Vijay 02-09-2023

In [544]:

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

In [545]:

```
from sklearn.linear_model import LogisticRegression
a=pd.read_csv(r"C:\USERS\user\Downloads\C6_bmi.csv")
a
```

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
...	...	...	...	...
495	Female	150	153	5
496	Female	184	121	4
497	Female	141	136	5
498	Male	150	95	5
499	Male	173	131	5

500 rows × 4 columns

In [546]:

```
a=a.head(50)
a
```

Out[546]:

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
5	Male	189	104	3
6	Male	147	92	5
7	Male	154	111	5
8	Male	174	90	3
9	Female	169	103	4
10	Male	195	81	2
11	Female	159	80	4
12	Female	192	101	3
13	Male	155	51	2
14	Male	191	79	2
15	Female	153	107	5
16	Female	157	110	5
17	Male	140	129	5
18	Male	144	145	5
19	Male	172	139	5
20	Male	157	110	5
21	Female	153	149	5
22	Female	169	97	4
23	Male	185	139	5
24	Female	172	67	2
25	Female	151	64	3
26	Male	190	95	3
27	Male	187	62	1
28	Female	163	159	5
29	Male	179	152	5
30	Male	153	121	5
31	Male	178	52	1
32	Female	195	65	1
33	Female	160	131	5
34	Female	157	153	5
35	Female	189	132	4
36	Female	197	114	3
37	Male	144	80	4
38	Female	171	152	5

	Gender	Height	Weight	Index
39	Female	185	81	2
40	Female	175	120	4
41	Female	149	108	5
42	Male	157	56	2
43	Male	161	118	5
44	Female	182	126	4
45	Male	185	76	2
46	Female	188	122	4
47	Male	181	111	4
48	Male	161	72	3
49	Male	140	152	5

In [547]:

```
from sklearn.linear_model import LogisticRegression
```

In [548]:

```
a.columns
```

Out[548]:

```
Index(['Gender', 'Height', 'Weight', 'Index'], dtype='object')
```

In [549]:

```
b=a[[ 'Height', 'Weight', 'Index']]
b
```

Out[549]:

	Height	Weight	Index
0	174	96	4
1	189	87	2
2	185	110	4
3	195	104	3
4	149	61	3
5	189	104	3
6	147	92	5
7	154	111	5
8	174	90	3
9	169	103	4
10	195	81	2
11	159	80	4
12	192	101	3
13	155	51	2
14	191	79	2
15	153	107	5
16	157	110	5
17	140	129	5
18	144	145	5
19	172	139	5
20	157	110	5
21	153	149	5
22	169	97	4
23	185	139	5
24	172	67	2
25	151	64	3
26	190	95	3
27	187	62	1
28	163	159	5
29	179	152	5
30	153	121	5
31	178	52	1
32	195	65	1
33	160	131	5
34	157	153	5
35	189	132	4
36	197	114	3
37	144	80	4
38	171	152	5

	Height	Weight	Index
39	185	81	2
40	175	120	4
41	149	108	5
42	157	56	2
43	161	118	5
44	182	126	4
45	185	76	2
46	188	122	4
47	181	111	4
48	161	72	3
49	140	152	5

In [550]:

```
c=b.iloc[:,0:5]
d=b.iloc[:,-1]
```

In [551]:

```
c.shape
```

Out[551]:

```
(50, 3)
```

In [552]:

```
d.shape
```

Out[552]:

```
(50,)
```

In [553]:

```
from sklearn.preprocessing import StandardScaler
fs=StandardScaler().fit_transform(c)
fs
```

