Loss Functions

What is a loss function?

A loss function, also referred to as an error function, is a function that compares the target and the predicted output values. In other words, a loss function measures how well the neural network models the training data.
We use loss functions to assess the performance of a model in a specific task, which in most cases, is regression or classification.

Loss functions, along with optimizers, are responsible for fitting the model to the given data. When training the model, our goal is to minimize the value of the loss function to reduce the loss between the predicted and target outputs.
There are different types of loss functions; namely Mean Squared Error (MSE), Mean Absolute Error (MAE), Binary Cross-Entropy and Categorical Cross-Entropy.

How do loss functions work?

Although there are different types of loss functions, fundamentally, they all operate by quantifying the difference between a model's predictions and the desired target value in the data. This difference is often referred to as the prediction error.The learning algorithm in a machine learning model is optimized to minimize the prediction error. This means that after each round of calculating the prediction error, the learning algorithm uses this information to update the values of the parameters of the model, namely weights and biases, which most likely leads to a lower prediction error in the next round of training.

Types of loss functions

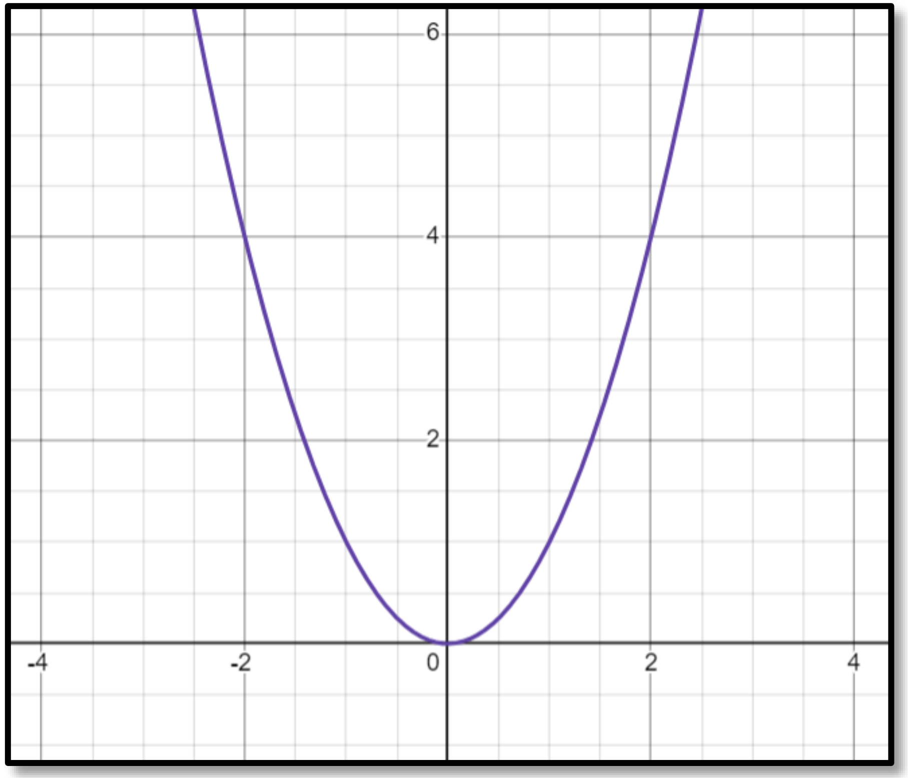
1. Mean Squared Error (MSE)

Mean Squared Error (MSE) is one of the most popular loss functions in machine learning models, particularly suitable for regression problems. MSE calculates the average of the squared differences between the target and predicted outputs. The difference between the target and predicted values is squared, which means it does not matter whether the difference is greater or less than zero. However, squaring the difference results in a higher penalty assigned to larger deviations from the target value. This function has a clearly defined global minimum, which facilitates the use of gradient descent optimization to adjust the weight values effectively.

