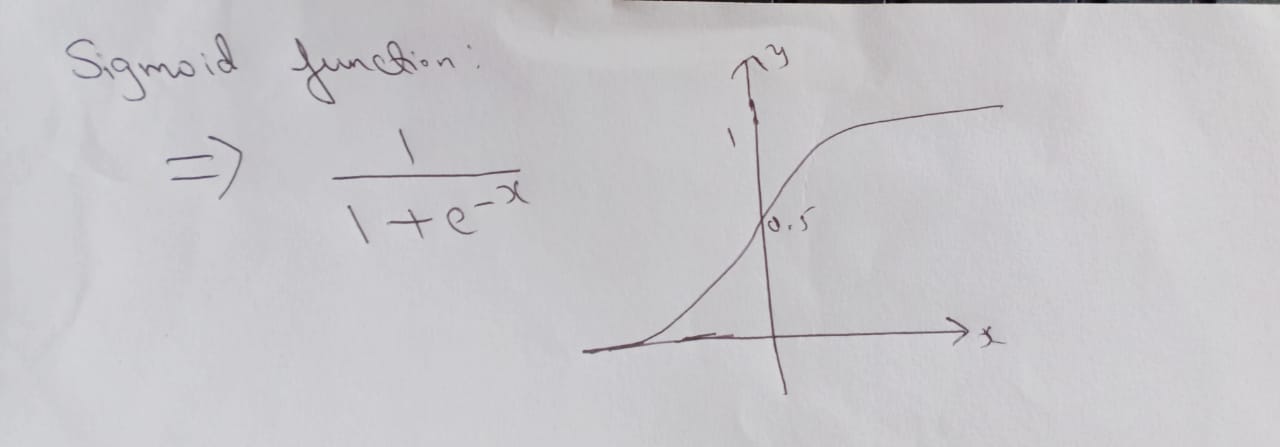
**Activation functions**

1. **Sigmoid function**

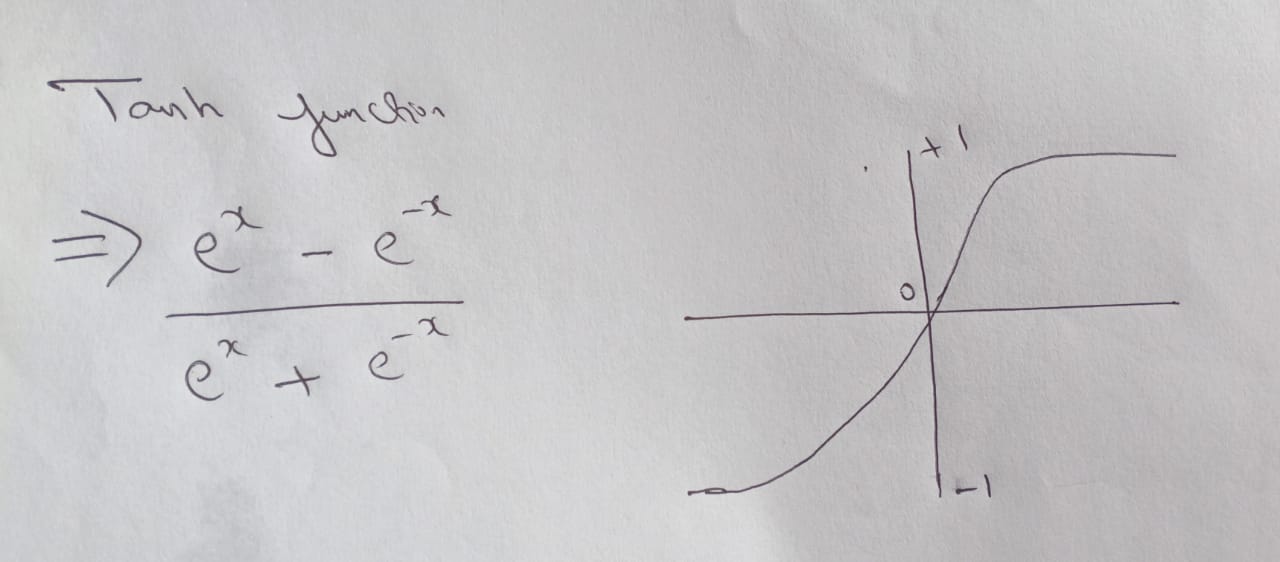


Used in Binary classification. Output is either 0 or 1.

Disadvantages:-

* + As it is exponential based operation it takes more time.
  + After a certain time, value is same (Vanishing Gradient problem).
  + Function outputs are non-zero centric.

1. **Tanh function**



* + Mostly used in binary classification.
  + Function output are zero centric.
  + Helps to normalize values.
  + Output ranges between -1 to +1.

