机器学习导论
习题三

学号：141130077
作者姓名：邱梓豪
邮箱：2957606241@qq.com

2017 年 4 月 26 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. 此处用于写证明(中英文均可)
（1）由决策树算法可知，决策树的每个内部结点是对于属性的判断，属性相同的样本被划分到相同的子节点；当当前样本集合属于同一类或其所有属性相同时停止划分。这就使得如果样本属性相同则必然进入相同的叶节点，所以如果训练集不含冲突数据，那么必存在训练误差为0的决策树。
（2）若使用"最小训练误差"作为决策树划分选择，则很有可能出现决策树对样本的过度学习，从而造成过拟合，使得决策树的泛化能力下降

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。

Solution. 此处用于写解答(中英文均可)
根节点中正例占：$p_1 = \frac{1}{2}$，反例占$p_2 = \frac{1}{2}$信息熵：

$$Ent(D) = -\sum_{k=1}^{2} p_k log_2 p_k = -(\frac{1}{2}log_2\frac{1}{2} + \frac{1}{2}log_2\frac{1}{2}) = 1 \tag{2.1}$$

表 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

以特征A进行划分：
$D^1 : \{1,3,5\}$，正例：$\frac{1}{3}$，反例：$\frac{2}{3}$
$D^2 : \{2,4,6\}$，正例：$\frac{2}{3}$，反例：$\frac{1}{3}$

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{3}log_2\frac{1}{3} + \frac{2}{3}log_2\frac{2}{3}) = 0.918 \\ Ent(D^2) &= -(\frac{1}{3}log_2\frac{1}{3} + \frac{2}{3}log_2\frac{2}{3}) = 0.918 \end{aligned} \tag{2.2}$$

故该特征A的信息增益为：

$$Gain(D,A) = 1 - (\frac{1}{2} \times 0.918 + \frac{1}{2} \times 0.918) = 0.082 \tag{2.3}$$

以特征B进行划分：
$D^1 : \{1,2,4,5\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$
$D^2 : \{3,6\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{2}log_2\frac{1}{2} + \frac{1}{2}log_2\frac{1}{2}) = 1 \\ Ent(D^2) &= -(\frac{1}{2}log_2\frac{1}{2} + \frac{1}{2}log_2\frac{1}{2}) = 1 \end{aligned} \tag{2.4}$$

故该特征B的信息增益为：

$$Gain(D,B) = 1 - (\frac{2}{3} \times 1 + \frac{1}{3} \times 1) = 0 \tag{2.5}$$

以特征C进行划分：
$D^1 : \{1,2,6\}$，正例：$\frac{1}{3}$，反例：$\frac{2}{3}$
$D^2 : \{3,4,5\}$，正例：$\frac{2}{3}$，反例：$\frac{1}{3}$

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{3}log_2\frac{1}{3} + \frac{2}{3}log_2\frac{2}{3}) = 0.918 \\ Ent(D^2) &= -(\frac{1}{3}log_2\frac{1}{3} + \frac{2}{3}log_2\frac{2}{3}) = 0.918 \end{aligned} \tag{2.6}$$

故该特征C的信息增益为：

$$Gain(D,C) = 1 - (\frac{1}{2} \times 0.918 + \frac{1}{2} \times 0.918) = 0.082 \tag{2.7}$$

此时特征A和C的增益相同，这里选择特征A来划分，所以此时根节点的两个子节点分别为$\{1,3,5\}$和$\{2,4,6\}$，先对$\{1,3,5\}$进行递归：

以特征B进行划分：
$D^1:\{1,5\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$
$D^2:\{3\}$，正例：0，反例：1

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{2}log_2\frac{1}{2}+\frac{1}{2}log_2\frac{1}{2}) = 1 \\ Ent(D^2) &= 0 \end{aligned} \tag{2.8}$$

故该特征B的信息增益为：

$$Gain(D,B) = 0.918 - (\frac{2}{3}\times 1+\frac{1}{3}\times 0) = 0.251 \tag{2.9}$$

