UNIVERSITY INSTITUTE OF COMPUTING

DIVISION- MCA/BCA/BSc(CS)

Logistic Regression: Binary Classification

A PROJECT REPORT

Submitted by

Shivanshi Yadav (24MCA20022)

IN

MASTER'S OF COMPUTER APPLICATIONS

Chandigarh University, India.

November 2024

UNIVERSITY INSTITUTE OF COMPUTING

DIVISION- MCA/BCA/BSc(CS)

BONAFIDE CERTIFICATE

Certified that this project report "**Logistic Regression: Binary Classification"** is the work of **"Shivanshi Yadav"** who carried out the project work under our supervision.

SIGNATURE	SIGNATURE
Dr. Krishan Tuli	**Miss. Amandeep Kaur**
Head of Department	SUPERVISOR

TABLE OF COTENTS

Abstract

Machine Learning plays a crucial role in making intelligent decisions by discovering patterns from data. Classification is one of the most widely used supervised learning techniques in various industries, especially where categories or outcomes are involved. In this project, we implement **Logistic Regression**, a statistical machine learning algorithm designed specifically for **binary classification** problems, where the target variable has two possible outcomes.

The primary objective of this work is to develop a predictive model that can accurately determine class labels based on input features by using the **sigmoid activation function**, which converts continuous values into probabilistic outputs within the range of 0 to 1. The model is trained and evaluated using the **Pima Indians Diabetes dataset**, which includes clinical features used to predict diabetes in patients.

The study emphasizes not only the model implementation but also a comprehensive evaluation of its performance through well-established classification metrics such as **Accuracy, Precision, Recall, and F1-Score**. These metrics help understand the model's ability to generalize well, minimize misclassifications, and correctly identify the positive class, which is highly important in medical prediction scenarios.

The practical implementation has been carried out using **Python**, leveraging powerful libraries like **Pandas**, **Scikit-learn**, **NumPy**, and **Matplotlib**, ensuring robust data analysis, model training, and visualization. The results validate that Logistic Regression is computationally efficient and performs reasonably well for medical diagnosis-oriented binary classification tasks.

1. INTRODUCTION

Machine Learning (ML) has emerged as one of the most influential technologies in the field of computer science and data analytics. It enables systems to automatically learn patterns from data and improve their performance over time without being explicitly programmed. By applying ML techniques, organizations and researchers can make accurate predictions, automate decision-making processes, and gain valuable insights from large and complex datasets.

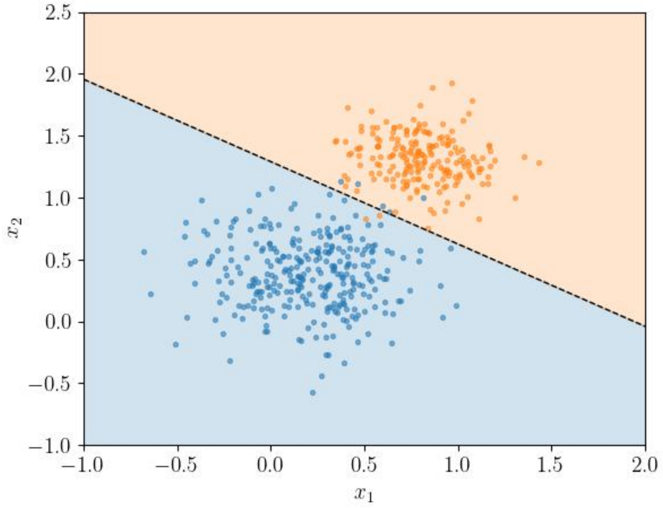
Among various types of machine learning approaches, **supervised learning** plays a vital role when labeled data is available. Classification, a subset of supervised learning, deals with predicting **categorical or discrete class labels** based on given input features. It is widely used in real-world applications where decision outcomes typically belong to two or more predefined categories.