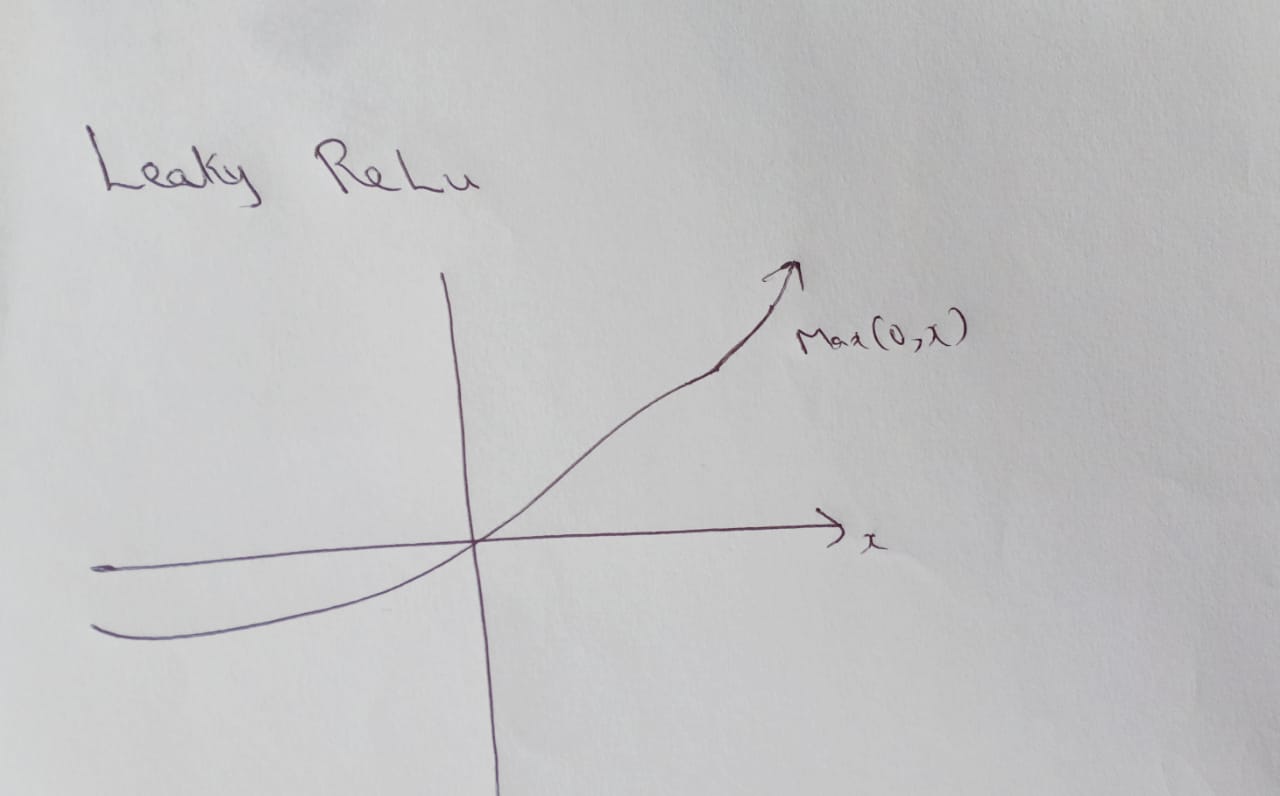
1. **ReLu function(Rectified Linear Unit)**
   * + Max(0,x)

No gradient saturation for +ve number.

Calculation is faster than Tanh and Sigmoid.