以特征C进行划分：
$D^1:\{1\}$，正例：0，反例：1
$D^2:\{3,5\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$

$$\begin{aligned} Ent(D^1) &= 0 \\ Ent(D^2) &= -(\frac{1}{2}log_2\frac{1}{2}+\frac{1}{2}log_2\frac{1}{2}) = 1 \end{aligned} \tag{2.10}$$

故该特征C的信息增益为：

$$Gain(D,C) = 0.981 - (\frac{1}{3}\times 0+\frac{2}{3}\times 1) = 0.251 \tag{2.11}$$

此时特征B和C的增益相同，这里选择特征B来划分，所以此时根节点的两个子节点分别为$\{1,5\}$和$\{3\}$，后者已经满足递归结束条件，所以对$\{1,5\}$进行递归，此时这两个元素仅是特征C不同，所以按照C进行划分即可。
再对$\{2,4,6\}$进行递归：
以特征B进行划分：
$D^1:\{2,4\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$
$D^2:\{6\}$，正例：1，反例：0

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{2}log_2\frac{1}{2}+\frac{1}{2}log_2\frac{1}{2}) = 1 \\ Ent(D^2) &= 0 \end{aligned} \tag{2.12}$$

故该特征B的信息增益为：

$$Gain(D,B) = 0.918 - (\frac{2}{3}\times 1+\frac{1}{3}\times 0) = 0.251 \tag{2.13}$$

以特征C进行划分：
$D^1:\{2,6\}$，正例：$\frac{1}{2}$，反例：$\frac{1}{2}$
$D^2:\{4\}$，正例：1，反例：0

$$\begin{aligned} Ent(D^1) &= -(\frac{1}{2}log_2\frac{1}{2}+\frac{1}{2}log_2\frac{1}{2}) = 1 \\ Ent(D^2) &= 0 \end{aligned} \tag{2.14}$$

故该特征C的信息增益为：

$$Gain(D,C) = 0.981 - (\frac{1}{3}\times 0+\frac{2}{3}\times 1) = 0.251 \tag{2.15}$$

此时特征B和C的增益相同，这里选择特征B来划分，所以此时根节点的两个子节点分别为$\{2,4\}$和$\{6\}$，后者已经满足递归结束条件，所以对$\{2,4\}$进行递归，此时这两个元素仅是特征C不同，所以按照C进行划分即可。
最后得到的递归树如下所示：

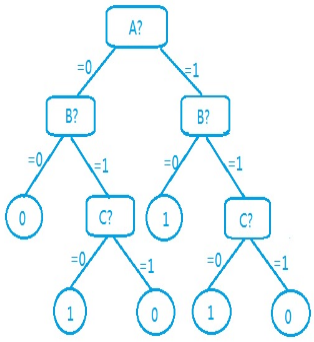

图1 第二题中的决策树

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. 此处用于写解答(中英文均可)
实现这个算法之后，我对BP的理解又加深了一层。对于只有一个隐层的神经网络而言，其学到的内容就隐藏在两个连接权矩阵和两个阈值向量中，BP算法就利用当前的连接权和阈值算出一个输出，然后算出这个输出和真实值直接的差异，进而算出每个连接权和阈值的梯度，这样反复迭代直到精度达到要求。
通过这次编程，原本“高大上”的神经网络在我的眼里已不再神秘，原来就是一个数学模型。我感觉一切问题归根结底都是数学问题，所以我在今后的学习中要注意提升自己的数学水平。

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)
利用开源库的精度略有下降（我写的精度为92.8%，利用开源库的为91.7%），由于两个神经网络的结构差别较大，激活函数也不同，所以很难说两个模型性能上差别有多大。
这次实践遇到的主要问题是开源库的使用，网上的评论大多认为Keras的文档清晰易懂，但我在看文档时却有点云里雾里，比如dense()这个函数是什么意思。后来我在google上找到一篇别人写的tutorial，才算解决了问题。所以有时找找tutorial能更好地入手一个陌生的库。