Out[553]:

```
array([[ 0.23938063, -0.28077719,  0.23354968],
       [ 1.12379184, -0.58450252, -1.32344821],
       [ 0.88794885,  0.19168443,  0.23354968],
       [ 1.47755633, -0.01079912, -0.54494926],
       [-1.23463805, -1.46193123, -0.54494926],
       [ 1.12379184, -0.01079912, -0.54494926],
       [-1.35255954, -0.41576622,  1.01204863],
       [-0.93983431,  0.22543169,  1.01204863],
       [ 0.23938063, -0.48326074, -0.54494926],
       [-0.0554231 , -0.04454638,  0.23354968],
       [ 1.47755633, -0.78698607, -1.32344821],
       [-0.64503058, -0.82073332,  0.23354968],
       [ 1.30067409, -0.1120409 , -0.54494926],
       [-0.88087356, -1.79940382, -1.32344821],
       [ 1.24171334, -0.85448058, -1.32344821],
       [-0.99879506,  0.09044265,  1.01204863],
       [-0.76295207,  0.19168443,  1.01204863],
       [-1.76528477,  0.83288234,  1.01204863],
       [-1.52944179,  1.37283847,  1.01204863],
       [ 0.12145914,  1.17035492,  1.01204863],
       [-0.76295207,  0.19168443,  1.01204863],
       [-0.99879506,  1.50782751,  1.01204863],
       [-0.0554231 , -0.24702993,  0.23354968],
       [ 0.88794885,  1.17035492,  1.01204863],
       [ 0.12145914, -1.25944768, -1.32344821],
       [-1.11671655, -1.36068946, -0.54494926],
       [ 1.18275259, -0.31452445, -0.54494926],
       [ 1.00587035, -1.42818398, -2.10194715],
       [-0.40918759,  1.84530009,  1.01204863],
       [ 0.53418437,  1.60906928,  1.01204863],
       [-0.99879506,  0.56290427,  1.01204863],
       [ 0.47522362, -1.76565656, -2.10194715],
       [ 1.47755633, -1.3269422 , -2.10194715],
       [-0.58606983,  0.90037685,  1.01204863],
       [-0.76295207,  1.64281654,  1.01204863],
       [ 1.12379184,  0.93412411,  0.23354968],
       [ 1.59547782,  0.32667346, -0.54494926],
       [-1.52944179, -0.82073332,  0.23354968],
       [ 0.06249839,  1.60906928,  1.01204863],
       [ 0.88794885, -0.78698607, -1.32344821],
       [ 0.29834138,  0.52915701,  0.23354968],
       [-1.23463805,  0.12418991,  1.01204863],
       [-0.76295207, -1.63066753, -1.32344821],
       [-0.52710908,  0.46166249,  1.01204863],
       [ 0.71106661,  0.73164056,  0.23354968],
       [ 0.88794885, -0.95572236, -1.32344821],
       [ 1.0648311 ,  0.59665153,  0.23354968],
       [ 0.65210587,  0.22543169,  0.23354968],
       [-0.52710908, -1.09071139, -0.54494926],
       [-1.76528477,  1.60906928,  1.01204863]])
```

In [554]:

```
logr=LogisticRegression()
logr.fit(fs,d)
```

Out[554]:

```
LogisticRegression()
```

In [557]:
```
e=[[77,9,55]]
```

In [558]:
```
prediction=logr.predict(e)
prediction
```

Out[558]:
```
array([4], dtype=int64)
```

In [559]:
```
logr.classes_
```

Out[559]:
```
array([1, 2, 3, 4, 5], dtype=int64)
```

In [560]:
```
logr.predict_proba(e)[0][0]
```

Out[560]:
```
3.415792265699517e-51
```

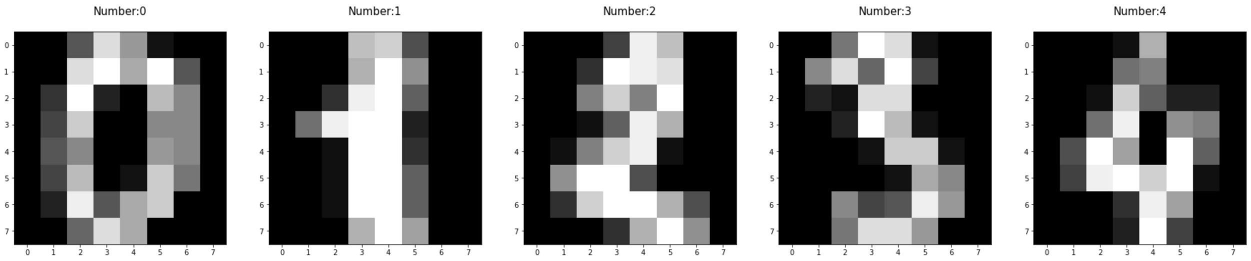
In [561]:
```
import re
from sklearn.datasets import load_digits
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [562]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
```

