说说随机森林

https://zhuanlan.zhihu.com/p/22097796

None Wed Aug, 12 22:07

作者：郭瑞东

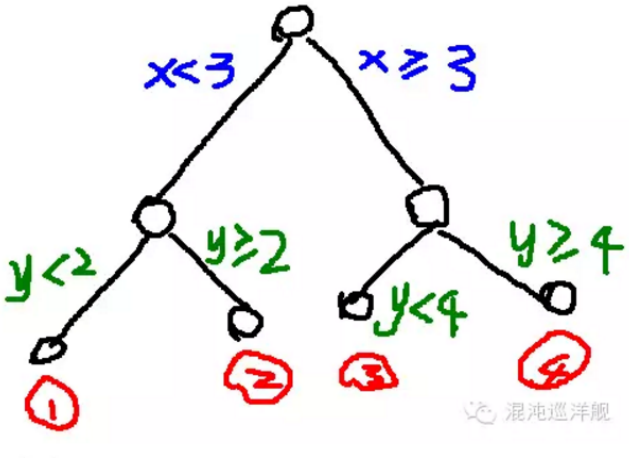
随机森林是机器学习中的一种常用方法，而随机森林背后的思想，更是与群体智慧，甚至“看不见的手”相互映照。

上世纪八十年代Breiman等人发明分类树的算法（Breiman et al. 1984），通过反复二分数据进行分类或回归，计算量大大降低。2001年Breiman把分类树组合成随机森林（Breiman 2001a），即在变量（列）的使用和数据（行）的使用上进行随机化，生成很多分类树，再汇总分类树的结果。随机森林在运算量没有显著提高的前提下提高了预测精度。随机森林对多元公线性不敏感，结果对缺失数据和非平衡的数据比较稳健，可以很好地预测多达几千个解释变量的作用（Breiman 2001b），被誉为当前最好的算法之一（Iverson et al. 2008）。

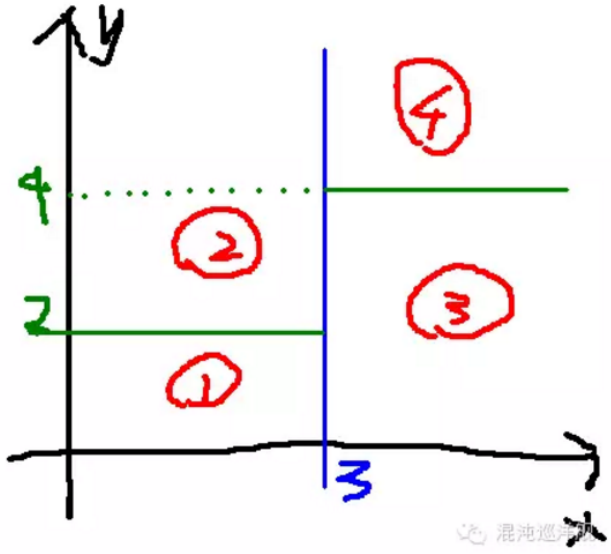
随机森林顾名思义，是用随机的方式建立一个森林，森林里面有很多的决策树组成，随机森林的每一棵决策树之间是没有关联的。在得到森林之后，当有一个新的输入样本进入的时候，就让森林中的每一棵决策树分别进行一下判断，看看这个样本应该属于哪一类（对于分类算法），然后看看哪一类被选择最多，就预测这个样本为那一类。随机森林可以既可以处理属性为离散值的量，比如ID3算法，也可以处理属性为连续值的量，比如C4.5算法。另外，随机森林还可以用来进行无监督学习聚类和异常点检测。

决策树（decision tree）是一个树结构（可以是二叉树或非二叉树）。其每个非叶节点表示一个特征属性上的测试，每个分支代表这个特征属性在某个值域上的输出，而每个叶节点存放一个类别。使用决策树进行决策的过程就是从根节点开始，测试待分类项中相应的特征属性，并按照其值选择输出分支，直到到达叶子节点，将叶子节点存放的类别作为决策结果。

随机森林由决策树组成，决策树实际上是将空间用超平面进行划分的一种方法，每次分割的时候，都将当前的空间一分为二，比如说下面的决策树（其属性的值都是连续的实数）：



这一颗树将样本空间划分为成的样子为：



下面是随机森林的构造过程：

1. 假如有N个样本，则有放回的随机选择N个样本(每次随机选择一个样本，然后返回继续选择)。这选择好了的N个样本用来训练一个决策树，作为决策树根节点处的样本。

2. 当每个样本有M个属性时，在决策树的每个节点需要分裂时，随机从这M个属性中选取出m个属性，满足条件m << M。然后从这m个属性中采用某种策略（比如说信息增益）来选择1个属性作为该节点的分裂属性。

3. 决策树形成过程中每个节点都要按照步骤2来分裂（很容易理解，如果下一次该节点选出来的那一个属性是刚刚其父节点分裂时用过的属性，则该节点已经达到了叶子节点，无须继续分裂了）。一直到不能够再分裂为止。注意整个决策树形成过程中没有进行剪枝。

