Loss Function

Junkun Yuan
Zhejiang University
yuanjk@zju.edu.cn

1. Introduction

A **loss function**/cost function/error function is a function that maps an event or values of one or more variables onto a real number intuitively representing "cost" associated with the event. In statistics, typically a loss function is used for parameter estimation, and the event is function of difference between estimated and true values of an instance of data.[1]

We summarize **basic loss functions** with key phrases:

- **L1/MAE:** mean absolute error of regression.
- **MSE:** mean squared error of regression/classification.
- **0-1:** error rate of binary classification, non-convex, discontinuous.
- **Hinge:** maximum-margin classification, SVMs.
- **BCE:** binary cross entropy of binary classification.
- **CrossEntropy:** cross entropy of classification.
- **KL: Kullback-Leibler divergence of classification.**

We summarize **loss functions for image segmentation**:

- **BCE:** binary classification of pixels, balanced classes.
- **W-BCE:** weighted binary cross entropy, skewed data, tune false negatives and false positives.
- **B-BCE:** balanced binary cross-entropy, skewed data.
- **Focal:** down-weights easy examples.
- **Sensitivity Specificity:** weighted sum of sensitivity and specificity, TP is important.
- **Dice:** overlap ratio.
- **Tversky:** add a weight to to FP and FN in Dice.

[1]From wikipedia on loss function: https://en.wikipedia.org/wiki/Loss_function.

2. Basic Loss Functions

2.1. Regression

2.1.1 L1/MAE

L1/MAE loss function measures the mean absolute error of regression with prediction $\hat{y}$ and target y.

$$\ell_{\text{L1/MAE}} = \frac{1}{N}\sum_{n=1}^{N}|\hat{y}_n - y_n| \tag{1}$$

2.1.2 MSE

MSE loss measures the mean squared error of regression/classification with prediction $\hat{y}$ and target y.

$$\ell_{\text{MSE}} = \frac{1}{N}\sum_{n=1}^{N}(\hat{y}_n - y_n)^2 \tag{2}$$

2.2. Classification

2.2.1 0-1

0-1 loss function measures the error rate of binary classification with prediction $\hat{y}$ and target y.

$$\ell_{\text{0-1}} = \frac{1}{N}\sum_{n=1}^{N}\begin{cases}1, & y_n \neq \hat{y}_n\\ 0, & y_n = \hat{y}_n\end{cases} \tag{3}$$

But it is non-convex and discontinuous.

2.2.2 Hinge

Hinge loss, notably for SVMs, is used for maximum-margin classification with prediction $\hat{y}$, label $y \in \{-1, 1\}$:

$$\ell_{\text{Hinge}} = \frac{1}{N}\sum_{i=1}^{N}\max(0, 1 - y\hat{y}) \tag{4}$$

2.2.3 BCE

BCE loss measures binary cross entropy of binary classification with prediction $\hat{y}$ and target y.

$$\begin{aligned}\ell_{\text{BCE}} =& \mathbb{E}_{P(y)} - \log P(\hat{y}) \\ =& -\frac{1}{N}\sum_{n=1}^{N} y_n \log \hat{y}_n + (1-y_n)\log(1-\hat{y}_n)\end{aligned} \tag{5}$$

2.2.4 CrossEntropy

CrossEntropy loss computes the cross entropy of classification with prediction $\hat{y}$ and target y.

$$\begin{aligned}\ell_{\text{CrossEntropy}} =& \mathbb{E}_{P(y)} - \log P(\hat{y}) \\ =& \frac{1}{N}\sum_{n=1}^{N}\left[-\sum_{c=1}^{C} y_{n,c}\log \hat{y}_{n,c}\right]\end{aligned} \tag{6}$$

$y_{n,c}$/$\hat{y}_{n,c}$ represents the c-th class dimension of the prediction/target of the n-th example.

Discussion: Why do we often use CrossEntropy rather than MSE for classification? Assume that the prediction $\hat{y}$ is calculated by $\hat{y} = \sigma(wx + b)$ with weight parameter w, bias parameter b, and activation function σ. With MSE loss, we update the parameters by calculating their gradients:

$$\begin{aligned}\frac{\partial \ell_{\text{MSE}}}{\partial w} =& \frac{\partial \ell_{\text{MSE}}}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial w} = 2(\hat{y}-y)\sigma' x \\ \frac{\partial \ell_{\text{MSE}}}{\partial b} =& \frac{\partial \ell_{\text{MSE}}}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial b} = 2(\hat{y}-y)\sigma'\end{aligned} \tag{7}$$

It could slow down parameter update and optimization convergence when the activation function σ is easily saturated with large input like Sigmoid. Similarly and simply, considering CrossEntropy loss with two classes:

$$\begin{aligned}\frac{\partial \ell_{\text{CrossEntropy}}}{\partial w} =& \frac{\partial \ell_{\text{CrossEntropy}}}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial w} \\ =& -\left(\frac{y}{\hat{y}} + \frac{y-1}{1-\hat{y}}\right)\sigma' x \\ =& \left(\frac{y(\hat{y}-1) - \hat{y}(y-1)}{\hat{y}(1-\hat{y})}\right)\underbrace{\hat{y}(1-\hat{y})}_{\text{Sigmoid gradient}}\ x \\ =& (\hat{y}-y)x \\ \frac{\partial \ell_{\text{CrossEntropy}}}{\partial b} =& \frac{\partial \ell_{\text{CrossEntropy}}}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial b} = (\hat{y}-y)\end{aligned} \tag{8}$$

The gradients are not relevant to activation function σ, but to the distance between the prediction $\hat{y}$ and the target y. It speeds up the parameter update when the distance is large.