In [563]:
```
digits=load_digits()
digits
```

```
  'pixel_1_3',
  'pixel_1_4',
  'pixel_1_5',
  'pixel_1_6',
  'pixel_1_7',
  'pixel_2_0',
  'pixel_2_1',
  'pixel_2_2',
  'pixel_2_3',
  'pixel_2_4',
  'pixel_2_5',
  'pixel_2_6',
  'pixel_2_7',
  'pixel_3_0',
  'pixel_3_1',
  'pixel_3_2',
  'pixel_3_3',
  'pixel_3_4',
  'pixel_3_5',
  'pixel_3_6',
  'pixel_3_7'
```

In [564]:
```
plt.figure(figsize=(50,25))
for index,(image,label) in enumerate(zip(digits.data[0:8],digits.target[0:5])):
    plt.subplot(1,8,index+1)
    plt.imshow(np.reshape(image,(8,8)),cmap=plt.cm.gray)
    plt.title('Number:%i\n'%label,fontsize=15)
```



In [567]:
```
x_train,x_test,y_train,y_test=train_test_split(digits.data,digits.target,test_size=0.
```

In [568]:
```
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(898, 64)
(899, 64)
(898,)
(899,)
```

In [569]:
```
logre=LogisticRegression(max_iter=10000)
logre.fit(x_train,y_train)

