**不同设备部件故障的关联分析模型使用说明**

1. 模型用途

设备某一部件发生故障后，判断它会导致哪些部件也发生故障。输出其他发生故障的设备部件。

1. 数据说明

输入数据用变量x表示，部件个数为n，共有n个特征，x(n)表示该部件是否发生故障（1表示发生故障，0表示未发生故障）。

1. 算法原理

不同设备部件故障的关联分析模型的核心思想是利用FP-growth算法只对数据库进行两次遍历，从而高效地发现频繁项集，构建FP树。有了FP树之后，就可以抽取频繁项集了。首先从单元素项集合开始，然后在此基础上逐步构建更大的集合。

从FP树中抽取频繁项集的三个基本步骤如下：

1. 从FP树中获得条件模式基；

2. 利用条件模式基，构建一个条件FP树；

3. 迭代重复步骤1步骤2，直到树包含一个元素项为止。

（一）数据预处理

不同设备部件故障的关联分析模型的输入数据共有七个维度，需要将每一个维度的数据和输出结果进行离散化，再进行one-hot编码，以适应关联分析算法本身和python工具包的要求。

（二）数学建模

FP-growth算法将数据存储在一种称为FP树的紧凑数据结构中。FP代表频繁模式（Frequent Pattern）。一棵FP树看上去与计算机科学中的其他树结构类似，但是它通过链接（link）来连接相似元素，被连起来的元素项可以看成一个链表。与搜索树不同的是，一个元素项可以在一棵FP树种出现多次。FP树辉存储项集的出现频率，而每个项集会以路径的方式存储在数中。存在相似元素的集合会共享树的一部分。只有当集合之间完全不同时，树才会分叉。 树节点上给出集合中的单个元素及其在序列中的出现次数，路径会给出该序列的出现次数。相似项之间的链接称为节点链接（node link），用于快速发现相似项的位置。为构建FP树，需要对原始数据集扫描两遍。第一遍对所有元素项的出现次数进行计数。数据库的第一遍扫描用来统计出现的频率，而第二遍扫描中只考虑那些频繁元素。

1. 使用方法

运行model.py文件得到不同部件发生故障之间的规则。

具体输入命令：

**python model.py --train\_dir 训练数据存放地址 --rules\_dir 结果保存地址 --support 最低支持度 --confidence 最低置信度**

例如：

**python model.py --train\_dir ./trainDataGZ.txt --rules\_dir ./rulesDataGZ.txt --surport 0.05 --confidence 0.2**

分析结果会存放在**结果保存地址**中