Types of loss functions

1. Mean Squared Error (MSE)

We can interpret a lower MSE as an indication that the model's predictions are closer to the actual values, implying better accuracy. Conversely, a higher MSE suggests that the model's predictions deviate further from the actual values, indicating poorer performance.



Types of loss functions

1. Mean Squared Error (MSE)

Here is the mathematical equation for MSE:

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

Where:

n: number of data points
y_i: target value of data point i
$\hat{y}_i$: predicted value of data point i

MSE is widely utilized in most regression tasks because it directs the model to minimize the squared differences between the predicted and target values. However, one disadvantage of this loss function is its sensitivity to outliers. This means that if a predicted value is significantly greater than or less than its target value, the loss will increase substantially, which may pose challenges in the optimization process.

Types of loss functions

1. Mean Squared Error (MSE)

Here is a numerical example in which we use MSE to calculate the loss based on our predicted and target values.
In the following table, we present the predicted and actual average temperatures for each month of the year.
The error and squared error are calculated for each month.

Month	Actual	Predicted	Error	Squared Error
January	42	46	−4	16
February	51	48	3	9
March	53	55	−2	4
April	68	73	−5	25
May	74	77	−3	9
June	81	83	−2	4
July	88	87	1	1
August	85	85	0	0
September	79	75	4	16
October	67	70	−3	9
November	58	55	3	9
December	43	41	2	4

Types of loss functions

1. Mean Squared Error (MSE)

Now, let us calculate the sum of squared errors:

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$
$$= 16 + 9 + 4 + 25 + 9 + 4 + 1 + 0 + 16 + 9 + 9 + 4 = 106$$

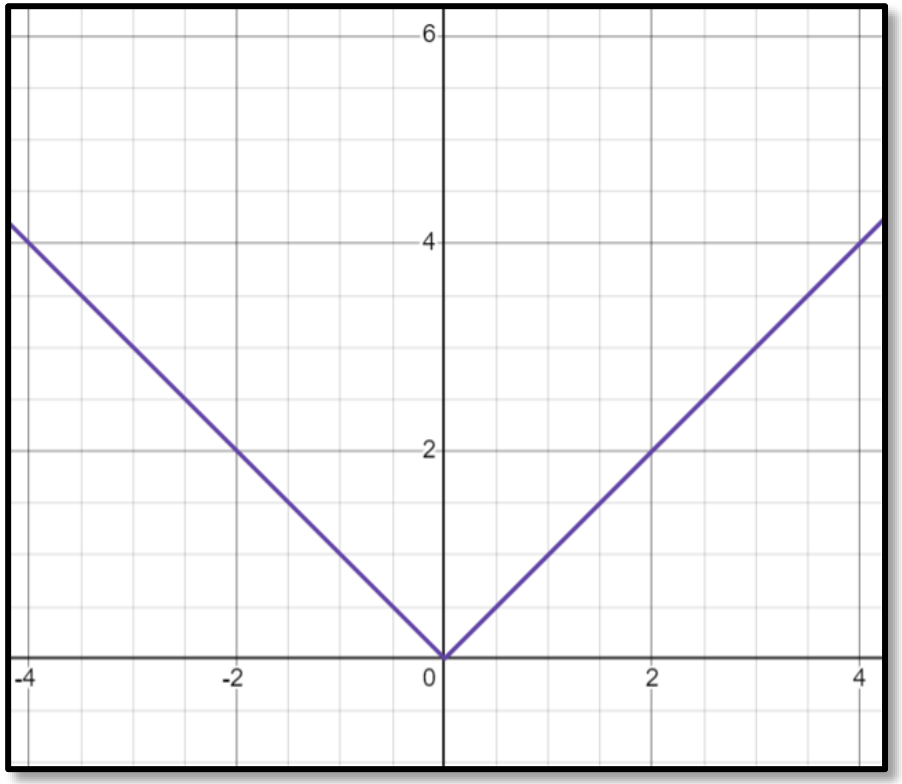
Now that we have the sum of differences, we can calculate the mean squared error.

