**RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD**

****

**Lab 10**

**Bachelors of Computer science – 6th semester**

**Subject:** Artificial Intelligence Lab

**Submitted to:** Ma’am Ayesha

**Submitted by:** Areeba Sadaqat

**Sap Id:** 47633

**Date:** 15th April, 2025

**Question 01:** Implement KNN classifier with Iris dataset.

import numpy as np

from sklearn.datasets import load\_iris

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

from sklearn.metrics import classification\_report

from collections import Counter

iris = load\_iris()

X, y = iris.data, iris.target

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

class KNearestNeighbors:

def \_init\_(self, k=3):

self.k = k

def fit(self, X\_train, y\_train):

self.X\_train = np.array(X\_train)

self.y\_train = np.array(y\_train)

def \_calculate\_distance(self, point1, point2):

return np.sqrt(np.sum((point1 - point2) \*\* 2))

def predict(self, X\_test):

return [self.\_predict\_single(sample) for sample in X\_test]

def \_predict\_single(self, sample):

distances = [self.\_calculate\_distance(sample, train\_sample) for train\_sample in self.X\_train]

nearest\_indices = np.argsort(distances)[:self.k]

nearest\_labels = [self.y\_train[idx] for idx in nearest\_indices]

return Counter(nearest\_labels).most\_common(1)[0][0]

knn\_model = KNearestNeighbors(k=3)

knn\_model.fit(X\_train, y\_train)

y\_pred = knn\_model.predict(X\_test)

accuracy = np.mean(y\_pred == y\_test)

print(f"Model Accuracy: {round(accuracy \* 100, 2)}%")

print("\nClassification Report:")

print(classification\_report(y\_test, y\_pred))





