Cost Function is No Rocket Science!

Become a GenAI Professional: 10+ Projects, 26+ Tools ,
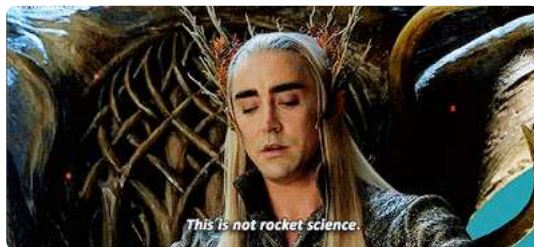
Analytics Vidhya

🔍   **V**

Home › Advanced › Cost Function is No Rocket Science!

[Saily Shah](#)
17 Nov, 2023 • 7 min read

This article was published as a part of the [Data Science Blogathon](#).

Credit: https://gifer.com/en/GxIE

The 2 main questions that popped up in my mind while working on this article were "Why am I writing this article?" & "How is my article different from other articles?" Well, the cost function is an important concept to understand in the fields of data science but while pursuing my post-graduation, I realized that the resources available online are too general and didn't address my needs completely.

I had to refer to many articles & see some videos on YouTube to get an intuition behind cost functions. As a result, I wanted to put together the "What," "When," "How," and "Why" of Cost functions that can help to explain this topic more clearly. I hope that my article acts as a one-stop-shop for cost functions!

Dummies guide to the Cost function 💁‍♀️

What is a loss/Cost function? 'Loss' in Machine learning

Cost Function is No Rocket Science!

in those supervised learning algorithms that use optimization techniques. Notable examples of such algorithms are regression, [logistic regression](#), etc. The terms cost function & loss function are analogous.

Loss function: Used when we refer to the error for a single training example.
Cost function: Used to refer to an average of the loss functions over an entire training dataset.

But, like, *why* use a cost function?

Why on earth do we need a cost function? Consider a scenario where we wish to classify data. Suppose we have the height & weight details of some cats & dogs. Let us use these 2 features to classify them correctly. If we plot these records, we get the following scatterplot:
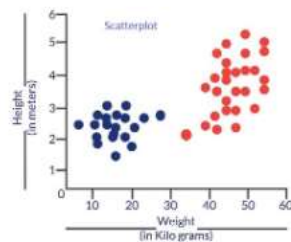

Fig 1: Scatter plot for height & weight of various dogs & cats

Blue dots are cats & red dots are dogs. Following are some solutions to the above classification problem.
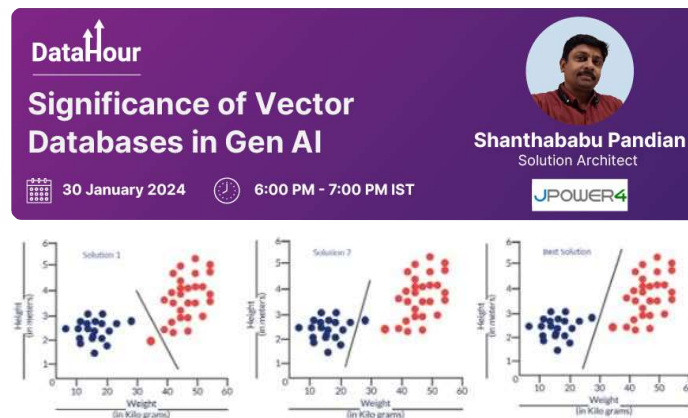

Fig: Probable solutions to our classification problem

Essentially all three classifiers have very high accuracy

Cost Function is No Rocket Science!

between the two groups, and not closer to any one of the groups. This is where the concept of cost function comes in. Cost function helps us reach the optimal solution. The cost function is the technique of evaluating "the performance of our algorithm/model".

It takes both predicted outputs by the model and actual outputs and calculates how much wrong the model was in its prediction. It outputs a higher number if our predictions differ a lot from the actual values. As we tune our model to improve the predictions, the cost function acts as an indicator of how the model has improved. This is essentially an optimization problem. The optimization strategies always aim at "minimizing the cost function".

Types of the cost function

There are many cost functions in machine learning and each has its use cases depending on whether it is a regression problem or classification problem.

