

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [2]: df=pd.read_csv(r"C4_framingham.csv")
df
```

Out[2]:

	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	t
1	39	4.0	0	0.0	0.0	0	0	0	
0	46	2.0	0	0.0	0.0	0	0	0	
1	48	1.0	1	20.0	0.0	0	0	0	
0	61	3.0	1	30.0	0.0	0	1	0	
0	46	3.0	1	23.0	0.0	0	0	0	
.	
1	50	1.0	1	1.0	0.0	0	1	0	
1	51	3.0	1	43.0	0.0	0	0	0	
0	48	2.0	1	20.0	NaN	0	0	0	
0	44	1.0	1	15.0	0.0	0	0	0	
0	52	2.0	0	0.0	0.0	0	0	0	

× 16 columns



```
In [3]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to 4237
Data columns (total 16 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   male                  4238 non-null   int64  
 1   age                   4238 non-null   int64  
 2   education             4133 non-null   float64 
 3   currentSmoker         4238 non-null   int64  
 4   cigsPerDay            4209 non-null   float64 
 5   BPMeds                4185 non-null   float64 
 6   prevalentStroke       4238 non-null   int64  
 7   prevalentHyp          4238 non-null   int64  
 8   diabetes              4238 non-null   int64  
 9   totChol               4188 non-null   float64 
10   sysBP                 4238 non-null   float64 
11   diaBP                 4238 non-null   float64 
12   BMI                   4219 non-null   float64 
13   heartRate             4237 non-null   float64 
14   glucose               3850 non-null   float64 
15   TenYearCHD            4238 non-null   int64  
dtypes: float64(9), int64(7)
memory usage: 529.9 KB
```

```
In [4]: df=df.dropna()
```

```
In [5]: df.isnull().sum()
```

```
Out[5]: male                0
age                0
education          0
currentSmoker      0
cigsPerDay         0
BPMeds             0
prevalentStroke    0
prevalentHyp       0
diabetes           0
totChol            0
sysBP              0
diaBP              0
BMI                0
heartRate          0
glucose            0
TenYearCHD         0
dtype: int64
```

In [6]: `df.describe()`

Out[6]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalen
count	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000
mean	0.443654	49.557440	1.979759	0.489059	9.022155	0.030361	0.030361
std	0.496883	8.561133	1.022657	0.499949	11.918869	0.171602	0.171602
min	0.000000	32.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	42.000000	1.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	49.000000	2.000000	0.000000	0.000000	0.000000	0.000000
75%	1.000000	56.000000	3.000000	1.000000	20.000000	0.000000	0.000000
max	1.000000	70.000000	4.000000	1.000000	70.000000	1.000000	1.000000

In [8]: `df.columns`

Out[8]: Index(['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds', 'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP', 'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD'], dtype='object')

In [7]: `df["TenYearCHD"].value_counts()`

Out[7]: 0 3099
1 557
Name: TenYearCHD, dtype: int64

In [9]: `df1=df[['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds', 'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP', 'diaBP', 'BMI', 'heartRate', 'glucose', 'TenYearCHD']]`

In [10]: `x=df1.drop("TenYearCHD",axis=1)`
`y=df1["TenYearCHD"]`

In [11]: `from sklearn.model_selection import train_test_split`
`x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.70)`

In [12]: `from sklearn.ensemble import RandomForestClassifier`
`rfc=RandomForestClassifier()`
`rfc.fit(x_train,y_train)`

Out[12]: RandomForestClassifier()

In [13]: `parameters={'max_depth':[1,2,3,4,5],`
`'min_samples_leaf':[5,10,15,20,25],`
`'n_estimators':[10,20,30,40,50]}`

```
In [14]: from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accuracy")
grid_search.fit(x_train,y_train)
```

```
Out[14]: GridSearchCV(cv=2, estimator=RandomForestClassifier(),
                      param_grid={'max_depth': [1, 2, 3, 4, 5],
                                   'min_samples_leaf': [5, 10, 15, 20, 25],
                                   'n_estimators': [10, 20, 30, 40, 50]},
                      scoring='accuracy')
```

```
In [15]: grid_search.best_score_
```

```
Out[15]: 0.8475969996090695
```

```
In [17]: parameters={'max_depth':[1,2,3,4,5],
                      'min_samples_leaf':[5,10,15,20,25],
                      'n_estimators':[10,20,30,40,50]}
```

```
In [18]: rfc_best=grid_search.best_estimator_
```

```
In [19]: from sklearn.tree import plot_tree  
plt.figure(figsize=(80,40))  
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','No'])
```

