相关知识补充

陈星铭

2018.10.10

集成学习技巧

- 单一弱模型的效果可能不是很好
- 如何使用一定的技巧，训练多个模型，将这些模型联合起来?
- 比如随机森林、Adaboost、GBDT
- 但是要记住，**技巧是通用的**，不是针对任何模型的

- **Bagging：** 随机森林
- **Boosting：** Adaboost，GBDT

Bagging

- 也叫 Bootstrap aggregation
- 原始数据集为 $X = \{x_1, x_2, \ldots, x_N\}$
- **有放回地**抽取 N 个样本，构成新的 bootstrap 数据集 X_B
- 该数据集是**有可能出现重复的样本**的
- 这样构成的数据集，理论上是与原数据集同分布的，但是实际肯定会有区别
- 生成 M 个这样的 bootstrap 数据集，用这些数据集分别训练对应的模型，**对于不同的模型进行不同的融合**，即可得到比单一模型表现要好的融合模型

Bagging

- 分类：
- 多数投票，权重投票，…

- 回归：
- 取均值，权重均值，…

考虑这样一个例子

- 原数据集：　　{(x1,y1), (x2,y2), (x3,y3), (x4,y4)} u_0 = (1, 1, 1, 1)
- 采样1：　　{(x2,y2), (x2,y2), (x3,y3), (x4,y4)} u_1 = (0, 2, 1, 1)
- 采样2：　　{(x1,y1), (x1,y1), (x1,y1), (x4,y4)} u_2 = (3, 0, 0, 1)

- $E^u = \frac{1}{N}\sum_{n=1}^{N} u_n err(y_n, h(x_n))$

- u_i可以视为每个数据点的权重

Boosting

- Bagging的可能弊端？

  大家学的东西都是一样的！

- 能不能控制权重u，让各个子模型学到不同的规则？

Boosting的基本思想：一个一个地训练子模型，在下一次训练的时候，当前**分类错误**的点的**权重增大，分类正确**的点的**权重减小**。那么下一次训练的时候就可以**更加关注当前分类错误的这些点**。

Adaboost

- 控制权重u，让各个子模型学到不同的规则

  子模型1：对了8个，错了2个，错误率e为0.2

  放大错误样本的比例：$2 \times 8 = 16$

  缩小正确样本的比例：$8 \times 2 = 16$

- 现在对于子模型1，新的错误率就变为0.5。而一个模型能得出0.5的错误率，等同于直接随机。那么子模型2在使用这个新的数据集来学习的话，学到的东西就能保证和子模型1的是不一样的。

Adaboost

- 控制权重u，让各个子模型学到不同的规则

$$u\text{-incorrect} \propto 1\text{-}e$$

$$u\text{-correct} \propto e$$

新的e为0.5

- 实际操作中，我们令$s = \sqrt{\frac{1-e}{e}}$

$$u\text{-incorrect} \leftarrow u\text{-incorrect} \times s$$

$$u\text{-correct} \leftarrow u\text{-correct} / s$$

Adaboost

- 现在我们学到了N个子模型，怎么合起来？

　　因为每个子模型的表现肯定都不一样，直接每个子模型投票，肯定是不太合理的。我们需要使用某种加权求和的规则来衡量每一个子模型的票数。

- 怎么判断每个子模型的票数？

　　每次算子模型的时候，每个子模型都持有一个错误率e，直观的来想，e越偏离0.5，这个子模型的票数应该越高。

Adaboost

- 怎么判断每个子模型的票数？

  实际操作中，我们使用 $\alpha = \ln(s)$ 来作为每一个子模型的权重。s为前面介绍的参数$s = \sqrt{\frac{1-e}{e}}$。

e = 0.5 → s = 1 → α = 0

e = 0.1 → s = 3 → α = ln(3)

e = 0.9 → s = 1/3 → α = ln(1/3) = -ln(3)

Adaboost

- **步骤：**

  初始化：u = [1/N, 1/N, 1/N, ...]（样本权重）

  For i = 1 to T：

  计算出在这个u下的子模型g_i

  计算出这个子模型在u下的错误率e，以及对应的s（放缩系数）和α_i（子模型权重）

  更新权重u

  return $G(x) = sign(\sum_{i=1}^{T} \alpha_i \, g_i)$

Adaboost

- 总结：

Adaboost = 弱模型g （学生）
+ 权重调整因子s （老师）
+ 加权求和α （班级）

- Bagging：有放回地抽样，每个子模型的训练数据集不同
- 随机森林：除了对数据样本抽样以为，对特征也进行抽样，每个子模型的训练数据集、用到的特征都不同。
- Boosting：每个子模型训练后，对训练失败的样本赋予更大的权重，让后续的子模型集中学习。
- Adaboost：迭代时除了更新样本权重以外，还会更新模型权重。
- GBDT：每个子模型的损失函数跟上一个子模型有关。