Disadvantages:-

1. For negative values the output is zero.
2. Not a zero centric function.
3. **Leaky ReLu**

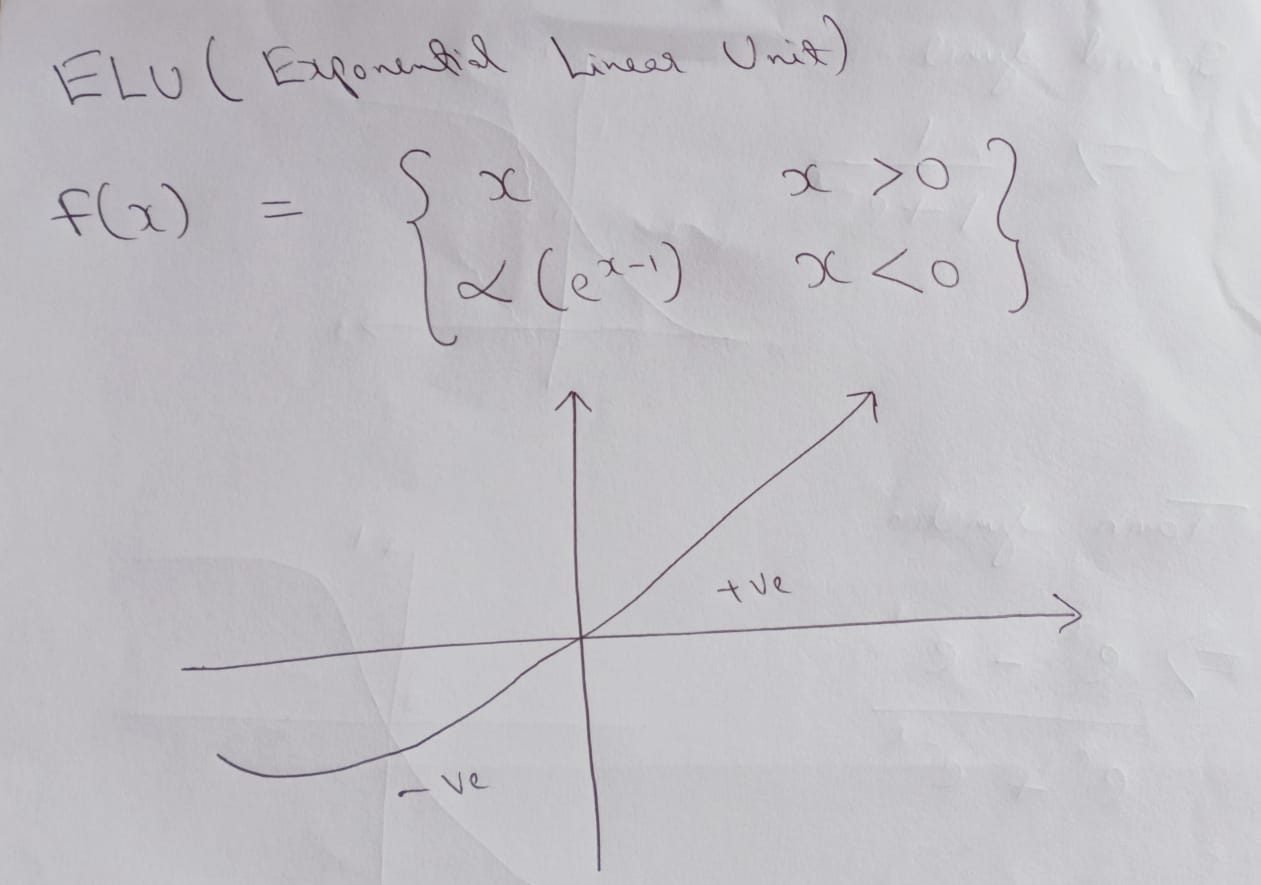
****

**Advantages:**

1. Simple Calculation.
2. –ve axis is adjusting to the gradient.
3. Gradient clipping will disappear

**Disadvantages:**

1. For –ve number we get a very less value like 0.01, so it is not much helpful
2. **ELU(Exponential Linear Unit)**

****

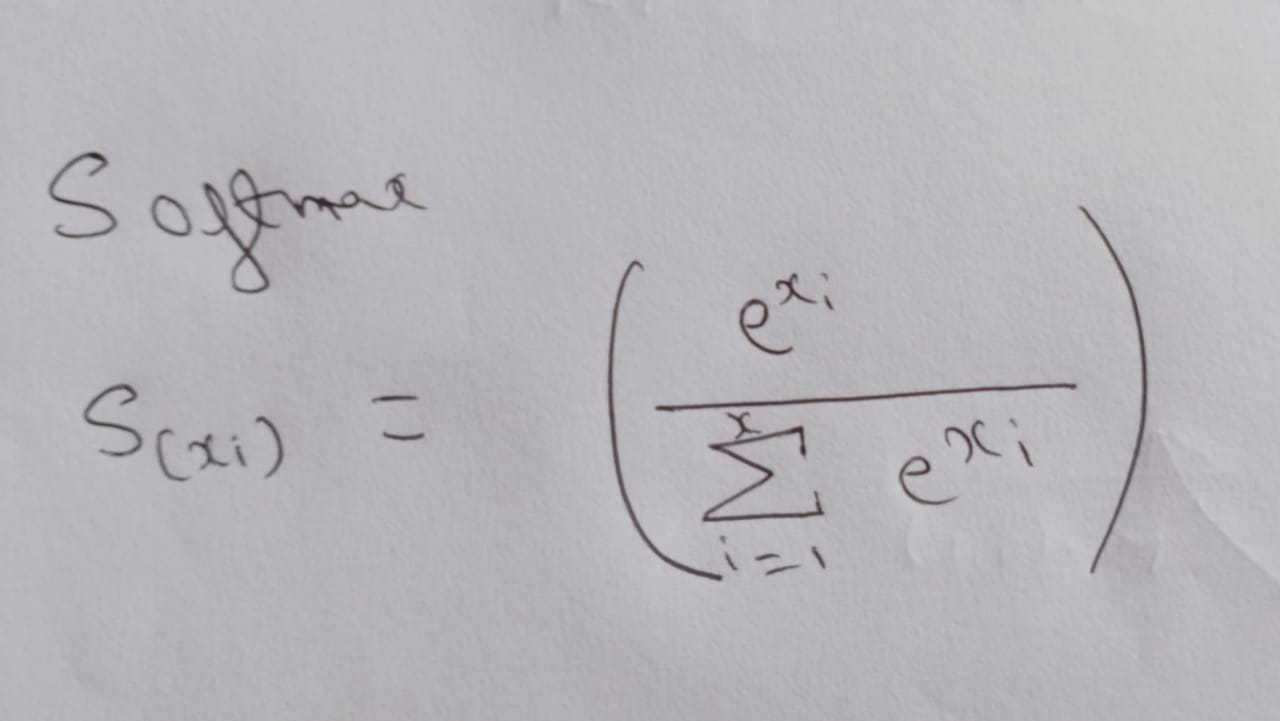
**Advantages:**

1. No gradient clipping.
2. The mean of the output is close to zero.
3. It is Zero Centric.

**Disadvantages:**

1. For –ve dataset it is little compute extensive.
2. **Softmax**

Gives the probability of occurrence of numbers.



Mostly used in multi class classification.

1. **PRelu(Parametric ReLu)**

f(x) = { x x>0,

αx x<= 0}

Value will never zero.

α is learnable parameter.