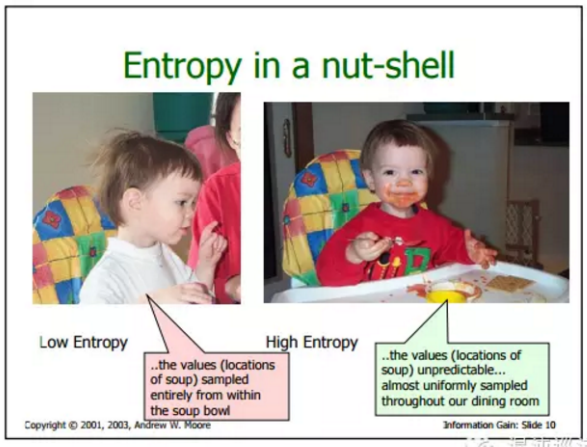
4. 按照步骤1~3建立大量的决策树，这样就构成了随机森林了。

在建立每一棵决策树的过程中，有两点需要注意采样与完全分裂。

首先是两个随机采样的过程，random forest对输入的数据要进行行、列的采样。对于行采样，采用有放回的方式，也就是在采样得到的样本集合中，可能有重复的样本。假设输入样本为N个，那么采样的样本也为N个。这样使得在训练的时候，每一棵树的输入样本都不是全部的样本，使得相对不容易出现over-fitting。然后进行列采样，从M个feature中，选择m个（m << M）。

之后就是对采样之后的数据使用完全分裂的方式建立出决策树，这样决策树的某一个叶子节点要么是无法继续分裂的，要么里面的所有样本的都是指向的同一个分类。一般很多的决策树算法都一个重要的步骤——剪枝，但是这里不这样干，由于之前的两个随机采样的过程保证了随机性，所以就算不剪枝，也不会出现over-fitting。

通过分类，子集合的熵要小于未分类前的状态，这就带来了信息增益（information gain）



决策树有很多的优点：

a. 在数据集上表现良好，两个随机性的引入，**使得随机森林不容易陷入过拟合**

b. 在当前的很多数据集上，相对其他算法有着很大的优势，两个随机性的引入，使得随机森林具有**很好的抗噪声能力**

c. 它能够**处理很高维度（feature很多）的数据，并且不用做特征选择，**对数据集的适应能力强：既能处理离散型数据，也能处理连续型数据，数据集无需规范化

d. 可生成一个Proximities=（pij）矩阵，用于**度量样本之间的相似性**： pij=aij/N, aij表示样本i和j出现在随机森林中同一个叶子结点的次数，N随机森林中树的颗数

e. 在创建随机森林的时候，对generlization error使用的是**无偏估计**

f. **训练速度快**，可以得到变量重要性排序（两种：基于OOB误分率的增加量和基于分裂时的GINI下降量

g. 在训练过程中，**能够检测到feature间的互相影响**

h. **容易做成并行化方法**

i. 实现比较简单

随机森林主要应用于回归和分类。本文主要探讨基于随机森林的分类问题。随机森林和使用决策树作为基本分类器的（bagging）有些类似。以决策树为基本模型的bagging在每次bootstrap放回抽样之后，产生一棵决策树，抽多少样本就生成多少棵树，在生成这些树的时候没有进行更多的干预。而随机森林也是进行bootstrap抽样，但它与bagging的区别是：在生成每棵树的时候，每个节点变量都仅仅在随机选出的少数变量中产生。因此，不但样本是随机的，连每个节点变量（Features）的产生都是随机的。

许多研究表明，组合分类器比单一分类器的分类效果好，**随机森林（random forest）是一种利用多个分类树对数据进行判别与分类的方法，它在对数据进行分类的同时，还可以给出各个变量（基因）的重要性评分，评估各个变量在分类中所起的作用。**

随机森林算法得到的随机森林中的每一棵都是很弱的，但是大家组合起来就很厉害了。我觉得可以这样比喻随机森林算法：每一棵决策树就是一个精通于某一个窄领域的专家（因为我们从M个feature中选择m让每一棵决策树进行学习），这样在随机森林中就有了很多个精通不同领域的专家，对一个新的问题（新的输入数据），可以用不同的角度去看待它，最终由各个专家，投票得到结果。而这正是群体智慧（swarm intelligence），经济学上说的看不见的手，也是这样一个分布式的分类系统，由每一自己子领域里的专家，利用自己独有的默会知识，去对一项产品进行分类，决定是否需要生产。随机森林的效果取决于多个分类树要相互独立，要想经济持续发展，不出现overfiting（就是由政府主导的经济增长，但在遇到新情况后产生泡沫），我们就需要要企业独立发展，独立选取自己的feature 。

本文首发于微信公众号混沌巡洋舰（chaoscruiser）

欢迎关注混沌巡洋舰，追寻自然界复杂下的简单，带你学习各路跨界干货。