### 太平反欺诈：算法

本文档将从

1. 算法的背景，即以前算法的总体思路和结构设计
2. 算法的改进，即现在算法的逻辑计算
3. 算法的不足，即以后可以改进的地方

这三个方面来展开讨论。

#### 算法的背景

太平反欺诈的总体思路分为两个步骤：

1. 对于某疾病，将该病下的总样本依据某些**条件**划分成若干子样本，使得划分后的样本有：子样本之间的**费用**分布有差异，呈水平；子样本内部的**费用**分布保持稳定和趋同，以正态分布为目标。  
   对于划分后的子样本，寻找这些样本除上述**条件**以外的共同特征，具体为：在医院做过的项目，并发症的种类和频率，投的是哪些保险。将这些共同特征以表格和数据的形式存储起来，组成一套（四个）模板。
2. 现有一批新增的数据，我们根据这批数据中每一个主体的**费用**和对应于模板的信息，来判断这个主体应该属于哪一个模板，并和该模板的相似度有多高。得出的相似度为：该主体与其最匹配的模板的相似程度。

该模型的总体思路是没有错误的，但是具体的计算过程中，存在许多逻辑错误。为配合该项目的具体情况，如不能改变其算法流程和表结构，后续的算法编写保持该总体思想不变，但是会更改具体的实现方式。

#### 算法的改进

对于原来的计算方法，首先更改的是：如何将总样本根据**条件**划分成子样本。使用切割的方法是不可取的，因为其涉及到数量巨大的切割顺序，无法完全模拟。此外，使用切割的方法将无法获取某些特定的子样本。

##### 划分样本

## Warning: package 'dplyr' was built under R version 3.3.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

本次所使用的方法为：依据**条件**，将总样本切割为不可切割的最小状态，即不能再分的最小样本。考虑如下的假设数据集：

## 'data.frame': 1000 obs. of 5 variables:  
## $ ID : num 1 2 3 4 5 6 7 8 9 10 ...  
## $ A : Factor w/ 3 levels "1","2","3": 2 3 1 1 3 3 2 2 2 3 ...  
## $ B : Factor w/ 2 levels "1","2": 2 1 2 2 1 2 2 1 1 2 ...  
## $ C : Factor w/ 3 levels "1","2","3": 3 3 2 2 1 1 3 3 3 1 ...  
## $ ZFY: num 9408 10045 10263 10265 10041 ...

该假设数据集中，可用于将总样本划分为子样本的**条件**为数据集的中间三列（A、B、C），即三个**条件**，每个**条件**下有2个以上的水平，如下：

A: A1 A2 A3  
B: B1 B2  
C: C1 C2 C3

以ID为主键，每个ID代表一个主体，从数据库中抽取出该主体的三个**条件**（A、B、C）的信息，合并到一张表格中，其结构应该与上述假设数据集类似。

## ID A B C ZFY  
## 1 1 2 2 3 9407.921  
## 2 2 3 1 3 10044.699  
## 3 3 1 2 2 10262.862  
## 4 4 1 2 2 10265.188  
## 5 5 3 1 1 10040.563  
## 6 6 3 2 1 9946.733

每一行即为一个主体。第一列*ID*为主体对应的索引。主体必然属于三个条件下若干水平的组合中的一种。每一个主体在*ZFY*列中对应有一条费用记录，即**费用**。

该数据集理论上的**最小样本**数，等于理论上该数据集的三个条件的所有水平下可能的组合数：

最小样本数 = 水平组合数 = 3(A水平数) \* 2（B水平数） \* 3(C水平数) = 18

然而，存在某种水平组合不存在的情况，特别是当样本偏少的时候。但是可以得知，任何样本都可以根据其**条件**划分为可穷尽的**最小样本**。我们对上述假设数据集进行整合，得出其18个**最小样本**。

## index A B C freq  
## 1 1 1 1 1 61  
## 2 2 1 1 2 68  
## 3 3 1 1 3 56  
## 4 4 1 2 1 59  
## 5 5 1 2 2 50  
## 6 6 1 2 3 56  
## 7 7 2 1 1 51  
## 8 8 2 1 2 64  
## 9 9 2 1 3 50  
## 10 10 2 2 1 60  
## 11 11 2 2 2 51  
## 12 12 2 2 3 60  
## 13 13 3 1 1 54  
## 14 14 3 1 2 57  
## 15 15 3 1 3 55  
## 16 16 3 2 1 61  
## 17 17 3 2 2 49  
## 18 18 3 2 3 38

每一行代表一个**最小样本**，故每一行还对应一个字符串向量和一个数值向量。前者存储该**最小样本**对应的**主体**ID是哪一些，后者存储这些ID的**费用**为多少。 如第一行（A = 1, B = 1, C = 1下的**最小样本**）所对应的两个向量为:

print(atom.data[1, 'ID.data'])

## [1] "12,33,36,45,104,114,119,155,158,172,181,227,248,253,261,262,293,319,339,383,384,395,401,427,429,432,434,441,448,452,460,479,483,490,568,576,577,592,595,615,620,627,635,636,652,657,658,662,679,685,734,742,758,764,795,801,803,825,863,923,970"

print(atom.data[1, 'ZFY.data'])