```

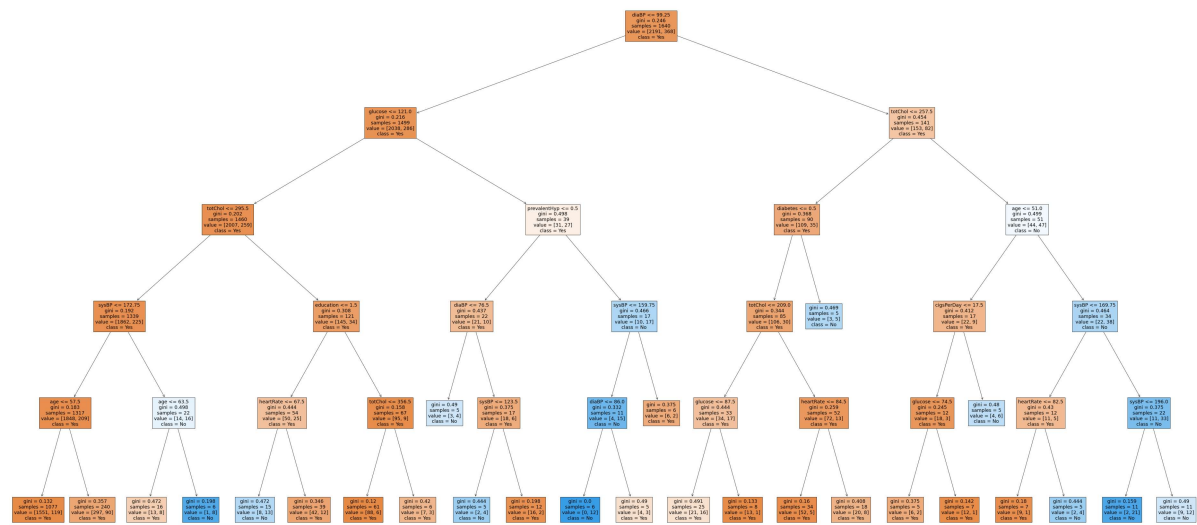
Out[19]: [Text(2396.8636363636365, 1993.2, 'diaBP <= 99.25\ngini = 0.246\nsamples = 16
40\nvalue = [2191, 368]\nclass = Yes'),
Text(1420.3636363636363, 1630.8000000000002, 'glucose <= 121.0\ngini = 0.216
\nsamples = 1499\nvalue = [2038, 286]\nclass = Yes'),
Text(811.6363636363636, 1268.4, 'totChol <= 295.5\ngini = 0.202\nsamples = 1
460\nvalue = [2007, 259]\nclass = Yes'),
Text(405.8181818181818, 906.0, 'sysBP <= 172.75\ngini = 0.192\nsamples = 133
9\nvalue = [1862, 225]\nclass = Yes'),
Text(202.9090909090909, 543.5999999999999, 'age <= 57.5\ngini = 0.183\nsampl
es = 1317\nvalue = [1848, 209]\nclass = Yes'),
Text(101.45454545454545, 181.19999999999982, 'gini = 0.132\nsamples = 1077\n
value = [1551, 119]\nclass = Yes'),
Text(304.3636363636364, 181.19999999999982, 'gini = 0.357\nsamples = 240\nva
lue = [297, 90]\nclass = Yes'),
Text(608.7272727272727, 543.5999999999999, 'age <= 63.5\ngini = 0.498\nsampl
es = 22\nvalue = [14, 16]\nclass = No'),
Text(507.27272727272725, 181.19999999999982, 'gini = 0.472\nsamples = 16\nva
lue = [13, 8]\nclass = Yes'),
Text(710.1818181818181, 181.19999999999982, 'gini = 0.198\nsamples = 6\nvalu
e = [1, 8]\nclass = No'),
Text(1217.4545454545455, 906.0, 'education <= 1.5\ngini = 0.308\nsamples = 1
21\nvalue = [145, 34]\nclass = Yes'),
Text(1014.5454545454545, 543.5999999999999, 'heartRate <= 67.5\ngini = 0.444
\nsamples = 54\nvalue = [50, 25]\nclass = Yes'),
Text(913.0909090909091, 181.19999999999982, 'gini = 0.472\nsamples = 15\nval
ue = [8, 13]\nclass = No'),
Text(1116.0, 181.19999999999982, 'gini = 0.346\nsamples = 39\nvalue = [42, 1
2]\nclass = Yes'),
Text(1420.3636363636363, 543.5999999999999, 'totChol <= 356.5\ngini = 0.158
\nsamples = 67\nvalue = [95, 9]\nclass = Yes'),
Text(1318.909090909091, 181.19999999999982, 'gini = 0.12\nsamples = 61\nvalu
e = [88, 6]\nclass = Yes'),
Text(1521.8181818181818, 181.19999999999982, 'gini = 0.42\nsamples = 6\nvalu
e = [7, 3]\nclass = Yes'),
Text(2029.090909090909, 1268.4, 'prevalentHyp <= 0.5\ngini = 0.498\nsamples
= 39\nvalue = [31, 27]\nclass = Yes'),
Text(1724.7272727272727, 906.0, 'diaBP <= 76.5\ngini = 0.437\nsamples = 22\n
value = [21, 10]\nclass = Yes'),
Text(1623.2727272727273, 543.5999999999999, 'gini = 0.49\nsamples = 5\nvalue
= [3, 4]\nclass = No'),
Text(1826.1818181818182, 543.5999999999999, 'sysBP <= 123.5\ngini = 0.375\ns
amples = 17\nvalue = [18, 6]\nclass = Yes'),
Text(1724.7272727272727, 181.19999999999982, 'gini = 0.444\nsamples = 5\nval
ue = [2, 4]\nclass = No'),
Text(1927.6363636363635, 181.19999999999982, 'gini = 0.198\nsamples = 12\nva
lue = [16, 2]\nclass = Yes'),
Text(2333.4545454545455, 906.0, 'sysBP <= 159.75\ngini = 0.466\nsamples = 17
\nvalue = [10, 17]\nclass = No'),
Text(2232.0, 543.5999999999999, 'diaBP <= 86.0\ngini = 0.332\nsamples = 11\n
value = [4, 15]\nclass = No'),
Text(2130.5454545454545, 181.19999999999982, 'gini = 0.0\nsamples = 6\nvalue
= [0, 12]\nclass = No'),
Text(2333.4545454545455, 181.19999999999982, 'gini = 0.49\nsamples = 5\nvalu
e = [4, 3]\nclass = Yes'),
Text(2434.909090909091, 543.5999999999999, 'gini = 0.375\nsamples = 6\nvalue
= [6, 2]\nclass = Yes'),
Text(3373.3636363636365, 1630.8000000000002, 'totChol <= 257.5\ngini = 0.454

```

