# Experiment No. 4

| | |
|---|---|
| Expt No. 4 | Implementation of Multi-Class Classification |
| Date : | |

## Aim:

To implement and understand the working of Logistic Regression for multiclass classification problems in Machine Learning.

## System Requirements:

- Operating System: Windows 8 or above / Linux / macOS
- Memory (RAM): Minimum 4 GB
- Processor: Minimum 2.33 GHz (Dual Core or higher)

## Software/Tools Required:

Jupyter Notebook / Anaconda Navigator / Google Colaboratory / Spyder
Python 3.x with the following libraries:

- NumPy
- Pandas
- Matplotlib
- Seaborn
- Scikit-learn

## Purpose of the Experiment:

- To implement and understand the principles of Multiclass Logistic Regression for classification problems.

- To gain hands-on experience in implementing and evaluating a multiclass logistic regression model using the Iris dataset.
- To explore the significance of data preprocessing, feature selection, and model evaluation in multiclass classification tasks.

## Expected Outcomes:

- Understand the fundamentals of logistic regression as a multiclass classifier.
- Gain insights into how logistic regression can handle more than two classes in classification problems.
- Evaluate the performance of the model using metrics like accuracy, confusion matrix, and classification report.
- Extend the logistic regression method for more complex real-world multiclass problems.

## Theory:

Logistic Regression is a machine learning algorithm based on supervised learning.
It is a statistical method that is used for predicting probability of target variable.
Logistic regression makes probability for classification problems that are discrete in nature.
Example:
Binary Classification: win or loss, survived or not survived
Multiclass Classification : Onion or tomato or potato, lily or sunflower or rose

Logistic Regression is a machine learning algorithm used for predicting the probability of a target variable. While traditionally used for binary classification (2 classes), it can be extended to multiclass classification using techniques like one-vs-rest (OvR) or softmax for multi-class prediction.

Binary Classification: Classifying instances into two categories (e.g., 0 or 1, True or False).

Multiclass Classification: Classifying instances into more than two categories (e.g., Iris species: setosa, versicolor, virginica).

## Real-World Applications of Logistic Regression

Here are four real-world applications of logistic regression:

1. Medical Diagnosis: Predicting the type of disease based on patient data (e.g., types of cancer, diseases).
2. Handwriting Recognition: Classifying handwritten digits or letters (e.g., MNIST dataset).
3. Sentiment Analysis: Categorizing reviews or tweets into positive, negative, or neutral classes.
4. Image Classification: Classifying different objects or scenes in images (e.g., categorizing animals, scenes, etc.)

### 1. Problem Statement

The task is to classify the Iris dataset into three categories: Setosa, Versicolor, and Virginica based on four features: sepal length, sepal width, petal length, and petal width. We aim to implement a multiclass logistic regression model to predict the species of an Iris flower

### 2. Analytical Approach

This is a multiclass classification problem where the target variable is categorical (species of Iris). The goal is to predict which species a given flower belongs to based on its features.

- Dependent Variable: Species (Setosa, Versicolor, Virginica)
- Independent Variables: Sepal Length, Sepal Width, Petal Length, Petal Width.

### 3. Data Collection

The dataset used for this experiment is the Iris dataset, which contains the following features for 150 samples of Iris flowers:

1. sepal_length: Length of the sepal (in cm)
2. sepal_width: Width of the sepal (in cm)
3. petal_length: Length of the petal (in cm)
4. petal_width: Width of the petal (in cm)
5. species: Target variable (Setosa, Versicolor, Virginica) Parch: Number of parents or children aboard the Titanic.

You can download the dataset from the following link:
https://github.com/25-Madhuri/Dataset/blob/main/Iris.csv

## Sample Code

## Import Libraries

```
# Importing necessary libraries for data manipulation, visualization, and modeling

import pandas as pd

# Load the Iris dataset from a CSV file

dataset = pd.read_csv("/content/Iris.csv")

# Print the unique values in the 'Species' column

print(dataset['Species'].unique())

# # Replace categorical species names with numeric labels

# 'Iris-setosa' becomes 1, 'Iris-versicolor' becomes 2, 'Iris-virginica' becomes 3

dataset['Species'].replace({'Iris-setosa': 1, 'Iris-versicolor': 2, 'Iris-virginica': 3}, inplace=True)

dataset
# Importing the train_test_split function from scikit-learn's model_selection module
# This function splits the dataset into training and testing sets for model evaluation

from sklearn.model_selection import train_test_split
```

**# Len(X_train) helps you understand the size of the training data used for model training**

```
len(X_train)
```

***# len(X_test) helps you understand the size of the testing data used for model evaluation***

```
len(X_test)
```

***# Importing the LogisticRegression class from scikit-learn's linear_model module***
***# This is used to create a logistic regression model for classification tasks***

```
from sklearn.linear_model import LogisticRegression
```

