Pattern Recognition

第六次作业

牛李金梁
201928014629008

2019 年 12 月 18 日

第一部分：问题、计算与证明

1 Adaboost 算法的设计思想

寻找弱分类器要比寻找精确的分类规则要简单得多。Adaboost 通过改变训练样本的权重反复学习得到一系列弱分类器，并将这些弱分类器进行组合来作为一个强分类器。

在反复学习中，提高那些前一轮弱分类器分错的样本的权重，降低已被正确分类的样本的权重，则错分的样本将在下一轮弱分类器中得到更多关注。

弱分类器的组合则采用弱分类器加权表决的方法，加大分类错误率较小的弱分类器的权重，使其在表决中起更大的作用。

2 模型选择的基本原则

对于在训练集上表现同样良好的 2 个分类器，更倾向于简单的那个。

- **没有免费的午餐定理** 对寻找代价函数极值的算法，在平均到所有可能的代价函数上时其表现恰好相同。不存在一个与具体应用无关的、普遍适用的最优分类器。算法必须要引入一些与问题领域有关的假设。要想在某些指标上得到性能的提高，必须在另一些指标上付出相应的代价。
- **丑小鸭定理** 不存在分类的客观标准, 一切分类的标准都是主观的。不存在与问题无关的最优的特征/属性集合。不存在与问题无关的模式之间的相似性度量。
- **Occam 剃刀原理**如无必要，勿增实体。如果对训练数据分类的效果相同，简单的分类器往往优于复杂的分类器。相比复杂的假设，我们更倾向于选择简单的、参数少的假设。
- **最小描述长度原理**我们必须使模型的算法复杂度以及与该模型相适应的训练数据的描述长度之和最小。即应该选择尽可能简单的分类器或模型。

3 分类器集成的基本方法

常用技术手段有处理训练数据、特征、类别标号，改进学习方法。分类器集成算法可以按照基本分类器类型是否相同分为异态集成和同态集成。典型的异态集成为层叠泛化，把前一层的输出作为这一层的输入。集成算法也可以按照训练数据处理方式分为 Bagging,Random subspace, Boosting/adaboost, 随机森林。

- **Bagging** 训练一组基分类器，每个基分类器通过一个 boostrap 训练样本集来训练，获得基本分类器之后，bagging 通过投票进行统计，被投票最多的类则确定为预测类。
- **Random Subspace** 对每一个分类器，选择部分子特征来构建一个训练集合，同时学习一个分类器。对于新样本，通过多数投票法来预测其类别。
- **Adaboost** 从弱学习算法出发，反复学习，得到一系列弱分类器；然后组合这些弱分类器，构成一个强分类器。每一轮中提高被前一轮弱分类器分错的样本的权重，降低已经被正确分类的样本的权重，错分的样本将在下一轮弱分类器中得到更多的关注。

4 Hard-Margin SVM 的优化目标

线性 SVM 判别函数形式为：

$$f\left(\boldsymbol{x},\boldsymbol{\omega},b\right)=sign\left(\boldsymbol{\omega}^{T}\boldsymbol{x}+b\right)$$

任一数据点到该判别面的距离为：

$$d(\boldsymbol{x}) = \frac{\left|\boldsymbol{\omega}^T\boldsymbol{x} + b\right|}{\sqrt{\|\boldsymbol{x}\|_2^2}} = \frac{\left|\boldsymbol{\omega}^T\boldsymbol{x} + b\right|}{\sqrt{\sum_{i=1}^{d}\boldsymbol{\omega}_i^2}}$$

margin 是数据点到判别面的最短距离，即，

$$margin = \arg\min d(\boldsymbol{x}) = \arg\min \frac{\left|\boldsymbol{\omega}^T\boldsymbol{x} + b\right|}{\sqrt{\sum_{i=1}^{d}\boldsymbol{\omega}_i^2}}$$

而 SVM 的训练目标是最大化 margin，于是，

$$\begin{aligned}&\arg\max_{\boldsymbol{\omega},b} \arg\min_{x_i \in D} \frac{\left|\boldsymbol{\omega}^T\boldsymbol{x}+b\right|}{\sqrt{\sum_{i=1}^{d}\boldsymbol{\omega}_i^2}}\\ &s.t. \forall x_i \in D, y_i\left(\boldsymbol{\omega}^T\boldsymbol{x} + b\right) \geq 0\end{aligned}$$

加强 margin 约束最小为 1，则该优化可以转化为，

$$\begin{aligned}&\arg\min_{\boldsymbol{\omega},b} \sqrt{\sum_{i=1}^{d}\boldsymbol{\omega}_i^2}\\ &s.t. \forall x_i \in D, y_i\left(\boldsymbol{\omega}^T\boldsymbol{x} + b\right) \geq 1\end{aligned}$$

5 Hinge Loss 在 SVM 中的意义

Hinge Loss 为：

$$\arg\min_{f \in H} \frac{1}{n}\sum_{i=1}^{n}\left(1 - y_i f(x_i)\right)_+ + \lambda\|f\|_H^2$$

对于线性不可分的两类数据，需要使用 soft-margin SVM。也就是在对约束条件进行一个松弛，即 $\forall x_i \in D, y_i\left(\boldsymbol{\omega}^T\boldsymbol{x} + b\right) \geq 1 - \epsilon$，且要在目标函数中最小化 $\sum_{i=1}^{n}\varepsilon_i$。但对于 margin 之外的数据点，进行松弛会使目标函数趋于负无穷，所以要对 ϵ 进行限制，使得对于 margin 外的数据 loss 为 0，margin 内的数据满足松弛条件。

第二部分：计算机编程

运行环境：python3.7.5

依赖库：scikit-learn numpy

实验中选取 3 和 8 两类数字进行分类，并按 4：6 的比例划分测试集与训练集。之后使用 scikit-learn 中的 svm 进行训练，参数按照默认参数，即使用 RBF 核，$\gamma = 1/n\_features$。

得到的实验结果如图 1 显示：

	precision	recall	f1-score	support
3	0.99	0.99	0.99	4218
8	0.99	0.99	0.99	4162
accuracy			0.99	8380
macro avg	0.99	0.99	0.99	8380
weighted avg	0.99	0.99	0.99	8380

图 1: SVM 运行结果

准确率达到了 0.99。