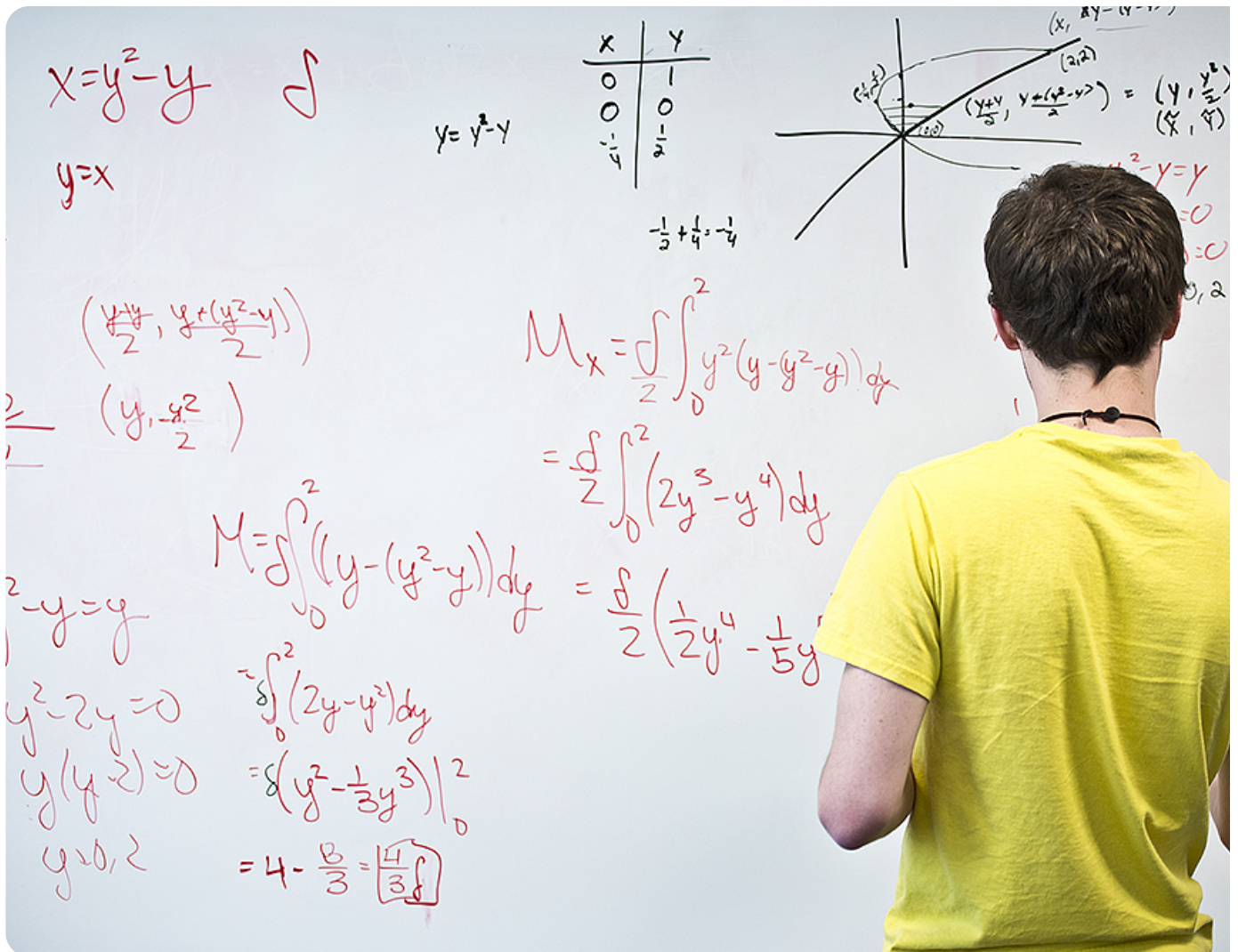


# A Beginners' Guide to Cross-Entropy in Machine Learning

Machine learning and deep learning models are normally used to solve regression and classification problems. In a supervised learning problem, during the training process, the model learns how to map the input to the realistic probability output.

by Vijaysinh Lendave



Join Us

Follow Us



Today we have many real-world applications which are based on machine learning such as [churn modeling](#), [image classification](#), [customer segmentation](#), etc. For all of these kinds of applications, businesses need to optimize their models, obtain the model's optimum accuracy and efficiency model. Therefore it is a bit critical to obtain a higher-performing model by tuning a certain

number of parameters. One such parameter is a [loss function](#) and among which mostly used one is cross-entropy. In this article, we will be discussing cross-entropy functions and their importance in machine learning, especially in classification problems. The important concepts that we will discuss here in this article are listed below.

