北京邮电大学软件学院

2019-2020 学年第一学期实验报告

课程名称： 数据挖掘

项目名称： 实验二：分类

项目完成人：

姓名： 刘子豪 **学号：** 2017211971

指导教师： 牛琨

日　　期： 2019 年 10 月 28 日

一、 实验内容

1. 实验内容如下：对给定数据集进行分类任务，并建立相应的分类器（如决策树）。分析分类结果指标，比较不同的实验结果，以生成最佳模型。
2. 解释你的模型。

二、 实验环境

系统环境：Windows 10

软件环境：WEKA 3.6.9

三、 实验结果

1. 类别标识选取

本实验中给出的数据集并没有给出已知的类别标识，因此作为替代，考虑选择数据集中的一个离散属性来作为此次分类的依据。

在选择类别标识的时候，不是任意选取的，需要从中进行筛选。与上一个实验——聚类相似，这次的分类实验要根据手机的各项配置信息，得到对于整个手机产品集的整体性描述。

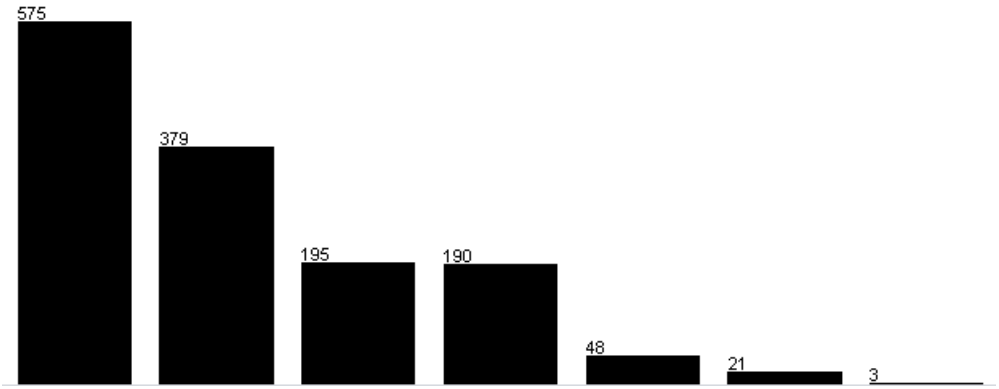
综上所述，**在选择类别标识的时候，应该保证这个属性能够对手机产品进行一般化的描述**。

″品牌″“上市时间”等显然不能作为类别标识，因为这些属性不是对手机配置信息的描述，只是对每个手机的标识。

“蓝牙”“FM 广播”等与手机具体配置信息有关的属性也不应该作为类别的标识，尽管用它们的时候分类效果很好。

依据这种原则，筛选出了下面几个属性：智能系统，市场定位，零售价格。

✧ **市场定位：**



该属性的数据是离散类型的，取值为 0~7。

根据这个属性的描述来看，它可以比较好的对手机产品进行一般化的描述，一种类别代表设计风格，硬件相似的一组手机。

但是虽然这个属性有 7 个取值，“5”“6”“7”这三类的数据量过少，甚至低于 50. 分类算法在建立这几个类的分类器时，缺少足够的输入数据支持，因此也就无法训练出一个很好的分类器了。所以，这个属性不适合作为本数据集的分类标识。

✧ 零售价格

No.	Label	Count	Weight
1	'(-inf-459.5]'	353	353.0
2	'(459.5-787]'	352	352.0
3	'(787-1292.5]'	353	353.0
4	'(1292.5-inf)'	353	353.0