```

Out[569]:
```
LogisticRegression(max_iter=10000)
```

In [570]:
```
print(logre.predict(x_test))
```

```
[0 9 4 1 5 4 9 5 9 7 4 1 1 6 2 5 8 8 1 6 4 9 8 7 5 8 3 1 0 7 1 1 1 3 5 2 9
 4 2 1 2 7 4 9 9 1 3 8 8 2 4 2 8 9 9 0 0 2 5 5 0 3 0 8 3 5 9 8 0 3 1 8 4 0
 4 8 1 7 8 4 0 9 5 6 1 7 4 0 1 3 7 6 1 0 7 7 6 6 1 5 5 7 1 3 6 8 8 3 2 0 5
 7 4 7 2 9 6 9 4 6 6 7 5 2 9 2 5 2 5 1 9 9 8 7 1 8 5 6 7 3 9 5 0 5 6 8 5 1
 5 1 7 9 0 2 4 0 5 5 5 7 0 4 2 5 0 5 3 9 8 0 4 9 1 2 0 3 2 1 5 1 8 0 8 4 7
 5 3 3 5 4 7 2 5 5 8 2 0 6 5 8 0 1 6 0 4 4 7 3 9 0 4 5 1 9 2 3 6 1 7 3 7 7
 6 4 0 3 9 8 5 9 0 6 8 2 9 8 2 5 2 5 2 6 6 9 3 9 1 1 7 3 5 4 5 5 8 1 1 2 1
 8 5 9 0 0 0 7 0 4 8 1 7 4 1 4 8 3 6 2 7 2 7 4 5 8 6 0 0 2 7 1 8 7 9 3 3 3
 1 1 0 6 0 5 9 4 5 9 2 4 7 8 9 2 0 3 5 1 0 7 2 7 0 4 4 3 6 1 9 8 2 4 0 0 7
 5 6 8 2 5 0 0 2 8 3 4 6 9 5 1 8 1 3 6 0 1 2 7 3 8 8 7 6 6 4 9 1 7 5 8 7 8
 3 5 8 8 4 2 8 7 9 4 7 3 8 1 1 4 4 6 7 8 5 8 6 3 0 3 3 1 3 6 2 8 6 8 1 8 9
 3 9 2 7 6 8 5 0 1 2 1 8 5 6 5 4 4 0 4 0 8 6 1 0 7 3 9 2 4 3 7 0 6 4 8 9 1
 9 9 6 7 8 9 7 3 2 8 8 4 2 7 3 3 6 2 1 7 9 1 2 4 4 5 4 6 2 5 9 2 1 4 9 6 6
 3 8 6 5 4 9 2 2 4 8 2 1 5 3 5 1 4 2 0 3 3 7 2 5 2 2 1 2 3 8 2 4 2 3 9 5 1
 3 8 8 7 8 1 8 0 5 0 3 0 8 3 0 1 1 4 0 9 4 3 2 9 2 2 0 4 7 4 2 7 6 1 4 2 0
 6 9 3 9 8 0 4 9 5 0 2 9 6 5 3 5 1 2 6 8 4 1 6 9 6 3 2 9 1 6 3 4 9 6 8 8 8
 1 4 0 7 6 8 8 1 2 7 2 6 8 3 0 6 1 1 9 8 4 2 8 2 4 9 8 1 4 6 5 8 9 9 3 2 2
 3 7 8 4 6 6 4 1 1 0 2 2 8 0 9 1 3 6 1 9 3 1 7 1 6 7 5 0 2 1 5 9 2 9 8 1 8
 1 2 3 4 5 0 5 1 1 8 8 3 3 4 9 8 4 0 3 3 5 6 7 3 7 5 3 5 5 1 6 0 5 4 6 2 6
 5 7 5 8 2 8 9 4 8 4 4 7 9 9 9 2 6 0 1 4 2 9 3 1 4 3 5 1 1 2 6 3 9 6 6 6 4
 2 4 6 0 0 7 2 1 9 4 2 5 9 8 3 9 1 3 1 9 9 3 4 3 0 3 6 0 3 0 5 5 8 0 0 5 5
 3 2 5 5 4 3 2 2 3 8 5 7 4 9 7 7 9 9 5 8 7 5 3 7 5 7 3 6 3 7 8 3 4 9 1 7 9
 5 6 4 9 7 3 3 0 1 9 1 2 4 0 3 2 7 2 3 6 0 4 3 2 9 1 0 7 2 7 4 9 6 8 2 9 0
 8 8 8 6 3 2 4 3 0 2 6 9 7 0 0 5 2 7 1 9 4 7 6 2 1 2 4 6 8 6 5 9 5 5 1 9 4
 2 5 4 9 4 6 8 9 4 9 9]
```

In [526]:

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

In [571]:

```
a=pd.read_csv(r"C:\USERS\user\Downloads\C6_bmi.csv")
```

In [572]:

```
a=a.head(50)
a
```

Out[572]:

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
5	Male	189	104	3
6	Male	147	92	5
7	Male	154	111	5
8	Male	174	90	3
9	Female	169	103	4
10	Male	195	81	2
11	Female	159	80	4
12	Female	192	101	3
13	Male	155	51	2
14	Male	191	79	2
15	Female	153	107	5
16	Female	157	110	5
17	Male	140	129	5
18	Male	144	145	5
19	Male	172	139	5
20	Male	157	110	5
21	Female	153	149	5
22	Female	169	97	4
23	Male	185	139	5
24	Female	172	67	2
25	Female	151	64	3
26	Male	190	95	3
27	Male	187	62	1
28	Female	163	159	5
29	Male	179	152	5
30	Male	153	121	5
31	Male	178	52	1
32	Female	195	65	1
33	Female	160	131	5
34	Female	157	153	5
35	Female	189	132	4
36	Female	197	114	3
37	Male	144	80	4
38	Female	171	152	5

	Gender	Height	Weight	Index
39	Female	185	81	2
40	Female	175	120	4
41	Female	149	108	5
42	Male	157	56	2
43	Male	161	118	5
44	Female	182	126	4
45	Male	185	76	2
46	Female	188	122	4
47	Male	181	111	4
48	Male	161	72	3
49	Male	140	152	5

In [574]:

```
b=a[['Gender', 'Height', 'Weight', 'Index']]
b
```

Out[574]:

