📊 *Logistic Regression: Solving Binary Classification Problems* ☑✖

Logistic Regression is a go-to algorithm for solving binary classification problems. Despite its name, it's a classification algorithm, not a regression one. Let's dive into its core concepts and how it works!

Summary

Logistic Regression predicts the probability of an instance belonging to a class (e.g., 0 or 1). It's particularly effective for problems like spam detection, loan default prediction, and medical diagnoses.

Highlights

🔢 ***Binary Output:*** The model outputs probabilities between 0 and 1. A threshold (commonly 0.5) is used to classify the result as 0 or 1.
📐 ***Decision Boundary:*** Logistic Regression fits a linear boundary to separate the two classes in feature space.
📈 ***Sigmoid Function:*** Uses the logistic (sigmoid) function to map predicted values to probabilities.

Key Insights

🚀 *Why Logistic Regression?*

Simple yet powerful for linearly separable data.
Outputs interpretable probabilities for decision-making.
⚖️ ***Threshold Tuning:*** Adjust the threshold (e.g., 0.4, 0.6) based on the problem's sensitivity and specificity requirements.

📊 *Evaluation Metrics:*

Accuracy: Overall correctness.
Precision: How many positive predictions were correct?
Recall: How many actual positives were identified?
F1-Score: Balance between precision and recall.

Key Takeaways

1️⃣ ***Simple and Intuitive:*** Logistic Regression works well for binary classification when the data is linearly separable.

2️⃣ ***Metrics Matter:*** Use precision, recall, and F1-score to evaluate performance, especially for imbalanced datasets.

3️⃣ ***Thresholding:*** Customize the threshold based on problem requirements for better decision-making.

GitHub code : https://github.com/NafisAnsari786/Machine-Learning-Algorithms/blob/main/7%20Logistic%20Regression/Binary%20Classification/Logistic%20Regression%20Binary%20classification.ipynb