## Table of Contents

1. Need of Cross-Entropy
2. What is Entropy?
3. What is cross-entropy
4. Cross entropy as loss function
5. Code Glimpses

Let's begin our discussion one by one.

## Need of Cross-Entropy

Machine learning and deep learning models are normally used to solve regression and classification problems. In a supervised learning problem, during the training process, the model learns how to map the input to the realistic probability output. As we already know, the model adjusts its parameters incrementally during the training phase of supervised learning so that prediction gets closer to closer as expected values (ground truth). Let's take an example of three images, which represents three class of vehicles as shown below and each image is encoded in the binary representation below;

		
Car	Excavator	Tank
Class	Labels	
Car	[1,0,0]	
Excavator	[0,1,0]	
Tank	[0,0,1]	

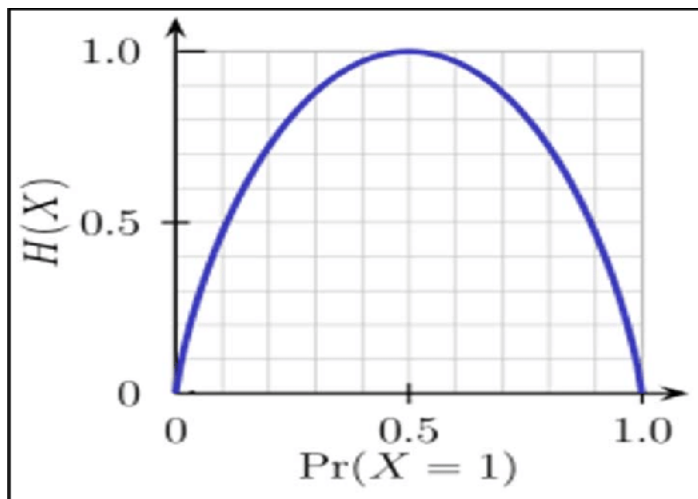
The above is the actual representation of our training data which we fed to our model as input and output class. When we give an image of the excavator to our model, the model tries to generalize parameters for it and return a probability distribution for all the three classes like [0.36,0.65,0.45] which is completely different from what we actually want. The model should report a higher probability for the ground truth. Well in our case it has been reported not much higher as we can see in the rest of the two there is close ambiguity between the excavator and tank.

Well in the next iterative process the model tries to improve its prediction by changing the output from  $y'$  to  $y$ . This task is undertaken by the various loss functions. It depends on which problem we are dealing with. For multi-class classification, the cross-entropy loss function is used, which basically tells our model in which direction the prediction is closer to the ground truth.

## What is Entropy?

*Claude Shannon*, a mathematician, and electrical engineer was trying to figure out how to deliver a communication message without losing a piece of information. He was thinking in terms of average message length, which meant he was attempting to encode a message with the fewest possible bits. Aside from that, he expected that the decoder would be able to reconstruct

the message losslessly, meaning that no data would be lost in this way he invented the idea of entropy.



*Entropy on Decision Tree*

Here entropy is defined as the smallest average encoding size per transmission with which a source can send a message to a destination efficiently and without losing any data. Entropy can be defined mathematically using the probability distribution denoted as H. when we are talking about the categorical variables the formula looks like this

$$H(X) = - \sum_{i=1}^n P(x_i) \log P(x_i)$$

*Source*

## What is Cross-Entropy?

Assume we have two distributions of data and need to be compared. Cross entropy employs the concept of entropy which we have seen above. Cross entropy is a measure of the entropy difference between two probability distributions. Assume the first probability distribution is denoted by A and the second probability distribution is denoted by B.

