实验四：决策树

姓名：　　学号：

- 实验目的

理解和掌握决策树原理，包括划分选择中三种经典指标信息增益、增益率和基尼指数的优缺点，剪枝处理方法及作用、连续值与缺失值处理等。

- 实验要求

基于给定数据集，采用决策树模型对问题进行分类。通过准确率指标值度量模型性能，对比不同划分选择标准的性能表现。

- 实验环境

python

numpy

matplotlib

sklearn

- 实验代码

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier, plot_tree


# 读取数据
load_data_testing = np.loadtxt('experiment_04_testing_set.csv', delimiter=',')
load_data_training = np.loadtxt('experiment_04_training_set.csv', delimiter=',')


# 分割
```

```
X_train = load_data_training[:, :-1]
Y_train = load_data_training[:, -1]
X_test = load_data_testing[:, :-1]
Y_test = load_data_testing[:, -1]

# 训练模型
tree = DecisionTreeClassifier(criterion='entropy', max_depth=3, random_state=1)
# tree = DecisionTreeClassifier(criterion='gini', max_depth=1, random_state=1)
# tree = DecisionTreeClassifier(criterion='gini', max_depth=2, random_state=1)
# tree = DecisionTreeClassifier(criterion='gini', max_depth=3, random_state=1)
# tree = DecisionTreeClassifier(criterion='entropy', max_depth=1, random_state=1)
# tree = DecisionTreeClassifier(criterion='entropy', max_depth=2, random_state=1)
tree.fit(X_train, Y_train)

# 画图
plot_tree(tree)
plt.show()

# 计算精度
predict = tree.predict(X_test)
accuracy = (np.sum((predict == Y_test).astype(int)) / np.size(Y_test, 0))
print(accuracy)
```

● 结果分析

使用 sklearn 中 tree.DecisionTreeClassifier 构建决策树，设置 random_state=1（消除随机性，多次实验结果相同），划分标准依次选择 criterion = 'gini'和 criterion = 'entropy'，决策树最大层数依次设置 max_depth = 1，max_depth = 2，max_depth = 3，填写如下实验结果。可以采用 sklearn 中 tree.plot_tree()方法绘制决策树。

(1) 测试集上精度（accuracy）为

准则\层数	1	2	3
基尼指数（gini）	0.655172	0.896552	0.948276
信息熵（entropy）	0.586207	0.948276	0.965517

(2) 得到的六个决策树依次为（tree.plot_tree()描绘决策树）：

图 1 准则基尼指数，最大层数 1

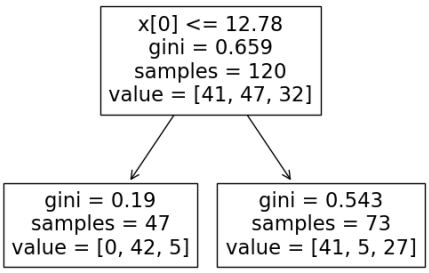

图 2 准则基尼指数，最大层数 2

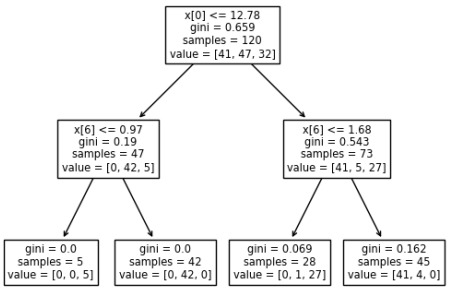

图 3 准则基尼指数，最大层数 3

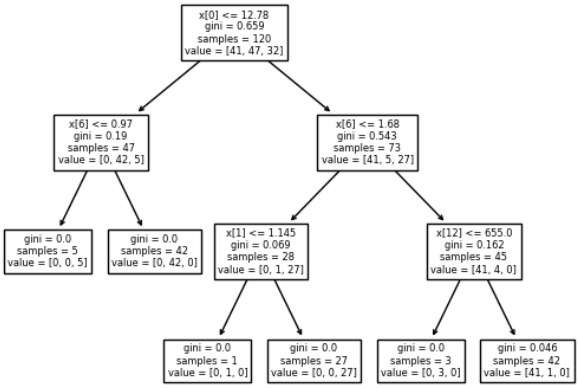

图 4 准则信息熵，最大层数 1

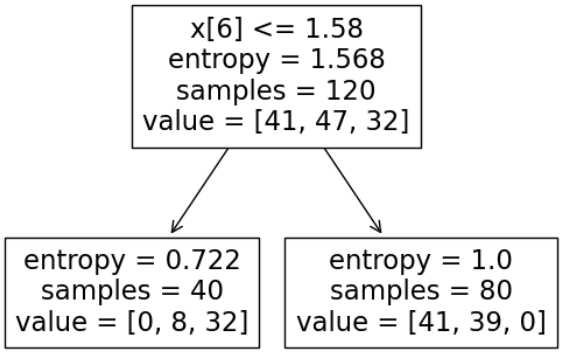

图 5 准则信息熵，最大层数 2

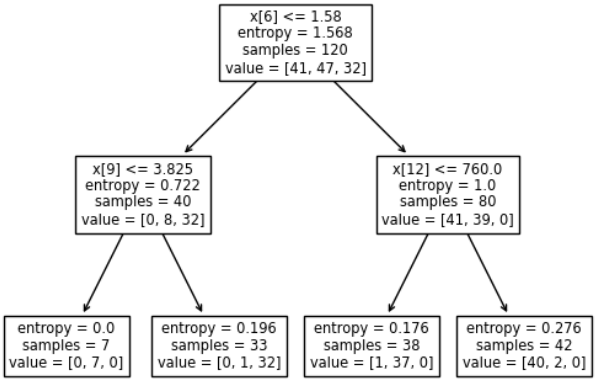

图 6 准则信息熵，最大层数 3