd
import numpy as np
```

In [2]:
```
data = pd.read_csv("D:\Detecting parkinsons disease\cleaned-data.csv")
data.head()
```

Out[2]:

	Unnamed: 0	name	MDVP:Fo(Hz)	MDVP:Fhi(Hz)	MDVP:Flo(Hz)	MDVP:Jitter(%)	MDVP:Jitter(Abs)	MDVP:RAP	MDVP:PPQ	Jitter:DI
0	0	phon_R01_S01_1	119.992	157.302	74.997	0.00784	0.000070	0.00370	0.00554	0.011
1	1	phon_R01_S01_2	122.400	148.650	113.819	0.00968	0.000080	0.00465	0.00576	0.013
2	2	phon_R01_S01_3	116.682	131.111	111.555	0.01050	0.000090	0.00544	0.00576	0.016
3	3	phon_R01_S01_4	116.676	137.871	111.366	0.00997	0.000090	0.00502	0.00576	0.015
4	4	phon_R01_S01_5	116.014	141.781	110.655	0.01101	0.000037	0.00593	0.00576	0.017

5 rows × 25 columns

In [3]:
```
#data information
data.columns
```

Out[3]:
```
Index(['Unnamed: 0', 'name', 'MDVP:Fo(Hz)', 'MDVP:Fhi(Hz)', 'MDVP:Flo(Hz)',
       'MDVP:Jitter(%)', 'MDVP:Jitter(Abs)', 'MDVP:RAP', 'MDVP:PPQ',
       'Jitter:DDP', 'MDVP:Shimmer', 'MDVP:Shimmer(dB)', 'Shimmer:APQ3',
       'Shimmer:APQ5', 'MDVP:APQ', 'Shimmer:DDA', 'NHR', 'HNR', 'status',
       'RPDE', 'DFA', 'spread1', 'spread2', 'D2', 'PPE'],
      dtype='object')
```

In [4]:
```
data.describe
```

Out[4]:
```
<bound method NDFrame.describe of      Unnamed: 0            name  MDVP:Fo(Hz)  MDVP:Fhi(Hz)  MDVP:Flo(Hz)  \
0             0  phon_R01_S01_1      119.992       157.302        74.997
1             1  phon_R01_S01_2      122.400       148.650       113.819
2             2  phon_R01_S01_3      116.682       131.111       111.555
3             3  phon_R01_S01_4      116.676       137.871       111.366
4             4  phon_R01_S01_5      116.014       141.781       110.655
..          ...             ...          ...           ...           ...
190         190  phon_R01_S50_2      174.188       230.978        94.261
191         191  phon_R01_S50_3      209.516       253.017        89.488
192         192  phon_R01_S50_4      174.688       240.005        74.287
193         193  phon_R01_S50_5      198.764       396.961        74.904
194         194  phon_R01_S50_6      214.289       260.277        77.973

     MDVP:Jitter(%)  MDVP:Jitter(Abs)  MDVP:RAP  MDVP:PPQ  Jitter:DDP  ...  \
0           0.00784          0.000070   0.00370   0.00554     0.01109  ...
1           0.00968          0.000080   0.00465   0.00576     0.01394  ...
2           0.01050          0.000090   0.00544   0.00576     0.01633  ...
3           0.00997          0.000090   0.00502   0.00576     0.01505  ...
4           0.01101          0.000037   0.00593   0.00576     0.01778  ...
..              ...               ...       ...       ...         ...  ...
190         0.00459          0.000030   0.00263   0.00259     0.00790  ...
191         0.00564          0.000030   0.00331   0.00292     0.00994  ...
192         0.01101          0.000080   0.00593   0.00564     0.01778  ...
193         0.00740          0.000040   0.00370   0.00390     0.01109  ...
194         0.00567          0.000030   0.00295   0.00317     0.00885  ...

     Shimmer:DDA      NHR     HNR  status      RPDE       DFA   spread1  \
0         0.06545  0.02211  21.033       1  0.414783  0.815285 -4.813031
1         0.09403  0.01929  19.085       1  0.458359  0.819521 -4.075192
2         0.08270  0.01309  20.651       1  0.429895  0.825288 -4.443179
3         0.08771  0.01353  20.644       1  0.434969  0.819235 -4.117501
4         0.10470  0.01767  19.649       1  0.417356  0.823484 -3.747787
..            ...      ...     ...     ...       ...       ...       ...
190       0.07008  0.02764  19.517       0  0.448439  0.657899 -6.538586
191       0.04812  0.01810  19.147       0  0.431674  0.683244 -6.195325
192       0.03804  0.10715  17.883       0  0.407567  0.655683 -6.787197
193       0.03794  0.07223  19.020       0  0.451221  0.643956 -6.744577
194       0.03078  0.04398  21.209       0  0.462803  0.664357 -5.724056

      spread2        D2       PPE
0    0.266482  2.301442  0.284654
1    0.335590  2.486855  0.368674
2    0.311173  2.342259  0.332634
3    0.334147  2.405554  0.368975
4    0.234513  2.332180  0.410335
..        ...       ...       ...
190  0.121952  2.657476  0.133050
191  0.129303  2.784312  0.168895
192  0.158453  2.679772  0.131728
193  0.207454  2.138608  0.123306
194  0.190667  2.555477  0.148569

[195 rows x 25 columns]>
```

In [5]:
```
#information about columns
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 195 entries, 0 to 194
Data columns (total 25 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Unnamed: 0        195 non-null    int64
 1   name              195 non-null    object
 2   MDVP:Fo(Hz)       195 non-null    float64
 3   MDVP:Fhi(Hz)      195 non-null    float64
 4   MDVP:Flo(Hz)      195 non-null    float64
 5   MDVP:Jitter(%)    195 non-null    float64
 6   MDVP:Jitter(Abs)  195 non-null    float64
 7   MDVP:RAP          195 non-null    float64
 8   MDVP:PPQ          195 non-null    float64
 9   Jitter:DDP        195 non-null    float64
 10  MDVP:Shimmer      195 non-null    float64
 11  MDVP:Shimmer(dB)  195 non-null    float64
 12  Shimmer:APQ3      195 non-null    float64
 13  Shimmer:APQ5      195 non-null    float64
 14  MDVP:APQ          195 non-null    float64
 15  Shimmer:DDA       195 non-null    float64
 16  NHR               195 non-null    float64
 17  HNR               195 non-null    float64
 18  status            195 non-null    int64
 19  RPDE              195 non-null    float64
 20  DFA               195 non-null    float64
 21  spread1           195 non-null    float64
 22  spread2           195 non-null    float64
 23  D2                195 non-null    float64
 24  PPE               195 non-null    float64
dtypes: float64(22), int64(2), object(1)
memory usage: 38.2+ KB
```

In [6]:
```
#independent variable
X=data.drop(['name','status'],axis=1)
```

In [7]:
```
#dependent variable
y=data['status']
```

In [8]:
```
#importing libraries
from sklearn import model_selection
from sklearn.ensemble import BaggingClassifier
from sklearn.tree import DecisionTreeClassifier

#making subsets
k_folds=model_selection.KFold(n_splits=10,random_state=42,shuffle=True)


#instantiating DecisionTreeClassifier
classifier=DecisionTreeClassifier(criterion="entropy", max_depth=3)
number_of_trees=100


model=BaggingClassifier(base_estimator=classifier,n_estimators=number_of_trees,random_state=42)

results = model_selection.cross_val_score(model, X, y, cv=k_folds)
print("Accuracy:",results.mean())
```

```
Accuracy: 0.948421052631579
```

In [ ]: