Logistic Regression in Detail with Real-Life Example

1. What is Logistic Regression?

Logistic Regression is a supervised machine learning algorithm used for classification, especially binary classification.

It answers: What is the probability that the output is 1 (Yes, Pass, Spam, etc.) given some input features?

2. Real-Life Example: Email Spam Detection

You want to build a system that predicts whether an email is spam or not spam.

Features:

- Free_Word: 1 if the word "free" is present, 0 otherwise

- Exclamations: Number of exclamation marks

Target:

- Spam: 1 = Spam, 0 = Not Spam

3. Why Not Linear Regression?

Linear regression can give outputs like 1.8 or -0.7 which are not valid probabilities.

Probabilities must be between 0 and 1.

4. The Sigmoid Function

$$\text{sigmoid}(z) = 1 / (1 + e^{-z})$$

This maps any value to a range between 0 and 1.

5. Logistic Regression Model

$$P(y = 1 \mid x) = 1 / (1 + e^{-(w_1 x_1 + w_2 x_2 + ... + b)})$$

Classify as 1 if $P \geq 0.5$ else classify as 0.

6. Python Code Example (Spam Detection)

```python
import pandas as pd

from sklearn.linear_model import LogisticRegression

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, classification_report


data = {

    'Free_Word': [1, 0, 1, 0, 1, 0],

    'Exclamations': [5, 0, 3, 1, 4, 0],

    'Spam': [1, 0, 1, 0, 1, 0]

}

df = pd.DataFrame(data)

X = df[['Free_Word', 'Exclamations']]

y = df['Spam']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

model = LogisticRegression()

model.fit(X_train, y_train)

y_pred = model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))

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

7. predict_proba

model.predict_proba(X_test) gives the probability for class 0 and class 1.

8. Evaluation Metrics

- Accuracy

- Precision

- Recall

- F1-Score

- Confusion Matrix

- ROC-AUC

9. When to Use Logistic Regression

- Target variable is binary

- Interpretable coefficients are needed

- Features have a linear relationship (in log-odds)

- Need probability estimates

10. Summary

Type: Classification

Output: Probability (0 to 1)

Function: Sigmoid

Best for: Binary classification

Examples: Spam detection, medical diagnosis, fraud detection