GBDT 如何用于分类问题

阿泽的学习笔记 3天前

> 因为用树模型太习以为常了，以至于看到这个标题很容易觉得这很显然。但越简单的东西越容易出现知识盲区，仔细想一下好像确实有点疑问：GBDT 用的是回归树，是如何做的分类呢？
>
> - 作者：1直在路上1
> - https://www.cnblogs.com/always-fight/p/9400346.html

一 简介

GBDT 在传统机器学习算法里面是对真实分布拟合的最好的几种算法之一，在前几年深度学习还没有大行其道之前，GBDT 在各种竞赛是大放异彩。原因大概有几个

- 效果确实挺不错；
- 既可以用于分类也可以用于回归；
- 可以筛选特征。

这三点实在是太吸引人了，导致在面试的时候大家也非常喜欢问这个算法。

GBDT 是通过采用加法模型（即基函数的线性组合），以及不断减小训练过程产生的残差来达到将数据分类或者回归的算法。

GBDT 通过多轮迭代，每轮迭代产生一个弱分类器，每个分类器在上一轮分类器的残差基础上进行训练。对弱分类器的要求一般是足够简单，并且是低方差和高偏差的。因为训练的过程是通过降低偏差来不断提高最终分类器的精度。

二 GBDT如何用于分类的

第一步：**「训练的时候，是针对样本 X 每个可能的类都训练一个分类回归树」**。如目前的训练集共有三类，即 K = 3，样本 x 属于第二类，那么针对样本x的分类结果，我们可以用一个三维向量 [0,1,0] 来表示，0 表示不属于该类，1 表示属于该类，由于样本已经属于第二类了，所以第二类对应的向量维度为 1，其他位置为 0。

针对样本有三类的情况，我们实质上是在每轮的训练的时候是同时训练三颗树。第一颗树针对样本 x 的第一类，输入是 $(x,0)$，第二颗树针对样本x的第二类，输入是 $(x,1)$，第三颗树针对样本x的第三类，输入是 $(x,0)$。

在对样本 x 训练后产生三颗树，对 x 类别的预测值分别是 $f_1(x),f_2(x),f_3(x)$，那么在此类训练中，样本 x 属于第一类，第二类，第三类的概率分别是：

$$P_1(x)=exp(f_1(x))/\sum_{k=3}^{3}exp(f_k(x))P_2(x)=exp(f_2(x))/\sum_{k=3}^{3}exp(f_k(x))P_3(x)=$$

然后可以求出针对第一类，第二类，第三类的残差分别是：

$$y_{11}=0-P_1(x)y_{22}=1-P_2(x)y_{33}=0-P_3(x)$$

然后开始第二轮训练，针对第一类输入为 $(x,y_{11}(x))$，针对第二类输入为 $(x,y_{22}(x))$，针对第三类输入为 $(x,y_{33}(x))$，继续训练出三颗树。一直迭代M轮，每轮构建三棵树当训练完毕以后，新来一个样本 x_1，我们需要预测该样本的类别的时候，便产生三个值 $f_1(x),f_2(x),f_3(x)$，则样本属于某个类别 c 的概率为：

$$P_c(x)=exp(f_c(x))/\sum_{k=3}^{3}exp(f_k(x))$$

三 GBDT多分类举例说明

下面以 Iris 数据集的六个数据为例来展示 GBDT 多分类的过程

样本编号	花萼长度(cm)	花萼宽度(cm)	花瓣长度(cm)	花瓣宽度	花的种类
1	5.1	3.5	1.4	0.2	山鸢尾
2	4.9	3.0	1.4	0.2	山鸢尾
3	7.0	3.2	4.7	1.4	杂色鸢尾
4	6.4	3.2	4.5	1.5	杂色鸢尾
5	6.3	3.3	6.0	2.5	维吉尼亚鸢尾
6	5.8	2.7	5.1	1.9	维吉尼亚鸢尾

具体应用到 gbdt 多分类算法。我们用一个三维向量来标志样本的 label，[1,0,0] 表示样本属于山鸢尾，[0,1,0] 表示样本属于杂色鸢尾，[0,0,1] 表示属于维吉尼亚鸢尾。

gbdt 的多分类是针对每个类都独立训练一个 CART Tree。所以这里，我们将针对山鸢尾类别训练一个 CART Tree 1。杂色鸢尾训练一个 CART Tree 2 。维吉尼亚鸢尾训练一个 CART Tree 3，这三个树相互独立。

