数据挖掘大作业二：关联规则挖掘

姓名：孙澈

学号：2102171054

1. 数据源

从大作业一的两个数据集中任选一个进行分析。（我选择了数据集 2: San Francisco Building Permits 数据集合）

2. 要求

1. 对数据集进行处理，转换成适合关联规则挖掘的形式；
2. 找出频繁项集；
3. 导出关联规则，计算其支持度和置信度
4. 对规则进行评价，可使用 Lift，也可以使用教材中所提及的其它指标

3. 提交的内容

- 3.1 对数据集进行处理的源程序

采用 San Francisco Building Permits 数据集（下简称为 permits 数据集）本身保存为 csv 格式，根据数据集的属性解释得知该数据集有 43 个属性，根据作业一的初步分析，可以得出该数据集的属性 Permit Type—Permit Type Definition、Existing Construction Type—Existing Construction Type Definition 和 Proposed Construction Type—Proposed Construction Type Definition 这三对儿属性是分别一一对应的，没有挖掘关联规则的必要，需要从数据库中剔除，除此之外，还需要将属性名称和属性内容连在一起（因为尽管一些属性都是用数值表示的，但因为属性不同而具有不同的意义）。处理好后，存储成以逗号隔开的 csv 文件便于后一步处理。

处理代码见 Sorter.m（使用 matlab 语言写的）,它读取原始数据，然后做上面提到的处理。处理后的示例为 Permit Number:201505000000，冒号前面是属性名字，冒号后面是属性内容。

- 3.2 关联规则挖掘的源程序

采用 Apriori 算法，它是一种挖掘关联规则的频繁项集算法。（代码见 Apriori.py 文件，是用 python 语言写的）

首先找出所有的频繁项集，在代码中用函数 L1()实现，这些项集出现的频繁性大于等于预定义的最小支持度。接下来由频繁项集产生强关联规则（满足定义的最小支持度和可信度），在代码中由 generateCK()和 generateLK()函数实现。最后使用该期望规则产生只包含集合的项的规则（右边只有一项），保留大于等于可信度的规则，同时输出可信度、支持度以及提升度(Lift)，在代码中由 rulegeneratir()函数实现。具体总体代码见 Apriori.py 文件

- 3.3 挖掘结果及分析

因为数据集的数据量太大了，有 198900 条数据，属性处理后也有 39 个，这使得使用 Apriori 算法的时间和空间过于庞大，因此我在处理数据的时候通过随机采样的方法生成挖掘的样本，分别选择了 2000、5000 和 10000 个挖掘样本，观察是否有明显变化(结果分别存在 result_2000、result_5000 和 result_10000 文件下)。下面以 2000 大小的样本为例，展示挖掘结果。
频繁项集（>>>左侧是属性，右侧是支持度，只粘贴了部分结果，因为支持度设置的偏小，存储在 FItems.txt 文件中）
Issued Date:01/07/2013 >>> 127

Filed Date:01/14/2013 >>> 133

Permit Expiration Date:01/02/2014 >>> 95

Existing Use:1 family dwelling >>> 433

Estimated Cost:1 >>> 132

Permit Expiration Date:01/09/2014 >>> 80

Existing Use:apartments >>> 341

Description:street space >>> 348

First Construction Document Date:01/09/2013 >>> 94

('Plansets:0', 'Proposed Use:1 family dwelling', 'Street Suffix:St') >>> 99

('Existing Construction Type Description:wood frame (5)', 'Permit Number:201301000000', 'Supervisor District:5') >>> 86

('Filed Date:01/10/2013', 'Permit Creation Date:01/10/2013', 'Street Suffix:St') >>> 82

('Existing Construction Type Description:constr type 1', 'Proposed Units:0', 'Proposed Use:office') >>> 118

('First Construction Document Date:01/11/2013', 'Issued Date:01/11/2013', 'Permit Creation Date:01/11/2013') >>> 92

('Current Status:complete', 'Existing Construction Type Description:wood frame (5)', 'Proposed Use:apartments') >>> 185

('Current Status:complete', 'Existing Construction Type Description:wood frame (5)', 'Number of Proposed Stories:3') >>> 227

('First Construction Document Date:01/07/2013', 'Permit Type Definition:otc alterations permit', 'Street Suffix:St') >>> 82

('Description:reroofing', 'Number of Existing Stories:2', 'Plansets:0') >>> 89

('Number of Existing Stories:3', 'Proposed Use:2 family dwelling', 'Street Suffix:St') >>> 65
关联规则(->左侧小括号里的的数字表示左边的支持度，引号中的是属性值，->右侧中括号里面的属性值，后面小括号的数字表示支持度，两个中括号里面的数字分别表示置信度和提升度)
['Current Status:issued', 'Plansets:0', 'Proposed Construction Type Description:wood frame (5)', 'Street Suffix:St'] (79) -> ['Permit Type Definition:otc alterations permit'] (1852) [1.0][1.0]
['Current Status:complete', 'Existing Units:1', 'Number of Existing Stories:2', 'Permit Number:201301000000', 'Permit Type Definition:otc alterations permit', 'Street Suffix:Av'] (93) -> ['Proposed Units:1'] (419) [0.87][0.52]
['Current Status:complete', 'Number of Existing Stories:2', 'Permit Number:201301000000', 'Permit Type Definition:otc alterations permit', 'Proposed Units:1', 'Street Suffix:Av'] (93) -> ['Existing Units:1'] (430) [0.65][0.45]
可以看出来，后两条关联规则实质上很接近，只是'Proposed Units:1'和'Existing Units:1'在左右两边换了位置。