RSM AI Report 1: Regression Basics

By Adriteyo Das

1 Linear Regression

Linear Regression is a supervised learning algorithm used for predicting continuous values. The goal is to model the relationship between the dependent variable y and independent variables X.

1.1 Model Representation

The hypothesis for Linear Regression is:

$$h_\theta(x) = w \cdot x + b$$

where:

- w is the weight (slope).
- b is the bias (intercept).
- x is the input feature vector.

1.2 Cost Function

The cost function measures the error between predicted and actual values. For Linear Regression, the cost function is the Mean Squared Error (MSE):

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

where:

- m is the number of training examples.
- $h_\theta(x^{(i)})$ is the prediction for the i-th training example.
- $y^{(i)}$ is the actual output for the i-th training example.

1.3 Error Analysis

Errors in Linear Regression can be:

- **Bias**: Systematic error due to assumptions in the model.
- **Variance**: Sensitivity of the model to small fluctuations in the data.

2 Logistic Regression

Logistic Regression is used for binary classification problems. Instead of predicting a continuous value, it predicts probabilities.

2.1 Model Representation

The hypothesis for Logistic Regression uses the sigmoid function:

$$h_\theta(x) = \sigma(w \cdot x + b)$$

where:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

2.2 Cost Function

For Logistic Regression, the cost function is the Log Loss:

$$J(w, b) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right]$$

2.3 Why Sigmoid?

The sigmoid function maps predictions to probabilities in the range $[0, 1]$. It ensures a meaningful probabilistic interpretation for classification tasks.

3 Gradient Descent

Gradient Descent is an optimization algorithm used to minimize the cost function $J(w, b)$.

3.1 Algorithm

The weights and bias are updated iteratively:

$$w := w - \alpha \frac{\partial J}{\partial w}, \quad b := b - \alpha \frac{\partial J}{\partial b}$$

where:

- α is the learning rate.
- $\frac{\partial J}{\partial w}$ and $\frac{\partial J}{\partial b}$ are the gradients of the cost function.

3.2 Learning Rate

The learning rate controls the step size in the parameter space:

- **Too Small**: Slow convergence.
- **Too Large**: Divergence or oscillations.

4 Classification Metrics

Metrics evaluate the performance of classification models.

4.1 Accuracy

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

4.2 Precision, Recall, F1-Score

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$
$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$
$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.3 ROC-AUC

The Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR). The Area Under the Curve (AUC) summarizes the performance.

5 Overfitting and Underfitting

Overfitting and underfitting are issues that arise during model training.

5.1 Overfitting

A model overfits when it performs well on training data but poorly on unseen data. Causes include:

- High model complexity.
- Insufficient training data.

5.2 Underfitting

A model underfits when it cannot capture the underlying patterns in the data. Causes include:

- Low model complexity.
- Poor feature engineering.

5.3 Solutions

- Use regularization (e.g., L1, L2).
- Collect more data.
- Optimize model architecture.