A Comprehensive Guide to Linear Regression, Logistic Regression, and Multinomial Logistic Regression

Done by: Malika Alnaabi

1. Introduction:

Regression models are foundational in statistics and machine learning. This document explores three key regression models: linear regression, Logistic Regression, and Multinomial Logistic Regression,.

1. Linear Regression:

Definition:

Linear regression models the relationship between a dependent variable y and a single independent variable x using a straight line.

Mathematical Formula:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

Deriving the Formula of MSE:

We assume a model of the form:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

We aim to find parameters **β0** and **β1**that minimize the error. To do this, we use the **Ordinary Least Squares (OLS)** method.

Minimize:

$$\text{Loss} = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2$$

Solving the derivatives for β0 and β1 gives the optimal parameters.

Loss Function (Mean Squared Error - MSE):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

The loss function indicates how far the predictions are from actual values. It is a convex function, which means it has one global minimum. It is used to train the model by optimizing weights.

Assumptions of Linear Regression:

- **Linearity**: The relationship between the independent and dependent variable is linear.
- **Homoscedasticity**: The residuals (errors) have constant variance across all levels of the independent variable.
- **No Multicollinearity**: Although not critical for simple linear regression (with one feature), this becomes important when interpreting relationships or extending to multiple linear regression. The feature must not be highly correlated with other potential explanatory variables.
- **Independence of Errors**: Residuals should be independent of each other.
- **Normality of Errors**: Residuals should be approximately normally distributed for valid inference.

R-squared:

- Measures the proportion of the variance in the dependent variable that is predictable from the independent variable(s).
- Ranges from 0 to 1; a higher value indicates a better fit.

Formula:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Coefficient of Determination (R Square)

$$R^2 = \frac{SSR}{SST}$$

$$SSR = \sum_i (\hat{y}_i - \bar{y})^2$$

$$SST = \sum_i (y_i - \bar{y})^2$$

Where,

- SSR is Sum of Squared Regression also known as variation explained by the model
- SST is Total variation in the data also known as sum of squared total
- y_i is the y value for observation i
- y_bar is the mean of y value
- y_bar_hat is predicted value of y for observation i

www.ashutoshtripathi.com

Source: https://ashutoshtripathi.com/2019/01/22/what-is-the-coefficient-of-determination-r-square/

Gradient Descent vs. OLS:

- **Ordinary Least Squares (OLS):** Solves the problem analytically by minimizing the sum of squared residuals; provides an exact solution for small-to-moderate datasets.
- **Gradient Descent:** An iterative optimization algorithm; used when the dataset is large or OLS is computationally

expensive. It updates coefficients gradually using gradients of the loss function.

Underfitting vs. Overfitting:

- **Underfitting:** The model is too simple and cannot capture the underlying trend. This results in high bias and poor performance on both training and test sets.
- **Overfitting:** The model is too complex and fits the training data too well, including noise. This results in low training error but high test error due to poor generalization.
- **A good linear regression model** should balance complexity and generalization using validation techniques such as cross-validation.

2- Logistic Regression:

Definition:

Logistic Regression is used for binary classification problems where the outcome variable y is categorical with two possible outcomes (0 or 1). Instead of modeling y directly, it models the log-odds (logit) of the probability of class 1 as a linear combination of input features.

Mathematical Formula:

We model the probability P(y=1|x) using the sigmoid (logistic) function:

$$\hat{y} = P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n)}}$$

This is equivalent to modeling the **log-odds** as a linear equation:

$$\log\left(\frac{P(y = 1|x)}{1 - P(y = 1|x)}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$$

Loss Function (Binary Cross-Entropy / Log Loss):

The goal is to find coefficients β that minimize the error between predicted probabilities y^\hat and true outcomes y:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)}\log(\hat{y}^{(i)}) + (1-y^{(i)})\log(1-\hat{y}^{(i)})\right]$$

Why This Loss Function?

- Derived from **Maximum Likelihood Estimation (MLE)**.
- Assumes Bernoulli distribution for binary outcome.
- Log-likelihood is easier to optimize and convex for logistic regression.

Maximizing the likelihood → Minimizing the negative log-likelihood → Cross-entropy loss.

