**Practical Number 7**

**Aim:** Naive Bayes using scikit-learn's built-in dataset, Iris Flowers.

**Software Used :** Pycharm Community Edition 2023.1, Python(3.9.12)

**Theory:**

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayestheorem and used for solving classification problems. It is mainly used in *text classification* that includes a high-dimensional training dataset. Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

**Code:**

import numpy as np

import matplotlib.pyplot as plt

from sklearn import datasets

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

from sklearn.naive\_bayes import GaussianNB

from sklearn.metrics import accuracy\_score

# Load the Iris dataset

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# Split the data into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=0)

# Train the classifier

gnb = GaussianNB()

gnb.fit(X\_train, y\_train)

# Predict the class labels for the test set

y\_pred = gnb.predict(X\_test)

# Calculate the accuracy

accuracy = accuracy\_score(y\_test, y\_pred)

print("Accuracy:", accuracy)

# Plot the actual values and predicted values

fig, ax = plt.subplots(1, 2, figsize=(12, 5))

ax[0].scatter(y\_test, y\_pred)

ax[0].set\_xlabel("Actual Values")

ax[0].set\_ylabel("Predicted Values")

# Plot the difference between the actual values and predicted values

difference = y\_test - y\_pred

ax[1].scatter(y\_test, difference)

ax[1].set\_xlabel("Actual Values")

ax[1].set\_ylabel("Difference")

# Create a box around any five difference points

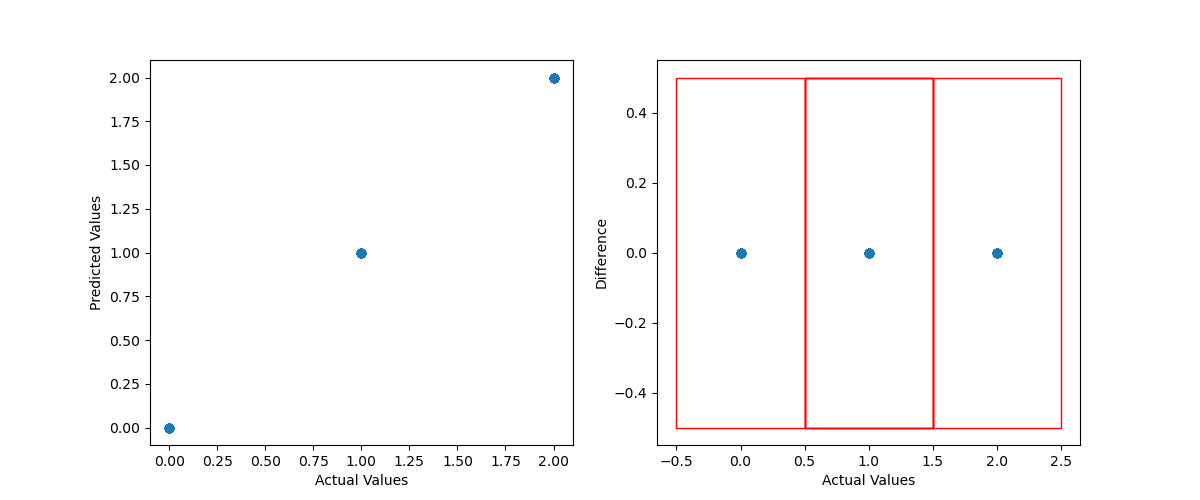
indices = np.random.choice(len(difference), 5, replace=False)

for i in indices:

ax[1].add\_patch(plt.Rectangle((y\_test[i]-0.5, difference[i]-0.5), 1, 1, color='r', fill=None))

plt.show()

**Results:**



**Accuracy:** 1.0

**Conclusion** : In this practical, we have successfully studied and implemented Naive Bayes using scikit-learn's built-in dataset, Iris Flowers.