Loss Function 损失函数

什么是 Loss Function

Loss Function 是用来评价模型的预测值 (Prediction Value) 和 现实值 (Actual Value) 不一样的程度。 Loss Function 越小，通常模型的表现越好。

一般 Regression (回归) 任务所用的 Loss Function

- MSE (Mean Squared Error)
- MAE (Mean Absolute Error)
- RMSE (Root Mean Square Error)
- MSLE (Mean Squared Logarithmic Error)
- RMSLE (Root Mean Squared Logarithmic Error)
- R^2
- Huber Loss

一般 Classification (分类) 任务所用的 Loss Function

- Cross Entropy Loss
- Log-Likelihood Loss
- Hinge Loss (SVM)

MSE (Mean Squared Error)

MSE 也叫 L2 Loss，是比较受欢迎的一个 Loss Function 当需要解决 Regression 问题的时候都会优先使用。This is because most variables can be modeled into a Gaussian Distribution. MSE 随着误差的减小，梯度也在减小，有利于 Converge。

$$MSE = \frac{\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2}{n}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$n \rightarrow total\ number\ of\ data\ points$$

MSE is the average of the squared differences between the prediction and the actual values.

MAE (Mean Absolute Error)

MAE 也叫 L1 Loss，simple yet robust loss function used for regression models. Regression problems may have variables that are Not Strictly Gaussian in Nature due to the presence of outlier 离群值 (values that are very different from the rest of the data). MSE would be an ideal option in such cases because it does not take into account the direction of the outliers 离群值 (unrealistically high positive or negative values).

$$MAE = \frac{\sum_{i=1}^{n} \left| Y_i - \hat{Y}_i \right|}{n}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$n \rightarrow total\ number\ of\ data\ points$$

MAE takes the average sum of the absolute differences between the actual and the prediction values.

RMSE (Root Mean Square Error)

RMSE represents the standard deviation of the residuals (Differences Between the Model Predictions and the True Values). RMSE 算出来的值是和 Output 的值一样的 Unit。Example 比如说模型需要预测房价，每平方是万元，算出来的 Loss 不 Square Root 的话，那么差值会很大。

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(Y_i - \hat{Y}_i \right)^2}{n}}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$n \rightarrow total\ number\ of\ data\ points$$

RMSE tells you how concentrated the data is around the line of best fit.

MSLE (Mean Squared Logarithmic Error)

Presence of Outliers (异常值的存在) can explode the error term to a very high value. In the case of RMSLE, outliers are drastically scaled down (大幅缩减) the error there for nullifying (作废) their effect.

$$MSLE = \frac{\sum_{i=1}^{n} \left(\log(Y_i) - \log(\hat{Y}_i) \right)^2}{n}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$n \rightarrow total\ number\ of\ data\ points$$

RMSLE (Root Mean Squared Logarithmic Error)

Presence of Outliers (异常值的存在) can explode the error term to a very high value. In the case of RMSLE, outliers are drastically scaled down (大幅缩减) the error there for nullifying (作废) their effect.

$$RMSLE = \sqrt{\frac{\sum_{i=1}^{n}\left(\log(Y_i + 1) - \log\,(\hat{Y}_i + 1)\right)^2}{n}}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$n \rightarrow total\ number\ of\ data\ points$$

RMSLE metric only considers the relative error between the Predicted value and the Actual Value, and the scale of the error is not significant. Internal part of RMSLE, it is the fundamentally a calculation relative error.

Example:

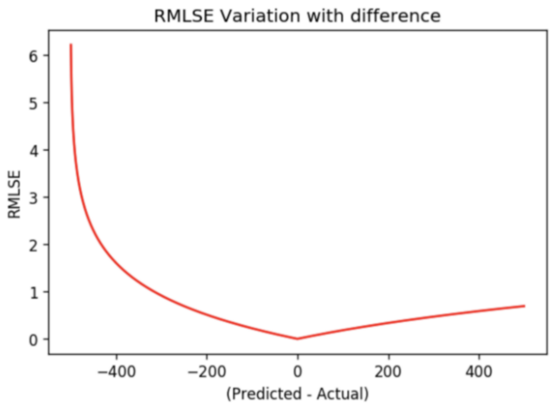
- Y = 90, $\hat{Y}$ = 100. Calculated RMSLE = 0.1053, Calculated RMSE = 10.
- Y = 9000, $\hat{Y}$ = 10000. Calculated RMSLE = 0.1053, Calculated RMSE = 1000.

