中国科学院大学

University of Chinese Academy of Sciences

# 模式识别与机器学习

## 081203M04004H

# Chap 11 课程作业解答

2022 年 12 月 26 号

*Professor:* 黄庆明


**学生: 周胤昌**
**学号: 202228018670052**
**学院: 网络空间安全学院**
**所属专业: 网络空间安全**
**方向: 安全协议理论与技术**

# Problem 1

模型复杂度过低/过高通常会导致 Bias 和 Variance 有怎样的问题?

**Solution:** 通常, 模型复杂度过低会导致偏差高且方差低 (即欠拟合), 而复杂度过高则会导致偏差小但方差大 (即过拟合).

# Problem 2

怎样判断且缓解过拟合/欠拟合问题?

**Solution:** 主要是通过验证集上的误差来判断. 当校验误差一直减小时, 则说明模型目前处于欠拟合; 当校验误差先减小而后增大时, 则说明模型目前处于过拟合状态. 当模型处于欠拟合状态时, 需要增加模型复杂度, 具体措施有:

- 增加模型的迭代次数;
- 增加更多特征;
- 降低模型正则化水平.

当模型处于过拟合状态时, 需要降低模型复杂度, 具体措施有:

- 及早停止迭代;
- 减少特征数量;
- 提高模型正则化水平;
- 扩大训练集.

# Problem 3

比较 Bagging 和 Boosting 算法的异同.

**Solution:** Bagging 和 Boosting 算法的异同:

- **不同点:** 这两类算法的不同点在于前者是对训练集做 $m$ 次有放回随机抽样来得到 $m$ 个子训练集, 从而分别**并行学习**得到 $m$ 个基模型. 而后者的 $m$ 个弱学习器是**按顺序进行学习**的, 并且有 $m$ 次的训练集转化.
- **相同点:** 相同点在于两者都分别利用 $m$ 个弱模型做出 $m$ 个预测, 并最终进行预测结果的整合.

# Problem 4

简述 Adaboosting 的流程.

**Solution:**Adaboosting 的流程如下算法1中所示:

**Algorithm 1** AdaBoost 算法流程

**Input:** 给定训练集 $(x_1, y_1), (x_2, y_2) \cdots, (x_m, y_m)$, 其中 $x_i \in X, y_i \in \{-1, +1\}$

**Output:** 强分类器 $H_{\text{final}}(x)$

初始化 $D_1(i) = 1/m, \forall i \in \{1, 2, \cdots, m\}$;

**for** $t = 1, 2, \cdots, T$ **do**

训练有误差的弱分类器 $h_t : X \to \{-1, +1\}$;

$\epsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i] < \frac{1}{2}$; ▷ 如果 error= $\frac{1}{2}$, 则学习器 $h_1$ 在训练集 $D_2$ 上的性能为随机猜测

$\alpha_t = 1/2 \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right) > 0$; ▷ 如果 error 越小, 则 $\alpha_t$ 越大

$\forall i \in \{1, 2, \cdots, m\}$, 做如下更新: ▷ 其中 $Z_t$ 为正则化因子

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t}\exp(-\alpha_t y_i h_t(x_i)) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t}, \text{若} y_i = h_t(x_i) \\ e^{\alpha_t}, \text{若} y_i \neq h_t(x_i) \end{cases}$$

**end for**

**return** 强分类器 $H_{\text{final}}(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$; ▷ 强分类器的权重较大

# Problem 5

随机森林更适合采用哪种决策树?

(A) .性能好, 深度较深 (B) .性能弱, 深度较浅

**Solution:** 选择 (A), 随机森林属于 Bagging 集成算法, 因为 Bagging 更适合对偏差低、方差高 (即过拟合) 的模型进行融合, 所以随机森林更适合性能好、深度较深 (即过拟合) 的决策树.

# Problem 6

基于树的 Boosting 更适合采用哪种决策树?

(A) .性能好, 深度较深 (B) .性能弱, 深度较浅

**Solution:** 选择 (B), 因为 Boosting 的基本思想是将弱学习器组合成强学习器. 故而基于树的 Boosting 更适合采用复杂度低的决策树, 即层数不深、性能弱的决策树.

# Problem 7

如果对决策树模型采用 Bagging 方式进行集成学习, 更适合采用哪种方法对决策树的超参 (比如树的深度) 进行调优?

(A) .交叉验证 (B) .包外估计

**Solution:** 选择 (B), 在 Bagging 中, 每个弱学习器只在原数据集的一部分上进行训练, 因此可以不用交叉验证而直接采用包外估计来进行超参调优.