## [1] "9965.79,10834.26,11019.41,10199.21,10684.98,10120.48,10849.74,10711.29,10872.9,10774.56,10563.39,10732.61,10981.5,10422.32,10823.35,11045.75,10836.12,10849.08,10925.13,10860.39,10788.64,11006.54,10459.34,10930.86,11327.44,10907.19,11173.09,10503.1,10859.96,10832.28,10169.98,10726.19,10767.43,10806.74,11167.47,10912.03,10996.49,10853.09,10882.03,11758.24,10890.68,10783.04,10856.5,11634.67,10856.78,11041.51,10997.24,10955.84,11020.95,11049.68,10939.36,11309.45,11044.74,11167.51,11233.86,11111.6,11429.27,10739.5,11142.1,11117.98,12130.03"

根据可穷尽的**最小样本**数据，以*index*为最小样本的索引，*ZFY.data*为每个最小样本下的总费用数据集。要将这些最小样本聚合起来，我们就必须要找到一种聚类的方法，其必须能处理两个长度不一样的数值向量。

##### 聚类的方法

一般的聚类方法依赖于距离的度量，需要聚类的对象之间必须有相同的属性，比如：两者均为二维数据。在本例中，两个样本之间聚类的标的是关于**费用**的数值向量，长度不等。于是本文能提供的聚类方法如下：

1. 不依赖传统的距离度量方法。令一个数值向量为一个主体，两个向量（A、B）之间的距离定义为:将两个向量合并后的方差值。以此构造出所有向量的方差矩阵，来找到组合后方差最小的向量组合，并将该组合对应的两个样本合并，形成新的子样本，并重新计算矩阵。迭代并使得最后子样本的个数达到指定值或满足某个条件。
2. 找到一种方法，让数值向量标准化，标准化后所有向量均拥有相同的属性。构造向量完成后，可以使用各种距离度量方法来聚合，迭代并使得最后子样本的个数达到指定值或满足某个条件。

鉴于第一种方法的计算开销太大，本文使用第二种方法来进行。

##### 构造向量

构造的向量必须在很大程度上能概括原有数值向量的信息，才能够用于聚类。故不可以用原数值向量的：均值、方差或概括性的数值。

本文使用的方法为：为每个数值向量构造一个轮廓向量，其主要概念是离散化及确定数值向量的分布模式。确定分布模式的目的是为了比较数值向量之间的差异，这就需要量化。将数值向量依据其大小分配到若干个等级中。

求出上述18个**最小样本**的轮廓矩阵（样例），如下：

## index 下极值 1 2 3 4 5 上极值  
## [1,] 1 0 7 15 67 8 3 0  
## [2,] 2 0 10 28 29 26 6 0  
## [3,] 3 0 7 21 41 21 9 0  
## [4,] 4 0 5 22 41 29 3 0  
## [5,] 5 0 8 14 36 36 6 0  
## [6,] 6 0 2 25 38 36 0 0  
## [7,] 7 0 0 27 43 24 6 0  
## [8,] 8 0 5 30 39 20 6 0  
## [9,] 9 0 12 26 40 20 2 0  
## [10,] 10 0 7 23 50 13 5 2  
## [11,] 11 2 4 24 41 27 2 0  
## [12,] 12 2 5 28 33 30 2 0  
## [13,] 13 0 4 28 31 22 13 2  
## [14,] 14 2 2 25 51 18 4 0  
## [15,] 15 0 9 25 42 11 13 0  
## [16,] 16 2 7 26 36 25 5 0  
## [17,] 17 0 4 8 49 37 2 0  
## [18,] 18 0 5 21 42 24 8 0

##### 距离度量

如何根据上述指标来求出两个模式之间的差异，即距离的选择，是聚类的关键之二。本次分析使用余弦相似度来度量，特点如下：

* 对异常值敏感
* 不需要去除量纲，即不受比例的影响

生成的距离矩阵如下：

## [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]  
## [1,] 0.00 0.23 0.08 0.13 0.20 0.20 0.11 0.12 0.10 0.03  
## [2,] 0.23 0.00 0.04 0.03 0.06 0.04 0.04 0.03 0.03 0.10  
## [3,] 0.08 0.04 0.00 0.02 0.05 0.05 0.01 0.02 0.02 0.02  
## [4,] 0.13 0.03 0.02 0.00 0.03 0.01 0.01 0.03 0.02 0.05  
## [5,] 0.20 0.06 0.05 0.03 0.00 0.03 0.07 0.09 0.08 0.13  
## [6,] 0.20 0.04 0.05 0.01 0.03 0.00 0.03 0.05 0.06 0.11  
## [7,] 0.11 0.04 0.01 0.01 0.07 0.03 0.00 0.01 0.03 0.04  
## [8,] 0.12 0.03 0.02 0.03 0.09 0.05 0.01 0.00 0.01 0.04  
## [9,] 0.10 0.03 0.02 0.02 0.08 0.06 0.03 0.01 0.00 0.03  
## [10,] 0.03 0.10 0.02 0.05 0.13 0.11 0.04 0.04 0.03 0.00

根据距离矩阵，使用层次聚类方法，将**最小样本**聚合成合适的类的个数。

##### 实际情况

在实际情况中，对应于上述的**条件**，我们选取以下条件：

## [1] "年龄" "事故原因代码" "事故结果"   
## [4] "年龄等级" "婚姻状况" "案件类型事故性质"  
## [7] "是否有驾照" "是否吸烟" "性别"   
## [10] "医院等级" "住院天数"

聚类完成后，根据ID，寻找每一类对应的并发症、项目、险种责任数据，并将其汇总，求出其出现频率。

#### 改进的空间