# 人工智能-第3次课程作业报告

授课教师：杨旭 作者：<徐子航>-<61520711>

## 1 问题描述

### 1.1题目介绍

利用决策树判断是否需要在餐馆等座位，本问题主要基于如下属性列表基础上决策的。

文本, 信件

描述已自动生成

### 1.2任务说明

利用restaurant\_willwait数据集中的75条数据训练决策树模型。数据的属性值如下表。

表1 属性值离散表示

|  |  |
| --- | --- |
| 属性 | 离散取值 |
| Alternate | 1 0 |
| Bar | 1 0 |
| Fri/sat | 1 0 |
| Hungry | 1 0 |
| Patrons | 0 0.5 1 |
| Price | 0 0.5 1 |
| Raining | 1 0 |
| Reservation | 1 0 |
| Type | 1 0.67 0.33 0 |
| Waitestimate | 0 0.17 0.5 1 |

表2 是否等待离散表示

|  |  |
| --- | --- |
| 属性 | 离散取值 |
| willwait | 1 0 |

### 1.3实验环境

Visual Stdio 2022

### 1.4评价标准

**正确率**

## 2 实验方案

这次实验需要完成的是决策树中的信息熵计算函数、importance函数，学习函数和分类预测函数。其中，分类预测函数classify\_rec函数的注释给的比较完善，其它三个函数是完成的重点。

### 2.1熵与Importance函数

熵衡量的其实是随机变量的不确定性，不确定性越高，熵越大，而随着信息的获取，确定性增加，熵也就减小。而决策树就是希望让熵值最快地下降。

下面是相关的计算公式，而Importance就是要选取信息收益最大的属性A。这样能提升分类的效果。



### 2.2 决策树学习算法

文本, 信件

描述已自动生成

上图是教材中给出的基本学习算法，在排除了三种特殊状况(example集为空, attribute集为空，所有样本均为同一类)后，根据Importance函数选择最优的属性进行构建。当然，对于算法中的*exs*(倒数第四行)，我认为也有必要进行是否为空的判定。

## 3实验结果

根据代码中的提示，正确的决策树应该由PATRONS作为第一个分类属性，观察这里的树的结构，确实符合这一要求。

文本

描述已自动生成

实验中，我也把给的数据进行了划分，分成了训练集和测试集。最终的测试结果可以达到93%的准确率。



## 4实验分析

|  |  |
| --- | --- |
| 数据占比 | 准确率 |
| 2:1 (train-50,test-25) | 92% |
| 4:1 (train-60,test-15) | 93.3% |

上面的实验数据都是我把75条数据的前50条，或前60条拿来做训练，剩下的做测试而得到的实验结果。但随后发现了这样得到的数据的偶然性，于是又进行了交叉验证,按照4:1进行数量划分,每次把分好的5份数据中拿一份作为测试集，剩下的为训练集。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 验证序号 | 1 | 2 | 3 | 4 | 5 |
| 准确率/% | 93.3 | 86.7 | 86.7 | 93.3 | 93.3 |

把这5次的准确率取均值，平均准确率为90.66%。

从这个数据波动中，可以看到决策树还是很容易受到数据的影响的，稍有不慎，就会过拟合。从结果反推，数据中下面几点很可能会影响决策树的效果

* 噪声数据：其实噪声一直是影响机器学习模型的一大因素，具体到决策树中，一旦以噪声为分割标准，就难以代表真实数据
* 代表性数据缺失：这是从数据的分布方面来进行分析。数据集很可能是符合“长尾分布”的，这样，一些类的数据较少，训练出的模型也就难以很好地对这些模型进行分类。

要在一定程度上解决这些问题，除了在数据收集的时候想办法，也可以考虑采取一些模型上的优化。一些技术虽然作业框架里没有使用，但确实能够提升模型的鲁棒性。对树的剪枝操作就非常重要。最简单的方法就像深度优先算法那样，设一个预定深度，当然这里还要设一个预定宽度。也可以使用一些更“高级”的算法，如CART等。

## 5 结论

在这一次实验中，我依照提示，实现了一个简单的决策树。在进一步理解了相关理论的同时，对于决策树的优点也有了认识。首先决策树这种模型的可视化非常容易，相较于现在深度学习中的各种模型要直观地多。 而且决策树对于数据量的需求也不大。这里给的数据只有75条，我自己进行测试时只用了50~60条数据，就获得了一个似乎还不错的模型。当然，一些缺点也很明显，对于一些更复杂的任务难以胜任，而且很容易过拟合，需要配合一些复杂的剪枝技术才能更好的运作。不过，联系到我正在学习的计算机视觉、自然语言处理等课程，我可以感受到这些“传统算法”的魅力与威力。神经网络、深度学习等等确实很有用，但这些传统算法的优势丝毫不弱。最近也有学者在把两者结合组建模型，或许这才是正确的方向。不论是传统算法还是深度学习算法，都应该基于实际问题去选择和设计。