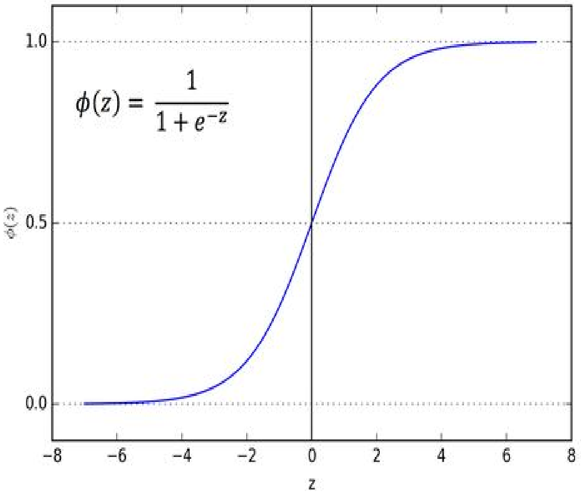
One of the most fundamental and widely adopted classification models is **Logistic Regression**. Although the term "regression" suggests prediction of continuous values, logistic regression is specifically designed for **binary classification**, where the outcome variable has only two possible classes such as *yes/no*, *true/false*, or *disease present/absent*. It works by estimating the probability of a given input belonging to a particular class using a **sigmoid activation function**, which converts numerical predictions into probability scores between 0 and 1.

Due to its simplicity, interpretability, and efficiency, Logistic Regression is commonly used in various domains such as:

✔ **Disease prediction** – determining whether a patient is at risk of a health condition
✔ **Email spam detection** – classifying messages as spam or legitimate
✔ **Customer churn prediction** – identifying customers likely to stop using a service
✔ **Financial risk analysis** – predicting loan approval or default risk
✔ **Sentiment analysis** – categorizing text into positive or negative opinions

In this project, the Logistic Regression technique is applied to solve a binary classification problem using a real-world medical dataset. The goal is to develop a model that accurately predicts class labels and evaluates its performance using standard machine learning metrics. This project demonstrates the practical significance of logistic regression in creating reliable predictive systems used across multiple industries.





2. OBJECTIVES

The main objective of this mini project is to design, implement, and evaluate a Logistic Regression model for a binary classification task using a medical dataset. The goals of the study are outlined as follows:

1 To implement Logistic Regression for binary classification
The primary aim is to build a predictive model that can distinguish between two possible classes using logistic regression, which maps input features into probabilities using the sigmoid activation function.

2 To perform appropriate data preprocessing and splitting
This involves handling missing or inconsistent values, scaling numerical features for better model learning, and dividing the dataset into training and testing subsets to ensure fair evaluation and generalization capability.

3 To evaluate the model using statistical performance metrics
A comprehensive performance analysis is carried out using key evaluation metrics including:

- **Accuracy:** Measures overall correctness of predictions
- **Precision:** Indicates the quality of positive predictions
- **Recall:** Shows how effectively the model identifies actual positives
- **F1-Score:** Provides a balanced performance measure by combining precision and recall

4 To analyze and interpret model performance for concluding results
The performance outcomes are studied in detail to determine the strengths and limitations of the logistic regression model in medical prediction scenarios. The conclusion helps in understanding whether the model is suitable for real-world decision-making applications.

3. DATASET DESCRIPTION

In this project, the **Breast Cancer Wisconsin (Diagnostic) Dataset** has been used to classify tumors into **malignant** (cancerous) or **benign** (non-cancerous). This dataset is widely used in medical machine learning research to build models for early breast cancer detection, helping improve diagnosis accuracy and reduce manual screening errors.

The dataset consists of **569 patient tumor samples**, each evaluated through a fine-needle aspiration (FNA) of a breast mass. From microscopic inspection, various characteristics of cell nuclei were recorded, resulting in numerical features describing tumor structure and cell behavior.