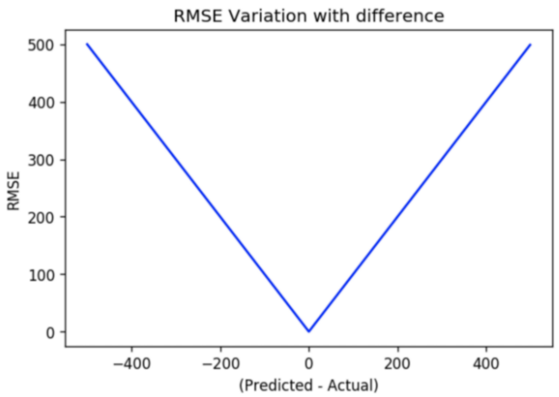
RMSLE incurs a large penalty for the underestimation of the Actual Variable than the Overestimation. In simple words, more penalty is incurred 招致 when the predicted Value is less than the Actual Value. 意思就是说 RMSLE 会 Underestimate 那个误差当 Actual Value 是大过 Prediction Value。

Example:

- Y = 600, $\hat{Y}$ = 1000. Calculated RMSLE = 0.510, Calculated RMSE = 400. (Underestimate)
- Y = 1400, $\hat{Y}$ = 1000. Calculated RMSLE = 0.33, Calculated RMSE = 400. (Overestimate)

适合用在 Predict 送达时间。当送达时间的预测是 Overestimate 一点是不伤大雅。但是如果送达时间是 Underestimate 会让送货员 miss the deadline。

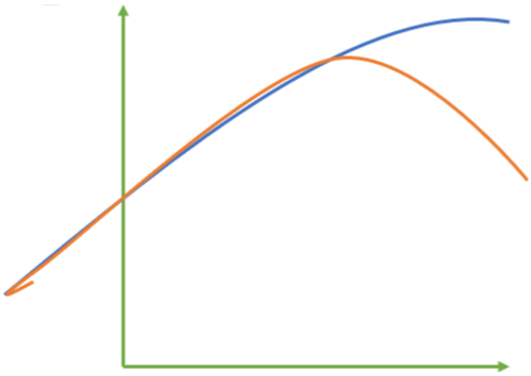




Predict – Actual 上图少过 0 代表 Underestimate，大过 0 代表 Overestimate。RMSLE 的图表示在 Underestimate 的情况 Error increases rapidly 迅速 as the underestimation of actual value increases. 在 Overestimate 的情况 Error is not increasing as rapidly 不迅速。

R^2 (R Squared)

在回归类算法，只探索数据预测是否准确是不足够的。



上图红色代表 Actual Values，蓝色代表拟合模型 (Fitting the model)。在图的前半段真实数据和拟合模型接近重叠，但是后半段有极大的差别。这时候如果用别的 Loss Function 来计算比如说 MSE，那么 MSE 计算出来的差值也是非常小因为前半段几乎 match 在一起。

R^2的值是在 0 – 1 之间。R^2能够判断 Prediction Value 是否正确之外，还能够判断我们的模型是否拟合了足够多的，数值之外的信息。

$$R^2 = \frac{SSR}{SST} = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y}_i)^2}{\sum_{i=1}^{n}\left(\hat{Y}_i - \bar{Y}_i\right)^2} = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2}{\sum_{i=1}^{n}\left(\hat{Y}_i - \bar{Y}_i\right)^2}$$

