Statistics and Application 统计学与应用, 2025, 14(7), 115-123
Published Online July 2025 in Hans. https://www.hanspub.org/journal/sa
https://doi.org/10.12677/sa.2025.147189

Hans汉斯

以得分优化为目标的损失函数

邹宇睿

福建师范大学数学与统计学院，福建 福州

收稿日期：2025年6月15日；录用日期：2025年7月6日；发布日期：2025年7月17日

摘　要

在机器学习的二分类任务中，损失函数对模型性能至关重要，但传统损失函数常难以直接优化准确率、F1 得分等关键评估指标。针对这一问题，本研究融合两种创新方法，提出基于近似 Heaviside 函数的得分导向损失函数(SOLH)。研究通过用梯度友好函数近似 Heaviside 阶跃函数，并将分类阈值视为随机变量，对近似阶跃函数取期望，实现评估指标的端到端可微，进而构建损失函数。理论分析表明，该损失函数满足 Lipschitz 连续性，适合优化。在艾滋病临床试验组数据集上的实验结果显示，以 F1 得分优化为目标时，SOLH 显著优于基线方法；在优化准确率方面，虽略逊于二元交叉熵损失函数，但仍优于其他基线方法。本研究成功整合两种前沿思路，搭建起训练与评估间的桥梁，不仅为损失函数性质提供理论依据，更通过实验验证其提升模型性能的有效性，为机器学习领域损失函数设计研究与实践应用开辟了新方向。

关键词

机器学习，二分类，损失函数，近似Heaviside函数

Loss Function Aimed at Score Optimization

Yurui Zou

Institute of Mathematics and Statistics, Fujian Normal University, Fuzhou Fujian

Received: Jun. 15th, 2025; accepted: Jul. 6th, 2025; published: Jul. 17th, 2025

Abstract

In binary classification tasks of machine learning, loss functions are pivotal in determining model performance. Nevertheless, traditional loss functions frequently fail to directly optimize critical evaluation metrics like accuracy and F1-score. Aiming at this problem, this study integrates two innovative approaches and proposes the Score-Oriented Loss with approximate Heaviside function (SOLH). The research approximates the Heaviside step function with a gradient-friendly function, regards the classification threshold as a random variable, and calculates the expectation of the

文章引用: 邹宇睿. 以得分优化为目标的损失函数[J]. 统计学与应用, 2025, 14(7): 115-123.
DOI: 10.12677/sa.2025.147189

approximate step function, thus achieving end-to-end differentiability of evaluation metrics and constructing the loss function. Theoretical analysis indicates that this loss function meets the Lipschitz continuity condition, rendering it suitable for optimization. Experiments conducted on the AIDS Clinical Trials Group dataset show that when optimizing for the F1-score, SOLH outperforms baseline methods significantly. When it comes to accuracy optimization, although SOLH lags slightly behind the binary cross-entropy loss function, it still surpasses other baseline methods. This study successfully combines two cutting-edge concepts, bridging the gap between training and evaluation. It not only offers a theoretical foundation for the properties of loss functions but also validates the effectiveness of improving model performance through experiments, opening up new avenues for the research and practical application of loss function design in the field of machine learning.

Keywords

Machine Learning, Binary Classification, Loss Function, Approximate Heaviside Function

1. 引言

在机器学习领域，尤其是二元分类任务中，损失函数的设计对于模型的性能至关重要。传统的损失函数，如交叉熵损失，虽然在许多场景下表现良好，但它们并不总是能够直接反映或优化模型在特定评估指标上的表现。例如，交叉熵损失并不能定向地优化 Accuracy 或 F1 得分等指标，而这些指标在实际应用中往往非常重要。因此，研究者们一直在探索如何设计损失函数，使其能够在训练过程中直接反映并优化这些关键的性能指标。

在这样的背景下，两篇具有创新性的研究文章引起了我的注意，F. Marchett (2022) [1]提出了一种基于期望混淆矩阵的损失函数，即得分导向损失(Score-Oriented Loss, SOL)函数。该函数允许在训练阶段先验地最大化技能得分，从而在训练过程中直接反映评估指标的性能。N. Tso (2022) [2]则从另一个角度出发，提出了一种基于软集混淆矩阵的统一训练和评估框架。该方法使用 Heaviside 阶跃函数的可微近似以及引入软集隶属度概念计算得到混淆矩阵值，从而使所需的评估指标可微，使得在训练过程中能够直接优化特定的性能指标。