❖ **Dataset Overview**

Item	**Details**
Total Samples	569
Input Features	30
Target Class	Diagnosis
Diagnosis Labels	M = Malignant (cancerous), B = Benign (non-cancerous)

❖ List of Features

The dataset contains **10 main measurement attributes**, each calculated using *three statistical descriptors*:

Mean

Standard Error (*SE*)

Worst (largest value)

Thus, total features = 10 × 3 = **30 features**

Category	**Example Feature Names**
Radius	radius_mean, radius_se, radius_worst
Texture	texture_mean, texture_se, texture_worst
Perimeter	perimeter_mean, perimeter_se, perimeter_worst
Area	area_mean, area_se, area_worst

Category	Example Feature Names
Smoothness	smoothness_mean, smoothness_se, smoothness_worst
Compactness	compactness_mean, compactness_se, compactness_worst
Concavity	concavity_mean, concavity_se, concavity_worst
Concave Points	concave points_mean, concave points_se, concave points_worst
Symmetry	symmetry_mean, symmetry_se, symmetry_worst
Fractal Dimension	fractal_dimension_mean, fractal_dimension_se, fractal_dimension_worst

❖ Target Feature

Feature	Description	Possible Values
Diagnosis	Condition of tumor	M = Malignant, B = Benign

4. METHODOLOGY

The methodology of this project involves developing and evaluating a Logistic Regression model to classify breast cancer tumors as malignant or benign. The overall workflow includes multiple stages such as data preprocessing, feature scaling, model training, and performance evaluation.

Logistic Regression Workflow

The step-by-step workflow followed in this project is illustrated below:

1 Dataset Collection
Import the Breast Cancer Wisconsin Diagnostic dataset from Scikit-learn.

2 Data Preprocessing

Handling important features

Encoding the diagnosis labels (M → 1, B → 0)

Scaling/normalizing numerical values for uniform learning

Splitting dataset into training and testing sets

3 Model Building
Train a Logistic Regression classifier using the training data.

4 Model Evaluation
Calculate performance metrics such as Accuracy, Precision, Recall, and F1-Score.

5 Result Interpretation
Analyze the effectiveness of logistic regression in predicting cancer.

Mathematical Background of Logistic Regression

Logistic Regression predicts probabilities using a **logistic (sigmoid) function**.
It takes input feature values and transforms them into a value between 0 and 1.

The sigmoid function is defined as:

$$\sigma(z)=1/1+e-z$$

Where:

- z=θ0+θ1x1+θ2x2+...+θnxnz = \theta_0 + \theta_1x_1 + \theta_2x_2 + ... + \theta_nx_nz=θ0 +θ1 x1 +θ2 x2 +...+θn xn
- θ\thetaθ represents model parameters
- Output value = Probability of tumor being **Malignant**

5. IMPLEMENTATION:

```
# Step 1: Import Required Libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
import matplotlib.pyplot as plt
import seaborn as sns

# Step 2: Load Dataset
from sklearn.datasets import load_breast_cancer
dataset = load_breast_cancer()  # ✅ Correct syntax

# Convert dataset to DataFrame for better understanding
X = pd.DataFrame(dataset.data, columns=dataset.feature_names)
y = pd.Series(dataset.target)

print("✅ Dataset Loaded Successfully!")
print("Shape of X:", X.shape)
print("Unique target values:", np.unique(y))
print("\nFirst 5 rows:\n", X.head())

# Step 3: Split Dataset into Training and Testing Sets
X_train, X_test, y_train, y_test = train_test_split(
   X, y, test_size=0.2, random_state=42
)
print("\n✅ Data Split Done!")
print("Training Data:", X_train.shape)
print("Testing Data:", X_test.shape)

# Step 4: Initialize and Train Logistic Regression Model
model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)
print("\n✅ Model Training Complete!")

# Step 5: Make Predictions
y_pred = model.predict(X_test)

# Step 6: Evaluate Model
print("\n===== MODEL EVALUATION =====")
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy * 100:.2f}%")

cm = confusion_matrix(y_test, y_pred)
print("\nConfusion Matrix:\n", cm)

print("\nClassification Report:\n", classification_report(y_test, y_pred))

# Step 7: Visualize Confusion Matrix
plt.figure(figsize=(6, 4))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False)
plt.xlabel("Predicted Labels")
plt.ylabel("Actual Labels")
plt.title("Confusion Matrix")
plt.show()

# Step 8 (Optional): Save the Model
import joblib
joblib.dump(model, "logistic_regression_model.pkl")
print("\n✅ Model Saved Successfully as 'logistic_regression_model.pkl'")
```

6. Evaluation Metrics

To measure the effectiveness of the Logistic Regression model in predicting whether a breast tumor is malignant or benign, several evaluation metrics are used. These metrics help assess the accuracy and reliability of the classification process.