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
5	Male	189	104	3
6	Male	147	92	5
7	Male	154	111	5
8	Male	174	90	3
9	Female	169	103	4
10	Male	195	81	2
11	Female	159	80	4
12	Female	192	101	3
13	Male	155	51	2
14	Male	191	79	2
15	Female	153	107	5
16	Female	157	110	5
17	Male	140	129	5
18	Male	144	145	5
19	Male	172	139	5
20	Male	157	110	5
21	Female	153	149	5
22	Female	169	97	4
23	Male	185	139	5
24	Female	172	67	2
25	Female	151	64	3
26	Male	190	95	3
27	Male	187	62	1
28	Female	163	159	5
29	Male	179	152	5
30	Male	153	121	5
31	Male	178	52	1
32	Female	195	65	1
33	Female	160	131	5
34	Female	157	153	5
35	Female	189	132	4
36	Female	197	114	3
37	Male	144	80	4
38	Female	171	152	5

	Gender	Height	Weight	Index
39	Female	185	81	2
40	Female	175	120	4
41	Female	149	108	5
42	Male	157	56	2
43	Male	161	118	5
44	Female	182	126	4
45	Male	185	76	2
46	Female	188	122	4
47	Male	181	111	4
48	Male	161	72	3
49	Male	140	152	5

In [578]:

```
b['Gender'].value_counts()
```

Out[578]:

```
Male      27
Female    23
Name: Gender, dtype: int64
```

In [579]:

```
x=b.drop('Gender',axis=1)
y=b['Gender']
print(b)
```

```
    Gender  Height  Weight  Index
0     Male     174      96      4
1     Male     189      87      2
2   Female     185     110      4
3   Female     195     104      3
4     Male     149      61      3
5     Male     189     104      3
6     Male     147      92      5
7     Male     154     111      5
8     Male     174      90      3
9   Female     169     103      4
10    Male     195      81      2
11  Female     159      80      4
12  Female     192     101      3
13    Male     155      51      2
14    Male     191      79      2
15  Female     153     107      5
16  Female     157     110      5
17    Male     140     129      5
18    Male     144     145      5
19    Male     172     139      5
20    Male     157     110      5
21  Female     153     149      5
22  Female     169      97      4
23    Male     185     139      5
24  Female     172      67      2
25  Female     151      64      3
26    Male     190      95      3
27    Male     187      62      1
28  Female     163     159      5
29    Male     179     152      5
30    Male     153     121      5
31    Male     178      52      1
32  Female     195      65      1
33  Female     160     131      5
34  Female     157     153      5
35  Female     189     132      4
36  Female     197     114      3
37    Male     144      80      4
38  Female     171     152      5
39  Female     185      81      2
40  Female     175     120      4
41  Female     149     108      5
42    Male     157      56      2
43    Male     161     118      5
44  Female     182     126      4
45    Male     185      76      2
46  Female     188     122      4
47    Male     181     111      4
48    Male     161      72      3
49    Male     140     152      5
```

In [580]:

```
g1={"Gender":{'g1':1}}
a=a.replace(g1)
print(a)
```

```
    Gender  Height  Weight  Index
0     Male     174      96      4
1     Male     189      87      2
2   Female     185     110      4
3   Female     195     104      3
4     Male     149      61      3
5     Male     189     104      3
6     Male     147      92      5
7     Male     154     111      5
8     Male     174      90      3
9   Female     169     103      4
10    Male     195      81      2
11  Female     159      80      4
12  Female     192     101      3
13    Male     155      51      2
14    Male     191      79      2
15  Female     153     107      5
16  Female     157     110      5
17    Male     140     129      5
18    Male     144     145      5
19    Male     172     139      5
20    Male     157     110      5
21  Female     153     149      5
22  Female     169      97      4
23    Male     185     139      5
24  Female     172      67      2
25  Female     151      64      3
26    Male     190      95      3
27    Male     187      62      1
28  Female     163     159      5
29    Male     179     152      5
30    Male     153     121      5
31    Male     178      52      1
32  Female     195      65      1
33  Female     160     131      5
34  Female     157     153      5
35  Female     189     132      4
36  Female     197     114      3
37    Male     144      80      4
38  Female     171     152      5
39  Female     185      81      2
40  Female     175     120      4
41  Female     149     108      5
42    Male     157      56      2
43    Male     161     118      5
44  Female     182     126      4
45    Male     185      76      2
46  Female     188     122      4
47    Male     181     111      4
48    Male     161      72      3
49    Male     140     152      5
```