2.2.5 KL

KL loss measures Kullback-Leibler divergence of classification with prediction $\hat{y}$ and target y.

$$\begin{aligned}\ell_{\text{KL}} =& \mathbb{E}_{P(y)} \log \frac{P(y)}{P(\hat{y})} \\ =& \frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} y_{n,c}\log\frac{y_{n,c}}{\hat{y}_{n,c}}\end{aligned} \tag{9}$$

Note that $\ell_{\text{KL}} = \ell_{\text{CrossEntropy}} - \ell_{\text{Entropy}}$, where ℓ_{Entropy} is the entropy of the target y.

3. Loss Functions for Image Segmentation

Image segmentation can be seen as a classification task on image pixels. We introduce the popular loss functions for image segmentation by following [3].

3.1. Distribution-Based Loss Functions

3.1.1 BCE

It applies Eq. (5) to pixels of image segmentation. It is suitable for image segmentation with balanced classes.

3.1.2 W-BCE

W-BCE is weighted binary cross entropy where positive examples are weighted, suited for skewed data.

$$\ell_{\text{W-BCE}} = -\left(\beta y \log \hat{y} + (1-y)\log(1-\hat{y})\right) \tag{10}$$

The hyper-parameter β tunes false negatives and false positives. Set $\beta > 1$ may reduce false negatives, and vice versa.

3.1.3 B-BCE

B-BCE is balanced binary cross-entropy with $\beta = 1 - \frac{y}{HW}$.

$$\ell_{\text{W-BCE}} = -\left(\beta y \log \hat{y} + (1-\beta)(1-y)\log(1-\hat{y})\right) \tag{11}$$

3.1.4 Focal

Focal loss [4] down-weights easy examples and focus on hard negatives using a focusing parameter $\gamma >= 0$ and a balance parameter α. Its original form [4] with label y is:

$$\ell_{\text{Focal}} = -\alpha_t(1-p_t)^{\gamma}\log(p_t), \tag{12}$$

$$\text{where } p_t = \begin{cases} p, & \text{if } y = 1 \\ 1-p, & \text{otherwise} \end{cases}. \tag{13}$$

p is the prediction used as $\hat{y}$ by us. We rewrite Eq. (12):

$$\ell_{\text{Focal}} = -\left(\alpha(1-\hat{y})^{\gamma} y \log \hat{y} + (1-\alpha)\hat{y}^{\gamma}(1-y)\log(1-\hat{y})\right) \tag{14}$$

3.2. Region-Based Loss Functions

3.2.1 Sensitivity Specificity

Sensitivity specificity loss [2] is the weighted sum of sensitivity and specificity with weight w.

$$\ell_{\text{SensitivitySpecificity}} = w * \text{sensitivity} + (1 - w) * \text{specificity} \tag{15}$$

$$\begin{aligned} \text{where sensitivity} =& \frac{TP}{TP + FN} \\ \text{specificity} =& \frac{TN}{TN + FP} \end{aligned} \tag{16}$$

It is suitable to apply it to the cases where TP is important.

3.2.2 Dice

Dice loss [5] measures overlap ratio of segmentation images with the prediction $\hat{y}$ and label y:

$$\begin{aligned} \ell_{Dice} =& 1 - \frac{2|Y \cap \hat{Y}|}{|Y| + |\hat{Y}|} \\ =& 1 - \frac{2y\hat{y} + \epsilon}{y + \hat{y} + \epsilon} \end{aligned} \tag{17}$$

3.2.3 Tversky

Tversky loss adds a weight β to FP and FN in Dice loss:

$$\ell_{\text{Tversky}} = 1 - \frac{y\hat{y} + \epsilon}{y\hat{y} + \beta(1 - y)\hat{y} + (1 - \beta)y(1 - \hat{y}) + \epsilon} \tag{18}$$

3.2.4 Focal Tversky

Focal Tversky loss [1] also learns hard examples more:

$$\ell_{FocalTversky} = \sum_c (1 - TI_c)^\gamma \tag{19}$$

References

[1] Nabila Abraham and Naimul Mefraz Khan. A novel focal tversky loss function with improved attention u-net for lesion segmentation. In *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*, pages 683–687. IEEE, 2019. 3

[2] Seyed Raein Hashemi, Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, Sanjay P Prabhu, Simon K Warfield, and Ali Gholipour. Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: Application to multiple sclerosis lesion detection. *IEEE Access*, 7:1721–1735, 2018. 3

[3] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–7. IEEE, 2020. 2

[4] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[5] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 240–248. Springer, 2017. 3