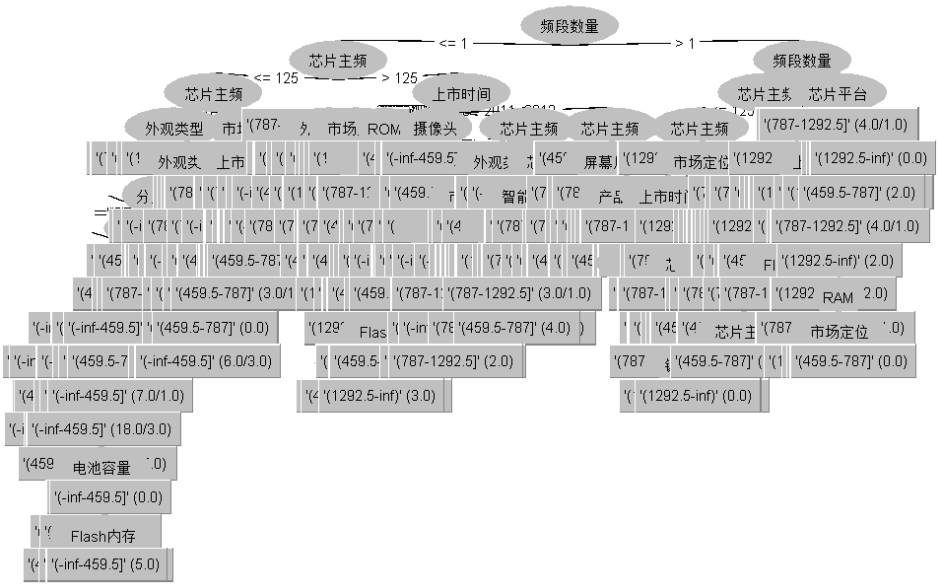
该属性的数据本身为连续数值类型，在本次实验中已将其离散化。

该属性可以在一定程度上反映手机的总体特征，零售价格高的手机在硬件配置上应该会高于价格低的手机。

使用决策树算法，以该属性为标识进行分类时，得出的测试集正确率为 53%，效果并不理想。

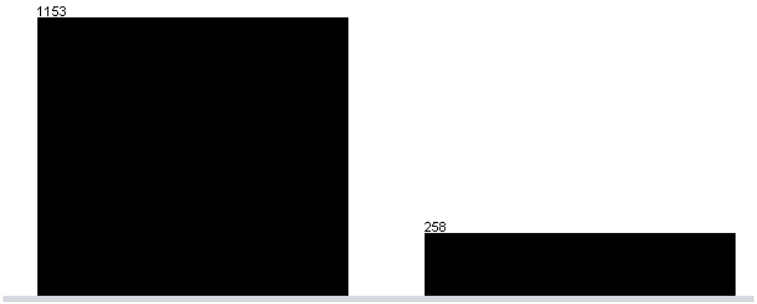
```
Correctly Classified Instances         750               53.1538 %
Incorrectly Classified Instances       661               46.8462 %
Kappa statistic                          0.3754
Mean absolute error                      0.2617
Root mean squared error                  0.4283
Relative absolute error                 69.7887 %
Root relative squared error             98.9049 %
Total Number of Instances             1411
```

而且算法生成的决策树模型也过于庞大，很难去理解分析。



综上所述，这个属性不适合作为该数据集的类别标识。

✧ 智能系统



该属性的数据为离散类型，取值为 0（无）和 1（有）。

虽然该属性描述的是手机是否具有智能的软件系统，但是该属性在一定程度上也能够反映手机的整体特征，配有智能系统的手机在硬件配置，功能等方面上会比一般的手机更强。

综上所述，选择的类别标识为“智能系统”。

2. 使用算法进行分类，并分析结果指标

本实验中，使用课上所讲的三种分类算法分别对该数据集进行分类，并比较这些算法的性能指标，最终确定最优的分类模型。

1. k 最近临近法 K-NN
   取 k=3，得到分类的结果如下：

```
Correctly Classified Instances         271               96.0993 %
Incorrectly Classified Instances        11                3.9007 %
Kappa statistic                          0.8533
Mean absolute error                      0.0664
Root mean squared error                  0.1924
Relative absolute error                 22.9127 %
Root relative squared error             51.9633 %
Total Number of Instances              282

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.983    0.152    0.971      0.983    0.977      0.854    0.955     0.983     0
                 0.848    0.017    0.907      0.848    0.876      0.854    0.955     0.879     1
Weighted Avg.    0.961    0.130    0.960      0.961    0.960      0.854    0.955     0.966

=== Confusion Matrix ===

   a   b   <-- classified as
 232   4 |   a = 0
   7  39 |   b = 1
```

2. 支持向量机 SVM 分类
   使用 weka 中的 SMO 方法（该方法能够为数据集构建 SVM），最后得到的结果如下：

```
Correctly Classified Instances         269               95.3901 %
Incorrectly Classified Instances        13                4.6099 %
Kappa statistic                          0.8382
Mean absolute error                      0.0461
Root mean squared error                  0.2147
Relative absolute error                 15.8958 %
Root relative squared error             57.9766 %
Total Number of Instances              282

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.962    0.087    0.983      0.962    0.972      0.840    0.937     0.977     0
               0.913    0.038    0.824      0.913    0.866      0.840    0.937     0.766     1
Weighted Avg.  0.954    0.079    0.957      0.954    0.955      0.840    0.937     0.943

=== Confusion Matrix ===

   a   b   <-- classified as
 227   9 |   a = 0
   4  42 |   b = 1
```

