5 GaussianNB classifier

Gaussian Naive Bayes is a machine learning classification technique based on a probablistic approach that assumes each class follows a normal distribution. It assumes each parameter has an independent capacity of predicting the output variable.

# GaussianNB

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.naive\_bayes import GaussianNB

nb = GaussianNB()

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

Y\_pred\_nb = nb.predict(X\_test)

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

print("The accuracy score achieved using Naive Bayes is: "+str(score\_nb)+" %")