The average number of bits required to send a message from distribution A to distribution B is referred to as cross-entropy. Cross entropy is a concept used in machine learning when algorithms are created to predict from the model. The construction of the model is based on a comparison of actual and expected results.

Mathematically we can represent cross-entropy as below:

$$H(p, q) = - \sum_{x \in \mathcal{X}} p(x) \log q(x)$$

*Source*

In the above equation, x is the total number of values and p(x) is the probability of distribution in the real world. In the projected distribution B, A is the probability distribution and q(x) is the probability of distribution. So working with two distributions how do we link cross-entropy to entropy? If the expected and actual values are the same then cross-entropy equals entropy.

In the real world, however, the predicted value differs from the actual value which is referred to as divergence, because they differ or diverge from the

actual value. As a result, cross-entropy is the sum of Entropy and KL divergence (type of divergence).

## Cross-Entropy as Loss Function

When optimizing classification models, cross-entropy is commonly employed as a loss function. The logistic regression technique and artificial neural network can be utilized for classification problems.

In classification, each case has a known class label with a probability of 1.0 while all other labels have a probability of 0.0. Here model calculates the likelihood that a given example belongs to each class label. The difference between two probability distributions can then be calculated using cross-entropy.

In classification, the goal of probability distribution  $P$  for an input of class labels 0 and 1 is interpreted as probability as *Impossible* or *Certain*. Because this probability includes no surprises (low probability event) they have no information content and have zero entropy.

When we are dealing with Two Class probability, the probability is modelled as Bernoulli distribution for the positive class. This means that the mode explicitly predicts the probability for class 1, while the probability for class 0 is given as  $1 - \text{projected probability}$ . For more clearly say;

- Class 1 = 1 (originally predicted)
- Class 0 = 1 – originally predicted

We are frequently concerned with lowering the model's cross-entropy throughout the entire training dataset. This can be done by taking the average cross-entropy of all training sets.

## Code Glimpses

Below we have used the python function to calculate the cross-entropy based on the above formula, later we checked whether the calculated cross-entropy is equal to the actual one or not by the logical function.

```
def cross_entropy(predicted, Ground_Truth, e=1e-12):
    """
    Computes CROSS_ENTROPY between Ground_Truth (encoded as
    one-hot vectors)
    and predicted value.
    Input: predicted value (N, k) ndarray
           Ground_Truth (N, k) ndarray
    Returns: fraction
    """

    predicted = np.clip(predicted, e, 1. - e)
    N = predicted.shape[0]
    CROSS_ENTROPY = -np.sum(Ground_Truth*np.log(predicted+1e-
9))/N
    N = predicted.shape[0]
    CROSS_ENTROPY = -np.sum(Ground_Truth*np.log(predicted+1e-
9))/N
    return CROSS_ENTROPY

predicted = np.array([[0.25,0.25,0.25,0.25],
                      [0.01,0.01,0.01,0.96]])
Ground_Truth = np.array([[0,0,0,1],
                          [0,0,0,1]])
ans = 0.71355817782 #Correct answer
x = cross_entropy(predicted, Ground_Truth)
```

```
print(np.isclose(x,ans)) # prints where calculated entropy is  
equal to the ans
```

In this case, the code returns *True* which means the calculated and actual values are equal to each other.

## Conclusion

In this article, we have seen the intuition theory of cross-entropy. When we are dealing with multiple classes and want our model to converge faster by countering the loss, cross-entropy plays a crucial role. All the major frameworks support this loss function, in Keras for binary classification we have the `Bnary_crossentropy` function and for multi-class, we have the `Categorical_crossentropy` function.

## References

- [Cross-entropy](#)
- <https://machinelearningmastery.com/cross-entropy-for-machine-learning/>

---

[Access all our open Survey & Awards Nomination forms in one place >>](#)

---



Vijaysinh Lendave

Vijaysinh is an enthusiast in machine learning and deep learning. He is skilled in ML algorithms, data manipulation, handling and visualization, model building.