我们以样本 1 为例：

- 针对 CART Tree1 的训练样本是 [5.1,3.5,1.4,0.2]，label 是 1，模型输入为 [5.1, 3.5, 1.4, 0.2, 1]
- 针对 CART Tree2 的训练样本是 [5.1,3.5,1.4,0.2]，label 是 0，模型输入为 [5.1, 3.5, 1.4, 0.2, 0]
- 针对 CART Tree3 的训练样本是 [5.1,3.5,1.4,0.2]，label 是 0，模型输入为[5.1, 3.5, 1.4, 0.2, 0]

下面我们来看 CART Tree1 是如何生成的，其他树 CART Tree2 ，CART Tree 3 的生成方式是一样的。CART Tree 的生成过程是从这四个特征中找一个特征做为 CART Tree1 的节点。

比如花萼长度做为节点。6 个样本当中花萼长度大于等于 5.1 cm 的就是 A 类，小于 5.1 cm 的是 B 类。生成的过程其实非常简单，问题

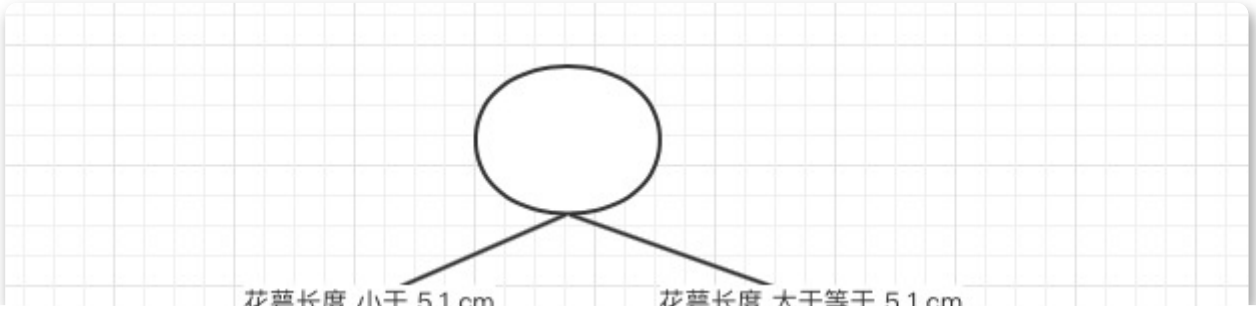
1. 是哪个特征最合适？

2. 是这个特征的什么特征值作为切分点？

即使我们已经确定了花萼长度做为节点。花萼长度本身也有很多值。在这里我们的方式是遍历所有的可能性，找到一个最好的特征和它对应的最优特征值可以让当前式子的值最小：

$$\min_{j,s}\left[\min_{\tau_1}\sum_{x_i\in R_1(j,s)}(y_i-c_1)^2+\min_{c_2}\sum_{x_1\in R_2(j,s)}(y_i-c_2)^2\right]$$

我们以第一个特征的第一个特征值为例。R1 为所有样本中花萼长度小于 5.1cm 的样本集合，R2 为所有样本中花萼长度大于等于 5.1cm 的样本集合，所以 $R_1=[2], R_2=[1,3,4,5,6]$ 。