3. 决策树分类（采用 C4.5 算法）

使用 weka 中的 J48 方法（即 C4.5 算法）进行决策树的生成，最后得到的结果如下：

```
=== Summary ===

Correctly Classified Instances         273               96.8085 %
Incorrectly Classified Instances         9                3.1915 %
Kappa statistic                          0.8861
Mean absolute error                      0.0565
Root mean squared error                  0.1735
Relative absolute error                 19.4887 %
Root relative squared error             46.8423 %
Total Number of Instances              282

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.975    0.065    0.987      0.975    0.981      0.887    0.960     0.985     0
               0.935    0.025    0.878      0.935    0.905      0.887    0.960     0.886     1
Weighted Avg.  0.968    0.059    0.969      0.968    0.968      0.887    0.960     0.969

=== Confusion Matrix ===

   a   b   <-- classified as
 230   6 |   a = 0
   3  43 |   b = 1
```

下面，比较这三个算法的各项性能指标：

算法	正确分类数	相对误差	TP	FP
K-NN 算法	96.1%	23.0%	98.3%/84.8%	15.2%/1.7%
SVM 算法	95.4%	15.8%	96.2%/91.3%	8.7%/3.8%
决策树算法	96.8%	19.4%	97.5%/93.5%	6.5%/2.5%

根据表格内容可知，使用决策树算法获得的分类器，比其他两个算法的准确率更高，总体的误判率也比较低，各项性能指标都比较出众，因此采用决策树算法作为分类算法。

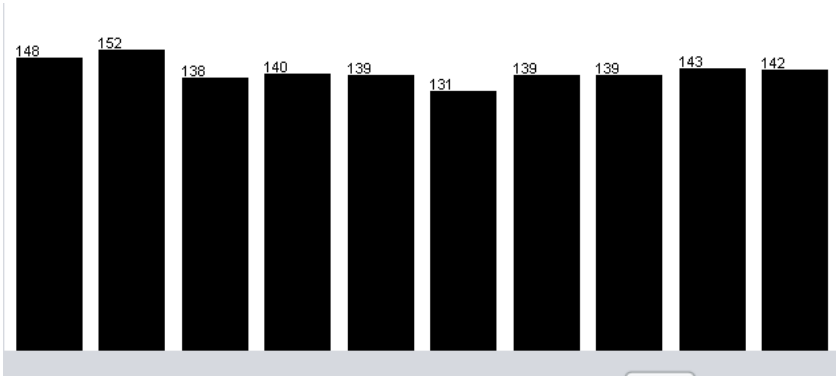
3. 算法的调优

在实验最开始的时候，数据集没有进行处理，使用这样的数据集分类的结果相对来说比较差：

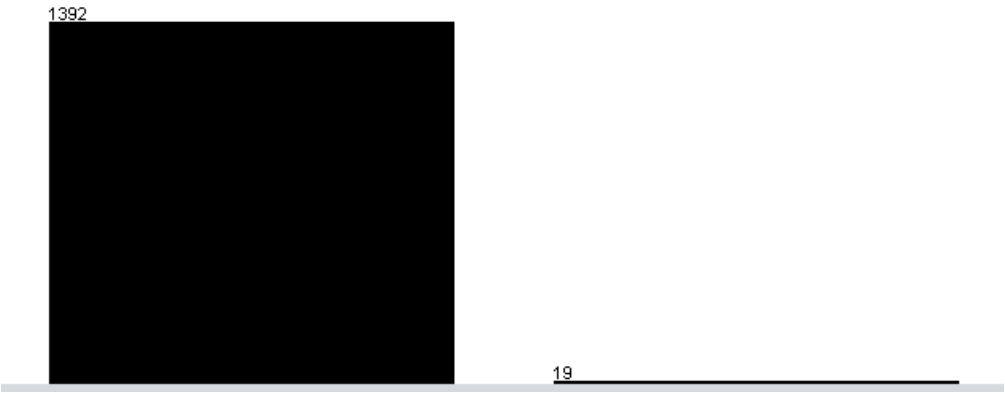
```
=== Summary ===

Correctly Classified Instances         262               92.9078 %
Incorrectly Classified Instances        20                7.0922 %
Kappa statistic                          0.8745
Mean absolute error                      0.0697
Root mean squared error                  0.1987
Relative absolute error                 18.1644 %
Root relative squared error             45.5214 %
Total Number of Instances              282
```