1. **Swiss function(Self gated function)**

Y = X \* sigmoid(X)

Advantages:

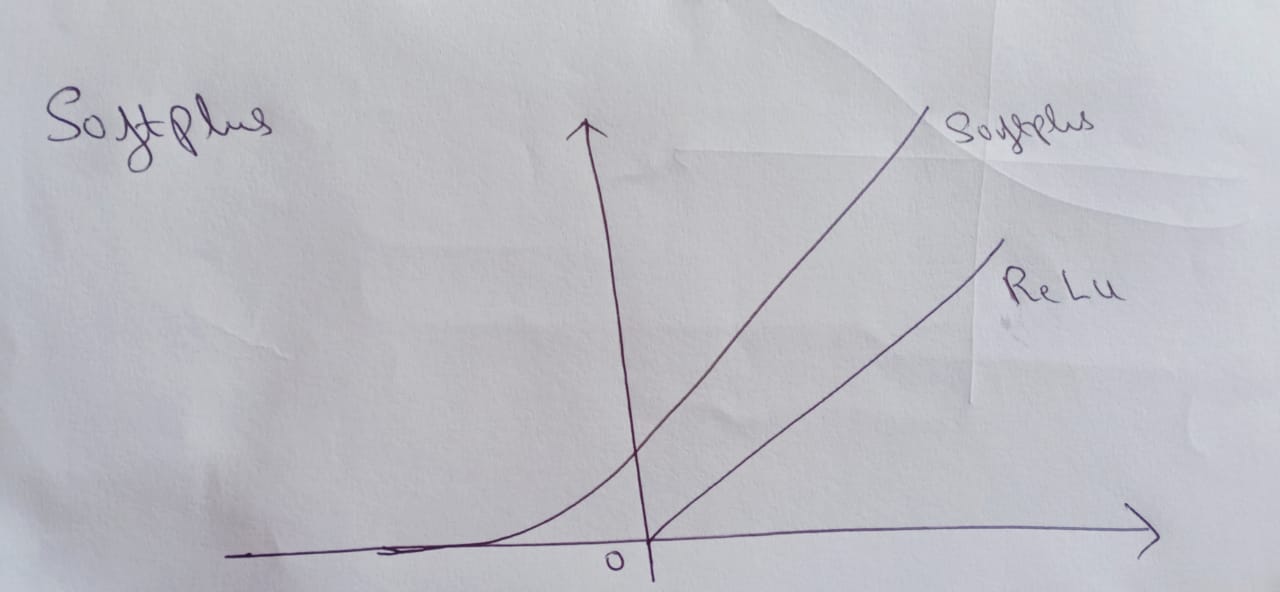
1. No Vanishing gradient problem
2. Always a smooth curve
3. Used in LSTM network
4. Always zero centric
5. **Maxout Function**

Max(w1x1 + b1 , w2x2 + b2)

It is combination of ReLu and Leaky ReLu.

It depends on weights and bias.

1. **Softplus Function**



f(x) = log(1+exp(x))

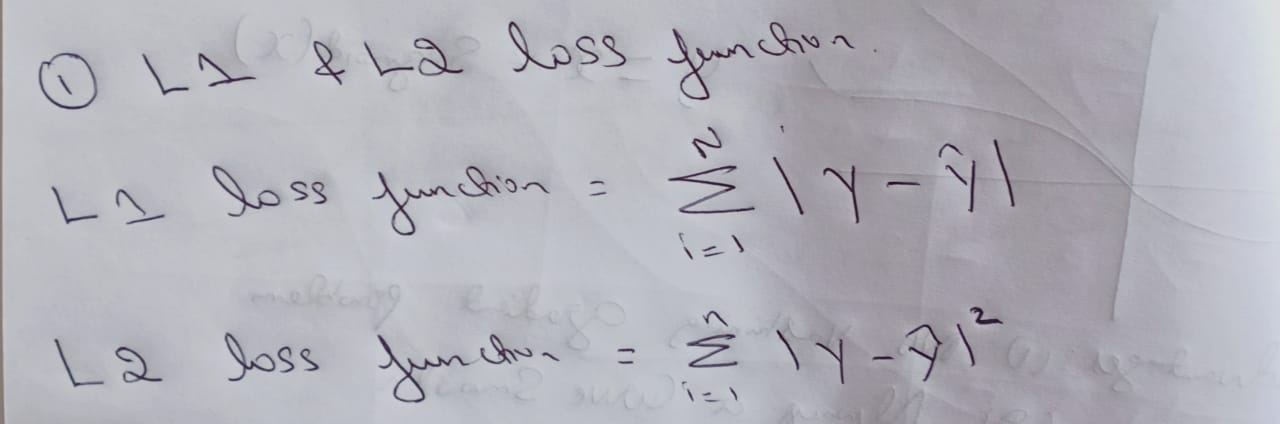
It is a smooth curve.

In backpropagation, activation function helps in finding derivate/gradient, not to decide output data.

In forward propagation, it tries to dominate given output or parallel dataset.