$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

$$\bar{Y}_i \rightarrow Mean\ Value$$

当R^2 接近 1 代表 Predicted Value 和 Actual Value 很接近，接近 0 代表模型不能很好的给出预测。

Huber Loss

MAE 的梯度太大，导致使用梯度下降的模型会遗漏最小值。MSE 梯度会随着 Loss 越小而变小，更容易找到最小值。在 Huber Loss 里使用了一个δ，当偏差值 ($|Y_i - \hat{Y}_i|$) 小于等于δ 的时候使用 MSE，当偏差值 ($|Y_i - \hat{Y}_i|$) 大于δ 的时候使用 MAE。 这种方法能够降低奇异数据点对于 Loss 计算的权重，避免模型过拟合 (Fit) 相比于最小二乘的线性回归。Huber Loss 降低了对离群点的惩罚程度，所以 Huber Loss 是一种常用的 Robust 的 Regression Loss Function。δ的选择非常关键。

$$Huber\ Loss = \begin{cases} \boldsymbol{\frac{1}{2}\left(Y_i - \hat{Y}_i\right)^2}\ \ For\ \left|Y_i - \hat{Y}_i\right| \le\ \delta \\ \boldsymbol{\delta\left|Y_i - \hat{Y}_i\right| - \frac{1}{2}\delta^2}\ Otherwise \end{cases}$$

$$Delta\ Value\ \delta \rightarrow Range\ for\ MAE\ and\ MSE\ (0, 1, 2, 3 \ldots n)$$

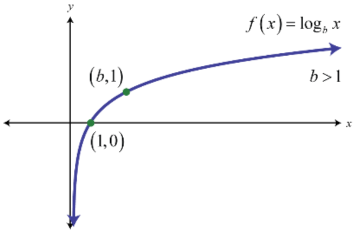
$$Y_i \rightarrow Predicted\ Value$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Ground\ Truth)$$

Cross-Entropy Loss

不使用 MSE 在 Classification 的任务是因为 MSE 采用梯度下降法进行学习，在模型一开始训练时，会发生学习的速率非常慢的情况。

Cross-Entropy Loss measures the performance of a Classification model whose output is a probability value between 0 and 1. Cross-Entropy Loss increases as the predicted probability diverges from the actual label.



Cross-Entropy Loss for Binary Classification

$$Cross\ Entropy\ Loss \rightarrow L(\hat{Y}, Y) = -\frac{\sum_{i=1}^{n} \hat{Y}_i \log(Y_i) + \left(1 - \hat{Y}_{i_i}\right) \log(1 - Y_i)}{n}$$

$$Y_i \rightarrow Predicted\ Value\ (Probability)$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Class\ 0\ or\ Class\ 1)$$

$$n \rightarrow total\ number\ of\ data\ points$$

When $\hat{Y} = 1$, $L(\hat{Y}, Y) = -\log(Y)$ → Want $\log(Y)$ Large, Want Y As Large as Possible

When $\hat{Y} = 0$, $L\left(\hat{Y}, Y\right) = -\log(1 - Y)$ → Want $\log(1 - Y)$ Large, Want Y As Small as Possible

Cross-Entropy Loss for Multiple Classification

$$Cross\ Entropy\ Loss \rightarrow L(\hat{Y}, Y) = -\frac{\sum - \sum_{c=1}^{M} \hat{Y}_i \log(Y_i)}{n}$$

$$Y_i \rightarrow Predicted\ Value\ (1\ for\ Class\ Object, 0\ for\ Non\ Class\ Object)$$

$$\hat{Y}_i \rightarrow Actual\ Value\ (Class\ 0\ or\ Class\ 1)$$

$$n \rightarrow total\ number\ of\ data\ points$$

$$M \rightarrow Number\ of\ Class$$

Example:

$$Sample\ 1\ Loss = -(0 * \log(0.3) + 0 * \log(0.3) + 1 * \log(0.4) = 0.91$$

$$Sample\ 2\ Loss = -(0 * \log(0.3) + 1 * \log(0.4) + 0 * \log(0.3) = 0.91$$

$$Sample\ 3\ Loss = -(1 * \log(0.1) + 0 * \log(0.2) + 0 * \log(0.7) = 2.30$$

$$Average\ Cross\ Entropy\ Loss = \frac{0.91 + 0.91 + 2.30}{3} = 1.37$$