Bahria University,

Karachi Campus

A logo with text on it

Description automatically generated

LAB EXPERIMENT NO.

12

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **01** | Develop a logistic regression model using KNIME that accurately predicts whether a passenger survived the Titanic disaster. The model will be evaluated based on its accuracy, precision. |
| **02** | You are a data analyst working for a healthcare organization, and your task  is to predict the likelihood of diabetes in patients using logistic regression in Python. The goal is to identify high -risk individuals to enable early intervention and better management of diabetes. |
|  |  |

Submitted On:

22-05-2024

(Date: DD/MM/YYYY)

**Task No. 01:** As a data scientist, you are part of a team that aims to analyze the Titanic dataset to predict passenger survival. The dataset includes attributes like class, sex,

age, and fare. Logistic regression is the chosen method because of its suitability for binary classification tasks and its ability to provide interpretable results.

Develop a logistic regression model using KNIME that accurately predicts whether a passenger survived the Titanic disaster. The model will be evaluated based on

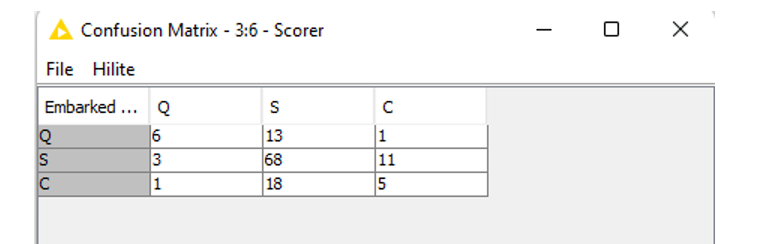
its accuracy, precision.

**Solution:**

A diagram of a logistic process

Description automatically generated

**Output:**

****

**A screenshot of a computer

Description automatically generated**

**Task No. 02:** You are a data analyst working for a healthcare organization, and your task

is to predict the likelihood of diabetes in patients using logistic regression in Python. The goal is to identify high -risk individuals to enable early intervention and better management of diabetes.

**Solution:**

import numpy as np

import pandas as pd

from sklearn.datasets import load\_diabetes

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

from sklearn.linear\_model import LogisticRegression

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

print(f"Accuracy: {accuracy:.2f}")

print("Confusion Matrix:")

print(conf\_matrix)

print("Classification Report:")

print(class\_report)

plt.figure(figsize=(8, 6))

sns.heatmap(conf\_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=['Below Mean', 'Above Mean'], yticklabels=['Below Mean', 'Above Mean'])

plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.title('Confusion Matrix')

plt.show()

report\_df = pd.DataFrame(class\_report\_dict).transpose()

plt.figure(figsize=(10, 6))

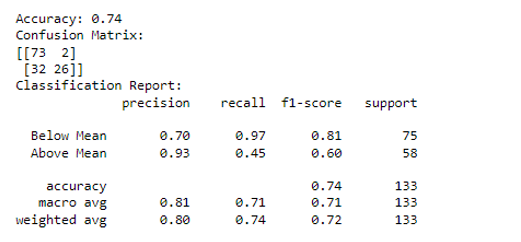
sns.heatmap(report\_df.iloc[:-1, :].T, annot=True, cmap='coolwarm', cbar=False)

plt.title('Classification Report')

plt.show()

A blue squares with white text

Description automatically generated**Output:**



A blue and red graph

Description automatically generated