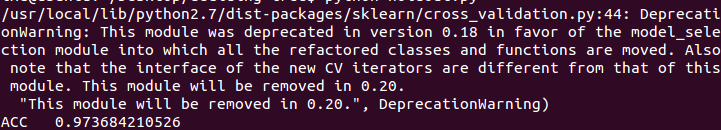
1. 决策树

说明：本实验使用python语言编写，在linux环境下运行，调用sklearn包实现部分功能，剩余功能通过手动编写，采用样本集为Iris测试集。

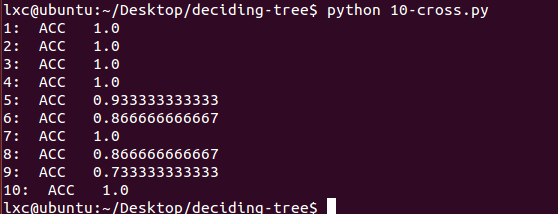
1. 使用holdout方法，10路交叉验证和bootstrap方法训练决策树：

首先是holdout方法，随机将数据分成训练集和测试集，调用sklearn包中的train\_test\_split函数，输入参数为特征列表和分类结果列表，输出为训练集特征，结果和测试集特征，结果。经过调用决策树生成函数生成决策树，并对模型进行模型评估，计算ACC的值，运行结果如下：



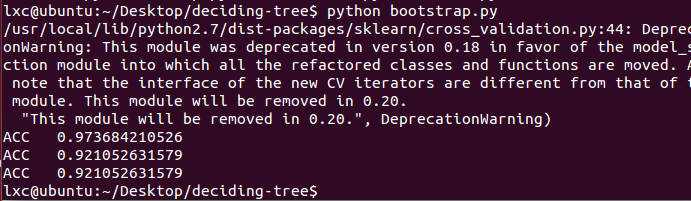
ACC值为0.97。

其次是10路交叉验证，将数据分成10份，其中9份作为训练集，1份作为测试集，调用Fold模块，按照API说明输出10份训练集和测试集， 经过调用决策树生成函数生成决策树，并对模型进行模型评估，计算ACC的值，运行结果如下：



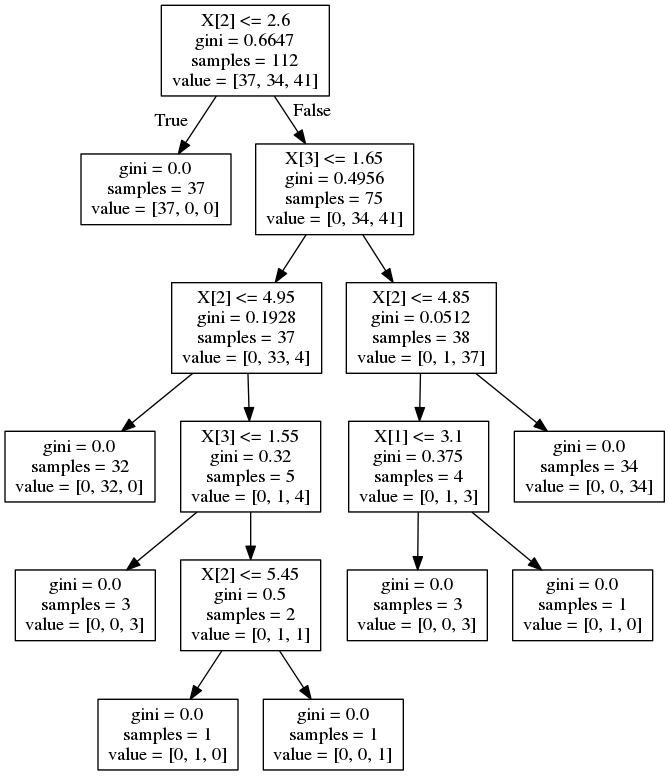
ACC值一共为10个，从0.73到1.0不等。

最后是bootstrap方法，对数据集进行重复采样，设定采样次数为3，调用sklearn包中的cross\_validation模块，经过调用决策树生成函数生成决策树，并对模型进行模型评估，计算ACC的值，运行结果如下：

