树模型

`ID3` `C4.5` `CART` `随机森林` `GBDT` `信息增益` `信息增益率` `基尼指数`

前置

GBM: Gradient Boosting Machine。
Categorical: 无序类别
GBRT: Gradient Boosted Regression Tree
GBDT: Gradient Boosted Decision Tree
MART: Multi Additive Regression Tree

决策树算法是一种比较直观和容易理解的算法，决策树模型类似我们平常在编写代码时写的判断逻辑，根据判断条件产生不同的分支。

ID3算法

节点分裂依据: 信息增益，特征只能是离散特征

C4.5算法

节点分类依据: 信息增益率，特征同样只能是离散特征

CART树

节点分类依据: 分类问题使用基尼指数、回归问题则使用最小平方误差

随机森林

树模型的Ensemble方法: Bagging

GBDT

树模型的Ensemble方法: Boosting

GBDT部分，可以参考腾讯Yafei Zhang发布的GBDT相关的两个PDF文档，写的非常简洁易懂了，所以这里不再展开描述。下面贴文档内的一个图看一下GBDT和LR的比较:

	GBDT	LR
linearity	nonlinear	linear
fitting capability	good	less good
feature selection	naturally	only l1-reg
feature combination	naturally	no
regression	yes	no
classification	yes	yes
multi-class classification	support	support
output probability	support	yes
parallelization	hard and less beneficial	easy

LR与GBDT对比的复杂度

	GBDT	LR
speed	slow	fast
memory(training)	$O(N) + O(K\bar{J})$	$O(N) + O(M)$
cpu(training)	$O(K\bar{J}MN \log N)$	$O(KN)$
memory(predicting)	$O(K\bar{J})$	$O(M)$
cpu(predicting)	$O(K\bar{J})$	$O(M)$
N	large	huge
M	medium	huge

K在GBDT中代表K棵决策树，在LR中代表K轮迭代

N代表训练样本个数

M代表特征个数

J代表一棵树中叶子的个数

从我们的一些实际经验，GBDT所需的样本数其实并不是太多，基本百万级别就够了。

Yafei Zhang的文档: 重点推荐，可以说写的非常简洁易懂了。

gbdt.pdf

1.1 MB

gbdt-in-detail1.pdf

371.1 KB