**Loss Functions**

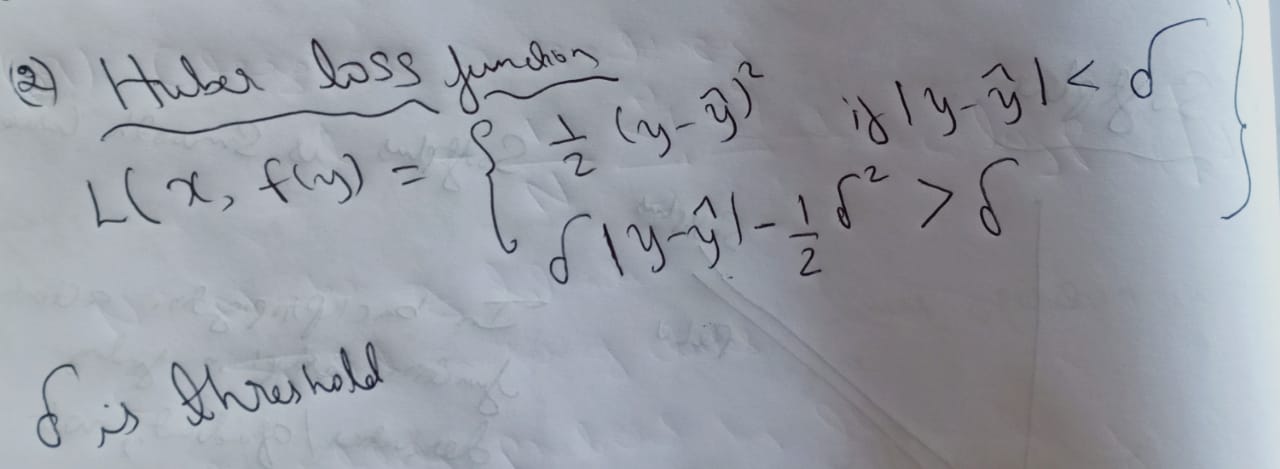
1. **L1 and L2 loss function**



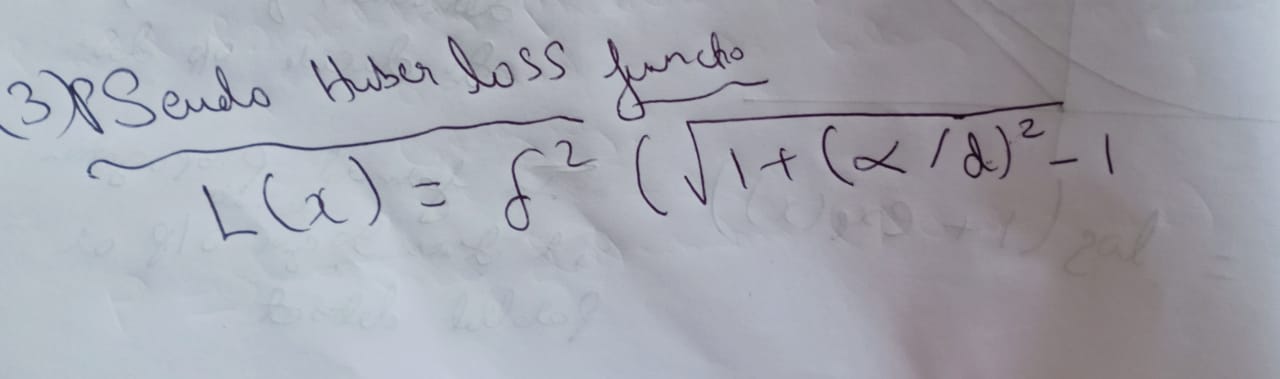
L1 is least absolute deviation

L2 is least squared error – Usually a smooth curve, output value is high, and minimize the error which is summation of all the absolute values and predicted values.

1. **Huber loss function**



1. **Pseudo Huber loss function**



1. **Hinge loss function**

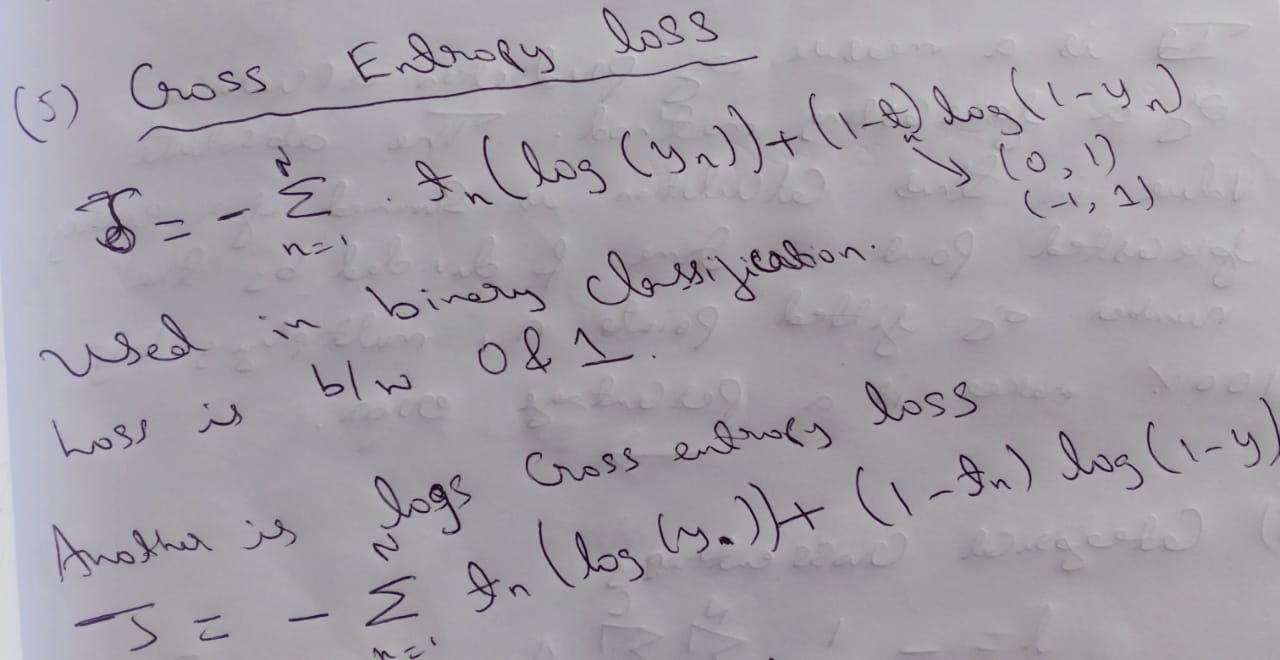
L(y) = max(0, 1 – t.y)

t is number of classes.