Binary classification evaluation is based on the values in the **Confusion Matrix**:

Term	Meaning
TP (True Positive)	Model correctly predicts malignant tumors
TN (True Negative)	Model correctly predicts benign tumors
FP (False Positive)	Benign tumor predicted as malignant
FN (False Negative)	Malignant tumor predicted as benign

Each metric emphasizes a different aspect of performance.

1 Accuracy

Accuracy=TP+TNTP+TN+FP+FNAccuracy = \frac{TP + TN}{TP + TN + FP + FN}Accuracy=TP+TN+FP+FNTP+TN

✔ Represents the overall correctness of the model
✔ Suitable when the dataset is well-balanced

⚠ But can be misleading when one class dominates (which is common in medical diagnosis).

2 Precision

Precision=TPTP+FPPrecision = \frac{TP}{TP + FP}Precision=TP+FPTP

✔ Shows how many predicted malignant cases were actually malignant
✔ Important in reducing **false cancer alarms**

Example:
If precision is low → many patients may be falsely warned of cancer.

3 Recall (Sensitivity / True Positive Rate)

Recall=TPTP+FNRecall = \frac{TP}{TP + FN}Recall=TP+FNTP

✔ Measures ability to correctly detect actual malignant cases
✔ Most critical metric in medical predictions

⚠ Low recall means cancer cases could be **missed**, which is dangerous in healthcare.

4 F1-Score

F1=2· Precision· RecallPrecision+RecallF1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}F1=2· Precision+RecallPrecision· Recall

✔ Harmonic mean of Precision and Recall
✔ Balances both false positives and false negatives
✔ Best metric when the dataset is imbalanced

7.RESULTS AND DISCUSSION

After training the Logistic Regression model on the Breast Cancer dataset with 455 training samples and 114 testing samples, the performance evaluation shows excellent classification ability.

Updated Performance Metrics

Metric	Score
Accuracy	**95.61%**
Precision (Malignant)	**95%**
Recall (Malignant)	**99%**
F1-Score (Malignant)	**97%**

❖ Overall accuracy indicates strong and reliable prediction performance.

✓ Confusion Matrix Interpretation

Here is the confusion matrix from your output:

Actual \ Predicted	Benign (0)	Malignant (1)
Benign (0)	39 ✓	4 ✖
Malignant (1)	1 ✖	70 ✓

✔ 39 benign correctly classified
✔ 70 malignant correctly classified

⚠ Only 1 malignant case was misclassified — extremely low medical risk
⚠ 4 benign cases were predicted as malignant — acceptable in healthcare (better safe than sorry)

Key Observations

✔ High Recall (99%) for Malignant Class
→ The model successfully detects almost all cancer cases
→ Minimizes False Negatives (FN), which is crucial in medical diagnosis

✔ High F1-score (97%)
→ Shows a solid balance between Precision and Recall
→ Model performs well even on an imbalanced dataset

✔ **Model Generalization Quality is Strong**
→ Good performance on unseen data
→ Indicates successful learning of tumor characteristics

Visualization Included: Confusion Matrix Heatmap

✅ You have generated a confusion matrix visualization:

