

机器学习导论第三次作业

韦俊林 (201928016029023)

2020 年 5 月 11 日

简述题

- (1) 试析随机森林为何比决策树 Bagging 集成的训练速度更快。

答：传统决策树在选择划分属性时，在当前节点的属性集合中选择一个最有属性。而随机森林对基决策树的每个结点，先从该节点的属性集合中随机选择一个包含 k 个属性的自己，然后从该子集中选择一个最优属性用于划分。具体来说就是 Bagging 需要考察节点的所有属性，而随机森林只需随机考察一个属性子集。

- (2) 试比较 Gradient Boosting 与 AdaBoost 的异同。

答：首先知道 Boosting 是一族可将弱学习器提升为强学习期的算法，AdaBoost 算法就是其最具代表性的算法。AdaBoost 算法基本流程是，先从初始训练集训练出一个基学习器，再根据基学习器的表现对训练样本分布进行调整，使得先前基学习器做错的训练样本再后续收到更多关注，然后基于调整后的样本分布来训练下一个基学习器，就这样重复 M 次得到 M 个基学习器，最后将其加权结合。

AdaBoost 算法与 Gradient Boosting 不同在于前者只能针对指数型损失函数，对于一般的损失函数不适用。而 Gradient Boosting 算法可适用于任何可微分的损失函数，可看作 AdaBoost 算法的推广。具体思想是通过在函数空间，沿负梯度方向，迭代选取弱学习器来优化损失函数。

- (3) 试比较包裹式选择、过滤式选择与嵌入式选择的异同。

答：

包裹式选择：直接把最终将要使用的学习器的性能作为特征子集的评价准则，目的是为给定学习期选择最有利于其性能的特征子集。一般来说都能取得很好的效果，但计算量很大，因为需要频繁调用学习算法进行候选特征子集的评价。

过滤式选择：先对数据集进行特征选择，然后再训练学习器。因此特征选择过程与后续学习器无关。基本过程是，首先定义一个评价函数来度量某个给定特征与类别标签之间的相关度，最后选择具有最大相关度的 k 个特征作为选择结果，因此核心在于评价函数的定义，与包裹式选择方法相比，计算量降低了很多。

嵌入式选择：将分类器学习与特征选择融为一体，分类器训练过程自动完成了自动选择。基本思想是在学习 ω 的时候，对 ω 进行限制，使得 ω 不仅能满足训练样本误差的要求，同时使得 ω 中非零元素尽可能少。具体来说就是加入各种正则化范数。与上述两个特征选择方法另一个不同在于不能直接设置最终选择特征的个数 k ，只能通过设置正则化系数 λ 来隐式的控制 k 。

- (4) 试描述直接求解 L_0 范数正则化会遇到的困难。

答： L_0 正则化的值是模型中非零参数的个数，因此用 L_0 正则化是可以进行特征选择，防止过拟

合。但 L_0 范数不连续，因此不可导。并且非凸，所以凸函数的求解方法对它也使用，因此 L_0 范数是个 NP 难的问题。

(5) 试描述为什么基于 L_1 范数可以进行特征选择。

答： L_1 正则化能产生稀疏性，把不重要的特征置为 0，这样就将多余无用的特征给去掉，从而实现特征选择。从图形上看，实际杀昂是要在正则化的等值线与损失函数的等值线中寻找一个交点，使得两者的和最小。例如在二维空间中 L_1 正则化的等值线是一组菱形，这些交点容易落在坐标轴上。因此，另一个参数的值在这个点上就是 0，从而实现了稀疏化。

(6) 试比较 K-SVD 与 K-means 方法的异同。

答： K-SVD 算法是 K-means 聚类算法的推广形式，K-means 算法输出类似于 one-hot 形式，每个样本输出向量中只有一项不为 0，其余都为 0，即每个样本只有一个原子逼近，而 K-SVD 数据点的稀疏编码有多个非零点。具体的目标如下：

K-means:

$$\min_{D,X} \{ \|Y - DX\|_F^2 \} \quad \text{subject to } \forall i, x_i = e_k \text{ for some } k$$

K-SVD:

$$\min_{D,X} \{ \|Y - DX\|_F^2 \} \quad \text{subject to } \forall i, \|x_i\|_0 \leq T_0$$