在本文中，我将分析这两篇文章的方法，之后融合两种方法构建一种新的损失函数，并比较它们在艾滋病临床试验组研究数据集上的应用效果。我相信，通过深入理解这些方法，可以为机器学习领域的研究者和实践者提供有价值的参考和启示。

本文主要贡献如下：1) 通过融合期望混淆矩阵和软集混淆矩阵的方法，提出了一种新的损失函数；2) 对本文方法进行理论分析；3) 通过实验与几种基线方法相比，展现出本文方法的优越性。

2. 相关工作

2.1. 预备知识

在传统的二元分类问题中，混淆矩阵是一个关键工具，用于评估分类器的性能。它通过阈值将连续的预测概率转换为离散的类别标签，但这个阈值的选择往往对最终的性能评估有显著影响。

设 $\mathcal{P}=\{p_1,p_2,...,p_n\}$ 为模型输出的概率集合，$\mathcal{Y}=\{y_1,y_2,...,y_n\}$ 为对应的真实标签，$\tau\in(0,1)$ 为阈值，$H(p,\tau)$ 为 Heaviside 函数：

$$H(p,\tau)=\begin{cases}0 & \text{if } p<\tau,\\ 1 & \text{if } p\geq\tau.\end{cases} \tag{1}$$

传统的混淆矩阵元素可以表示为：

$$\begin{aligned} TP&=\sum_{i=1}^{n}y_iH(p_i,\tau), & FN&=\sum_{i=1}^{n}y_i\left(1-H(p_i,\tau)\right),\\ FP&=\sum_{i=1}^{n}(1-y_i)H(p_i,\tau), & TN&=\sum_{i=1}^{n}(1-y_i)\left(1-H(p_i,\tau)\right). \end{aligned} \tag{2}$$

其中 TP 和 FN 分别代表真阳性(True Positives)和假阴性(False Negatives)的值。 FP 和 TN 分别代表假阳性(False Positives)和真阴性(True Negatives)的值。

通过混淆矩阵可以得到分类器的性能得分：

$$Score=Score(TP,FN,FP,TN). \tag{3}$$

2.2. 基于期望混淆矩阵的方法

基于期望混淆矩阵的方法提出了一种创新的视角，将阈值视为一个随机变量，而不是固定的值。这种方法采用了期望混淆矩阵的概念，并通过选择合适的概率分布函数来影响阈值的选择。其核心在于构建一个可微分的损失函数，允许模型在训练过程中直接针对特定的性能指标进行优化。

阈值 $\tau\in(0,1)$ 被视为随机变量，其概率密度函数为 $f(\tau)$，概率分布函数为 $F(\tau)$，那么：

$$E_\tau\left[H(p,\tau)\right]=\int_0^1H(p,\tau)f(\tau)\mathrm{d}\tau=\int_0^pf(\tau)\mathrm{d}\tau=F(p). \tag{4}$$

因此，期望混淆矩阵的元素可以表示为：

$$\begin{aligned} E[TP]&=\sum_{i=1}^{n}y_iF(p_i), & E[FN]&=\sum_{i=1}^{n}y_i\left(1-F(p_i)\right),\\ E[FP]&=\sum_{i=1}^{n}(1-y_i)F(p_i), & E[TN]&=\sum_{i=1}^{n}(1-y_i)\left(1-F(p_i)\right). \end{aligned} \tag{5}$$

从而，可以定义基于期望混淆矩阵损失函数如下：

$$\mathcal{L}_{SOL}=1-Score\left(E[TP],E[FN],E[FP],E[TN]\right). \tag{6}$$

2.3. 基于软集混淆矩阵的方法

为了解决训练和评估神经网络二分类器之间的差距，基于软集混淆矩阵的方法提出，使用 Heaviside 阶跃函数的可微近似以及使用软集隶属度概念计算的混淆矩阵值。然后，通过软集混淆矩阵计算评估指标，而不是使用传统的混淆矩阵，这样可以使所需的评估指标可微，从而构造损失函数。

