Loss Function in Deep Learning

Loss Function is a method of evaluating how well your algorithm is modelling your dataset.

If output of Loss Function is Big , means Algo is performing bad

If output of Loss function is small, Means algo is performing good,

Loss Function is Just a Mathematical Function.

Loss Function ka Value change kaise hote hai?

By changing parameters of Algos.

This was general small revision.

Why is Loss Function Important ?

"You can't Improve What you can't Measure."

~ Peter Drucker.

Galti janna bhaut important hota hai .

A loss function, also known as a cost function or objective function, is a crucial component in the training of machine learning models. Its primary purpose is to measure the difference between the predicted values of the model and the actual values (ground truth) of the target variable. The goal during the training process is to minimize this difference, as it reflects how well the model is performing on the given task.

The choice of a loss function depends on the type of problem you are trying to solve (classification, regression, etc.) and the nature of your data. Here are a few common types of loss functions:

Loss Function	Cost Function
Measures the error between predicted and actual values in a machine learning model.	Quantifies the overall cost or error of the model on the entire training set.
Used to optimize the model during training.	Used to guide the optimization process by minimizing the cost or error.
Can be specific to individual samples.	Aggregates the loss values over the entire training set.
Examples include mean squared error (MSE), mean absolute error (MAE), and binary cross-entropy.	Often the average or sum of individual loss values in the training set.
Used to evaluate model performance.	Used to determine the direction and magnitude of parameter updates during optimization.
Different loss functions can be used for different tasks or problem domains.	Typically derived from the loss function, but can include additional regularization terms or other considerations.

The terms "loss function" and "cost function" are often used interchangeably, but in some contexts, they can have slightly different meanings. Let's explore the differences:

1. **Loss Function:**
    - The loss function is a term more commonly associated with individual data points or samples. It measures the error between the predicted output of the model and the actual target for a single data point.
    - In the context of training a machine learning model, the loss function quantifies how well the model is performing on a single training example. It is computed for each example independently.
    - For example, in the case of mean squared error (MSE) for regression, the loss for a single data point is the squared difference between the predicted and actual values.
2. **Cost Function:**
    - The cost function, on the other hand, is a term often used in the context of the entire training dataset. It represents the overall performance of the model across all training examples.
    - The cost function is the average of the individual losses over the entire dataset. It gives a global measure of how well the model is doing on average.
    - For example, in the context of mean squared error (MSE), the cost function is the average of the squared differences over all training examples.

In summary, while the terms are often used interchangeably, "loss function" tends to refer to the error for a single data point, and "cost function" tends to refer to the overall performance across the entire dataset. In practice, people often use either term without making a strict distinction between them.

Loss Function is Eye of Algorithm

Loss Function in Deep Learning

1. Regression
    1. MSE(Mean Squared Error)
    2. MAE(Mean Absolute Error)
    3. Hubber loss
2. Classification
    1. Binary cross-entropy
    2. Categorical cross-entropy
3. AutoEncoder
    1. KL Divergence
4. GAN
    1. Discriminator loss
    2. Minmax GAN loss
5. Object detection
    1. Focal loss
6. Word embeddings
    1. Triplet loss

- ## MSE (Mean Squared Error)

1. **Easy to Interpret**
2. **Always Differentiable. (G.D)**
3. **Only 1 Local Minima**

Disadvantages

- Error ka Unit Squared rehta hai, so sochna rehta normal
- It is not Robust to outlier jish vajeh se bhautlog use nahi krte.

MSE ke tym 1 chiz main sure krna hai

Jo Last neuron hai uska Activation function= “Linear” Rehna chaye tbhi Hum Mean Squared Error Apply karskte hai.

MAE (L1 Loss)

L = | yi – ^yi |

C = 1/n i=1 to n |y - ^yi |

Advantages:

1.) Intuitive and easy to understand

2.) Unit -> same – y

3.) Robust to Out Lier.

Disadvantages:

1.Graph is Not Differentiable , so we can’t use Gradient Descent Directly.

Huber Loss

MSE ispe outlier impact daalte hai.

MAE pe Outlier ka impact nahi hai.

Huber Loss

$$L = \begin{cases} \frac{1}{2}(y-\hat{y})^2 & \text{for } |y-\hat{y}| \leq \delta \\ \delta |y-\hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases}$$

Whole Idea of Huber Loss is yeah MAE & MSE k beech mai lie krta hai

Yeah tbh acha perform krta hai jbh Outlier hoo.

Binary Cross Entropy

It is only used when there is only 2 classes,

E.g logistic mai use hota hai.

Binary entropy use krrhe tou

Output layer mai humesa Activation function Sigmoid rkhna hai

Categorical Cross Entropy

Output mai utne hie neuron hote jitne classes hote hai ,

Activation function sbhka Softmax hogga.

Sparse Categorical Entropy