\* Name: Akshata Sunil Dharmadhikari

\* Sap ID: 60004220125

\* Batch: C22

\* Branch: Computer Engineering

\* Course: Machine Learning

\* Experiment 8: SVM

CODE:

from sklearn import datasets

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

from sklearn.svm import SVC

from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report

# Load dataset

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# Split dataset into training set and test set

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Define kernels

kernels = ['linear', 'poly', 'rbf', 'sigmoid']

# Iterate over each kernel

for kernel in kernels:

print("Kernel:", kernel)

# SVM model with current kernel

svm\_model = SVC(kernel=kernel)

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

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

# Accuracy

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

print("Accuracy:", accuracy)

# Confusion Matrix

confusion\_mat = confusion\_matrix(y\_test, y\_pred)

print("Confusion Matrix:")

print(confusion\_mat)

# Classification Report

class\_report = classification\_report(y\_test, y\_pred)

print("Classification Report:")

print(class\_report)

print("-------------------------\n")

OUTPUT:

Kernel: linear

Accuracy: 1.0

Confusion Matrix:

[[19 0 0]

[ 0 13 0]

[ 0 0 13]]

Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 19

1 1.00 1.00 1.00 13

2 1.00 1.00 1.00 13

accuracy 1.00 45

macro avg 1.00 1.00 1.00 45

weighted avg 1.00 1.00 1.00 45

-------------------------

Kernel: poly

Accuracy: 0.9777777777777777

Confusion Matrix:

[[19 0 0]

[ 0 12 1]

[ 0 0 13]]

Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 19

1 1.00 0.92 0.96 13

2 0.93 1.00 0.96 13

accuracy 0.98 45

macro avg 0.98 0.97 0.97 45

weighted avg 0.98 0.98 0.98 45

-------------------------

Kernel: rbf

Accuracy: 1.0

Confusion Matrix:

[[19 0 0]

[ 0 13 0]

[ 0 0 13]]

Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 19

1 1.00 1.00 1.00 13

2 1.00 1.00 1.00 13

accuracy 1.00 45

macro avg 1.00 1.00 1.00 45

weighted avg 1.00 1.00 1.00 45

-------------------------

Kernel: sigmoid

Accuracy: 0.2222222222222222

Confusion Matrix:

[[ 0 0 19]

[ 0 10 3]

[ 0 13 0]]

Classification Report:

precision recall f1-score support

0 0.00 0.00 0.00 19

1 0.43 0.77 0.56 13

2 0.00 0.00 0.00 13

accuracy 0.22 45

macro avg 0.14 0.26 0.19 45

weighted avg 0.13 0.22 0.16 45

-------------------------

CODE:

import pandas as pd

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

from sklearn.svm import SVC

from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report

# Load the dataset

data = pd.read\_csv("../content/UniversalBank (1).csv")

# Drop irrelevant columns if necessary and split features and target variable

X = data.drop(['ID', 'ZIP Code', 'Personal Loan'], axis=1)

y = data['Personal Loan']

# Split dataset into training set and test set

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Define kernels

kernels = ['linear', 'poly', 'rbf', 'sigmoid']

# Iterate over each kernel

for kernel in kernels:

print("Kernel:", kernel)

# SVM model with current kernel

svm\_model = SVC(kernel=kernel)

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

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

# Accuracy

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

print("Accuracy:", accuracy)

# Confusion Matrix

confusion\_mat = confusion\_matrix(y\_test, y\_pred)

print("Confusion Matrix:")

print(confusion\_mat)

# Classification Report

class\_report = classification\_report(y\_test, y\_pred)

print("Classification Report:")

print(class\_report)

print("-------------------------\n")

OUTPUT:

Kernel: linear

Accuracy: 0.9473333333333334

Confusion Matrix:

[[1334 9]

[ 70 87]]

Classification Report:

precision recall f1-score support

0 0.95 0.99 0.97 1343

1 0.91 0.55 0.69 157

accuracy 0.95 1500

macro avg 0.93 0.77 0.83 1500

weighted avg 0.95 0.95 0.94 1500

-------------------------

Kernel: poly

Accuracy: 0.904

Confusion Matrix:

[[1343 0]

[ 144 13]]

Classification Report:

precision recall f1-score support

0 0.90 1.00 0.95 1343

1 1.00 0.08 0.15 157

accuracy 0.90 1500

macro avg 0.95 0.54 0.55 1500

weighted avg 0.91 0.90 0.87 1500

-------------------------

Kernel: rbf

Accuracy: 0.8993333333333333

Confusion Matrix:

[[1336 7]

[ 144 13]]

Classification Report:

precision recall f1-score support

0 0.90 0.99 0.95 1343

1 0.65 0.08 0.15 157

accuracy 0.90 1500

macro avg 0.78 0.54 0.55 1500

weighted avg 0.88 0.90 0.86 1500

-------------------------

Kernel: sigmoid

Accuracy: 0.8626666666666667

Confusion Matrix:

[[1244 99]

[ 107 50]]

Classification Report:

precision recall f1-score support

0 0.92 0.93 0.92 1343

1 0.34 0.32 0.33 157

accuracy 0.86 1500

macro avg 0.63 0.62 0.63 1500

weighted avg 0.86 0.86 0.86 1500

-------------------------