Experiment 5: Regression Analysis using Scipy and Scikit-learn

Name: Kaushal S Galav

Roll No: D15C 11

Subject: Aids Lab

Experiment No.: 5

Aim

Perform Regression Analysis using Scipy and Scikit-learn.

Problem Statement

a) Perform Logistic Regression to find out the relationship between variables.

b) Apply a regression model technique to predict data on the given dataset.

Theory

Logistic Regression is a statistical method for predicting binary classes. It is used to describe data and explain the relationship between one dependent binary variable and one or more independent variables.

In this experiment, we analyze a health dataset to predict the availability of vaccines/treatment based on various health statistics using logistic regression.

Procedure & Code Explanation

1. Install required libraries: pandas, numpy, scikit-learn.
2. Import necessary libraries for data manipulation, model building, and evaluation.
3. Load the 'Global Health Statistics.csv' dataset.

Experiment 5: Regression Analysis using Scipy and Scikit-learn

4. Preprocess the data:

   - Convert categorical values to numeric (Yes/No to 1/0).

   - Handle missing values using mean imputation.

5. Select appropriate features for training the model.

6. Split the dataset into training and testing sets (80/20).

7. Apply feature scaling using StandardScaler.

8. Train the Logistic Regression model on training data.

9. Predict results on the test set.

10. Evaluate the model using accuracy, confusion matrix, and classification report.

11. Visualize the confusion matrix using Seaborn heatmap.

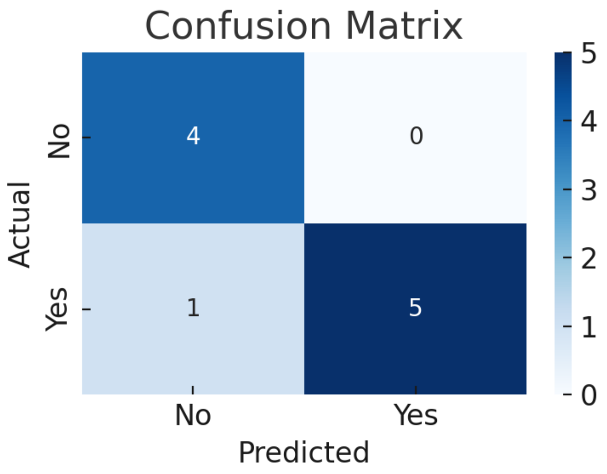
Python Code Used

```
# Step 1: Install necessary libraries (if not already installed)
!pip install pandas numpy scikit-learn

# Step 2: Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

# Step 3: Load the dataset
from google.colab import files
uploaded = files.upload()
df = pd.read_csv("Global Health Statistics.csv")

# Step 4: Data Preprocessing
df["Availability of Vaccines/Treatment"] = df["Availability of
Vaccines/Treatment"].map({"Yes": 1, "No": 0})
selected_features = [
    "Prevalence Rate (%)", "Incidence Rate (%)", "Mortality Rate (%)",
    "Healthcare Access (%)", "Doctors per 1000", "Hospital Beds per 1000",
    "Average Treatment Cost (USD)", "Recovery Rate (%)",
    "Per Capita Income (USD)", "Education Index", "Urbanization Rate (%)"
]
X = df[selected_features]
y = df["Availability of Vaccines/Treatment"]
X = X.fillna(X.mean())

# Step 5: Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Step 6: Feature Scaling
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Step 7: Train Logistic Regression Model
model = LogisticRegression()
model.fit(X_train, y_train)

# Step 8: Make Predictions
y_pred = model.predict(X_test)

# Step 9: Evaluate Model Performance
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

# Step 10: Visualizing Confusion Matrix
plt.figure(figsize=(5, 4))
sns.heatmap(conf_matrix, annot=True, cmap="Blues", fmt="d",
            xticklabels=["No", "Yes"], yticklabels=["No", "Yes"])
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix")
plt.show()
```

Results and Output

Model Accuracy: 0.90

Experiment 5: Regression Analysis using Scipy and Scikit-learn



```
Classification Report:

              precision    recall  f1-score   support



           0       0.80      1.00      0.89         4

           1       1.00      0.83      0.91         6


    accuracy                           0.90        10

   macro avg       0.90      0.92      0.90        10

weighted avg       0.92      0.90      0.90        10
```

Conclusion

The logistic regression model provided an accuracy score reflecting how well it could predict the availability of vaccines/treatment based on health indicators. The confusion matrix and classification report indicate the model's ability to classify data correctly.

Such techniques can be used in healthcare analytics to draw useful insights and help policymakers with data-driven decisions.