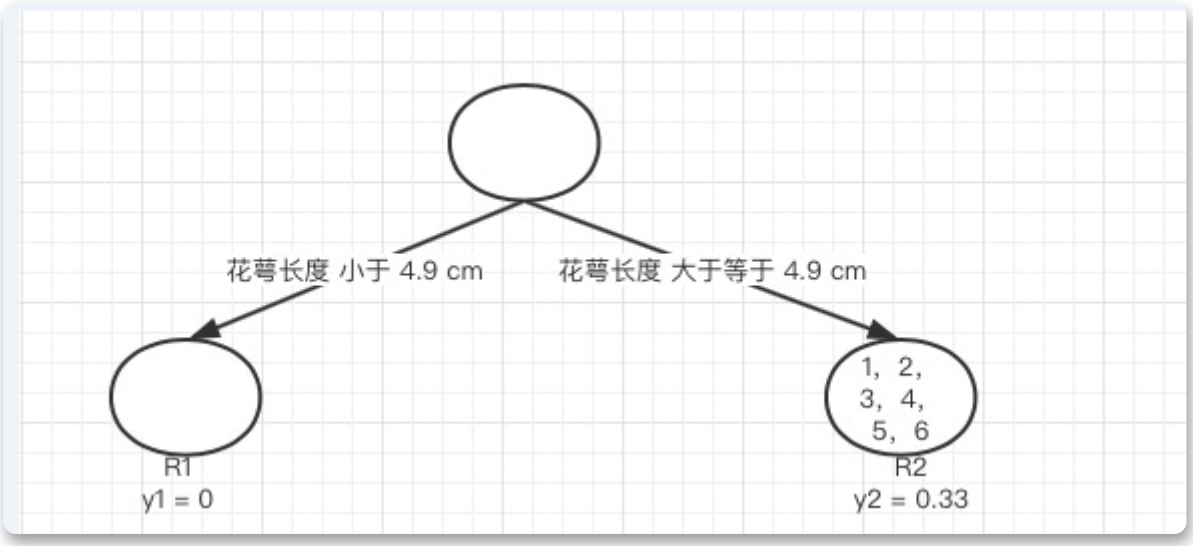
$y1$ 为 R1 所有样本label的均值：$1/1=1$，y_2 为 R2 所有样本 label 的均值：

$$(1+0+0+0+0)/5=0.2$$

下面计算损失函数的值，采用平方误差，分别计算 R1 和 R2 的误差平方和，样本 2 属于 R1 的误差：$(1-1)^2=0$，样本 1，3，4，5，6 属于 R2 的误差和：

$$(1-0.2)^2+(0-0.2)^2+(0-0.2)^2+(0-0.2)^2+(0-0.2)^2=0.8$$

接着我们计算第一个特征的第二个特征值，即 R1 为所有样本中花萼长度小于 4.9 cm 的样本集合，R2 为所有样本当中花萼长度大于等于 4.9 cm 的样本集合，
$R_1=[],R_2=[1,2,3,4,5,6]$，y_1 为 R1 所有样本 label 的均值：0，y_2 为 R2 所有样本 label 的均值：$(1+1+0+0+0+0)/6=0.3333$



计算所有样本的损失值，样本 1 和 2 属于 R2，损失值为：$(1-0.3333)^2+(1-0.3333)^2$，样本 3，4，5，6 也属于 R2，损失值为：
$(0-0.3333)^2+(0-0.3333)^2+(0-0.3333)^2+(0-0.3333)^2$，两组损失值和为 2.222，大于特征一的第一个特征值的损失值，所以我们不取这个特征的特征值。

「继续，这里有四个特征，每个特征有六个特征值，所有需要 6*4=24 个损失值的计算，我们选取值最小的分量的分界点作为最佳划分点，这里我们就不一一计算了，直接给出最小的特

征花萼长度，特征值为 5.1 cm。这个时候损失函数最小为 0.8。于是我们的预测函数此时也可以得到:」

$$f(x)=\sum_{x\in R_1} y_1 * I(x \epsilon R_1)+\sum_{x\in R_2} y_2 * I(x \in R_2)$$

「此例子中 $R_1=[2], R_2=[1,3,4,5,6], y_1=1, y_2=0.2$，训练完以后的最终式子为：」

$$f(x)=\sum_{x\in R_1} 1 * I(x \epsilon R_1)+\sum_{x\in R_2} 0.2 * I(x \in R_2)$$

由这个式子，我们得到对样本属于类别 1 的预测值：$f_1(x)=1*1+0.2*0=1$，同理我们可以得到对样本属于类别 2，3 的预测值 $f_2(x), f_3(x)$，样本属于类别 1 的概率：

$$P_1(x)=\exp(f_1(x))/\sum_{k=1}^{3}\exp(f_k(x))P_1(x)=\exp(f_1(x))/\sum_{k=1}^{3}\exp(f_k(x))$$