24.Question: K-Nearest Neighbors (KNN) Classifier

You are working on a classification problem to predict whether a patient has a certain medical

condition or not based on their symptoms. You have collected a dataset of patients with labeled

data (0 for no condition, 1 for the condition) and various symptom features.

Write a Python program that allows the user to input the features of a new patient and the value of k

(number of neighbors). The program should use the KNN classifier from the scikit-learn library to

predict whether the patient has the medical condition or not based on the input features.

Code :

import pandas as pd

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

from sklearn.neighbors import KNeighborsClassifier

from sklearn.preprocessing import StandardScaler

from sklearn.metrics import accuracy\_score

# Step 1: Load dataset from Excel

file\_path = input(r"C:\Users\hares\Downloads\q24\_05.xlsx")

df = pd.read\_excel(file\_path)

# Step 2: Split features and target

X = df.drop('condition', axis=1)

y = df['condition']

# Step 3: Split into train-test sets

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

# Step 4: Standardize features

scaler = StandardScaler()

X\_train\_scaled = scaler.fit\_transform(X\_train)

X\_test\_scaled = scaler.transform(X\_test)

# Step 5: Input value of k from user

k = int(input("Enter the value of k (number of neighbors): "))

knn = KNeighborsClassifier(n\_neighbors=k)

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

# Step 6: Predict and show accuracy

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

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

print(f"Model accuracy on test set: {accuracy \* 100:.2f}%")

# Step 7: Predict for a new patient

def predict\_new\_patient():

print("\nEnter symptom values for a new patient:")

new\_features = []

for col in X.columns:

value = float(input(f"{col}: "))

new\_features.append(value)

new\_scaled = scaler.transform([new\_features])

prediction = knn.predict(new\_scaled)

result = "has the medical condition" if prediction[0] == 1 else "does NOT have the medical condition"

print(f"\nPrediction: The patient {result}.")

predict\_new\_patient()

output :

Enter the Excel file path (e.g., patient\_data.xlsx): patient\_data.xlsx

Enter the value of k (number of neighbors): 3

Model accuracy on test set: 80.00%

Enter symptom values for a new patient:

symptom\_1: 2

symptom\_2: 3

symptom\_3: 4

Prediction: The patient has the medical condition.

Dataset :

|  |  |  |  |
| --- | --- | --- | --- |
| **symptom\_1** | **symptom\_2** | **symptom\_3** | **condition** |
| 1 | 3 | 4 | 0 |
| 2 | 3 | 3 | 1 |
| 2 | 2 | 4 | 0 |
| 3 | 1 | 5 | 1 |
| 1 | 3 | 2 | 0 |
| 2 | 2 | 4 | 0 |
| 1 | 2 | 3 | 1 |
| 3 | 1 | 2 | 1 |
| 1 | 1 | 5 | 0 |
| 2 | 2 | 3 | 1 |