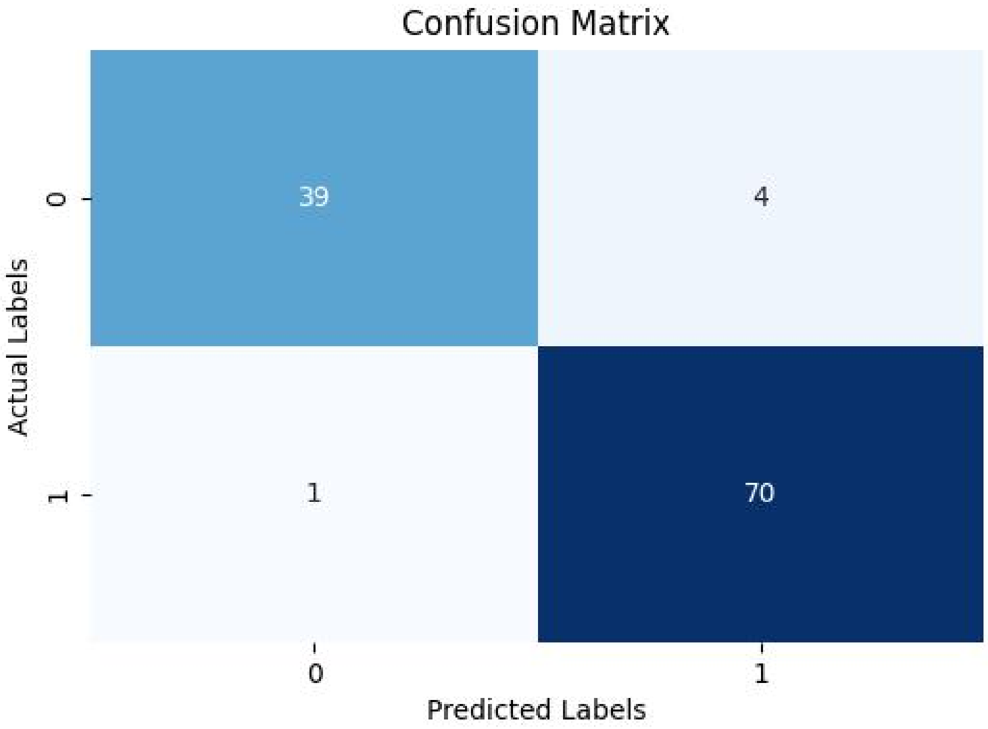
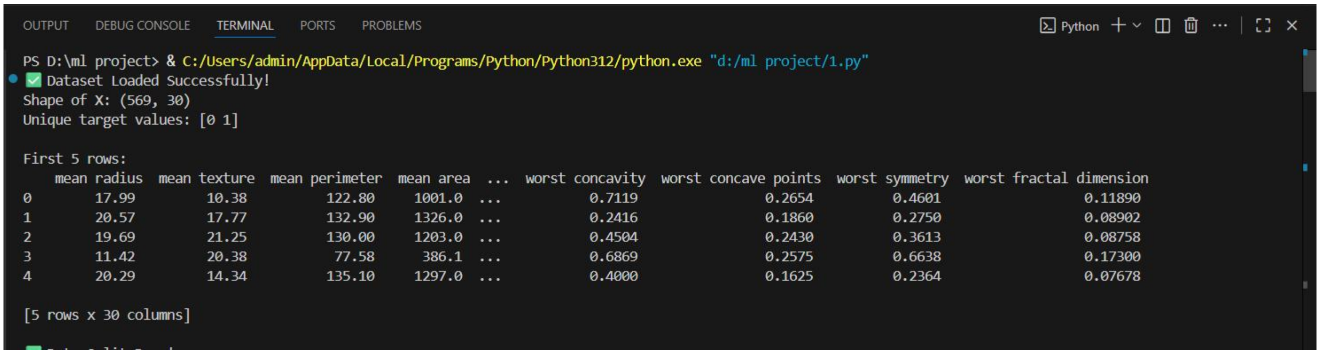


