1. 决策树的建立与分类

建立模型与分类过程如下：

1. 将原始数据集转化成csv文件并存储为data.csv



1. 读取csv数据并序列化



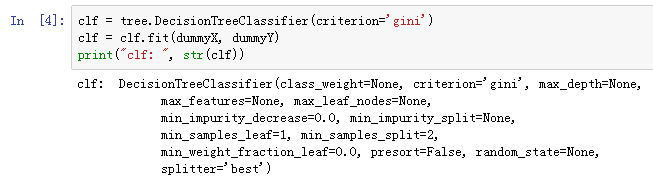
1. 将特征和类别编码为特征向量

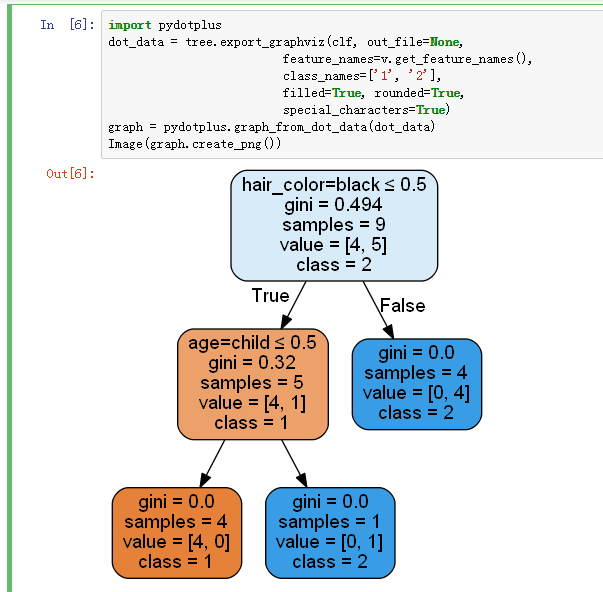


1. 建立决策树，采用CART算法。

第一步选取特征为“发色是否为黑色”，其gini不纯度为0.494，在所有的特征中最小，若发色为黑色，则分类为第二组，此时gini不纯度为0，该分支可停止；若发色非黑色，则继续询问“年龄是否为儿童”，其jini不纯度为0.32，若是儿童则分类为第一组，否则分类为第二组。至此，所有的叶结点的gini不纯度均为0，算法停止。