1. Regression cost Function

2. Binary Classification cost Functions

3. Multi-class Classification cost Functions

1. Regression cost Function:

Regression models deal with predicting a continuous value for example salary of an employee, price of a car, loan prediction, etc. A cost function used in the regression problem is called "Regression Cost Function". They are calculated on the distance-based error as follows:

Error = y-y'

Where,

Y – Actual Input

Cost Function is No Rocket Science!

... Mean Error (ME)

- In this cost function, the error for each training data is calculated and then the mean value of all these errors is derived.

- Calculating the mean of the errors is the simplest and most intuitive way possible.

- The errors can be both negative and positive. So they can cancel each other out during summation giving zero mean error for the model.

- Thus this is not a recommended cost function but it does lay the foundation for other cost functions of regression models.

1.2 Mean Squared Error (MSE)

- This improves the drawback we encountered in Mean Error above. Here a square of the difference between the actual and predicted value is calculated to avoid any possibility of negative error.

- It is measured as the average of the sum of squared differences between predictions and actual observations.

$$MSE = \frac{\sum_{i=0}^{n}(y-y')^2}{n}$$

MSE = (sum of squared errors)/n

- It is also known as L2 loss.

- In MSE, since each error is squared, it helps to penalize even small deviations in prediction when compared to MAE. But if our dataset has outliers that contribute to larger prediction errors, then squaring this error further will magnify the error many times more and also lead to higher MSE error.

- Hence we can say that it is less robust to outliers

1.3 Mean Absolute Error (MAE)

Cost Function is No Rocket Science!

between the actual and predicted value is calculated to avoid any possibility of negative error.

- So in this cost function, MAE is measured as the average of the sum of absolute differences between predictions and actual observations.

$$MAE = \frac{\sum_{i=0}^{n} |y - y\prime|}{n}$$

MAE = (sum of absolute errors)/n

- It is also known as L1 Loss.

- It is robust to outliers thus it will give better results even when our dataset has noise or outliers.

2. Cost functions for Classification problems

Cost functions used in classification problems are different than what we use in the regression problem. A commonly used loss function for classification is the cross-entropy loss. Let us understand cross-entropy with a small example. Consider that we have a classification problem of 3 classes as follows.

Class(Orange,Apple,Tomato)

The machine learning model will give a probability distribution of these 3 classes as output for a given input data. The class with the highest probability is considered as a winner class for prediction.

Output = [P(Orange),P(Apple),P(Tomato)]

The actual probability distribution for each class is shown below.

Orange = [1,0,0]

Apple = [0,1,0]

Tomato = [0,0,1]

Cost Function is No Rocket Science!

actual probability distribution of Tomato. If the predicted probability distribution is not closer to the actual one, the model has to adjust its weight. This is where cross-entropy becomes a tool to calculate how much far the predicted probability distribution from the actual one is. In other words, Cross-entropy can be considered as a way to measure the distance between two probability distributions. The following image illustrates the intuition behind cross-entropy:
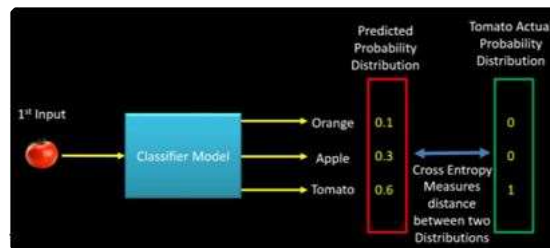

FIg 3: Intuition behind croos-entropy (credit – machinelearningknowledge.ai)

This was just an intuition behind cross-entropy. It has its origin in information theory. Now with this understanding of cross-entropy, let us now see the classification cost functions.

2.1 Multi-class Classification cost Functions

This cost function is used in the classification problems where there are multiple classes and input data belongs to only one class. Let us now understand how cross-entropy is calculated. Let us assume that the model gives the probability distribution as below for 'n' classes & for a particular input data D.

$$P\ (D) = \begin{bmatrix} p1 \\ p2 \\ pn \end{bmatrix}$$

And the actual or target probability distribution of the data D is

$$Y\ (D) = \begin{bmatrix} y1 \\ y2 \\ vn \end{bmatrix}$$

Cost Function is No Rocket Science!