```

\nsamples = 141\nvalue = [153, 82]\nclass = Yes'),
  Text(2942.181818181818, 1268.4, 'diabetes <= 0.5\ngini = 0.368\nsamples = 90\nvalue = [109, 35]\nclass = Yes'),
  Text(2840.7272727272725, 906.0, 'totChol <= 209.0\ngini = 0.344\nsamples = 85\nvalue = [106, 30]\nclass = Yes'),
  Text(2637.818181818182, 543.5999999999999, 'glucose <= 87.5\ngini = 0.444\nsamples = 33\nvalue = [34, 17]\nclass = Yes'),
  Text(2536.3636363636365, 181.19999999999982, 'gini = 0.491\nsamples = 25\nvalue = [21, 16]\nclass = Yes'),
  Text(2739.272727272727, 181.19999999999982, 'gini = 0.133\nsamples = 8\nvalue = [13, 1]\nclass = Yes'),
  Text(3043.6363636363635, 543.5999999999999, 'heartRate <= 84.5\ngini = 0.259\nsamples = 52\nvalue = [72, 13]\nclass = Yes'),
  Text(2942.181818181818, 181.19999999999982, 'gini = 0.16\nsamples = 34\nvalue = [52, 5]\nclass = Yes'),
  Text(3145.090909090909, 181.19999999999982, 'gini = 0.408\nsamples = 18\nvalue = [20, 8]\nclass = Yes'),
  Text(3043.6363636363635, 906.0, 'gini = 0.469\nsamples = 5\nvalue = [3, 5]\nclass = No'),
  Text(3804.5454545454545, 1268.4, 'age <= 51.0\ngini = 0.499\nsamples = 51\nvalue = [44, 47]\nclass = No'),
  Text(3550.909090909091, 906.0, 'cigsPerDay <= 17.5\ngini = 0.412\nsamples = 17\nvalue = [22, 9]\nclass = Yes'),
  Text(3449.4545454545455, 543.5999999999999, 'glucose <= 74.5\ngini = 0.245\nsamples = 12\nvalue = [18, 3]\nclass = Yes'),
  Text(3348.0, 181.19999999999982, 'gini = 0.375\nsamples = 5\nvalue = [6, 2]\nclass = Yes'),
  Text(3550.909090909091, 181.19999999999982, 'gini = 0.142\nsamples = 7\nvalue = [12, 1]\nclass = Yes'),
  Text(3652.3636363636365, 543.5999999999999, 'gini = 0.48\nsamples = 5\nvalue = [4, 6]\nclass = No'),
  Text(4058.181818181818, 906.0, 'sysBP <= 169.75\ngini = 0.464\nsamples = 34\nvalue = [22, 38]\nclass = No'),
  Text(3855.272727272727, 543.5999999999999, 'heartRate <= 82.5\ngini = 0.43\nsamples = 12\nvalue = [11, 5]\nclass = Yes'),
  Text(3753.818181818182, 181.19999999999982, 'gini = 0.18\nsamples = 7\nvalue = [9, 1]\nclass = Yes'),
  Text(3956.7272727272725, 181.19999999999982, 'gini = 0.444\nsamples = 5\nvalue = [2, 4]\nclass = No'),
  Text(4261.090909090909, 543.5999999999999, 'sysBP <= 196.0\ngini = 0.375\nsamples = 22\nvalue = [11, 33]\nclass = No'),
  Text(4159.636363636364, 181.19999999999982, 'gini = 0.159\nsamples = 11\nvalue = [2, 21]\nclass = No'),
  Text(4362.545454545454, 181.19999999999982, 'gini = 0.49\nsamples = 11\nvalue = [9, 12]\nclass = No')]

```



```
In [ ]:
```