

用户行为预测

一、小组成员

赵颖 2620160012

王晓媛 2620160007

李昱燃 2620160009

二、概述

我们欲分三步解决实验提出的问题：数据预处理，分类算法的选择，分类算法的实现。首先，对已有的 10000 条数据进行预处理，选择合适的特征变量，方便分类器的训练，我们使用 java 编程实现了这一过程。其次，出于提高准确率、加快训练速度的目的，需要选择适当的分类算法，最终确定了随机森林、与 AdaBoost 结合的决策树、梯度提升决策树三种算法。最后，实现分类算法并进行分类器的训练，这一步骤还未完成。

三、项目进展

1、实验环境

JDK+eclipse：用 java 语言进行数据预处理和特征选取。

Python：用数据挖掘中常用的 python 进行训练、分类和预测。

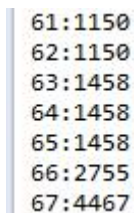
2、数据预处理

给定的训练集共 10000 条数据，每条数据分别包括 230 个特征值，特征值的前 190 个为数值类型，后 40 个为 string 字符串类型，特征值之间以逗号（,）分隔，无特征值时空（" "）。

从 230 个特征值中选择特征值出现次数最多，较为完整的 40 余特征值进行训练，其余数据由于完整度不够被删去。选择特征值的方法如下：

- a) 使用 java 编写程序，从 train.txt 中按行读入数据，每行记为 lineTxt，由于数据之间严格采用“,”分隔，故可调用函数 split(), 得到 tempArray 数组，其元素为字符串类型，其长度为 230，其中的值为空或为某一特征值。
- b) 定义全局变量 sum 数组用于统计每个特征值的缺失次数，sum 为 int[230] 类型，初始值皆为 0，对 tempArray 数组中的元素逐个进行判断，若 tempArray[i] 值为空，则令 sum[i] 加 1，否则不变。

- c) 统计结束后，为了在获得其中缺失次数的信息的同时保存其所在列的信息，将数组复制到另一数组 s 中，对 s 进行排序，调用 Arrays.sort(s) 函数并输出，对输出数据进行观察统计，可得到下图结果：



```
61:1150
62:1150
63:1458
64:1458
65:1458
66:2755
67:4467
```

图 1 数组 s 的标号与对应值

- d) 由图中统计数字不难发现，在缺失次数超过 1500 次后，缺失次数过多，且急剧增长，不适合用于分类器的训练，因此仅考虑缺失次数不超过 1500 次的特征值所在列，在限定缺失次数不超过 1500 次的情况下对 sum 数组再次进行遍历，从而得到缺失次数较少的特征值所在列。结果如下（数字表示列号）：

[5, 6, 12, 20, 21, 23, 24, 27, 34, 37, 43, 56, 64, 71, 72, 73, 75, 77, 80, 82, 84, 93, 108, 111, 112, 118, 122, 124, 125, 131, 132, 133, 139, 142, 143, 148, 152, 159, 162, 172, 180]。

3、算法选择

首先对实验中数据集的特性进行了分析。在使用的数据集中，训练数据和测试数据各 10000 个，数量比较大；而且该数据集中的 230 个特征变量并不是同一类型的，而是 190 个数值型变量和 40 个类别型变量的组合；而且数据集中有大量数据缺失。而后分析了竞赛中选手们选用的算法。参赛者使用最多的几种分类算法降序排列如下：决策树集合类的分类算法（包括随机森林、与 AdaBoost 结合的决策树算法、梯度提升决策树等算法）、线性分类算法（特别是逻辑回归算法）、支持向量机。可见这几种算法对于该数据集的处理有一定优势。同时，结合决策树集合分类算法的几个优势：在大数据集上的表现良好，运行速度快；处理不同类型的变量比较容易；有大比例的数据缺失时，也能保持较高的准确度；实现简单、鲁棒性好。我们最终选择了与 AdaBoost 结合的决策树、梯度提升决策树、随机森林三种算法进行实验。

四、待完成的工作

之后，我们将完成分类算法的实现工作。基于所选的梯度提升决策树、与 AdaBoost 结合的决策树、随机森林三种算法，由训练数据生成对应的分类器，用测试集分别对客户的忠诚度、消费欲和增值服务倾向性做出二元判别，并计算各分类器的准确率，最后将各个结果进行比较分析。