机器学习导论
习题三

141220132, 银琦, 141220132@smail.nju.edu.cn

2017 年 4 月 24 日

1 [30pts] Decision Tree Analysis

决策树是一类常见的机器学习方法，但是在训练过程中会遇到一些问题。

(1) **[15pts]** 试证明对于不含冲突数据(即特征向量完全相同但标记不同)的训练集，必存在与训练集一致(即训练误差为0)的决策树;

(2) **[15pts]** 试分析使用“最小训练误差”作为决策树划分选择的缺陷。

Solution.

(1)因为决策树是通过属性来划分，相同属性的样本最终肯定会进入相同的叶节点。一个叶节点只有一个分类，如果样本属性相同而分类不同，必然产生训练误差。反之，决策树只会在当前样本集合是同一类或者所有属性相同时才会停止划分，最终得到训练误差为0的决策树。

(2)”最小训练误差”容易引起过度学习，而过度学习样本特性最终可能导致严重的过拟合，没有泛化能力，这对决策树的划分带来很大的误差。

2 [30pts] Training a Decision Tree

考虑下面的训练集：共计6个训练样本，每个训练样本有三个维度的特征属性和标记信息。详细信息如表1所示。

请通过训练集中的数据训练一棵决策树，要求通过“信息增益”(information gain)为准则来选择划分属性。请参考书中图4.4，给出详细的计算过程并画出最终的决策树。

Table 1: 训练集信息

序号	特征**A**	特征**B**	特征**C**	标记
1	0	1	1	0
2	1	1	1	0
3	0	0	0	0
4	1	1	0	1
5	0	1	0	1
6	1	0	1	1

Solution. 此处用于写解答(中英文均可)
(1)首先计算根节点的信息熵，开始时根节点包含D的所有样例，其中正例占$p_1 = \frac{3}{6}$，反例占$p_2 = \frac{3}{6}$，于是根据公式可得：

$$Ent(D) = -\sum_{k=1}^{2} p_k log_2 p_k = -(\frac{3}{6} log_2 \frac{3}{6} + \frac{3}{6} log_2 \frac{3}{6}) = 1$$

(2)计算当前属性集合A,B,C中每个属性的信息增益

对于属性A，若对其进行划分，则可得到两个子集，分别记为$D^1(A = 0)$，$D^2(A = 1)$，子集$D^1(A = 0)$包含编号为1,3,5的三个样例，其中正例占$p_1 = \frac{1}{3}$，反例占$p_2 = \frac{2}{3}$，子集$D^2(A = 1)$包含编号为2,4,6的三个样例，其中正例占$p_1 = \frac{2}{3}$，反例占$p_2 = \frac{1}{3}$，根据公式计算出用属性A划分之后所获得的两个分支节点的信息熵为：

$$Ent(D^1) = -(\frac{1}{3} log_2 \frac{1}{3} + \frac{2}{3} log_2 \frac{2}{3}) = 0.918$$

$$Ent(D^2) = -(\frac{2}{3} log_2 \frac{2}{3} + \frac{1}{3} log_2 \frac{1}{3}) = 0.918$$

根据公式可计算出属性A的信息增益为：

$$Gain(D, A) = Ent(D) - \sum_{v=1}^{2} \frac{|D^v|}{|D|} Ent(D^v) = 1 - (\frac{1}{2} \times 0.918 + \frac{1}{2} \times 0.918) = 0.082$$

对于属性B，若对其进行划分，则可得到两个子集，分别记为$D^1(B = 0)$，$D^2(B = 1)$，子集$D^1(B = 0)$包含编号为1,2,4,5的四个样例，其中正例占$p_1 = \frac{2}{4}$，反例占$p_2 = \frac{2}{4}$，子集$D^2(B = 1)$包含编号为3,6的两个样例，其中正例占$p_1 = \frac{1}{2}$，反例占$p_2 = \frac{1}{2}$，根据公式计算出用属性B划分之后所获得的两个分支节点的信息熵为：

$$Ent(D^1) = -(\frac{2}{4} log_2 \frac{2}{4} + \frac{2}{4} log_2 \frac{2}{4}) = 1$$

$$Ent(D^2) = -(\frac{1}{2} log_2 \frac{1}{2} + \frac{1}{2} log_2 \frac{1}{2}) = 1$$