In [581]:

```
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 [582]:

```
from sklearn.ensemble import RandomForestClassifier
```

In [583]:

```
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[583]:

```
RandomForestClassifier()
```

In [584]:

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

In [585]:

```
from sklearn.model_selection import GridSearchCV
```

In [586]:

```
grid_search=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accuracy")
grid_search.fit(x_train,y_train)
```

Out[586]:

```
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 [587]:

```
grid_search.best_score_
```

Out[587]:

```
0.6029411764705883
```

In [588]:

```
rfc_best=grid_search.best_estimator_
```

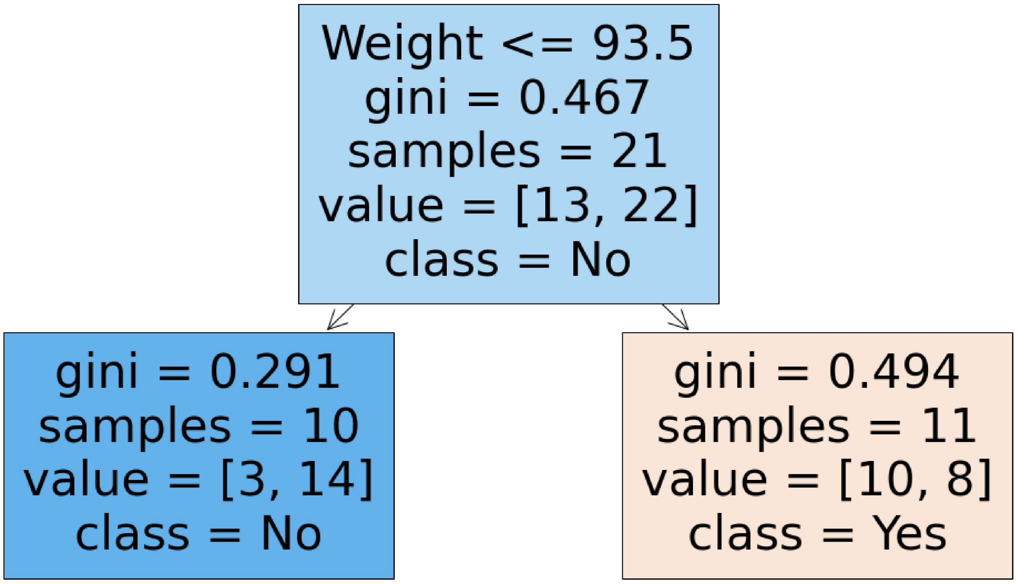
In [589]:

```
from sklearn.tree import plot_tree
```

In [590]:
```
plt.figure(figsize=(20,10))
plot_tree(rfc_best.estimators_[5],feature_names=x.columns,class_names=['Yes','No'],fi

```

Out[590]:
```
[Text(558.0, 407.70000000000005, 'Weight <= 93.5\ngini = 0.467\nsamples = 21\nvalue = [1
3, 22]\nclass = No'),
 Text(279.0, 135.89999999999998, 'gini = 0.291\nsamples = 10\nvalue = [3, 14]\nclass = N
o'),
 Text(837.0, 135.89999999999998, 'gini = 0.494\nsamples = 11\nvalue = [10, 8]\nclass = Y
es')]
```


Weight <= 93.5
gini = 0.467
samples = 21
value = [13, 22]
class = No
gini = 0.291
samples = 10
value = [3, 14]
class = No
gini = 0.494
samples = 11
value = [10, 8]
class = Yes