Gradient Descent in Logistic Regression:

Since there's no closed-form solution like in linear regression, we use gradient descent to update weights:

$$\beta_j := \beta_j - \alpha \cdot \frac{\partial \mathcal{L}}{\partial \beta_j}$$

Interpretation:

- Coefficients represent a change in **log-odds** for a unit change in the feature.
- Output y^\hat is a **probability** between 0 and 1.
- Decision boundary: often set at 0.5 to classify 0 or 1.

Assumptions of Logistic Regression:

- Binary outcome (extension needed for multi-class).
- Linearity in the logit.
- Independence of observations.
- No multicollinearity among independent variables.
- Large sample size is preferred for stable estimation.

Evaluation Metrics:

- **Accuracy**: Proportion of correct predictions.
- **Precision**: TP / (TP + FP)
- **Recall**: TP / (TP + FN)
- **F1-Score**: Harmonic mean of Precision and Recall.
- **ROC-AUC**: Performance across different thresholds.

3-Multinomial Logistic Regression:

Definition:

Multinomial Logistic Regression is a generalization of binary logistic regression used when the target variable has more than two classes (e.g., Class A, B, C). It models the probability that an observation belongs to each of the possible categories using the softmax function.

It is used for multi-class classification problems where the classes are mutually exclusive.

Mathematical Formula:

$$P(y = k|x) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{for } k = 1, 2, \ldots, K$$

This is known as the **softmax function**, and it ensures that:

- All probabilities lie between 0 and 1.
- The probabilities sum to 1 across all classes.

Loss Function (Categorical Cross-Entropy): We use the negative log-likelihood of the true class labels over the predicted class probabilities:

$$\mathcal{L} = -\sum_{i=1}^{n}\sum_{k=1}^{K} y_k^{(i)} \log(\hat{y}_k^{(i)})$$

Why This Loss Function?

- Derived using **maximum likelihood estimation** for the multinomial distribution.
- Penalizes the model heavily if the probability of the correct class is low.
- Encourages the model to assign high confidence to the correct class.

Parameter Estimation (Gradient Descent): There's no analytical (closed-form) solution. We optimize the parameters using **gradient-based optimization** techniques like:

- Gradient Descent
- Stochastic Gradient Descent (SGD)
- Adam

Each weight vector βk is updated by computing the gradient of the loss function with respect to each class's parameters.

Interpretation:

- The model assigns **probabilities** to each class.
- Prediction is the class with the highest probability.
- Coefficients are interpreted relative to a **baseline class** (reference class).

Assumptions of Multinomial Logistic Regression:

- **Mutually exclusive classes** (only one correct label per instance).
- **Linearity in the log-odds** for each class.
- **Independence of irrelevant alternatives (IIA)**: the odds between two categories are not affected by the presence or absence of other categories.
- **No multicollinearity** among features.

Evaluation Metrics:

- Accuracy
- Precision / Recall / F1-Score (macro, micro, weighted)
- Confusion Matrix
- Log Loss for multi-class
- Cross-Entropy

Comparison Between Linear, Logistic, and Multinomial Logistic Regression

Aspect	Linear Regression	Logistic Regression	Multinomial Logistic Regression
Problem Type	Regression	Binary Classification	Multi-class Classification
Output	Continuous value (e.g., price, age)	Probability (between 0 and 1), then 0 or 1 class	Probabilities for each class (sum to 1)
Equation	$y = \beta_0 + \beta_1 x + \varepsilon$	$\hat{y} = P(y=1\|x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_n x_n)}}$	$P(y=k\|x) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$
Loss Function	Mean Squared Error (MSE)	Binary Cross-Entropy	Categorical Cross-Entropy
Function Type	Linear	Sigmoid (S-shaped curve)	SoftMax(multi-class extension of sigmoid)
Interpretation of Coefficients	Change in output value for unit change in feature	Change in log-odds of class 1	Change in log-odds relative to baseline class
Solution Method	Analytical (OLS) or Gradient Descent	Gradient Descent (No closed form)	Gradient Descent (One weight vector per class)
Target Variable Type	Continuous (Real numbers)	Binary (0 or 1)	Categorical (3 or more unordered classes)

Evaluation Metrics	RMSE, R^2, MAE	Accuracy, Precision, Recall, AUC	Accuracy, F1-score (macro/micro), Log Loss
Assumptions	Linearity, Normality, Homoscedasticity, Independence	Linearity in log-odds, Independence, No multicollinearity	Linearity in log-odds, IIA assumption, No multicollinearity
Common Use Cases	House price prediction, demand forecasting	Email spam detection, medical diagnosis	Sentiment analysis, image classification, topic prediction

Key Conceptual Differences

1. Prediction Type:

- Linear Regression gives **exact numerical predictions**.
- Logistic Regression gives **probabilities** used to classify into 0/1.
- Multinomial Logistic Regression gives **probabilities for multiple classes**, allowing classification into one of several outcomes.

