**AML Algorithm #15 : k-Nearest Neighbors implementation using iris dataset**

import numpy as np

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

from sklearn.neighbors import KNeighborsClassifier

from sklearn import datasets

from sklearn import metrics

# Load the Iris dataset

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# Split the dataset into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.27, random\_state=25)

# Create k-NN classifier

knn\_classifier = KNeighborsClassifier(n\_neighbors=4) # You can adjust the number of neighbors (k) here

# Train the classifier

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

# Make predictions on the test set

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

# Evaluate the model

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

print(f"Accuracy: {accuracy}")

# Example usage for predicting a new sample

new\_sample = np.array([[5.0, 3.5, 1.5, 0.2]]) # You can modify the values for a different sample

predicted\_class = knn\_classifier.predict(new\_sample)

print(f"Predicted Class for the new sample: {predicted\_class}")

**OUTPUT :**

Accuracy: 0.926829268292683

Predicted Class for the new sample: [0]