Mostly used inside a classifier, support vector machine.

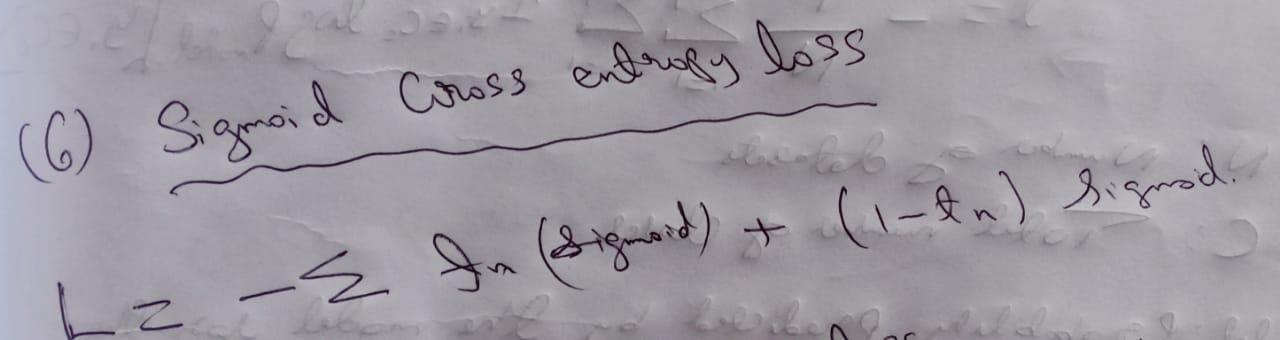
Not recommended for regression.

1. **Cross Entropy loss function**

****

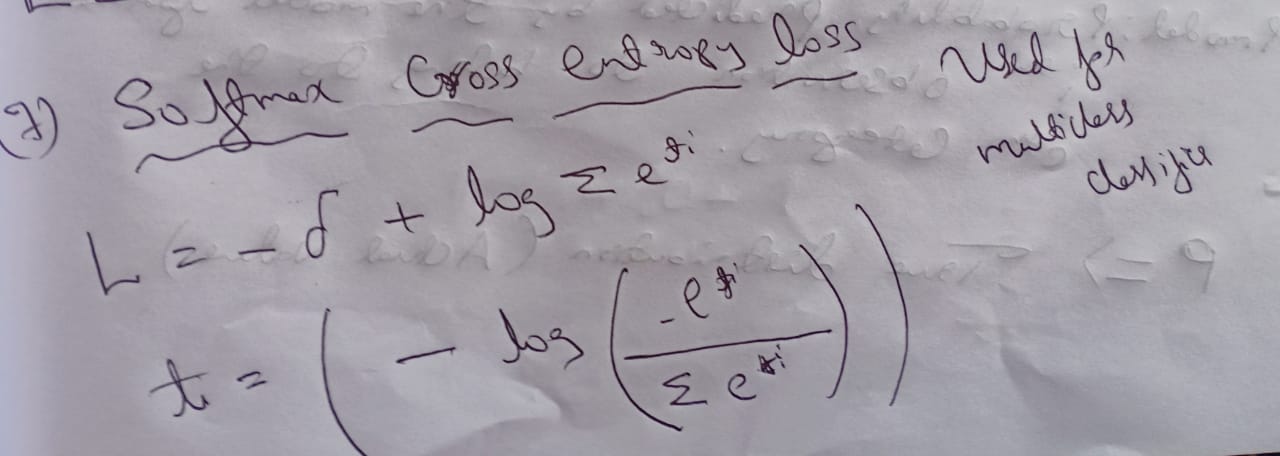
Used in binary classification, loss is between 0 and 1.