代码和决策树可视化如下：





1. 分类（矮，黑色，成年）

首先询问“发色是否为黑色”，结果为发色是黑色，因此分类为第二组。分类结束。

1. 朴素贝叶斯分类

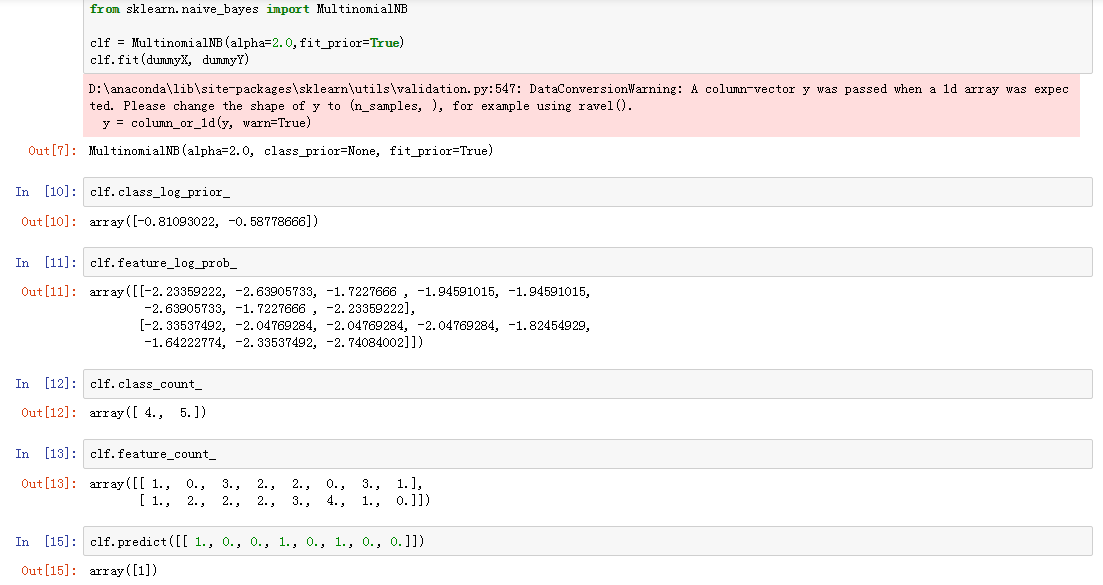
建立模型与分类过程如下：

代码输出中的所有概率均为取对数结果，采用伯努利离散朴素贝叶斯分类器，所有特征均为二值分布，且相互独立。

先验概率为[4/9, 5/9]，各个特征的后验概率的对数值可见代码中特征分布的输出。

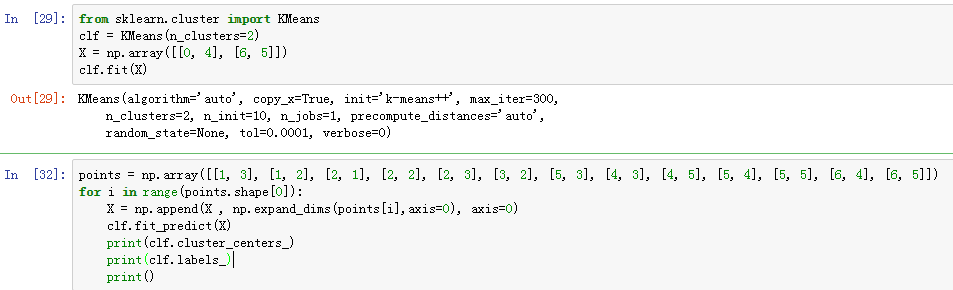
对给定样本的后验概率分别为：

1/2 \* 0 \* 1/4和2/5 \* 4/5 \* 1/5，因此根据朴素贝叶斯分类器，选取最大化后验概率作为分类结果，样本被分到第二组。



三、k-means聚类

初始化并逐个增加数据点到聚类模型中：



逐渐聚类的类中心和聚类结果如下，聚类的过程为对每个新输入的点根据其距离最近的类中心进行分类，然后归入该类，之后对所有的点迭代重复该操作，直到类中心的变化不超过阈值或到达迭代次数上限：

[[ 6. 5. ]

[ 0.5 3.5]]

[1 0 1]

[[ 0.66666667 3. ]

[ 6. 5. ]]

[0 1 0 0]

[[ 1. 2.5]

[ 6. 5. ]]

[0 1 0 0 0]

[[ 1.2 2.4]

[ 6. 5. ]]

[0 1 0 0 0 0]

[[ 1.33333333 2.5 ]

[ 6. 5. ]]

[0 1 0 0 0 0 0]

[[ 1.57142857 2.42857143]

[ 6. 5. ]]

[0 1 0 0 0 0 0 0]

[[ 1.57142857 2.42857143]

[ 5.5 4. ]]

[0 1 0 0 0 0 0 0 1]

[[ 1.57142857 2.42857143]

[ 5. 3.66666667]]

[0 1 0 0 0 0 0 0 1 1]

[[ 1.57142857 2.42857143]

[ 4.75 4. ]]

[0 1 0 0 0 0 0 0 1 1 1]

[[ 1.57142857 2.42857143]

[ 4.8 4. ]]

[0 1 0 0 0 0 0 0 1 1 1 1]

[[ 1.57142857 2.42857143]

[ 4.83333333 4.16666667]]

[0 1 0 0 0 0 0 0 1 1 1 1 1]

[[ 5. 4.14285714]

[ 1.57142857 2.42857143]]

[1 0 1 1 1 1 1 1 0 0 0 0 0 0]

[[ 1.57142857 2.42857143]

[ 5.125 4.25 ]]

[0 1 0 0 0 0 0 0 1 1 1 1 1 1 1]

对最后的聚类结果进行可视化如下：