Cross-entropy loss$(y,p) = -y^T \log(p)$

$= -(y_1 \log(p_1) + y_2 \log(p_2) + \ldots\ldots y_n \log(p_n))$

$$\therefore \text{Cross-entropy loss}(y,p) = -[y1 \ y2 \ yn] \begin{bmatrix} \log(p1) \\ \log(p2) \\ \log(pn) \end{bmatrix}$$

Let us now define the cost function using the above example (Refer cross entropy image -Fig3),

p(Tomato) = [0.1, 0.3, 0.6]

y(Tomato) = [0, 0, 1]

Cross-Entropy$(y,P) = -(0*\text{Log}(0.1) + 0*\text{Log}(0.3)+1*\text{Log}(0.6)) = 0.51$

The above formula just measures the cross-entropy for a single observation or input data. The error in classification for the complete model is given by categorical cross-entropy which is nothing but the mean of cross-entropy for all N training data.

Categorical Cross-Entropy = (Sum of Cross-Entropy for N data)/N

2.2 Binary Cross Entropy Cost Function

Binary cross-entropy is a special case of categorical cross-entropy when there is only one output that just assumes a binary value of 0 or 1 to denote negative and positive class respectively. For example-classification between cat & dog.

Let us assume that actual output is denoted by a single variable y, then cross-entropy for a particular data D is can be simplified as follows –

Cross-entropy$(D) = -y*\log(p)$ when y = 1

Cost Function is No Rocket Science!

mean of cross-entropy for all N training data.

Binary Cross-Entropy = (Sum of Cross-Entropy for N data)/N

Frequently Asked Questions

Q1. What is cost function?

A. A cost function is a measure of how well a machine learning model performs by quantifying the difference between predicted and actual outputs. Its goal is to be minimized by adjusting the model's parameters during training.
The choice of cost function depends on the type of problem being solved, such as mean squared error for regression problems and cross-entropy for classification problems. The cost function determines the model's performance.

Q2. What is total and average cost function?

A. The total cost function calculates the total cost of production for a given quantity of output, including fixed and variable costs. It is represented as $TC(Q) = FC + VC(Q)$. The average cost function calculates the cost per unit of output, and is represented as $AC(Q) = TC(Q) / Q$. Average cost is the total cost divided by the quantity of output, and is used to determine the optimal level of production that minimizes costs.

Conclusion

I hope you found this article helpful! Let me know what you think, especially if there are suggestions for improvement. You can connect with me on LinkedIn: https://www.linkedin.com/in/saily-shah/ and here's my GitHub profile: https://github.com/sailyshah

The media shown in this article are not owned by

We use cookies on Analytics Vidhya websites to deliver our services, analyze web traffic, and improve your experience on the site. By using Analytics Vidhya, you agree to our Privacy Policy and Terms of Use.   Accept

Cost Function is No Rocket Science!

Saily Shah
17 Nov 2023

Advanced   Maths   Regression   Statistics

Responses From Readers

What are your thoughts?...

Submit reply

Alessandro
24 May, 2023

Shouldn't this " Error = y-y' Where, Y – Actual Input Y' –
Predicted output " have Y - Actual output? Thank you for
the article

Related Articles

Cost Function is No Rocket Science!

Data Science

Participate in Analytics Vidhya's 30th Edition of Blogathon So, start writing yo...

Apache Hadoop   AWS

Yana Khare
21 Mar, 2023 • 3 min read

Should Know ...

Do you want to be a data scientist? In this article we will learn about 10 usefu...

blogathon

functions in python

CHIRAG GOYAL
02 Aug, 2022 • 6 min read

Data Science Blogathon Winners are Here and Announcing Blogathon 2!

Here are the winners for the first Data Science Blogathon plus the official laun...

blogathon

blogathon winners

Pranav Dar
20 Oct, 2020 • 5 min read

Scribe, Shine, Succeed →

Write, captivate, and earn accolades and rewards for your work

✓   Reach a Global Audience

✓   Get Expert Feedback

Cost Function is No Rocket Science!

✓ Level Up Your Data Science Game

Barney Darlington
5

Suvojit Hore
9

Company

About Us

Contact Us

Careers

Learn

Free courses

Learning path

BlackBelt program

Gen AI

Contribute

Contribute & win

Become a speaker

Become a mentor

Become an instructor

Discover

Blogs

Expert session

Podcasts

Comprehensive Guides

Engage

Community

Hackathons

Events

Daily challenges

Enterprise

Our offerings

Case studies

Industry report

quexto.ai

Cost Function is No Rocket Science!

Google Play   App Store