决策树实验报告

191250111 裴为东

决策树算法原理:

1.首先将所有的特征看成一个个节点，创建出根节点;

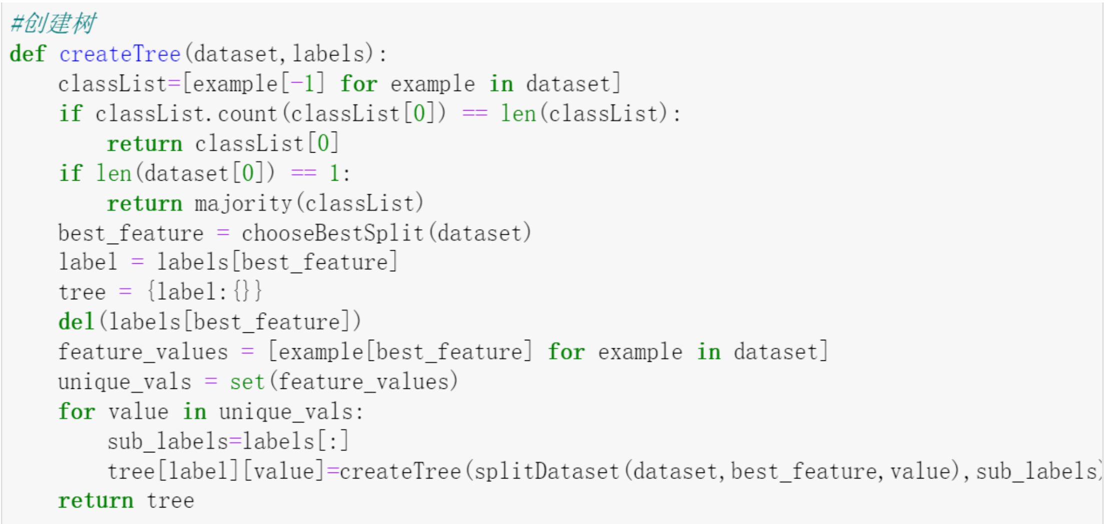
2.然后遍历所有的特征，在每一次到某个特征时遍历当前特征的所有分割方式，找到最好的分割点，讲数据划分为不同的子节点，计算划分后子节点的信息熵

3.在遍历的所以特征中，比较寻找最优的特征以及最优特征的最优划分方式，选择信息增益最高的特征作为节点进行分割操作，产生子树

4．重复 2-3 步，直到子节点中只有一种类型或为空，或者当前节点中样本数小于某个值，同时迭代次数达到指定值

核心代码:

创建决策树



```python
#创建树
def createTree(dataset,labels):
    classList=[example[-1] for example in dataset]
    if classList.count(classList[0]) == len(classList):
        return classList[0]
    if len(dataset[0]) == 1:
        return majority(classList)
    best_feature = chooseBestSplit(dataset)
    label = labels[best_feature]
    tree = {label:{}}
    del(labels[best_feature])
    feature_values = [example[best_feature] for example in dataset]
    unique_vals = set(feature_values)
    for value in unique_vals:
        sub_labels=labels[:]
        tree[label][value]=createTree(splitDataset(dataset,best_feature,value),sub_labels)
    return tree
```

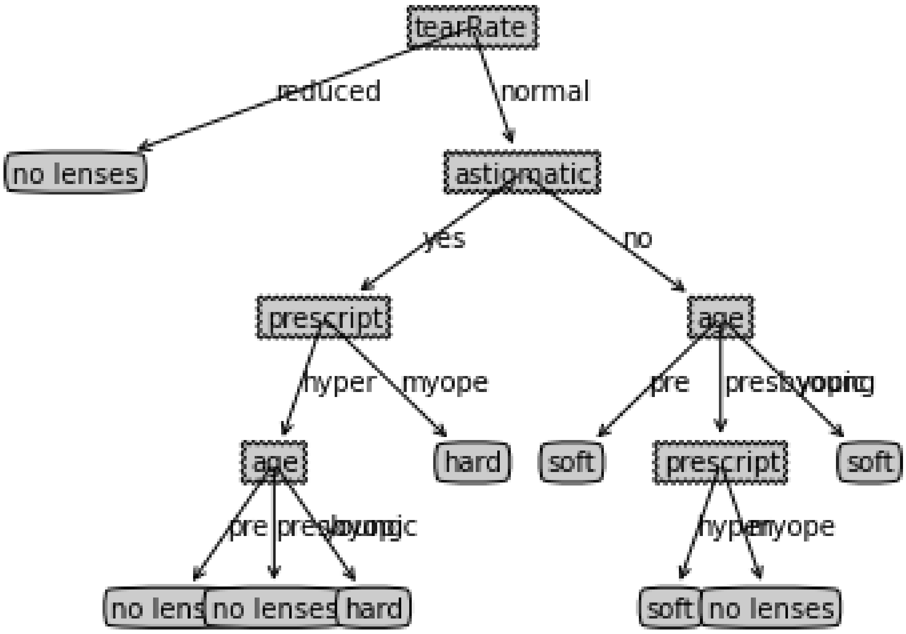
计算香浓熵

```
#计算香浓熵
def calShannon(dataset):
    size = len(dataset)
    label={}
    for feature in dataset:
        current_label=feature[-1]
        if current_label not in label.keys():
            label[current_label]=0

        label[current_label] +=1

    shannon=0.0
    for key in label:
        probability=float(label[key])/size
        shannon -= probability * log(probability,2)
    return shannon
```

选择最优的划分

```
#选择最好的数据集划分方式
def chooseBestSplit(dataset):
    num_features = len(dataset[0])-1
    base = calShannon(dataset)
    best_info_gain = 0.0
    best_feature = -1
    for i in range(num_features):
        feature_list = [example[i] for example in dataset]
        unique_vals = set(feature_list)
        new = 0.0
        for value in unique_vals:
            sub_dataset = splitDataset(dataset,i,value)
            probability = len(sub_dataset)/float(len(dataset))
            new += probability * calShannon(sub_dataset)
        info_gain = base-new
        if info_gain > best_info_gain:
            best_info_gain = info_gain
            best_feature = i

    return best_feature
```

实验结果:


tearRate
reduced
normal
no lenses
astigmatic
yes
no
prescript
age
hyper
myope
pre
presbyopic
young
hard
soft
prescript
soft
age
pre
presbyopic
young
hyper
myope
no lens
no lenses
hard
soft
no lenses