近似 Heaviside 阶跃函数 H^l 为

$$H^l=\begin{cases} p\cdot m_1 & \text{if } p<\tau-\dfrac{\tau_m}{2},\\ p\cdot m_3+\left(1-\delta-m_3\left(\tau+\dfrac{\tau_m}{2}\right)\right) & \text{if } p>\tau+\dfrac{\tau_m}{2},\\ p\cdot m_2+(0.5-m_2\tau) & \text{otherwise.}\end{cases} \tag{7}$$

$$m_1=\frac{\delta}{\tau-\frac{\tau_m}{2}},\qquad m_2=\frac{1-2\delta}{\tau_m},\qquad m_3=\frac{\delta}{1-\tau-\frac{\tau_m}{2}},\qquad \tau_m=\min\{\tau,1-\tau\}.$$

其参数为给定的阈值 $\tau \in (0,1)$ 和斜率参数 δ，它将三个斜率为 m_1， m_2 和 m_3 的线段连接起来。软集混淆矩阵的元素可以表示为：

$$\begin{aligned} TP_s &= \sum_{i=1}^{n} y_i H^l(p_i,\tau), & FN_s &= \sum_{i=1}^{n} y_i (1-H^l(p_i,\tau)), \\ FP_s &= \sum_{i=1}^{n} (1-y_i) H^l(p_i,\tau), & TN_s &= \sum_{i=1}^{n} (1-y_i)(1-H^l(p_i,\tau)). \end{aligned} \tag{8}$$

从而，可以定义基于软集混淆矩阵的损失函数如下：

$$\mathcal{L}_s = 1 - Score(TP_s, FN_s, FP_s, TN_s). \tag{9}$$

3. 方法论

本文提出了一种融合期望混淆矩阵和软集混淆矩阵的方法，更好地统一神经网络二元分类器的训练和评估步骤。本文的方法主要有两个步骤，首先，用公式(7)中的梯度友好的分段函数 H^l 进行近似 Heaviside 阶跃函数。然后，将阈值视为一个随机变量，对近似阶跃函数求期望。这样就可以将近似阶跃函数的期望代入计算基于混淆矩阵的指标。这种方法使指标具有端到端的可微性且考虑到阈值的分布，因此它们可以用作构造损失函数。

阈值 $\tau \in (0,1)$ 被视为随机变量，其概率密度函数为 $f(\tau)$，概率分布函数为 $F(\tau)$，那么：

$$\begin{aligned} E_\tau\left[H^l(p,\tau)\right] &= \int_0^1 H^l(p,\tau) f(\tau) dx \\ &= \begin{cases} 2p\delta\left(\int_{2p}^{\frac{1}{2}} \frac{f(\tau)}{\tau} d\tau + \int_{\frac{1}{2}}^{1} \frac{f(\tau)}{3\tau-1} d\tau\right), & 0 \le p < \frac{1}{4}, \\ 2p\delta \int_{\frac{2p+1}{3}}^{1} \frac{f(\tau)}{3\tau-1} d\tau, & \frac{1}{4} \le p \le 1, \end{cases} \\ &+ \begin{cases} \int_0^{\frac{2p}{3}} \left[(2p\delta - 2\delta)\frac{1}{2-3\tau} + 1\right] f(\tau) d\tau, & 0 \le p < \frac{3}{4}, \\ \int_0^{\frac{1}{2}} \left[(2p\delta - 2\delta)\frac{1}{2-3\tau} + 1\right] f(\tau) d\tau + \int_{\frac{1}{2}}^{2p-1} \left[(2p\delta - 2\delta)\frac{1}{1-\tau} + 1\right] f(\tau) d\tau, & \frac{3}{4} \le p \le 1, \end{cases} \\ &+ \begin{cases} \int_{\frac{2p}{3}}^{p} \left[p(1-2\delta)\frac{1}{\tau} + 2\delta - 0.5\right] f(\tau) d\tau, & 0 \le p < \frac{1}{4}, \\ \int_{\frac{2p}{3}}^{\frac{1}{2}} \left[p(1-2\delta)\frac{1}{\tau} + 2\delta - 0.5\right] f(\tau) d\tau + \int_{\frac{1}{2}}^{\frac{2p+1}{3}} \left[(1-2\delta)\frac{p-\tau}{1-\tau} + 0.5\right] f(\tau) d\tau, & \frac{1}{4} \le p < \frac{3}{4}, \\ \int_{2p-1}^{\frac{2p+1}{3}} \left[(1-2\delta)\frac{p-\tau}{1-\tau} + 0.5\right] f(\tau) d\tau, & \frac{3}{4} \le p \le 1. \end{cases} \end{aligned} \tag{10}$$