于是，在接下来的实验中着重对数据进行预处理，最大限度地排除数据质量对分类结果造成的影响。本次实验中，数据处理部分主要包括了下面的几个部分：

- 对连续数值的属性进行等频离散化，分成记录个数基本的几个区间（下图是对零售价格进行离散化的结果）：



- 删除一些对分类有干扰的属性，比如“品牌”“上市时间”等与手机配置信息无关的一些属性
- 删除一些二值属性，这些属性中两个取值的记录个数差距过于悬殊，比如“电视”“红外”（下图为“电视”属性）：



综合使用了上面的数据预处理方法，最终分类的准确率有一定的提高：

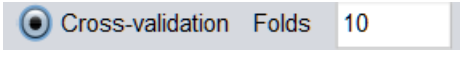
```
=== Summary ===

Correctly Classified Instances         273               96.8085 %
Incorrectly Classified Instances         9                3.1915 %
Kappa statistic                          0.8861
Mean absolute error                      0.0565
Root mean squared error                  0.1735
Relative absolute error                 19.4887 %
Root relative squared error             46.8423 %
Total Number of Instances              282
```

虽然准确率提高的幅度不是很大，但是在这个过程中，对数据的理解更加深刻，也能够更灵活使用数据预处理的方法，为以后的学习奠定基础。

4. 交叉验证集的使用

使用智能系统作为分类标识，最后结果中识别的准确率是非常高的，但这可能会出现过拟合的问题。为了防止过拟合的问题，使用了交叉验证集。

在 weka 中，使用 10 折交叉验证：



进行模型训练，得到的结果如下：

```
=== Summary ===

Correctly Classified Instances        1351               95.7477 %
Incorrectly Classified Instances        60                4.2523 %
Kappa statistic                          0.8537
Mean absolute error                      0.0633
Root mean squared error                  0.1967
Relative absolute error                 21.1691 %
Root relative squared error             50.8893 %
Total Number of Instances             1411
```

可以看到，模型识别的准确度相比之前的 96.8%有所下降，但是使用交叉验证集，能够保证分类器不是只对实验所用的数据集识别非常准确，而对一般的数据识别效果较差，从而提高分类器识别其他数据的准确度。

5. 最优模型结果解释

本实验中使用决策树算法训练，结合上文所说的数据预处理与交叉验证集的使用，作为该数据集最优的分类模型，该模型的各项性能指标如下：

```
=== Summary ===

Correctly Classified Instances        1351               95.7477 %
Incorrectly Classified Instances        60                4.2523 %
Kappa statistic                          0.8537
Mean absolute error                      0.0633
Root mean squared error                  0.1967
Relative absolute error                 21.1691 %
Root relative squared error             50.8893 %
Total Number of Instances             1411

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.982    0.151    0.967      0.982    0.974      0.855    0.919     0.963     0
                 0.849    0.018    0.913      0.849    0.880      0.855    0.919     0.868     1
Weighted Avg.    0.957    0.127    0.957      0.957    0.957      0.855    0.919     0.945

=== Confusion Matrix ===

    a    b   <-- classified as
 1132   21 |    a = 0
   39  219 |    b = 1
```

