机器学习导论
习题六

141210016, 刘冰楠, bingnliu@outlook.com

2017 年 6 月 9 日

1 [20pts] Ensemble Methods

(1) **[10pts]** 试说明 Boosting 的核心思想是什么，Boosting 中什么操作使得基分类器具备多样性？

(2) **[10pts]** 试析随机森林为何比决策树 Bagging 集成的训练速度更快。

Solution.

1.1 Problem (1)

通过对基学期器进行迭代训练，在每轮训练中，通过调整训练样本分布使得分类错误的样本后续收到更多关注，从而增加基学习期的多样性，最后预测使用所有基学习器加权结合的结果。

使得基学习器具备多样性的操作是权重调整。对样本权重调整后，新一轮的训练相当于基于不同的样本，从而训练出不同于之前的学习器。

1.2 Problem (2)

随机森林相对于 Baggin 决策树的关键区别在于，在选择划分属性时，首先随机选择一个属性集的子集，再在这个子集中寻找最优属性。

由于一般随机选择的属性子集规模比所有属性集小 (如推荐值 $k = \log_2 d$)，训练时只需考察这个较小的子集，从而训练速度更快。

2 [20pts] Bagging

考虑一个回归学习任务 $f : \mathbb{R}^d \to \mathbb{R}$。假设我们已经学得 M 个学习器 $\hat{f}_1(\mathbf{x}), \hat{f}_2(\mathbf{x}), \ldots, \hat{f}_M(\mathbf{x})$。我们可以将学习器的预测值看作真实值项加上误差项

$$\hat{f}_m(\mathbf{x}) = f(\mathbf{x}) + \epsilon_m(\mathbf{x}) \tag{2.1}$$

每个学习器的期望平方误差为 $\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})^2]$。所有的学习器的期望平方误差的平均值为

$$E_{av} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})^2] \tag{2.2}$$

M 个学习器得到的 Bagging 模型为

$$\hat{f}_{bag}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\hat{f}_m(\mathbf{x}) \tag{2.3}$$

Bagging 模型的误差为

$$\epsilon_{bag}(\mathbf{x}) = \hat{f}_{bag}(\mathbf{x}) - f(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\epsilon_m(\mathbf{x}) \tag{2.4}$$

其期望平均误差为

$$E_{bag} = \mathbb{E}_{\mathbf{x}}[\epsilon_{bag}(\mathbf{x})^2] \tag{2.5}$$

(1) **[10pts]** 假设 $\forall\, m \neq l$，$\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})] = 0$，$\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})\epsilon_l(\mathbf{x})] = 0$。证明

$$E_{bag} = \frac{1}{M}E_{av} \tag{2.6}$$

(2) **[10pts]** 试证明不需对 $\epsilon_m(\mathbf{x})$ 做任何假设，$E_{bag} \leq E_{av}$ 始终成立。（提示：使用 Jensen's inequality）

Proof.

2.1 Problem (1)

$$\begin{aligned}
E_{bag} =& \mathbb{E}_{\mathbf{x}}[\epsilon_{bag}(\mathbf{x})^2] \\
=& \mathbb{E}_{\mathbf{x}}\left[\frac{1}{M}\sum_{m=1}^{M}\epsilon_m(\mathbf{x})\right]^2 && \text{(definition)} \\
=& \mathbb{E}_{\mathbf{x}}\left[\frac{1}{M^2}\sum_{m=1}^{M}\sum_{n=1}^{M}\epsilon_m(\mathbf{x})\epsilon_n(\mathbf{x})\right] && \text{(expansion of square)} \\
=& \frac{1}{M^2}\sum_{m=1}^{M}\sum_{n=1}^{M}\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\epsilon_n(\mathbf{x})\right] && \text{(linearity of expectation)} \\
=& \frac{1}{M^2}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})^2] + \frac{1}{M^2}\sum_{1\leq m\neq n\leq M}\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})\epsilon_n(\mathbf{x})] && \text{(separation)} \\
=& \frac{1}{M^2}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}[\epsilon_m(\mathbf{x})^2] && \text{(hypothesis)} \\
=& \frac{1}{M}E_{av}. && \text{(definition)}
\end{aligned} \tag{2.7}$$