因此，基于近似阶跃函数的期望混淆矩阵的元素可以表示为：

$$\begin{aligned} E[TP_s] &= \sum_{i=1}^{n} y_i E_\tau\left[H^l(p_i,\tau)\right], & E[FN_s] &= \sum_{i=1}^{n} y_i \left(1 - E_\tau\left[H^l(p_i,\tau)\right]\right), \\ E[FP_s] &= \sum_{i=1}^{n} (1-y_i) E_\tau\left[H^l(p_i,\tau)\right], & E[TN_s] &= \sum_{i=1}^{n} (1-y_i)\left(1 - E_\tau\left[H^l(p_i,\tau)\right]\right). \end{aligned} \tag{11}$$

当 τ 服从均匀分布：

$$F(x) = x \cdot I\{0 < x < 1\}. \tag{12}$$

当 τ 服从升余弦分布：

$$F_{C(\mu,\delta)}(x) = \frac{1}{2}\left(1+\frac{x-\mu}{\delta}+\frac{1}{\pi}\sin\left(\frac{x-\mu}{\delta}\pi\right)\right)I\{\mu-\delta<x<\mu+\delta\}+I\{\mu+\delta\le x<1\}. \tag{13}$$

从而，可以定义 $\mathcal{L}_{SOLH}$ 损失(Score-Oriented Loss with approximate Heaviside funcation, SOLH)如下：

$$\mathcal{L}_{SOLH} = 1-Score\left(E[TP_s],E[FN_s],E[FP_s],E[TN_s]\right). \tag{14}$$

4. 理论基础

4.1. Lipschitz 连续性

定义 4.1.对于在实数集的子集的函数 $f:\mathcal{X}\subseteq\mathbb{R}\to\mathbb{R}$。若存在常数 $L\ge 0$，使得对 $\forall x_1,x_2\in\mathcal{X}$ 有：

$$\left|f(x_1)-f(x_2)\right|\le L\left|x_1-x_2\right|. \tag{15}$$

则称 f 是 Lipschitz 连续的，其中 L 称为 f 的 Lipschitz 常数。

定理 4.1. (N. Tso (2022)) [2]线性 Heaviside 函数近似 $H^l(p,\tau)$ 是 Lipschitz 连续的，其 Lipschitz 常数为 $M=\max\{m_1,m_2,m_3\}$，因此对 $\forall p_1,p_2\in[0,1]$ 有：

$$\left|H^l(p_1,\tau)-H^l(p_2,\tau)\right|\le M\left|p_1-p_2\right|. \tag{16}$$

由定理 4.1.可推导出，存在常数 $M>0$，使得对 $\forall p_1,p_2\in[0,1]$ 有：

$$\begin{aligned}\left|E_\tau\left[H^l(p_i,\tau)\right]-E_\tau\left[H^l(p_2,\tau)\right]\right| &\le \int_0^1\left|H^l(p_1,\tau)-H^l(p_2,\tau)\right|f(\tau)d\tau\\ &\le M\left|p_1-p_2\right|\int_0^1 f(\tau)d\tau\\ &= M\left|p_1-p_2\right|.\end{aligned} \tag{17}$$

所以 $E_\tau\left[H^l(p,\tau)\right]$ 是 Lipschitz 连续的。

混淆矩阵元素 $E[TP_s]$ 是 $E_\tau\left[H^l(p_i,\tau)\right]$ 的线性组合，Lipschitz 函数的线性组合保持 Lipschitz 连续性：

$$\begin{aligned}\left|E[TP_s](p_1)-E[TP_s](p_2)\right| &= \left|\sum_{i=1}^n y_i E_\tau\left[H^l(p_{2,i},\tau)\right]-\sum_{i=1}^n y_i E_\tau\left[H^l(p_{2,i},\tau)\right]\right|\\ &\le \sum_{i=1}^n y_i M\left|p_{1,i}-p_{2,i}\right|\\ &\le M\sum_{i=1}^n\left|p_{1,i}-p_{2,i}\right|\\ &\le M\left\|p_1-p_2\right\|_1.\end{aligned} \tag{18}$$