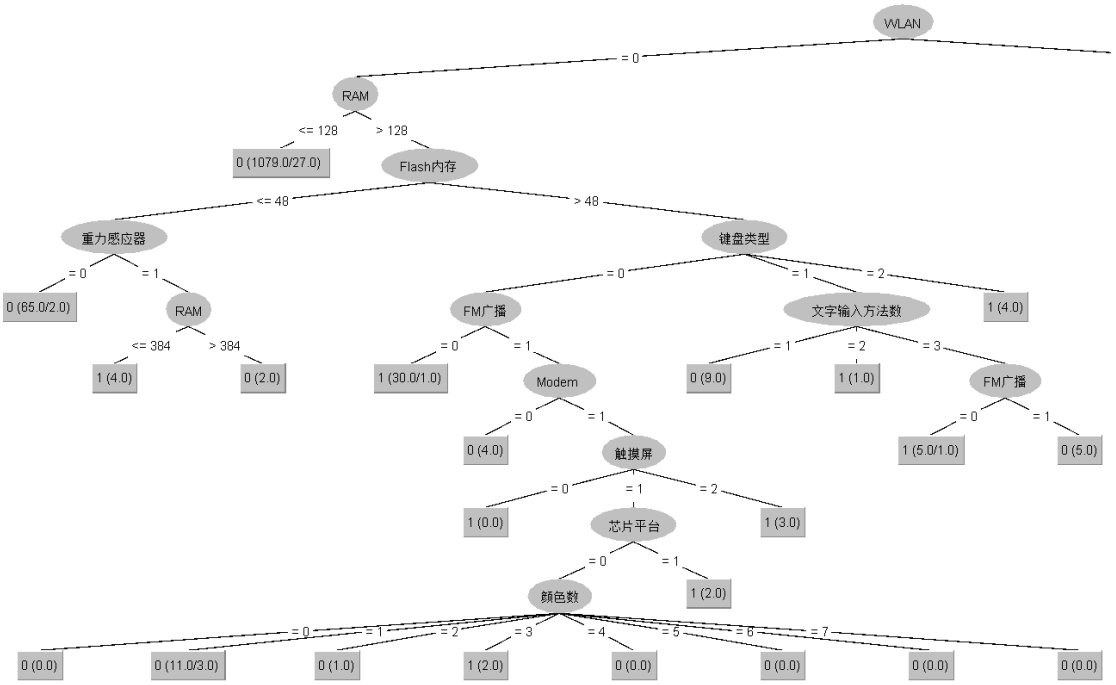
下面对 weka 给出的几个重要的性能指标进行分析（以上图的结果为例）：

- Correctly Classified Instance：测试集中分析正确的数据占比，本次分类中的准确率为 95.7%，可以说准确率较高。
- Relative absolute error：结合绝对误差与实例数得到的一个百分比，也能够衡量模

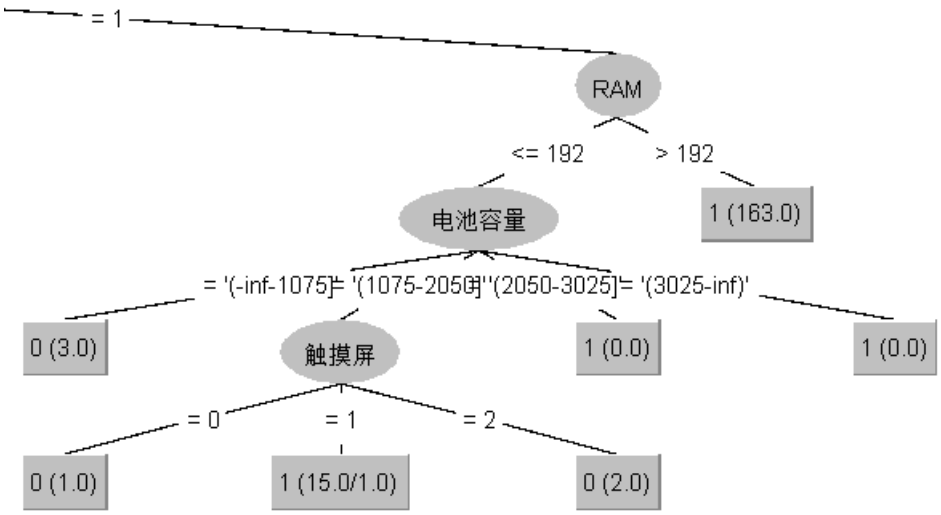
型测试的准确度，本次分类中该指标为 22%，比较理想。

- TP rate & FP rate：识别率和误判率，本次分类中两个类的误判率接近 0，说明基本上没有出现误判的情况。
- Confusion Matrix：以矩阵的形式给出了多少实例正确地被分到某个类中，错误地被分到某个类中。本次分类中，只有 39 个实例被错误地分到类 0 中，21 个实例被错误地分到类 1 中

生成的决策树如下：



决策树的左半部分



决策树的右半部分

对于一个配有智能系统的手机，肯定会含有 WLAN，触摸屏等基本功能器件，而且硬件配置各种参数也比较优良。在该决策数中，符合这样条件的手机产品被分到了类 1 中。因此，该决策树在现实角度上来看也是比较合理的，它比较好地给出了基于手机各项配置信息来判断是否具备智能软件系统的依据。

至此，对最优模型的解释结束。