***# Create an instance of the LogisticRegression model***

```
lr = LogisticRegression(max_iter=200)
```

***# Fit the model on the training data (X_train and y_train)***
***# This trains the logistic regression model to learn the relationship between features and the target variable***

```
lr.fit(X_train, y_train)
```

```
LogisticRegression()
```

***# Use the trained Logistic Regression model to make predictions on the test data (X_test)***

***# This will output the predicted class labels for each instance in the test set***

```
lr.predict(X_test)
```

```
X_test
```

```
# Evaluate the accuracy of the logistic regression model on the test set (X_test, y_test)
# This will return the proportion of correct predictions made by the model

lr.score(X_test, y_test)

# Importing the seaborn library and using pairplot to visualize pairwise relationships between features
# 'hue='Species'' colors the data points by their species, helping to distinguish between them in the plot

import seaborn as sns

sns.pairplot(dataset[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm', 'Species']], hue='Species')
```

**Observation**

I. Record the following in observation table:

| Sr.No | Metric | Description | Observations |
|---|---|---|---|
| 1. | Unique Species in Dataset | Record the unique values in the 'Species' column after loading the dataset. | |
| 2. | Species Mapping (Numeric Labels) | Record the numeric mapping for species names, i.e., how 'Iris-setosa', 'Iris-versicolor', and 'Iris-virginica' are represented numerically. | |
| 3. | Size of Training Dataset (X_train) | Record the number of samples in the training dataset after the train_test_split() function. | |
| 4. | Size of Testing Dataset (X_test) | Record the number of samples in the testing dataset after the train_test_split() function. | |
| 5. | Accuracy of Logistic Regression Model | Record the accuracy score of the model on the test data. The accuracy is returned by lr.score(X_test, y_test). | |

| | | | |
|---|---|---|---|
| 6. | Predictions for Test Set | Record the predicted species labels for the test set after running lr.predict(X_test). | |
| 7. | Pairplot of Features | Describe any visible patterns or relationships between the features (SepalLengthCm, SepalWidthCm, etc.) and species from the pairplot. | |

II. Perform the following tasks and record results by applying Logistic Regression. Use the Pima Indian Diabetes dataset for each task:

Problem Statement:

Use the Pima Indian Diabetes dataset(Link to download Dataset: https://github.com/25-Madhuri/Dataset/blob/main/pima-indians-diabetes-2.csv ) to build a Logistic Regression model that predicts whether a person has diabetes or not based on various health attributes.

Answer the following questions:

1. What are the unique values in the 'Outcome' column?
2. What is the size of the training dataset?
3. What is the size of the testing dataset?
4. What is the accuracy of the Logistic Regression model on the test set?
5. What do you observe from the pairplot of features and the target variable?

**Conclusion –**

The logistic regression model successfully classifies Iris species based on key features like petal length and width. The model shows high accuracy, confirming the importance of feature selection in effective classification.

## Viva Questions

| Sr.No | Question | CO |
|---|---|---|
| 1. | What is Logistic Regression and how does it differ from Linear Regression? | **1ICPC406_3** |
| 2. | When would you use Logistic Regression instead of Linear Regression? | **1ICPC406_3** |
| 3. | Explain the logistic function and how it is used in Logistic Regression? | **1ICPC406_3** |
| 4. | What is the range of the output of a Logistic Regression model and what does it represent? | **1ICPC406_3** |
| 5. | How do you handle the categorical predictors in Logistic Regression? | **1ICPC406_3** |
| 6. | What are some methods to assess the performance of a Logistic Regression model? | **1ICPC406_3** |
| 7. | What is the purpose of using a confusion matrix in the context of Logistic Regression? | **1ICPC406_3** |
| 8. | How can you handle imbalanced datasets in Logistic Regression? | **1ICPC406_3** |
| 9. | What is the difference between binary logistic regression and multinomial logistic regression? | **1ICPC406_3** |
| 10. | State a basic example for Logistic Regression? | **1ICPC406_3** |

References –


a. **Textbook –**
   i. Machine Learning with Python – An approach to Applied ML – Abhishek Vijayvargiya, BPB Publications, 1st Edition 2018
   ii. Machine Learning, Tom Mitchell, McGraw Hill Education, 1st Edition 1997

b. Online references –
   i. https://www.analyticsvidhya.com/blog/2021/08/conceptual-understanding-of-logistic- regression-for-data-science-beginners/
   ii. https://www.simplilearn.com/tutorials/machine-learning-tutorial/logistic-regression- in-python
   iii. https://www.kaggle.com/code/nargisbegum82/logistic-regression-in-machine- learning