类似地，其他元素 $E[FP_s],E[TN_s],E[FN_s]$ 也满足 Lipschitz 条件。

根据 Krzysztof Dembczynski (2017) [3]的观点，基于混淆矩阵的指标如准确率、F_β-Score 等都是 Lipschitz 连续的。又因为 Lipschitz 连续函数的复合函数仍然是 Lipschitz 连续的，所以由满足 Lipschitz 条件的混淆矩阵值 $E[TP_s],E[FP_s],E[TN_s],E[FN_s]$ 组成的损失函数 $\mathcal{L}_{SOLH}$ 也是 Lipschitz 连续的。

4.2. 梯度分析

设 $c(x)$ 为样本 x 的特征表示，w 为模型的权重，则模型输出的概率为 $p=\sigma\left(w^T c(x)\right)$，其中 σ 是 sigmoid 函数。

通过链式法则，损失函数的梯度为：

$$\nabla_w\mathcal{L}_{SOLH} = -\left[\frac{\partial\text{Score}}{\partial E[TP_s]}\nabla_w E[TP_s]+\frac{\partial\text{Score}}{\partial E[FN_s]}\nabla_w E[FN_s]+\frac{\partial\text{Score}}{\partial E[FP_s]}\nabla_w E[FP_s]+\frac{\partial\text{Score}}{\partial E[TN_s]}\nabla_w E[TN_s]\right]. \tag{19}$$

混淆矩阵中各元素的梯度：

$$\begin{aligned}&\nabla_w E\left[TP_s\right]=\sum_{i=1}^{n} y_i \nabla_w E_\tau\left[H^l\left(p_i,\tau\right)\right], \qquad \nabla_w E\left[FN_s\right]=-\sum_{i=1}^{n} y_i \nabla_w E_\tau\left[H^l\left(p_i,\tau\right)\right],\\&\nabla_w E\left[FP_s\right]=\sum_{i=1}^{n}\left(1-y_i\right)\nabla_w E_\tau\left[H^l\left(p_i,\tau\right)\right], \nabla_w E\left[TN_s\right]=-\sum_{i=1}^{n}\left(1-y_i\right)\nabla_w E_\tau\left[H^l\left(p_i,\tau\right)\right].\end{aligned} \tag{20}$$

其中

$$\begin{aligned}\nabla_w E_\tau\left[H^l\left(p_i,\tau\right)\right]&=\nabla_p E_\tau\left[H^l\left(p_i,\tau\right)\right]\nabla_w p_i\\&=\nabla_p E_\tau\left[H^l\left(p_i,\tau\right)\right] p_i\left(1-p_i\right)c\left(x\right).\end{aligned} \tag{21}$$

