Tipos de optimizers[[1]](#footnote-1)

|  |  |  |
| --- | --- | --- |
| Nome | Características | Mais indicado para... |
| Gradient Descent (GD) | directly uses the derivative of the loss function and learning rate to reduce the loss…It is easy to implement and interpret the results …The weights are updated when **the whole dataset gradient is calculated**, which slows down the process … | Ideia inicial das RNAs com backpropagation mas superada pelos novos algoritmos |
| Stochastic Gradient Descent | This is a changed version of the GD method, where the model parameters are updated on every iteration …These frequent updates result in converging to the minima in less time, but it comes at the cost of increased variance that can make the model overshoot the required position…But an advantage of this technique is low memory requirement |  |
| Mini-Batch Gradient Descent | Another variant of this GD approach is mini-batch, where the model parameters are updated in small batch sizes…the model is proceeding towards minima in fewer steps without getting derailed often. This results in less memory usage and low variance in the model. |  |
| Momentum Based Gradient Descent | … it can speed the whole process and this is what momentum means in this optimizer. This element depends on the previous value, learning rate, and a new parameter called gamma, which controls this **history update**. |  |
| Nesterov Accelerated Gradient (NAG) | The momentum-based GD gave a boost to the currently used optimizers by converging to the minima at the earliest, but it introduced a new problem. This method takes a lot of u-turns and oscillates in and out in the minima valley adding to the total time. The time taken is still way too less than normal GD, but this issue also needs a fix and this is done in NAG.  The approach followed here was that the parameters update would be made with the history element first and then only the derivative is calculated which can move it in the forward or backward direction. This is called the look-ahead approach, and it makes more sense because if the curve reaches near to the minima, then the derivative can make it move slowly so that there are fewer oscillations and therefore saving more time. |  |
| Adagrad | … adaptive learning rate that can change according to the input provided. Adagrad optimizer tries to offer this adaptiveness by decaying the learning rate in proportion to the updated history of the gradients… **One disadvantage of this approach is that the learning rate decays aggressively and after some time it approaches zero.** | For a sparse feature input where most of the values are zero, we can afford a higher learning rate which will boost the dying gradient resulted from these sparse features. |
| RMSProp | **It is an improvement to the Adagrad optimizer.** This aims to reduce the aggressiveness of the learning rate by taking an exponential average of the gradients instead of the cumulative sum of squared gradients. Adaptive learning rate remains intact as now exponential average will punish larger learning rate in conditions when there are fewer updates and smaller rate in a higher number of updates. |  |
| Adam | **Adaptive Moment Estimation combines the power of RMSProp (root-mean-square prop) and momentum-based GD**. In Adam optimizers, the power of momentum GD to hold the history of updates and the adaptive learning rate provided by RMSProp makes Adam optimizer a powerful method**. It also introduces two new hyper-parameters beta1 and beta2** which are usually kept around 0.9 and 0.99 but you can change them according to your use case. |  |

Tipos de loss functions[[2]](#footnote-2)

|  |  |  |
| --- | --- | --- |
| Nome | Tipo (Regression loss or Classification Loss) | Descrição |
| Mean Squared Error | Regression | Mean Squared Error is the mean of squared differences between the actual and predicted value. If the difference is large the model will penalize it as we are computing the squared difference. |
| Mean Squared Logarithmic Error Loss | Regression | Suppose we want to reduce the difference between the actual and predicted variable **we can take the natural logarithm of the predicted variable then take the mean squared error**. This will overcome the problem possessed by the Mean Square Error Method. The model will now penalize less in comparison to the earlier method. |
| Mean Absolute Error Loss | Regression | Sometimes there may be some data points which far away from rest of the points i.e outliers, in case of cases Mean Absolute Error Loss will be appropriate to use as it calculates the average of the absolute difference between the actual and predicted values. |
| Binary Cross Entropy Loss | Binary Classification | It gives the probability value between 0 and 1 for a classification task. Cross-Entropy calculates the average difference between the predicted and actual probabilities. |
| Hinge Loss | Binary Classification | This type of loss is used when the target variable has 1 or -1 as class labels. It penalizes the model when there is a difference in the sign between the actual and predicted class values. |
| Categorical Cross Entropy Loss | Multi-Class Classification | These are similar to binary classification cross-entropy, used for multi-class classification problems. |
| Kullback Leibler Divergence Loss | Multi-Class Classification | Kullback Leibler Divergence Loss calculates how much a given distribution is away from the true distribution. These are used to carry out complex operations like autoencoder where there is a need to learn the dense feature representation. |

Metrics

**Evaluation Metrics**

We will evaluate the performance of the model using Root Mean Squared Error (RMSE), a commonly used metric for regression problems. In simple terms, RMSE measures the average magnitude of the residuals or error. Mathematically, it is computed as the square root of the average of squared differences between predicted and actual values

1. https://www.upgrad.com/blog/types-of-optimizers-in-deep-learning/ [↑](#footnote-ref-1)
2. https://analyticsindiamag.com/loss-functions-in-deep-learning-an-overview/ [↑](#footnote-ref-2)