1. **Sigmoid Cross Entropy loss function**



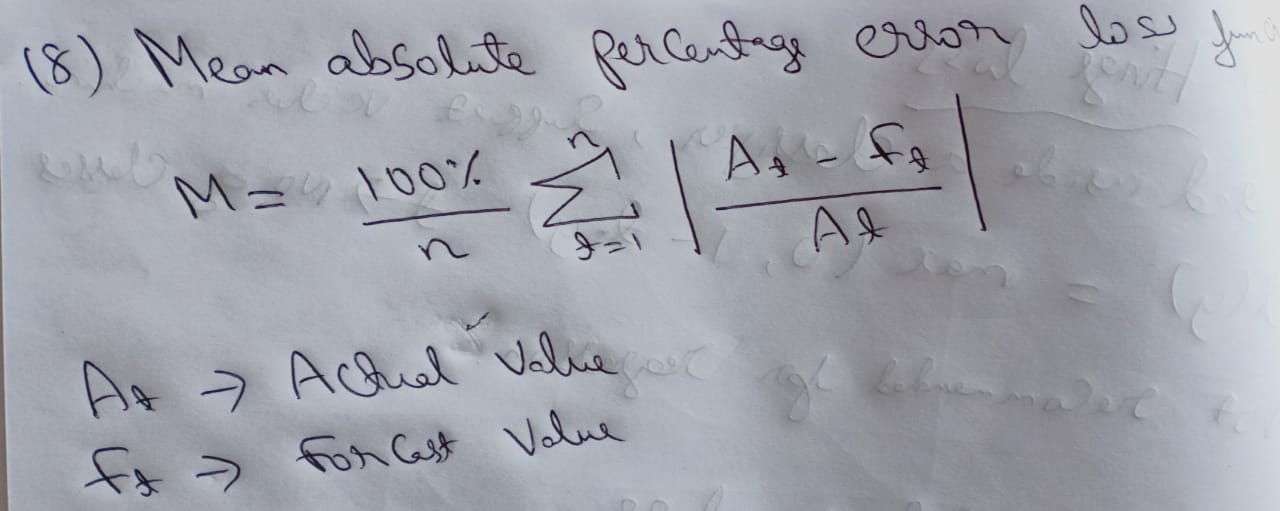
1. **Softmax Cross Entropy loss function**