$\nabla_p E_\tau\left[H^l\left(p,\tau\right)\right]$ 表示如下：

$$\begin{aligned}&\nabla_p E_\tau\left[H^l\left(p,\tau\right)\right]\\&=\begin{cases}2\delta\left[\int_{2p}^{\frac{1}{2}}\frac{f\left(\tau\right)}{\tau}d\tau-f\left(2p\right)+\int_{\frac{1}{2}}^{1}\frac{f\left(\tau\right)}{3\tau-1}d\tau\right], & 0\le p<\frac{1}{4},\\2\delta\left[\int_{\frac{2p+1}{3}}^{1}\frac{f\left(\tau\right)}{3\tau-1}d\tau-\frac{1}{3}f\left(\frac{2p+1}{3}\right)\right], & \frac{1}{4}\leqslant p\leqslant 1,\end{cases}\\&+\begin{cases}2\delta\int_{0}^{\frac{2p}{3}}\frac{f\left(\tau\right)}{2-3\tau}d\tau+\frac{2}{3}f\left(\frac{2p}{3}\right)\left(1-\delta\right), & 0\leqslant p<\frac{3}{4},\\2\delta\left[\int_{0}^{\frac{1}{2}}\frac{f\left(\tau\right)}{2-3\tau}d\tau+\int_{\frac{1}{2}}^{2p-1}\frac{f\left(\tau\right)}{1-\tau}d\tau\right]+2\left(1-\delta\right)f\left(2p-1\right), & \frac{3}{4}\leqslant p\leqslant 1,\end{cases}\\&+\begin{cases}\int_{\frac{2p}{3}}^{2p}\left(1-2\delta\right)\frac{1}{\tau}f\left(\tau\right)d\tau+2\delta f\left(2p\right)+\frac{2}{3}f\left(\frac{2p}{3}\right)\left(\delta-1\right), & 0\leqslant p<\frac{1}{4},\\\int_{\frac{2p}{3}}^{\frac{1}{2}}\left(1-2\delta\right)\frac{1}{\tau}f\left(\tau\right)d\tau+\int_{\frac{1}{2}}^{\frac{2p+1}{3}}\left(1-2\delta\right)\frac{1}{1-\tau}f\left(\tau\right)d\tau+\frac{2}{3}\left(\delta-1\right)f\left(\frac{2p}{3}\right)+\frac{2}{3}\delta f\left(\frac{2p+1}{3}\right), & \frac{1}{4}\leqslant p<\frac{3}{4},\\\int_{2p-1}^{\frac{2p+1}{3}}\frac{1-2\delta}{1-x}f\left(\tau\right)d\tau+\frac{2}{3}\delta f\left(\frac{2p+1}{3}\right)+\left(2\delta-2\right)f\left(2p-1\right), & \frac{3}{4}\leqslant p<1.\end{cases}\end{aligned} \tag{22}$$

以准确率为目标的损失函数的梯度：

$$\nabla_w \mathcal{L}_{SOLH}=-\frac{1}{n}\sum_{i=1}^{n}\left(2y_i-1\right)\nabla_p E_\tau\left[H^l\left(p_i,\tau\right)\right] p_i\left(1-p_i\right)c\left(x\right). \tag{23}$$

以 F1 Score 为目标的损失函数的梯度：

$$\nabla_w \mathcal{L}_{SOLH}=-\frac{2B\left(\sum_{i=1}^{n} y_i \nabla_p E_\tau\left[H^l\left(p_i,\tau\right)\right] p_i\left(1-p_i\right)c\left(x\right)\right)-2A\left(\sum_{i=1}^{n}\left(1-2y_i\right)\nabla_p E_\tau\left[H^l\left(p_i,\tau\right)\right] p_i\left(1-p_i\right)c\left(x\right)\right)}{\left(2A+B\right)^2}, \tag{24}$$

其中：

$$\begin{aligned}A&=\sum_{i=1}^{n} y_i E_\tau\left[H^l\left(p_i,\tau\right)\right],\\B&=\sum_{i=1}^{n}\left(1-y_i\right)E_\tau\left[H^l\left(p_i,\tau\right)\right]+y_i\left(1-E_\tau\left[H^l\left(p_i,\tau\right)\right]\right).\end{aligned} \tag{25}$$

5. 实验结果

本次实验所用到的数据来源于艾滋病临床试验组研究数据集[4]。数据集包含有关被诊断患有 AIDS 的患者的医疗保健统计数据和分类信息。该数据集最初于 1996 年发布。预测任务是预测每个患者是否在某个时间窗口内死亡。该数据集涵盖了 2139 个样本，其中正类的占比为 24.36%。

本文使用了一个前馈神经网络，包括了三个全连接层(线性层)，每层之后跟随 ReLU 激活函数，引入非线性，并且加入 Dropout 层用于正则化。本文采用了 Adam 优化算法[5]，该算法有以下优点：易于实现、计算效率高、内存需求少、参数更新大小不随梯度的对角缩放而变化、适合非平稳目标和包含噪声或稀疏梯度的问题、超参数有直观解释通常无需调整等。

本文在 500 个 epoch 上进行训练，当 7 个 epoch 内验证损失没有得到改善(评估损失差值小于等于 0.0001)时，实验施加了一个早期停止条件。至于目标性能得分，本文考虑了准确率(Accuracy)和 F1 得分(F1-Score)它们被定义为：

$$\begin{aligned}\text{Accuracy} &= \frac{TP+TN}{TP+TN+FP+FN},\\ \text{F1-Score} &= \frac{2TP}{2TP+FP+FN}.\end{aligned} \tag{26}$$