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n} = \frac{106}{12} = 8.83$$

Types of loss functions

2. Mean Absolute Error (MAE)

MAE is another loss function commonly used in regression problems. MAE calculates the average of the absolute differences between the target and predicted values.
This loss function is sometimes employed as an alternative to MSE. As mentioned previously, MSE is highly sensitive to outliers, which can dramatically affect the loss. In cases where the data contains a large number of outliers, MAE is often preferred over MSE.



Types of loss functions

2. Mean Absolute Error (MAE)

Here is the mathematical equation for MAE:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$$

Where:

n: number of data points
y_i: target value of data point i
$\hat{y}_i$: predicted value of data point i

While MAE is sometimes used as an alternative to MSE, it does not receive much attention in deep learning due to a significant disadvantage. As the average distance approaches zero, gradient descent optimization ceases to function effectively. This is because the function's derivative at zero is undefined, which leads to a division by zero error.

Types of loss functions

3. Binary Cross-Entropy / Log Loss

Binary Cross-Entropy is a loss function used in binary classification models, where the model takes an input and classifies it into one of two pre-set categories.

In binary classification, there are only two possible actual values for y, 0 or 1.

To accurately determine the loss between the actual and predicted values, we need to compare the actual value with the probability that the input aligns with that category.

Now, let us delve deeper into the terminology.

Entropy: A calculation of the degree of randomness or disorder in a system. In other words, it measures the uncertainty of an event.

Cross-Entropy: A term commonly used in information theory, which measures the differences between two probability distributions that can be used to identify an observation.

It is important to note that Entropy is a significant topic in machine learning and could be explored further. However, in this lecture, we will adhere to the provided definition.

Types of loss functions

3. Binary Cross-Entropy / Log Loss

Here is the mathematical equation for Binary Cross-Entropy Loss, also known as Log Loss:

$$L(y, \hat{y}) = -(y.\log(\hat{y}) + (1-y).\log(1-\hat{y}))$$

Where:

y: true binary label
$\hat{y}$: predicted probability of the positive class

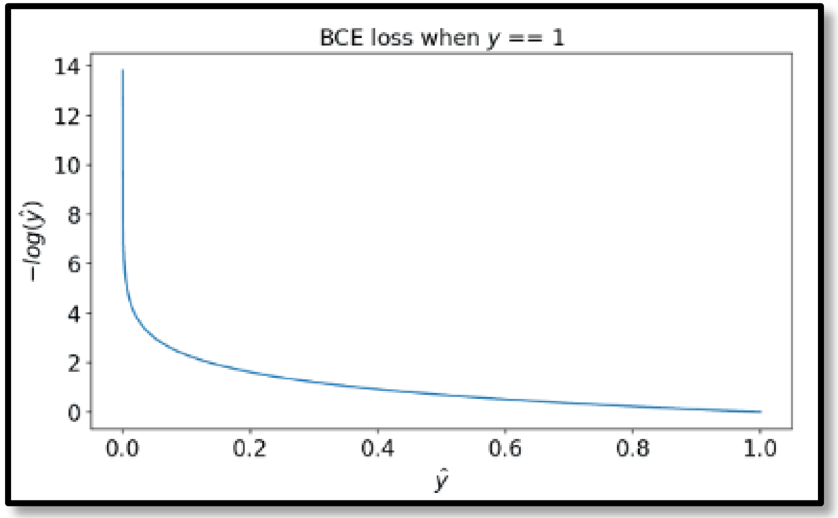
BCE is used in logistic regression problems and in training artificial neural networks designed to predict the likelihood of a data sample belonging to a class. These models leverage the Sigmoid activation function internally.
The goal is to minimize the BCE loss. Smaller values of BCE indicate a better and more accurate model, while larger values represent a less accurate one.

