# Classification

* The process of dividing the datasets into different categories or groups by adding label.
* It adds the data point to a particular labelled group on the basis of some condition.
* Without condition we cannot work with Classification.

# Decision Tree Algorithm

* Supervised Algorithm
* Solves both Regression and Classification problems.



# Coding

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.datasets import load\_iris

from sklearn.tree import DecisionTreeClassifier

from sklearn.cross\_validation import train\_test\_split

from sklearn.metrics import accuracy\_score

data= load\_iris()

df=pd.DataFrame(data.data, columns=data.feature\_names)

#df["Type of Flower"] = data.target

df["Type of Flower"] = pd.Categorical.from\_codes(data.target, data.target\_names)

df.head()

X=df.drop("Type of Flower", axis=1)

Y = df["Type of Flower"]

X,Y

X.shape[0],Y.shape[0]

X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(X,Y,random\_state=1, test\_size=0.25)

model = DecisionTreeClassifier(criterion = 'entropy', )

model.fit(X\_train,Y\_train)

Y\_pred = model.predict(X\_test)

print("Accuracy score for predicted values: ",(accuracy\_score(Y\_test, Y\_pred)\*100) )

Accuracy score for predicted values: 97.36842105263158

we will tune the parameters of the decision tree to increase its accuracy. One of those parameters is 'min\_samples\_split', which is the minimum number of samples required to split an internal node. Its default value is equal to 2 because we cannot split on a node containing only one example/ sample.