考虑均匀和上升的余弦分布 $\mathcal{C}(\mu,\delta)$ (在 $\mu = 0.5$ 和 $\delta = 0.1$，0.3，0.5 的情况下)，本文比较了常用的二元交叉熵损失(Binary Cross-Entropy Loss)、基于软集混淆矩阵的损失 $\mathcal{L}_s$、基于期望混淆矩阵的损失 $\mathcal{L}_{SOL}$ 以及融合两种方法的损失 $\mathcal{L}_{SOLH}$。本文通过评估默认阈值 0.5 和后验分数最大化选择的最佳阈值 τ^*的性能得分来报告在训练集和测试集上的表现。为消除随机波动，实验结果取三次重复实验的均值。

Table 1. Comparison of loss functions with accuracy as the target
表 1. 以 Accuracy 为目标的损失函数的比较

Loss	Train				Test	
	Epoch	Accuracy	τ^*	Accuracy (τ^*)	Accuracy	Accuracy (τ^*)
BCELoss	93	0.8988	0.4267	0.9042	0.8591	0.8630
$\mathcal{L}_s$	77	0.8956	0.0633	0.9073	0.8455	0.8484
$\mathcal{L}_{SOL}(\mathcal{C}(0.5,0.1))$	55	0.8559	0.4167	0.8590	0.7930	0.7959
$\mathcal{L}_{SOL}(\mathcal{C}(0.5,0.3))$	68	0.8575	0.2733	0.8629	0.8115	0.8066
$\mathcal{L}_{SOL}(\mathcal{C}(0.5,0.5))$	84	0.9065	0.2067	0.9097	0.8494	0.8503
$\mathcal{L}_{SOL}(\mu)$	79	0.8949	0.1767	0.9065	0.8387	0.8416
$\mathcal{L}_{SOLH}(\mathcal{C}(0.5,0.1))$	58	0.8590	0.1500	0.8614	0.8008	0.8076
$\mathcal{L}_{SOLH}(\mathcal{C}(0.5,0.3))$	87	0.8949	0.1200	0.9081	0.8455	0.8533
$\mathcal{L}_{SOLH}(\mathcal{C}(0.5,0.5))$	70	0.8917	0.1267	0.9034	0.8397	0.8455
$\mathcal{L}_{SOLH}(\mu)$	82	0.8925	0.0733	0.9073	0.8377	0.8377

本文基于表 1 和表 2 的实验数据，对以 Accuracy 和 F1-Score 为目标的损失函数性能进行了系统比较分析。在以 Accuracy 为目标的实验中(表 1)，$\mathcal{L}_{SOLH}(\mathcal{C}(0.5,0.3))$ 表现仅次于 BCELoss，其测试集准确率达

0.8455，最佳阈值下测试集准确率为 0.8533。

在以 F1-Score 为目标的实验中(表 2)，$\mathcal{L}_{SOLH}\left(\mathcal{C}\left(0.5,0.3\right)\right)$ 与 $\mathcal{L}_{SOLH}\left(\mu\right)$ 在测试集表现最佳，前者对于默认阈值的 F1-Score 得分为 0.7213，后者对于最佳阈值的 F1-Score 得分为 0.7307。

综合分析表明：本文提出的 $\mathcal{L}_{SOLH}$ 在 F1-Score 优化中优势明显，尽管在 Accuracy 的优化中不及 BCELoss，但是相较于基线 $\mathcal{L}_S$ 以及 $\mathcal{L}_{SOL}$ 有所提升。两类实验共同验证了改进损失函数在模型性能提升方面的有效性。

Table 2. Comparison of loss functions with f1-score as the target
表 2. 以 F1-Score 为目标的损失函数的比较