根据公式可计算出属性B的信息增益为：

$$Gain(D, B) = Ent(D) - \sum_{v=1}^{2} \frac{|D^v|}{|D|} Ent(D^v) = 1 - (\frac{1}{2} \times 1 + \frac{1}{2} \times 1) = 0$$

对于属性C，若对其进行划分，则可得到两个子集，分别记为$D^1(C = 0)$，$D^2(C = 1)$，子集$D^1(C = 0)$包含编号为3,4,5的三个样例，其中正例占$p_1 = \frac{2}{3}$，反例占$p_2 = \frac{1}{3}$，子集$D^2(C = 1)$包含编号为1,2,6的三个样例，其中正例占$p_1 = \frac{1}{3}$，反例占$p_2 = \frac{2}{3}$，根据公式计算出用属性C划分之后所获得的两个分支节点的信息熵为：

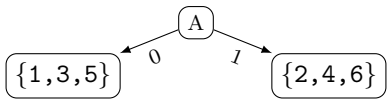
$$Ent(D^1) = -(\frac{2}{3} log_2 \frac{2}{3} + \frac{1}{3} log_2 \frac{1}{3}) = 0.918$$

$$Ent(D^2) = -(\frac{1}{3} log_2 \frac{1}{3} + \frac{2}{3} log_2 \frac{2}{3}) = 0.918$$

根据公式可计算出属性C的信息增益为：

$$Gain(D, C) = Ent(D) - \sum_{v=1}^{2} \frac{|D^v|}{|D|} Ent(D^v) = 1 - (\frac{1}{2} \times 0.918 + \frac{1}{2} \times 0.918) = 0.082$$

由于属性A与属性C均取得了最大增益，可任选其中之一作为划分属性，我选择属性A，此时得到了基于属性A对根节点的划分:

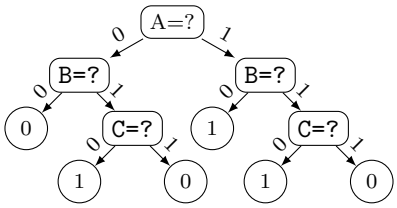


然后决策树学习算法将对每个分支节点做进一步划分，设第一个集合为$D^1(A=0)$，第二个集合为$D^2(A=1)$ 基于D^1, D^2分别计算出属性B、C的信息增益：

$$Gain(D^1, B) = 0.251; \qquad Gain(D^1, C) = 0.251.$$

$$Gain(D^2, B) = 0.251; \qquad Gain(D^2, C) = 0.251.$$

在集合D^1, D^2中，属性B、C均取得了最大的信息增益，所以任选其一即可，我均选择B，所以最后得出的决策树如下：



3 [40pts] Back Propagation

单隐层前馈神经网络的误差逆传播(error BackPropagation，简称BP)算法是实际工程实践中非常重要的基础，也是理解神经网络的关键。

请编程实现BP算法，算法流程如课本图5.8所示。详细编程题指南请参见链接：`http://lamda.nju.edu.cn/ml2017/PS3/ML3_programming.html`

在实现之后，你对BP算法有什么新的认识吗？请简要谈谈。

Solution.

(1)遇到的问题：一开始使用循环进行计算，但是效率太过低下，耗时很长，于是改成了矩阵相乘，但是计算出的精度一直小于1，查看了输出，发现输出的所有值几乎一样，经过调试发现是归一化出现了问题，从理论上来讲应该先对数据进行归一化，可能是我写的不对，归一化后精度十分小，最终取消了归一化，使得精度达到了88左右；激活函数可能精度不够高，因为输出了看到有很多0，我增加了“format long”，略微提高了一点精度；学习次数对精度也有影响，一开始设置的20次，后来增加到30次发现精度有所提高。

(2)认识：作业中完成的BP算法是一种监督学习，包括两个过程：正向传播和误差反向传播，它可以很方便的处理多分类问题，并且能够达到一定的精度和效率。

(3)参考文献：`http://blog.csdn.net/google19890102/article/details/32723459`

附加题[30pts] Neural Network in Practice

在实际工程实现中，通常我们会使用已有的开源库，这样会减少搭建原有模块的时间。

因此，请使用现有神经网络库，编程实现更复杂的神经网络。详细编程题指南请参见链接：
http://lamda.nju.edu.cn/ml2017/PS3/ML3_programming.html

和上一题相比，模型性能有变化吗？如果有，你认为可能是什么原因。同时，在实践过程中你遇到了什么问题，是如何解决的？

Solution. 此处用于写解答(中英文均可)