TITLE 1

NAME: Y.YAMINI REDDY

REG NO: 192211153

SUBJECT CODE: SPIC409

GUIDE NAME: DR.D.JAYALAKSHMI

APPROVAL ID: SSE\_26\_11\_153\_1 CODE

import pandas as pd

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

from sklearn.svm import SVC

from sklearn.naive\_bayes import GaussianNB

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

from sklearn.compose import ColumnTransformer

from sklearn.preprocessing import OneHotEncoder, StandardScaler

from sklearn.pipeline import Pipeline

csv\_path = "C:/Users/Arshad/Desktop/yamini main.csv"

df = pd.read\_csv(csv\_path)

selected\_features = ['RL', 'glaucoma', 'age', 'ocular\_pressure', 'Mean\_deviation',

'PSD', 'GHT','cornea\_thickness', 'RNFL4.mean']

features = df[selected\_features]

labels = df['glaucoma']

categorical\_features = ['RL', 'GHT']

preprocessor = ColumnTransformer(

transformers=[

('num', StandardScaler(), features.select\_dtypes(include='number').columns),

('cat', OneHotEncoder(handle\_unknown='ignore'), categorical\_features)

])

svm\_pipeline = Pipeline([

('preprocessor', preprocessor),

('classifier', SVC(C=0.15, kernel='rbf', gamma='scale', random\_state=52)) # Adjust parameters

])

X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, labels, test\_size=0.85, random\_state=92)

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

svm\_predictions = svm\_pipeline.predict(X\_test)

svm\_accuracy = accuracy\_score(y\_test, svm\_predictions)

print(f"SVM Accuracy: {svm\_accuracy}")

print("Classification Report (SVM):")

print(classification\_report(y\_test, svm\_predictions))

print("Confusion Matrix (SVM):")

print(confusion\_matrix(y\_test, svm\_predictions))

X\_train\_dense = preprocessor.fit\_transform(X\_train)

X\_test\_dense = preprocessor.transform(X\_test)

nb\_pipeline = Pipeline([

('classifier', GaussianNB())

])

X\_train\_nb, X\_test\_nb, y\_train\_nb, y\_test\_nb = train\_test\_split(features, labels, test\_size=0.85, random\_state=42)

nb\_pipeline.fit(X\_train\_dense, y\_train\_nb)

nb\_predictions = nb\_pipeline.predict(X\_test\_dense)

nb\_accuracy = accuracy\_score(y\_test\_nb, nb\_predictions)

print(f"Naive Bayes Accuracy: {nb\_accuracy}")

print("Classification Report (Naive Bayes):")

print(classification\_report(y\_test\_nb, nb\_predictions))

print("Confusion Matrix (Naive Bayes):")

print(confusion\_matrix(y\_test\_nb, nb\_predictions))