Loss	Train				Test	
	Epoch	F1-Score	τ^*	F1-Score (τ^*)	F1-Score	F1-Score (τ^*)
BCELoss	98	0.3167	0.7800	0.7800	0.7056	0.7203
$\mathcal{L}_S$	55	0.5367	0.7844	0.7844	0.6951	0.6926
$\mathcal{L}_{SOL}\left(\mathcal{C}\left(0.5,0.1\right)\right)$	64	0.4933	0.7585	0.7585	0.6928	0.6864
$\mathcal{L}_{SOL}\left(\mathcal{C}\left(0.5,0.3\right)\right)$	54	0.5233	0.7843	0.7843	0.6952	0.6952
$\mathcal{L}_{SOL}\left(\mathcal{C}\left(0.5,0.5\right)\right)$	67	0.6000	0.7821	0.7821	0.7042	0.6931
$\mathcal{L}_{SOL}\left(\mu\right)$	57	0.5467	0.7908	0.7908	0.7078	0.7169
$\mathcal{L}_{SOLH}\left(\mathcal{C}\left(0.5,0.1\right)\right)$	59	0.3633	0.7807	0.7807	0.7100	0.7179
$\mathcal{L}_{SOLH}\left(\mathcal{C}\left(0.5,0.3\right)\right)$	69	0.5600	0.7869	0.7869	0.7213	0.7123
$\mathcal{L}_{SOLH}\left(\mathcal{C}\left(0.5,0.5\right)\right)$	52	0.3733	0.8025	0.8025	0.7018	0.7082
$\mathcal{L}_{SOLH}\left(\mu\right)$	59	0.4000	0.7892	0.7892	0.7187	0.7307

6. 结论

两种方法在优化神经网络二分类器时具有不同的侧重点和技术手段，但它们都旨在解决训练和评估过程之间的不一致性。两种方法的共同目标都是通过在训练过程中直接优化评估指标，使训练和评估过程保持一致，从而提升分类器的性能。

基于期望混淆矩阵的方法通过将阈值视为随机变量，并基于其概率分布来构建损失函数，直接优化分类性能指标。其优点在于不需要明确设定单一阈值，而是通过概率分布捕捉不同阈值下的分类性能，有助于更全面地优化分类器。阈值的概率分布使得混淆矩阵的期望值也是可微的，从而能够进行梯度计算和参数优化。

基于软集混淆矩阵的方法通过引入可微的近似 Heaviside 函数和软集合，构建了一个在训练过程中可以计算和优化混淆矩阵的框架。其优点在于可以明确控制阈值的设定，并且通过软集合能够将混淆矩阵的值转化为连续可微的形式。Heaviside 函数的可微近似确保了在反向传播时的梯度计算，可以有效进行参数更新。

本文通过结合这两种方法构造了一种新的损失函数 $\mathcal{L}_{SOLH}$，此方法通过对软集混淆矩阵取期望，引入了阈值的概率分布，拓宽了损失函数参数的选择空间。理论分析表明了 $\mathcal{L}_{SOLH}$ 是 Lipschitz 连续的。此外，本文通过实验验证了该方法的可行性，并且展现了其对分类器的性能提升，尤其是当以 F1 得分为目标时性能提升效果明显。

未来，我将进一步研究该方法的其他方面。从方法论的角度来看，可以将该方法扩展到以加权评估

指标为目标的情况上。从应用的角度来看，可以考虑将该方法应用于多类别分类问题。

参考文献

[1] Marchetti, F., Guastavino, S., Piana, M. and Campi, C. (2022) Score-Oriented Loss (SOL) Functions. *Pattern Recognition*, **132**, Article ID: 108913. https://doi.org/10.1016/j.patcog.2022.108913

[2] Tsoi, N., Candon, K., Li, D., *et al*. (2022) Bridging the Gap: Unifying the Training and Evaluation of Neural Network Binary Classifiers. *Advances in Neural Information Processing Systems*, **35**, 23121-23134.

[3] Dembczyński, K., Kotłowski, W., Koyejo, O., *et al*. (2017) Consistency Analysis for Binary Classification revisited. International Conference on Machine Learning. PMLR, 961-969.

[4] Hammer, S.M., Katzenstein, D.A., Hughes, M.D., Gundacker, H., Schooley, R.T., Haubrich, R.H., *et al*. (1996) A Trial Comparing Nucleoside Monotherapy with Combination Therapy in HIV-Infected Adults with CD4 Cell Counts from 200 to 500 per Cubic Millimeter. *New England Journal of Medicine*, **335**, 1081-1090. https://doi.org/10.1056/nejm199610103351501

[5] Kingma, D.P. (2014) Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980.