

# 机器学习笔记

纳文琪<sup>1</sup>

2019 年 8 月 30 日

## 1 学习算法的性能度量 [1]

### 1.1 错误率和精度

**错误率** 指的是分类错误的样本占总样本数的比例，主要适用于二分类问题，也可用于多分类问题。

**精度** 指的是分类正确的样本数占总样本数的比例，同意适用于二分类和多分类问题。

错误率和精度简单、常用，但并不能满足所有需求。

### 1.2 查准率、查全率和 F1

**混淆矩阵** 对二分类问题，可以将真实类别与预测类别组合划分成 TP、TN、FP、FN 四种情形，分别表示预测正确的正例和反例、预测错误的正例和反例。分类结果可以使用一个“混淆矩阵”表示：

		Predicted class	
		$P$	$N$
Actual Class	$P$	True Positives (TP)	False Negatives (FN)
	$N$	False Positives (FP)	True Negatives (TN)

图 1: Confusion Matrix

**查准率（准确率，precision）** 是从预测结果（其数量作为分母）出发计算的精度，指的是预测为正例的样本中，有多少的预测正确的。其定义是：

$$P = \frac{TP}{TP + FP} \quad (1)$$

**查全率（召回率，recall）** 是从样本（其数量作为分母）出发计算的精度，指的是所有正例样本中，有多少被预测正确了。其定义是：

$$R = \frac{TP}{TP + FN} \quad (2)$$

查全率与查准率是一对矛盾的度量。

**P-R 曲线** 用于直观地显示学习器在样本总体上的查全率、查准率。

**F1 和  $F_\beta$**  F1 是综合考虑查全率和查准率的度量，定义为：

$$F1 = \frac{2 \times P \times R}{P + R} \quad (3)$$

一些应用中，对查准率和查全率的重视程度不同，此时需要用  $F_\beta$ ：

$$F_\beta = \frac{(1 + \beta^2) \times P \times R}{(\beta^2 \times P) + R} \quad (4)$$

$F_\beta$  是 F1 的一般形式，当  $\beta = 1$  时就是 F1；当  $\beta > 1$  时，查全率有更大影响；当  $\beta < 1$  时，查全率有更大影响。

### 1.3 ROC 与 AUC

ROC 全称是“受试者工作特征”，与 P-R 曲线类似，它也有两个坐标，其纵坐标表示的是“真正例率”（TPR），即正例的查全率，横坐标表示的是“假正例率”，及反例被判断错误的比率。两者定义为：

$$TPR = \frac{TP}{TP + FN} \quad (5)$$

$$FPR = \frac{FP}{TN + FP} \quad (6)$$

ROC 曲线的对角线对应的是“随机猜想”模型，而点 (0,1) 则代表“理想模型”。

AUC 进行学习器比较时，与 P-R 曲线类似，若一个学习器的 ROC 曲线被另一个学习器的曲线完全包住，则可断言后者的性能优于前者；若两个学习器的 ROC 曲线有交叉，则需要比较 ROC 曲线下的面积，即 AUC 来进行判断。

## 2 神经网络

**隐层参数**  $W_j \in \mathbb{R}^{d \times n}$  表示 j-th 层的参数，此层有 n 个神经元，接收 d 个来自的 (j-1)-th 层的输入。 $W_j$  的第 k 列，就是输入到第 k 个神经元的数据对应的参数。

**隐层输出**  $y_j \in \mathbb{R}^n$  表示 j-th 层的输出，n 个神经元有 n 个输出，计算公式为：  
 $y_j = W_j^T y_{j-1} + b_j$

### 3 Deep Neural Networks for Bot Detection[2]

**Motivation** 现有的系统都是在 account-level 进行 bots 的发现，需要根据特定帐号的一系列历史活动记录来确认帐号是否是 bot。这在进行检测的时候代价非常昂贵。论文希望通过仅仅一条 tweet 来判断是否是 bot。

论文将 bot 发现的方法分为用户级和 tweet 级，进行分类；tweet 级根据

**数据集** 论文使用的数据集存在不平衡问题，作者分别使用 SMOTE+ENN 和 SOMTE+TOMEK 的方法平衡数据。数据分为用户级和 tweet 级两类，用户级数据（用户元数据）包括 statuses count、followers count 等，tweet 级数据包括 retweet count、number of hashtags 等。

## 4 损失函数

### 4.1 softmax

**下溢 (underflow)** 当接近零的数被四舍五入为零后发生下溢。

**上溢 (overflow)** 当大数量级的数被近似为  $\infty$  或  $-\infty$  时，发生上溢。

**softmax 函数** 可对下溢和上溢进行数值稳定，定义为：

$$\text{softmax}(\mathbf{x})_i = \frac{\exp(x_i)}{\sum_{j=1}^n \exp(x_j)} \quad (7)$$