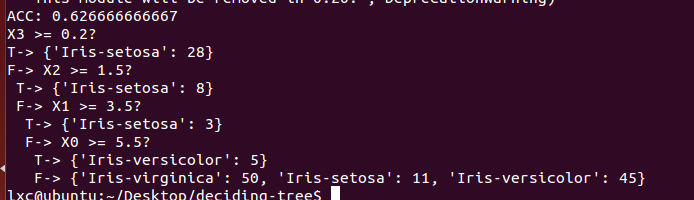


ACC值为3个，从0.92到0.97不等。

调用绘制决策树函数，生成一个png文件，如图：

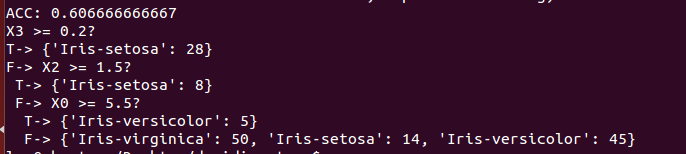


1. 通过基尼指数生成决策树，通过手动编写代码，计算生成决策树过程的不同决策导致的基尼指数变化，决定最优决策划分。模型评估结果如下：



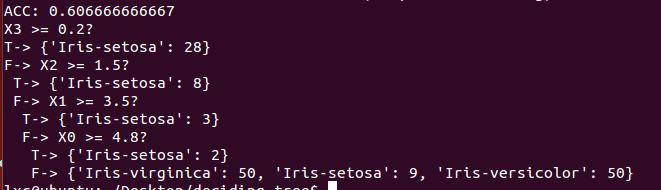
使用基尼指数进行决策，ACC值为0.62。

1. 通过错误率生成决策树，通过手动编写代码，计算生成决策树过程的不同决策导致的错误率变化，决定最优决策划分。模型评估结果如下：



使用错误率进行决策，ACC值为为0.60。

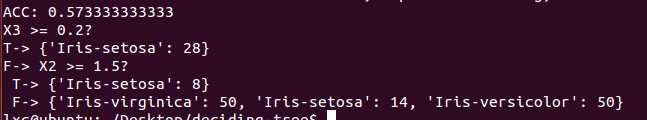
1. 通过信息熵生成决策树，通过手动编写代码，计算生成决策树过程的不同决策导致的信息熵变化，决定最优决策划分。模型评估结果如下：



使用信息熵进行决策，ACC值为0.60。

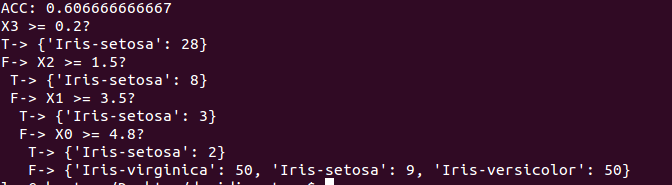
结论：因此可以看出，通过调用sklearn生成的决策树和自己写的决策树在ACC值上存在较大差异，归根到底是核心算法的不同，封装的决策树生成函数中每一个属性都是可以重复使用的，从图中可以看出部分属性出现了多次，然而在自己编写决策树算法时特别设置了每一个属性只允许出现一次，因此决策树的非叶子结点数量不会大于数据集的维度，从而导致ACC值有所下降，因此综合来说属性的可重复使用在一定程度上提高了决策树的性能。

1. 预剪枝与后剪枝。预剪枝为设立一个初始阈值，在决策树生成过程中信息增益大于这个阈值则停止生成，本实验中将预剪枝阈值设为0.1，以信息熵为例，预剪枝结果如下：



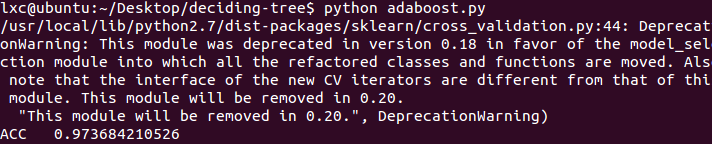
可以看出，ACC值发生变化，并且决策树结构也发生了变化，即预剪枝成功。

后剪枝为决策树生成之后，自下向上通过计算信息增益，若子节点信息熵的和大于父节点的信息熵则合并2个节点，即将此枝剪掉。后剪枝运行结果如下：



发现结果和后剪枝之前并没有什么不同，可能是因为数据集小的原因。

1. 使用adaboost算法训练决策树，调用AdaBoostClassifier模块训练决策树，AdaBoost算法为将多个弱分类器关联成一个强分类器，此模块默认使用决策树分类器，因此直接调用，经过调用决策树生成函数生成决策树，并对模型进行模型评估，计算ACC的值，运行结果如下：



ACC值为0.97。

1. K-means算法

说明：本实验使用matlab语言编写，运行环境为windows，没有调用包，使用的数据集为Iris数据集。

1. 说明不同K值对算法SSE的影响。k值对应k个初始聚类中心，通过设定不同K值比较SSE大小：

因此可以看出不同初始聚类中心数对K-means聚类算法影响较大。

1. 说明不同聚类中心初值对SSE的影响。通过对特征集中选取不同元组作为初始值比较聚类效果：

因此可以看出聚类中心初值对K-means算法影响较大。

1. 二分聚类，并比较二分聚类和普通K-means聚类的效果差异。为了方便和普通K-means聚类进行比较，初始聚类数为5，即为K-means聚类下SSE最小的情况，二分聚类通过比较不同分类条件下SSE的值来决定进行二分的簇，因此具有更好的聚类效果，程序运行结果如下：



可以看出，聚类结束后，将第三个簇二分，得到的最优结果为57.25，而K-means算法得到的最优SSE为77.33，明显二分聚类效果优于k-means聚类。