二元问题扩展成多元问题

- OVA - one versus all

一对多：把一个类别中的所有样本视为二元中的正样本，把其他类别中的所有样本视为二元中的负样本，进行二元问题的学习。

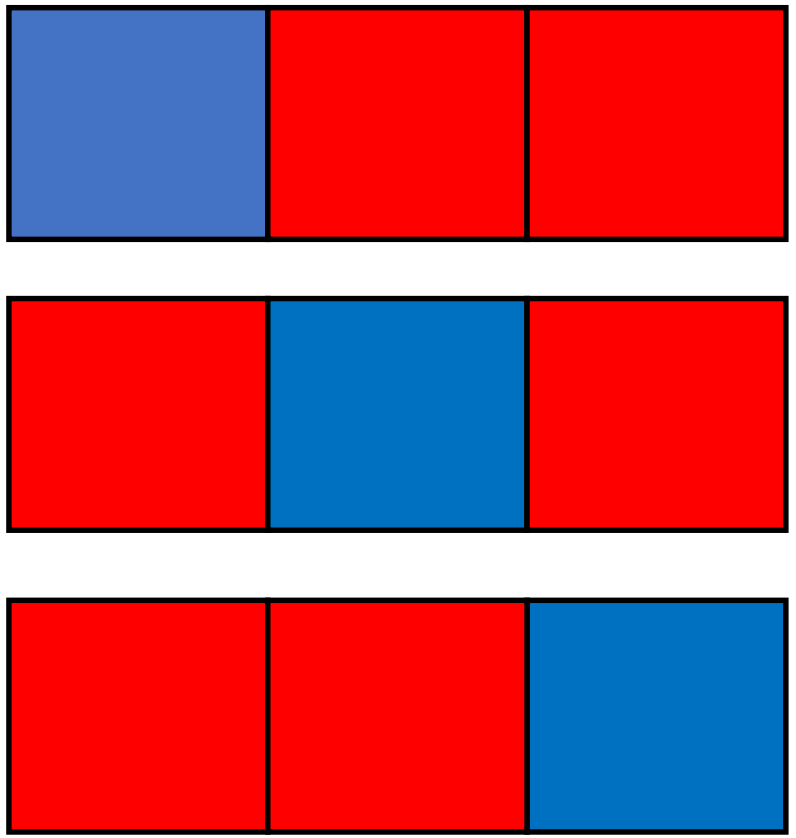
  - N个类别下就会有N个模型。
  - 每个模型训练时都会使用全部的数据。
  - 最后这N个模型通过投票来预测新样本。

- 可能的问题：

类别一多，正负样本数量就会不平衡。

二元问题扩展成多元问题

- OVA - one versus all



CART树：二叉树(处理属性值多元的情况)

有房者	婚姻状况	年收入	拖欠贷款者
是	单身	125K	否
否	已婚	100K	否
否	单身	70K	否
是	已婚	120K	否
否	离异	95K	是
否	已婚	60K	否
是	离异	220K	否
否	单身	85K	是
否	已婚	75K	否
否	单身	90K	是

	有房	无房
否	3	4
是	0	3

$Gini(t_1)=1-(3/3)^2-(0/3)^2=0$
$Gini(t_2)=1-(4/7)^2-(3/7)^2=0.4849$
$Gini=0.3\times0+0.7\times0.4898=0.343$

	单身或已婚	离异
否	6	1
是	2	1

$Gini(t_1)=1-(6/8)^2-(2/8)^2=0.375$
$Gini(t_2)=1-(1/2)^2-(1/2)^2=0.5$
$Gini=8/10\times0.375+2/10\times0.5=0.4$

	单身或离异	已婚
否	3	4
是	3	0

$Gini(t_1)=1-(3/6)^2-(3/6)^2=0.5$
$Gini(t_2)=1-(4/4)^2-(0/4)^2=0$
$Gini=6/10\times0.5+4/10\times0=0.3$

