Module: Understanding Logistic Regression through a Classification Problem

Welcome to the Logistic Regression module! While Linear Regression is powerful for predicting continuous values, many real-world problems involve predicting categories or class memberships (e.g., spam/not spam, customer churn yes/no, disease present/absent). Logistic Regression is a fundamental algorithm designed for these **classification problems**, specifically for estimating the *probability* that an instance belongs to a particular class.

Structure of this Module

This module will explore Logistic Regression in detail:

1. **Introduction to Logistic Regression** *(Current Section)*
2. Estimating Probabilities
3. Logistic Regression Cost Functions
4. Softmax Regression (for multi-class problems)
5. Performance Metrics (for classification)
6. ROC Curve and AUC
7. Optimising Logistic Regression Model (Project)

Understanding the Logit Model (Introduction to Logistic Regression)

* **Purpose:** In statistics, the **logistic model** (or **logit model**) is primarily used to **model the probability of a specific class or event occurring**. Examples include pass/fail, win/lose, alive/dead, or healthy/sick.
* **Logistic Regression Defined:** Logistic regression is a statistical method that, in its basic form, uses a specific mathematical function called the **logistic function** (or sigmoid function) to model a **binary dependent variable** (a variable with only two possible outcomes).
* **Parameter Estimation:** Similar to linear regression, the process involves **estimating the parameters** (coefficients) of the logistic model based on the input data.
* **Binary Dependent Variable:** Mathematically, a binary logistic model has a dependent variable Y that can take only two values, typically represented by an indicator variable coded as "0" and "1" (e.g., 0 for "fail", 1 for "pass"; 0 for "healthy", 1 for "sick").
* **The Core Idea - Log-Odds (Logit):** Instead of directly modeling the probability p = P(Y=1) as a linear combination of predictors (which could lead to probabilities outside the valid 0-1 range), logistic regression models the **log-odds** (also called the **logit**) of the probability as a linear combination.
  + **Odds:** The ratio of the probability of the event happening (p) to the probability of it not happening (1-p). Odds = p / (1-p).
  + **Log-Odds (Logit):** The natural logarithm (usually) of the odds. Log-Odds = ln(p / (1-p)).
  + **Linear Relationship:** Logistic regression assumes: log-odds = β₀ + β₁x₁ + β₂x₂ + ... + β<0xE2><0x82><0x99>x<0xE2><0x82><0x99>
    - Where β₀, β₁, ... are the model parameters (coefficients) and x₁, x₂, ... are the independent predictor variables (which can be binary, categorical encoded, or continuous).
* **From Log-Odds to Probability:** The corresponding probability p (the probability of the outcome labeled "1") can range only between 0 (certainly the value "0") and 1 (certainly the value "1"). The **logistic function** is the crucial mathematical link that converts the linear combination of predictors (the log-odds) back into this valid probability range [0, 1].
* **Characteristic Interpretation:** A defining characteristic, derived from this setup, is that increasing one of the independent variables (xᵢ) by one unit **multiplicatively scales the odds** of the outcome (Y=1) by a factor of exp(βᵢ), holding other variables constant.
* **Beyond Binary:**
  + For outputs with more than two *unordered* values (e.g., classifying fruit type as 'apple', 'orange', 'banana'), **multinomial logistic regression** is used.
  + For outputs with more than two *ordered* values (e.g., rating scale 'poor', 'fair', 'good', 'excellent'), **ordinal logistic regression** models (like the proportional odds model) are applicable.

Understanding the Math: From Log-Odds to Sigmoid

Let's see how the probability p is derived. Consider a model with two predictors, x₁ and x₂, and a binary response Y. We want to find p = P(Y=1).

1. **Assume Linearity in Log-Odds:** We start with the core assumption that the log-odds (logit), denoted by l, are linearly related to the predictors:
2. l = log\_b( p / (1-p) ) = β₀ + β₁x₁ + β₂x₂

(Where log\_b is the logarithm with base b. Usually, the natural logarithm, base e, is used, where e ≈ 2.71828.)

1. **Exponentiate to Get Odds:** To isolate the odds, we exponentiate both sides using the base b:
2. Odds = p / (1-p) = b^(β₀ + β₁x₁ + β₂x₂)

This shows that once the coefficients (βs) are fixed, we can compute the *odds* of Y=1 for a given observation (x₁, x₂).

1. **Solve for Probability (p):** Now, we perform algebraic manipulation to solve for p:
   * Let Z = β₀ + β₁x₁ + β₂x₂
   * p / (1-p) = b^Z
   * p = b^Z \* (1-p)
   * p = b^Z - p \* b^Z
   * p + p \* b^Z = b^Z
   * p \* (1 + b^Z) = b^Z
   * p = b^Z / (1 + b^Z)

We can rewrite this by dividing the numerator and denominator by b^Z:

p = 1 / ( (1/b^Z) + 1 ) = 1 / ( 1 + b^(-Z) )

Substituting Z back in:

p = 1 / ( 1 + b^-(β₀ + β₁x₁ + β₂x₂) )

1. **The Sigmoid Function:** This final form p = 1 / (1 + b^(-Z)) is known as the **Sigmoid function** (or logistic function) with base b, often denoted as S\_b(Z).
2. p = S\_b(β₀ + β₁x₁ + β₂x₂)
   * When base e is used (most common), p = 1 / (1 + e^-(β₀ + β₁x₁ + β₂x₂)).
   * This function takes any real-valued input Z (the linear combination of predictors) and squashes it into an output probability p between 0 and 1.

**The main use-case of a logistic model is to estimate the probability p = P(Y=1) given the values of the predictor variables (x₁, x₂, ...).** The model learns the parameters (β₀, β₁, ...) from the data that best achieve this estimation.