

## Decision Tree

```
In [1]: import pandas as pd
import matplotlib.pyplot as p
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn import tree
from sklearn.neural_network import MLPClassifier
```

```
In [2]: dataset = pd.read_csv('wifi_localization.csv', sep='\t' , header=0)
dataset
```

Out[2]:

	atb1	atb2	atb3	atb4	atb5	atr6	atb7	lable
0	-64	-56	-61	-66	-71	-82	-81	1
1	-68	-57	-61	-65	-71	-85	-85	1
2	-63	-60	-60	-67	-76	-85	-84	1
3	-61	-60	-68	-62	-77	-90	-80	1
4	-63	-65	-60	-63	-77	-81	-87	1
...	...	...	...	...	...	...	...	...
1995	-59	-59	-48	-66	-50	-86	-94	4
1996	-59	-56	-50	-62	-47	-87	-90	4
1997	-62	-59	-46	-65	-45	-87	-88	4
1998	-62	-58	-52	-61	-41	-90	-85	4
1999	-59	-50	-45	-60	-45	-88	-87	4

2000 rows × 8 columns

```
In [5]: #dataset.head()
#dataset.shape
#dataset.info()
#print(dataset['t'])
#y=(dataset['t'])
#x=dataset[['a','b','c','d','e','f','g','h','i','j','k','l']]
#print(x)
#print(y)
#dataset.hist(bins=50, figsize=(20,15))
#p.show()
train_set, test_set = train_test_split(dataset, test_size=0.3, random_state=40
)
#train_set.shape
#test_set.shape
#train_set.head
train_set_x = train_set.drop(['lable'], axis=1)
train_set_y = train_set['lable']
test_set_x = test_set.drop(['lable'], axis=1)
test_set_y = test_set['lable']

print("x_train: ",train_set_x.shape)
print("x_test: ",test_set_x.shape)
print("y_train: ",train_set_y.shape)
print("y_test: ",test_set_y.shape)

from sklearn.tree import DecisionTreeClassifier
dtree = DecisionTreeClassifier()
dtree.fit(train_set_x, train_set_y.ravel())
print("accuracy: ", dtree.score(test_set_x, test_set_y))
p.figure(figsize=(35,35))
temp = tree.plot_tree(dtree, fontsize=12)
p.show()
```

```
x_train: (1400, 7)
x_test: (600, 7)
y_train: (1400,)
y_test: (600,)
accuracy: 0.9716666666666667
```