Used for multiclass classifier

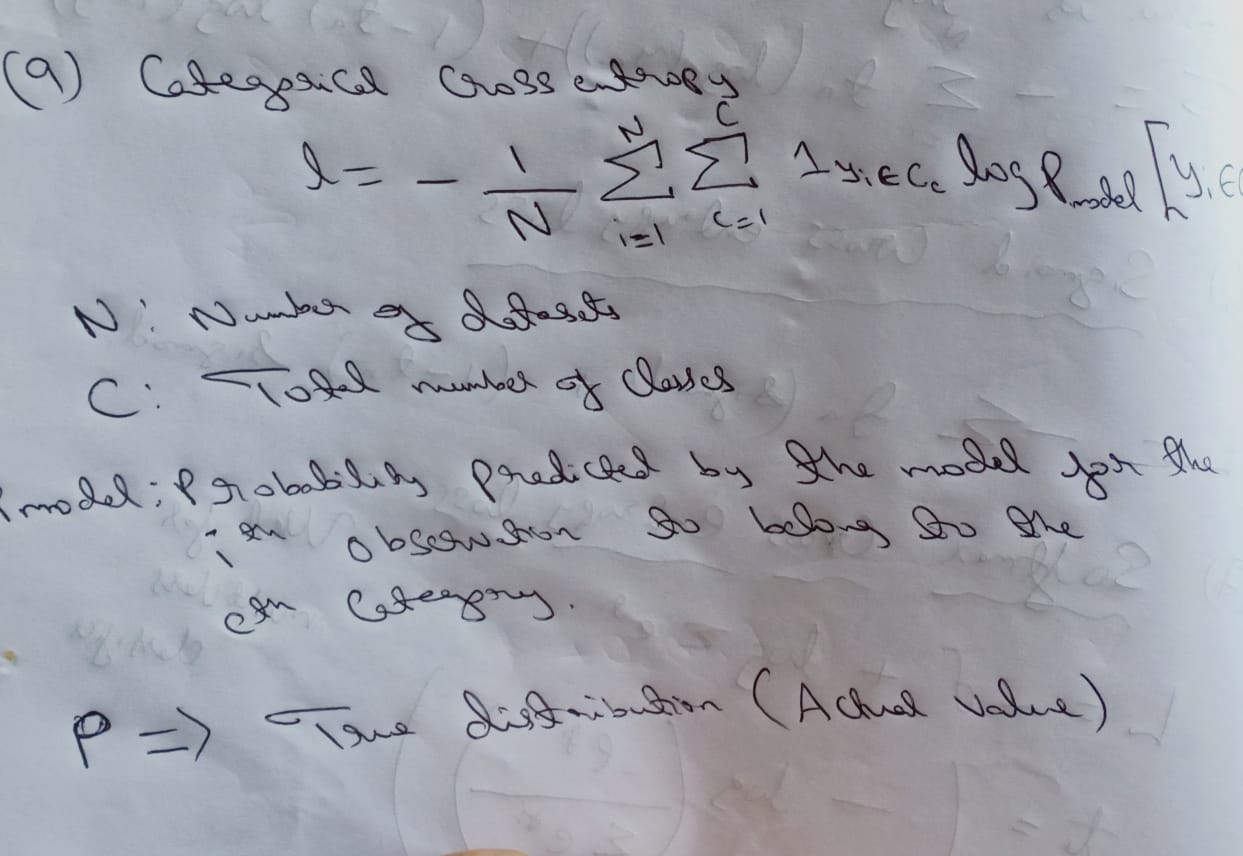


1. **Mean Absolute Percentage Error Loss Function**

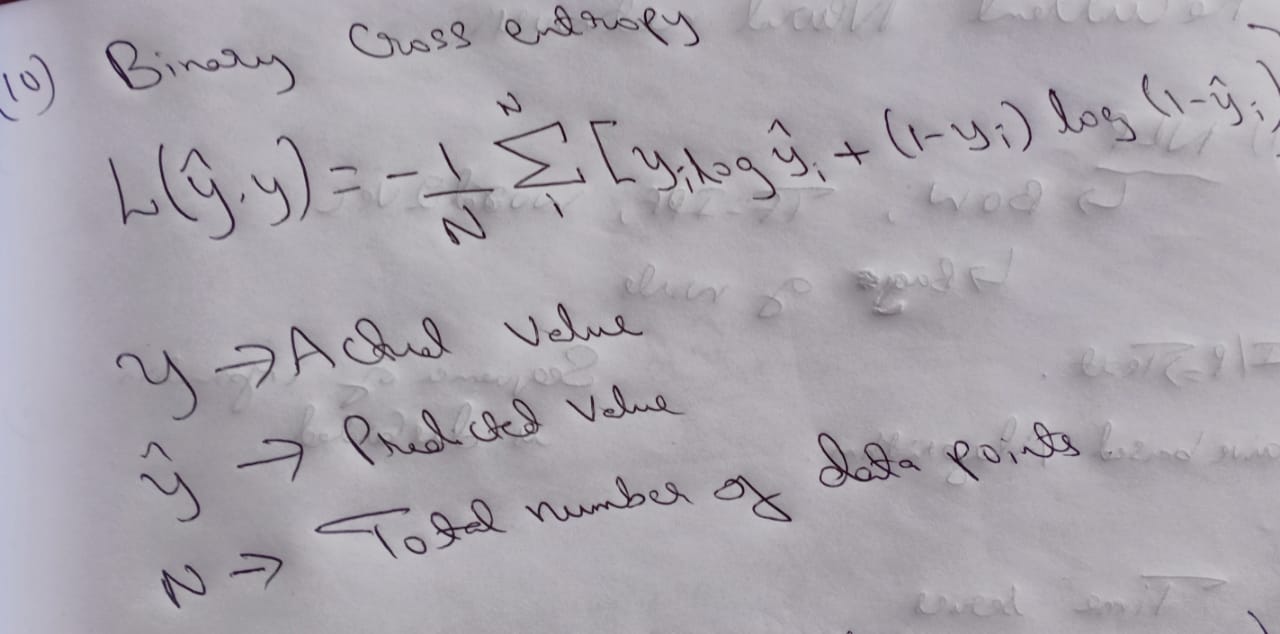
It is a measure of prediction accuracy of a forecasting method. The absolute value in this calculation is summed for every forecasted point in time and divided by the number of fitted points n, multiplying by 100% makes it a percentage error.



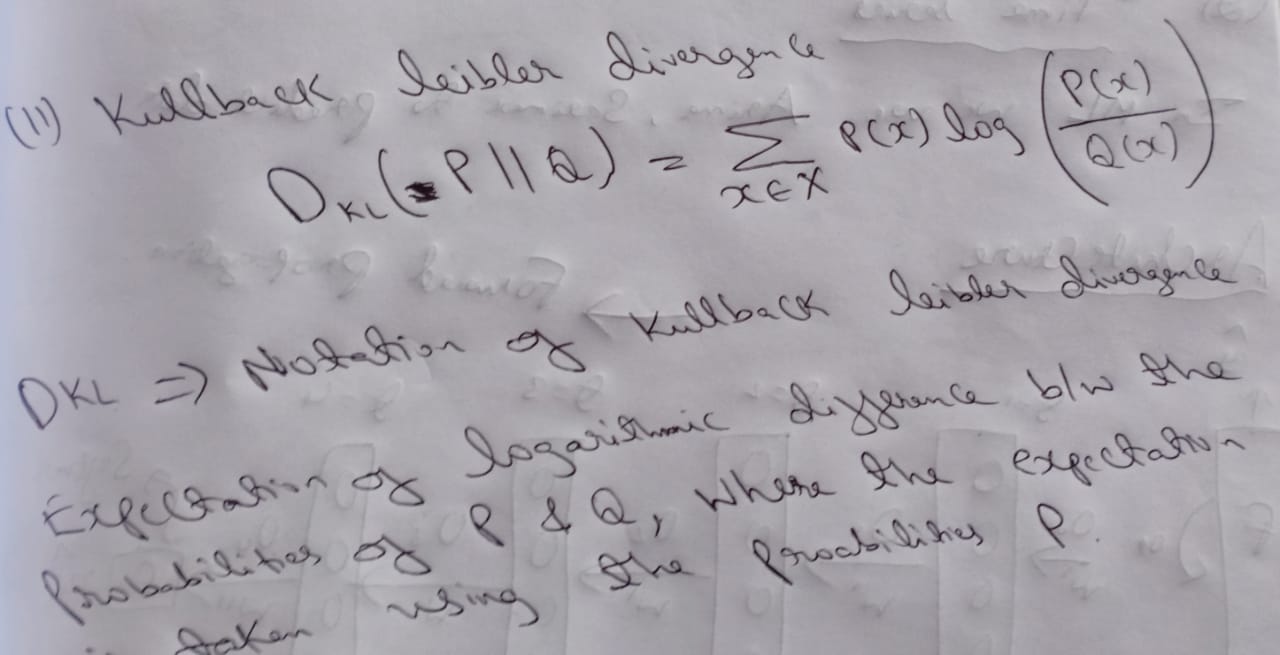
1. **Categorical Cross Entropy Loss Function**

****

1. **Binary Cross Entropy Loss Function**

****

1. **Kullback Leibler Divergence**

****