# task_8.2

# Machine Learning (WiSe 2025/2026)

Author: Suvansh Shukla
Matriculation No. 256245

## Assignment 8 Task 2

The function of an activation function of an output node is to give the final result of the processed input. The purpose of the loss function is to determine the difference between the predicted output processed by the node and the actual value of the output. The output of the activation function is what is used as the output term in the loss function. An example of this would be a Mean Square Error function.

Mean Square Error:

$$MSE = \frac{1}{n}\sum_{d}(t_d - o_d)^2$$

Here you can see that the $o_d$ that is the output from the node is used as
in the calculation of MSE.

## Linear Activation Function

The linear activation function is a simple straight line function which is directly proportional to the input, i.e. the weighted sum of neurons. It can be used for regression. Where we try to draw a line over scattered points in a 2D space. Here the error function would be euclidean distance, i.e. the distance between a point on the line and the actual point in space.

Euclidean distance:

$$Err = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

## Sigmoid Function

The sigmoid function enables smoother decision boundaries and also probabilistic outputs. The sigmoid function can map inputs to a continuous range of values. These range between 0 and 1. This is particularly useful for classification tasks where outputs represent probabilities.

Sigmoid Function:

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

The loss function used when our output layer has a sigmoid function is Binary Cross-Entropy Loss.
This is also known as Log loss. The BCE helps measure the dissimilarity between the predicted
probability and the true binary label of the instance.

Binary Cross-Entropy Loss Function:

$$L(y, \hat{y}) = -[ylog(\hat{y}) + (1-y)log(1-\hat{y})]$$

Where, $y$ is the true label and $\hat{y}$ is the predicted probability from the sigmoid function.

This combination of sigmoid function and BCE is particularly effective because:

- the derivative of the sigmoid and BSE functions results in a simpler and stronger
signal for backpropagation
- if the sigmoid and MSE loss function was used, the resulting derivative would be close to zero,
thus leading to the vanishing gradient problem meaning the neural network would learn too slowly.
- the BSE is derived from the principle of Maximum Likelihood Estimation, assuming the output follows
a Bernoulli probability distribution, which is the better choice mathematically

## Softmax Function

The Softmax activation function, is designed for multi-class classification and converts the raw output scores into a vector of probabilities that sum to 1.
The class with the highest probability is the network's prediction.

Softmax Function:

$$\hat{y}_k = \frac{e^{z_k}}{\sum_{j=1}^{C} e^{z_j}}$$

The choice of loss function for the Softmax activation function is the Categorical Cross-Entropy
Loss function. This CCE function measures the dissimilarity between this predicted probability
distribution from the Softmax function and the true probability distribution of the ground truth.

Categorical Cross-Entropy Loss Function:

$$L(y, \hat{y}) = -\sum_{i=1}^{C} y_i log(\hat{y}_i)$$

Where,

- C is the total number of classes
- $y_i$ is the true label for class $i$
- $\hat{y}_i$ is the predicted probability for class $i$ from the Softmax function

The combination of Softmax and CCE function is also good due to it's mathematical stability and efficiency.