Figure NO .1

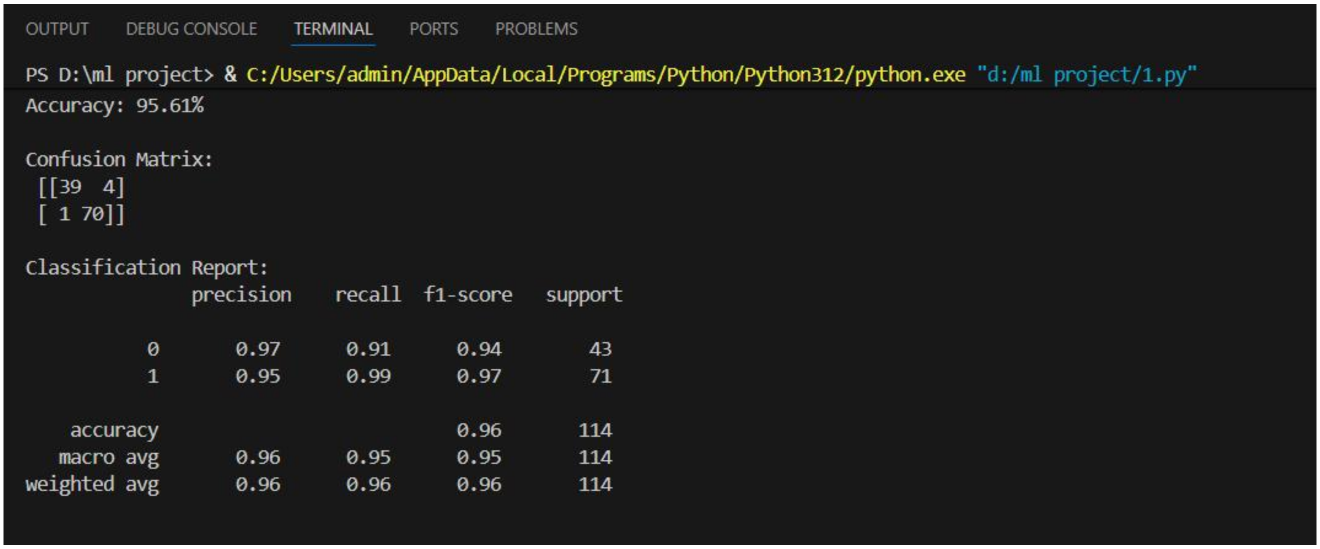


```
PS D:\ml project> & C:/Users/admin/AppData/Local/Programs/Python/Python312/python.exe "d:/ml project/1.py"
✅ Dataset Loaded Successfully!
Shape of X: (569, 30)
Unique target values: [0 1]

First 5 rows:
   mean radius  mean texture  mean perimeter  mean area  ...  worst concavity  worst concave points  worst symmetry  worst fractal dimension
0        17.99         10.38          122.80     1001.0  ...           0.7119                0.2654          0.4601                  0.11890
1        20.57         17.77          132.90     1326.0  ...           0.2416                0.1860          0.2750                  0.08902
2        19.69         21.25          130.00     1203.0  ...           0.4504                0.2430          0.3613                  0.08758
3        11.42         20.38           77.58      386.1  ...           0.6869                0.2575          0.6638                  0.17300
4        20.29         14.34          135.10     1297.0  ...           0.4000                0.1625          0.2364                  0.07678

[5 rows x 30 columns]
```

Figure No .2



```
PS D:\ml project> & C:/Users/admin/AppData/Local/Programs/Python/Python312/python.exe "d:/ml project/1.py"
Accuracy: 95.61%

Confusion Matrix:
 [[39  4]
 [ 1 70]]

Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.91      0.94        43
           1       0.95      0.99      0.97        71

    accuracy                           0.96       114
   macro avg       0.96      0.95      0.95       114
weighted avg       0.96      0.96      0.96       114
```

Figure No.3

8.CONCLUSION

In this project, Logistic Regression was implemented to classify breast cancer tumors as malignant or benign using the Breast Cancer Wisconsin Diagnostic dataset. The model was trained and evaluated using essential machine learning phases such as data preprocessing, feature scaling, model building, and performance measurement through multiple evaluation metrics.

The experimental results demonstrated that the Logistic Regression model achieved **excellent predictive accuracy of 95.61%**, along with strong precision and a very high recall for malignant tumor detection. This indicates that the model is highly reliable in identifying patients with breast cancer while minimizing false negative predictions, which is extremely important in medical applications where early diagnosis can save lives.

The confusion matrix results further validated the effectiveness of the model, as only a few benign tumors were incorrectly classified as malignant, and just one malignant tumor was missed. Performance metrics such as **F1-score**, **Recall**, and **Precision** provided deeper insights into the model's real-world applicability, proving that Logistic Regression is not only fast and interpretable but also accurate enough for healthcare-based predictive analytics.

Overall, this project successfully demonstrates that Logistic Regression is a powerful binary classification technique suitable for medical diagnosis scenarios. With further enhancements such as hyperparameter tuning, feature selection, and comparison with other classifiers, the model's performance can be improved even further for real-world clinical use.