	离异或已婚	单身
否	5	2
是	1	2

$Gini(t_1)=1-(5/6)^2-(1/6)^2=0.2778$
$Gini(t_2)=1-(2/4)^2-(2/4)^2=0.5$
$Gini=6/10\times0.2778+4/10\times0.5=0.3667$

	60	70	75	85	90	95	100	120	125	220
	65	72	80	87	92	97	110	122	172	
	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>	≤	>
是	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0		
否	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1		
Gini	0.400		0.375		0.343		0.417		0.400		0.300		0.343		0.375		0.400			

二元问题扩展成多元问题

- OVO - one versus one

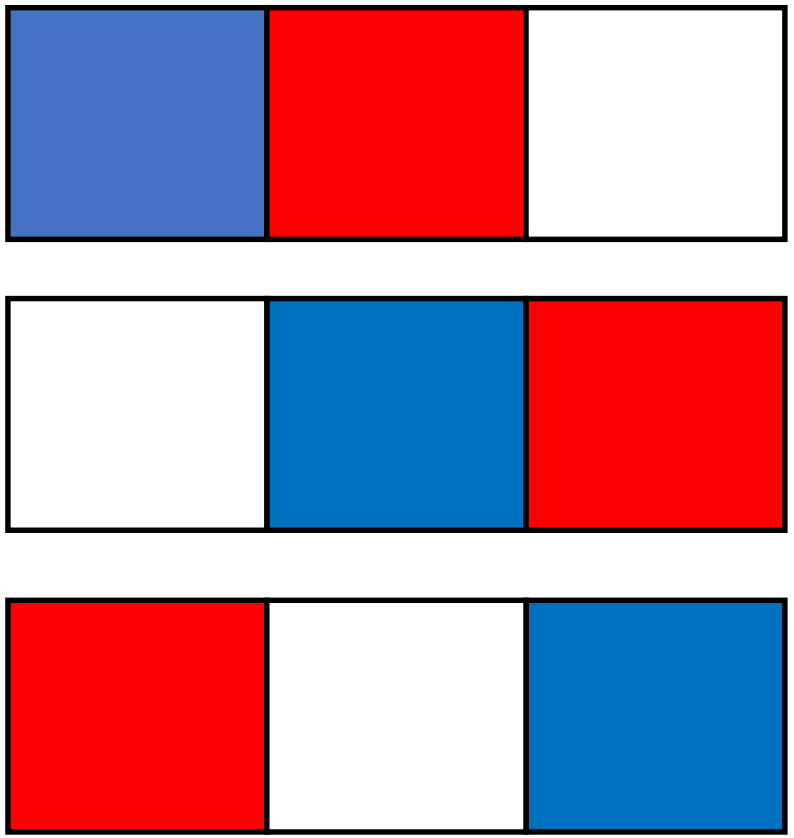
一对一：把一个类别中的样本视为正样本，把另一个类别中的样本视为负样本，进行二元问题的学习。

  - N个类别下就会有$\binom{N}{2}$个模型。
  - 每个模型训练时使用的是这两个类别下的数据。
  - 最后这$\binom{N}{2}$个模型通过投票来预测新样本。

- 一定程度上避免了OVA中样本不平衡的问题，但同样的也增大了计算量。

二元问题扩展成多元问题

- OVO - one versus one



一些Pro的建议

- 大部分时候，简单的模型调出一个好的参数，跑出来的结果不比那些看起来高大上的算法的结果差
- 比起我们每次帮你跑10个版本的表现，要学会自己线下通过验证集来判断自己模型的表现，这样更有效率，更方便，更自由。
- 觉得自己调参不好，提交各种版本，总会有表现好的时候。
- 记得控制版本和结果

评分标准

- 排名 30%（小组算分，同一个小组所有成员相同）
- 验收 30%（验收以小组为单位，但是每个人单独算分）
- 报告（40%）（个人为单位，单独算分）
  - 主要写自己的个人工作，小组工作简要概括下即可。
- 加分（5~20%）（小组算分）
  - 尝试新算法
  - 或者分数较高且算法有创新
  - 除此之外的一些优秀的情况

最终提交

- 报告提交DDL：（10月21日晚上11点）23:00:00
- 提交内容：
  - 实验报告，每个人一份，命名为：组号_学号_姓名拼音_report.pdf，如“2_16350000_xiaoming_report.pdf”
  - 无论之前是否已经有成绩，都要提交一份最后的结果跑最终rank。
  - 源码zip，包含多个文件，命名为：组号_code.zip，如“2_code.zip”，里面包含一个 **readme** 文件，阐述各文件用途