□

2.2 Problem (2)

Jensen's Inequality states that:
For a real convex function φ, numbers $x_1, x_2, \ldots, x_n$ in its domain, and positive weights a_i,

$$\varphi\left(\frac{\sum a_i x_i}{\sum a_i}\right) \le \frac{\sum a_i \varphi(x_i)}{\sum a_i}. \tag{2.8}$$

And if a_i is already normalized (sum is 1), then Eq.(2.8) is:

$$\varphi\left(\sum_i a_i x_i\right) \le \sum_i a_i \varphi(x_i). \tag{2.9}$$

We already know that

$$E_{bag} = \mathbb{E}_{\mathbf{x}}\left[\frac{1}{M}\sum_{m=1}^{M}\epsilon_m(\mathbf{x})\right]^2 = \mathbb{E}_{\mathbf{x}}\left[\sum_{m=1}^{M}\frac{1}{M}\epsilon_m(\mathbf{x})\right]^2. \tag{2.10}$$

In Eq.(2.9), let $\varphi(x) = x^2$, $a_i = 1/M$ and $x_i = \epsilon_m(\mathbf{x})$, then

$$\left[\sum_{m=1}^{M}\frac{1}{M}\epsilon_m(\mathbf{x})\right]^2 \le \sum_{m=1}^{M}\frac{1}{M}\epsilon_m(\mathbf{x})^2. \tag{2.11}$$

Since Eq.(2.11) is true for any $\mathbf{x} \in D$, from the *monotonicity* of expectations, we have

$$\mathbb{E}_{\mathbf{x}}\left[\sum_{m=1}^{M}\frac{1}{M}\epsilon_m(\mathbf{x})\right]^2 \le \mathbb{E}_{\mathbf{x}}\left[\sum_{m=1}^{M}\frac{1}{M}\epsilon_m(\mathbf{x})^2\right], \tag{2.12}$$

i.e. $E_{bag} \le E_{av}$.

□

3 [30pts] AdaBoost in Practice

(1) **[25pts]** 请实现以 Logistic Regression 为基分类器的 AdaBoost，观察不同数量的 ensemble 带来的影响。详细编程题指南请参见链接:`http://lamda.nju.edu.cn/ml2017/PS6/ML6_programming.html`

(2) **[5pts]** 在完成上述实践任务之后，你对 AdaBoost 算法有什么新的认识吗？请简要谈谈。

Solution.

3.1 不同数量 ensemble 带来的影响

随着 ensemble 数量的增加，基分类器的多样性增加，从而集成分类器的精度提高。对于较简单的分类器，可能较少数量的基分类器就能做到非常高的精度（甚至 100%），从而集成分类器的精度难以提高。

3.2 你对 AdaBoost 算法有什么新的认识吗

AdaBoost 算法需要和基分类器配合好使用，基分类器如果太强，则：1 不需要 adaboost 2 由于权重不变，没法通过 adaboost 提升 (当然一般问题不会 e=0，也就不会权重不变)

一个原本没注意到的问题：迭代中，错误率是在新分布上的错误率 (带权重)，如果写错了，ensemble 就不会有效。

3.3 为什么不同的基分类器参数设置会带来很不一样的效果

首先基分类器不同的参数会影响训练时间，进而影响集成学习的训练时间。

更重要的是，基分类器参数设置会影响基分类器的强弱，一般来说较强的分类器只需要较少的轮数就能达到较好的集成效果，继续提升基分类器数目，反而可能导致过拟合，从而导致集成分类器性能下降；而较弱的分类器则能“不断进步”。一般来说较强的基分类器会有更好的集成性能，然而也有例外，比如本数据集上，选择略大的 Logistic 回归惩罚系数 (从而较弱)，反而在集成性能上能超越最好的基分类器 (这里最好指基分类器回归惩罚系数较小从而精度较高)。