模式识别第 9 次作业

宁雪妃
2016310503

Problem 1

Gradient boosting decision tree 是用 gradient boosting 方法 ensemble 多个弱决策树。Boosting 类方法都是顺序(sequential)的 stagewise 的在不断改变的 fit 目标(weighted data(the same as fitting a exponential loss residual) or loss residual)上 fit 多个弱分类器，并且把这些分类器的结果做一个加性的 ensemble(在分类的时候可以是投票，也可以有分数定义其他的 loss)。

在 Gradient boosting decision tree 模型中：

- 弱分类器: decision tree，为了防止过拟合，通常每个弱分类器 decision tree 的深度会比较小
- 弱分类器 fit 方法：把已有的 N 个分类器得到的 prediction add 起来之后，与实际训练 label 有一个 loss 的定义，loss 在当前 predict 值处的 gradient 乘上一个较小的 learning rate 称为 pseudo residual, 通过这个 pseudo residual 用 recursive partition 等方法 grow 好 decision tree 的结构。然后用简单的方法可以对每个 leaf node 确立的 region 找到 optimal prediction constant。其中 learning rate 是对 negative gradient 做 shrink，防止 overfit。这里之所以要分两步，并且第一步用了 gradient 来 grow tree 划分出 region，是因为对于 general loss function 形式, jointly 优化 region 和 constant 最小化 add 之后的 loss 是很难的。

GBDT 是一种利用了决策树 fit 算法的 stage-wise additive 学习算法，其中利用了一般性的 loss function 的 gradient 来指导每个弱分类器决策树的 growing。

Problem 2

Decision tree 实现见 decision_tree.py, 调用见 run.py。采用的不纯度度量为 entropy, 支持设置 min_samples_split 和 max_depth。

结果：

随机将数据按照 9:1 划分为了 train 和 val 数据。试过几次不同划分，min_samples_split=30 的训练集精度基本在 82% ~ 86%范围内。

下表为固定划分(随即种子), 设置不同的 min_sample_split(不限制 max_depth)的几次实验结果，其中 min_samples_split 代表如果一个节点包含的训练数据小于该值就直接作为叶子节点，Node number 一列为得到的 decision tree 的节点个数，Max depth 一列为得到的树的深度，Train accuracy 为训练集精度，Val accuracy 为测试集精度。

min_samples_split	Node number	Max depth	Train accuracy	Val accuracy
40	953	35	0.815277777778	0.719444444444
30	1085	38	0.827083333333	0.720138888889

20	1305	40	0.846759259259	0.723611111111
10	1719	42	0.879398148148	0.715972222222

可以看到 decision tree 随着 min_samples_split 减小和深度的增加，对 train 集的拟合会越来越好，但是会 overfit，比如 min_samples_split＝10 已经有明显的过拟合了。