**Decision tree classifier**

****Decision Trees**** are a non-parametric supervised learning method used for [classification](https://scikit-learn.org/stable/modules/tree.html" \l "tree-classification) and [regression](https://scikit-learn.org/stable/modules/tree.html" \l "tree-regression). The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. The decision tree classifier creates the classification model by building a decision tree. Each node in the tree specifies a test on an attribute, each branch descending from that node corresponds to one of the possible values for that attribute.

#Decision tree classifier

import pandas as pd

import numpy as np

df = pd.read\_csv('/content/sample\_data/Pokemon.csv')

df.head()

df.drop(columns=['#','Name',  'Type 1', 'Type 2'],inplace=True)

df

df.Legendary.value\_counts()

from sklearn import preprocessing

label\_encoder = preprocessing.LabelEncoder()

df['Legendary']= label\_encoder.fit\_transform(df['Legendary']) df['Legendary'].unique()

df

df.Legendary.value\_counts()

from sklearn.model\_selection import train\_test\_split

pokemon\_features = df.drop("Legendary",axis=1)

target = df["Legendary"]

X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(pokemon\_features,target,test\_size=0.20,random\_state=0)

from sklearn.metrics import accuracy\_score

from sklearn.tree import DecisionTreeClassifier

max\_accuracy = 0

for x in range(200):

    dt = DecisionTreeClassifier(random\_state=x)

    dt.fit(X\_train,Y\_train)

    Y\_pred\_dt = dt.predict(X\_test)

    current\_accuracy = round(accuracy\_score(Y\_pred\_dt,Y\_test)\*100,2)

    if(current\_accuracy>max\_accuracy):

        max\_accuracy = current\_accuracy

        best\_x = x

dt = DecisionTreeClassifier(random\_state=best\_x)

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

Y\_pred\_dt = dt.predict(X\_test)

score\_dt = round(accuracy\_score(Y\_pred\_dt,Y\_test)\*100,2)

print("The accuracy score achieved using Decision Tree is: "+str(score\_dt)+" %")