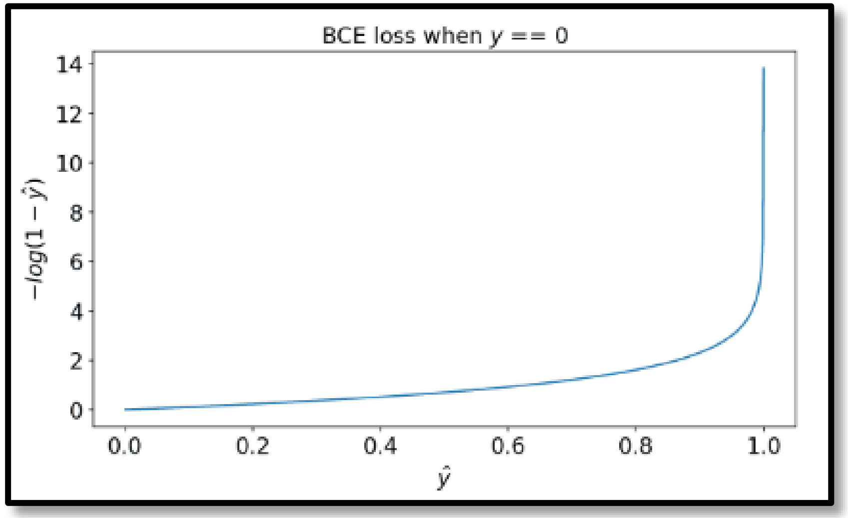
Types of loss functions

3. Binary Cross-Entropy / Log Loss

The Binary Cross-Entropy (BCE) loss function penalizes inaccurate predictions—those that significantly differ from the positive class. In other words, it quantifies the level of entropy associated with these predictions.

When BCE is utilized as a component within learning algorithms, it encourages the model to refine its predictions, which represent the probabilities for the appropriate class during training.





Types of loss functions

4. Categorical Cross-Entropy

Categorical Cross-Entropy loss, also known as SoftMax loss, is a widely used loss function for training models in multiclass classification problems, where the model takes an input and classifies it into one of multiple categories. In cases where the number of classes exceeds two, we utilize Categorical Cross-Entropy instead of Binary Cross-Entropy. However, this-

Loss function follows a very similar process to Binary Cross-Entropy in terms of terminology and functionality.

Types of loss functions

4. Categorical Cross-Entropy

Here is the mathematical equation for Categorical Cross-Entropy:

$$L = -\sum_{i=1}^{n} y_i \log(p_i)$$

Where:

n: number of classes
y_i: truth label
p_i: SoftMax probability for class i

In other words, to apply Cross-Entropy to a multiclass classification task, the loss for each class is calculated separately, and then summed to determine the total loss.

Types of loss functions

4. Categorical Cross-Entropy

Now, let us work on an example. Consider a 4-class classification task where an image is classified as either a dog, cat, horse or cheetah. Let us assume that the model's input is labeled as a dog; so the desired output is [1, 0, 0, 0]. We also assume that the model's output is [0.775, 0.116, 0.039, 0.070]. We want to calculate the Categorical Cross-Entropy. We are using the formula in the previous slide.

$$L = -\sum_{i=1}^{n} y_i \log(p_i)$$

$$= -[1\log(0.775) + 0\log(0.116) + 0\log(0.039) + 0\log(0.070)]$$

$$= -\log(0.775) = 0.3677$$

Types of loss functions

5. Hinge Loss

Hinge loss is a function used in machine learning to train classifiers aimed at increasing the margin between data points and the decision boundary. Therefore, it is primarily employed for maximum margin classifications. This loss function penalizes predictions from the model that are wrongly classified, which fall on the wrong side of the margin boundary. It also penalizes predictions that are correctly classified but are in close proximity to the decision boundary.

Using hinge loss can enhance the model's generalization capabilities, making it effective for accurately classifying data points with a high degree of certainty.

Types of loss functions

5. Hinge Loss

Here is the mathematical equation for Hinge Loss:

$$L(y,\hat{y}) = \max(0, 1 - y.\hat{y})$$

Where:

y: true label or target value (-1 or 1)
$\hat{y}$: predicted value or decision function output

Hinge loss is used in binary classification problems where the objective is to separate the data points in two classes, typically labeled as -1 and 1.