2. Mathematical Foundation:

- Linear regression minimizes squared differences (distance).
- Logistic models use **likelihood-based** loss (cross-entropy), aligning better with classification goals.

- Multinomial logistic regression expands this idea by **predicting probabilities across multiple competing categories**.

3. Model Output Space:

- Linear regression outputs are unbounded real values.
- Logistic regression is bounded between 0 and 1.
- Multinomial regression is bounded per class between 0 and 1, and all outputs sum to 1.

Practical Tip When Choosing:

- Use **Linear Regression** if the label is a number.
- Use **Logistic Regression** if the label is **binary**.
- Use **Multinomial Logistic Regression** if the label is **categorical with >2 classes**, and no inherent order (if there is an order, **Ordinal Regression** might be better).

Optimization in Machine Learning

What Is Optimization?

Optimization refers to the process of adjusting a model's parameters (weights and biases) to **minimize a loss function**. In simpler terms, it's how we make a model *learn* from data by reducing its error in predicting outcomes.

In supervised learning like regression or classification:

- We define a **loss function** that measures prediction error.
- Optimization finds the parameters (like β) that **minimize** this loss function.

1-Optimization in Linear Regression

Objective:

Minimize the **Mean Squared Error (MSE)**:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Methods:

1. Ordinary Least Squares (OLS) (Analytical Solution)

- Directly solves the optimal parameters using matrix algebra.
- Formula:

$$\beta = (X^T X)^{-1} X^T y$$

Very efficient for small and moderate datasets.

Not suitable when:

- The data is too large,
- The features are collinear,
- Inversion of X^T is computationally expensive.

2. Gradient Descent (GD) (Numerical Optimization)

- Iteratively update parameters to minimize the loss:

$$\beta_j := \beta_j - \alpha \cdot \frac{\partial MSE}{\partial \beta_j}$$

α\alphaα is the learning rate.

Each parameter is updated in the **opposite direction** of the gradient.

Derivation of Gradient for MSE:

$$\frac{\partial MSE}{\partial \beta_j} = -\frac{2}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)x_{ij}$$

2-Optimization in Logistic Regression:

Objective:

Minimize the **Binary Cross-Entropy (Log Loss)**:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

Where $\hat{y}_i = \frac{1}{1+e^{-z_i}}$, and $z_i = \beta_0 + \sum \beta_j x_{ij}$

Why Not OLS?

- Logistic regression is **non-linear** due to the sigmoid function.
- There is **no closed-form solution** like OLS.
- We **must use numerical methods** such as:

Optimization Methods:

1. Gradient Descent

- Loss is minimized by iteratively updating the weights.
- Gradient of the log-loss function:

$$\frac{\partial \mathcal{L}}{\partial \beta_j} = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i) x_{ij}$$

Update rule:

$$\beta_j := \beta_j - \alpha \cdot \frac{\partial \mathcal{L}}{\partial \beta_j}$$

2. Stochastic Gradient Descent (SGD)

- Updates weights using **one sample at a time**.
- Faster for large datasets, but more variance per step.

3. Mini-Batch Gradient Descent

- Compromise between full batch and SGD.
- Updates based on a small batch of examples.

4. Advanced Optimizers (used in logistic/multinomial models):

- **Adam** (Adaptive Moment Estimation)
- **RMSProp**
- **LBFGS** (Quasi-Newton method)

Summary of Optimization Concepts

Concept	Linear Regression	Logistic Regression
Loss Function	Mean Squared Error	Binary Cross-Entropy
Has Closed-Form Solution?	Yes (OLS)	No
Needs Iterative Solver?	Sometimes	Always
Common Optimizers	OLS, Gradient Descent	Gradient Descent, SGD, Adam, LBFGS