这个式子同样会产生溢出，例如，当  $\mathbf{x}$  是很小的负数时，分母会变成零；当  $\mathbf{x}$  是很大的数时，一样会发送上溢。这个问题可以通过计算  $\text{softmax}(\mathbf{z}), \mathbf{z} = \mathbf{x} - \max_i x_i$  同时解决。

## 5 A Discriminative Feature Learning Approach for Deep Face Recognition [3]

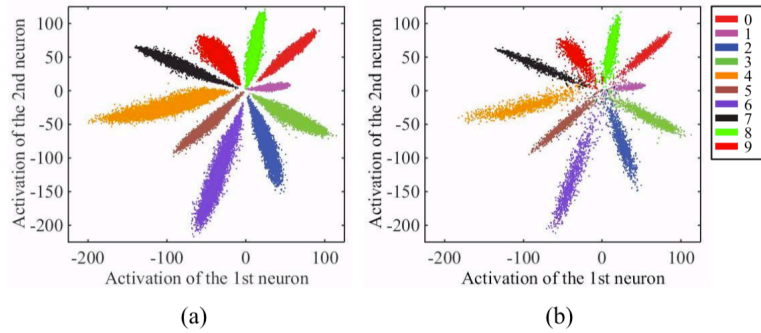
### 5.1 Introduction

**动机** 一般的物体识别，主要是闭集识别，利用 softmax loss 即可处理，但人脸识别不单需要特征可分离 (separable)，也需要可区分 (discriminative)，构建一个高效的 loss 可提升可区别性，为此，本文提出 center loss。

**Center loss** 为每一个 class 的特征维护一个 center，每次进行 SGD 的时候同时更新 center，并最小化 class 中样本特征到中心的距离。

### 5.2 The Proposed Approach

**toy example** 本文使用一个 toy example 来演示算法的效果。它最后一个隐层的维度是 2，以便于我们使用二维图像来显示特征分布。toy example 使用 softmax loss，最后输出的特征的分布表示为图3：



**Fig. 2.** The distribution of deeply learned features in (a) training set (b) testing set, both under the supervision of softmax loss, where we use 50K/10K train/test splits. The points with different colors denote features from different classes. **Best viewed in color.**

图 2:

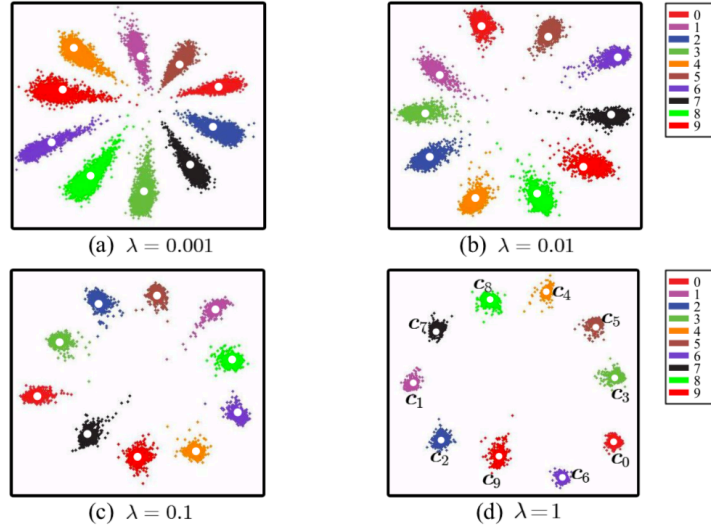
**Center loss** 为增强可区分度，需在保持特征分类的同时，最小化内部 class 的方差。center loss 定义为：

$$\mathcal{L}_C = \frac{1}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2 \quad (8)$$

理论上,  $c_{y_j}$  应当随着特征的变化而更新, 因此, 在每次更新更新参数的时候都要对它进行更新。

综上, 总的损失函数可以表示为:

$$\mathcal{L} = \mathcal{L}_S + \lambda \mathcal{L}_C \quad (9)$$



**Fig. 3.** The distribution of deeply learned features under the joint supervision of softmax loss and center loss. The points with different colors denote features from different classes. Different  $\lambda$  lead to different deep feature distributions ( $\alpha = 0.5$ ). The white dots ( $c_0, c_1, \dots, c_9$ ) denote 10 class centers of deep features. **Best viewed in color.**

图 3:

## 参考文献

- [1] 周志华, 机器学习, Qing hua da xue chu ban she, 2016. 1
- [2] Sneha Kudugunta and Emilio Ferrara, “Deep neural networks for bot detection”, arXiv preprint arXiv:1802.04289, 2018. 4
- [3] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, “A discriminative feature learning approach for deep face recognition”, in European conference on computer vision. Springer, 2016, pp. 499–515. 6