读书报告内容：（复制下面部分到一个新文档，名称命名为自己的名字，填写完交给组长，没做第四部分可以不复制第四部分）

1. （必填）自己提出的问题的理解（罗列全部）：
2. 提出的问题1：GSP算法很直白，但是也有缺点，就是每次计算序列的支持度时，都需要全表扫描数据集D，有没有改进的方法

讨论后的理解：可以使用，SPADE算法。SPADE与GSP算法大体相同，只不过由于它多了一个ID\_LIST记录，使得每一次的ID\_LIST根据上一次的ID\_LIST得到（从而得到支持度），而ID\_LIST的规模是随着剪枝的不断进行，而规模逐渐缩小的。所以也就解决了GSP算法多次扫描数据集D问题。

1. （必填）别人提出的问题的理解（选择几个问题罗列，并给出理解）：
2. 问题2：序列模式挖掘出频繁序列假设为 <A,B> ,这表明A发生后隔一段时间可能发生B，但隔一段时间到底是隔多长时间呢？换句话说，如何给序列挖掘添加时限约束

自己的理解：有两种约束方法，最大跨度约束及最小间隔和最大间隔约束1.最大跨度约束  
最大跨度约束指定整个序列中所允许的事件的最晚和最早发生时间的最大时间差。一般地，最大时间跨度maxspan越长，在数据序列中检测到模式的可能性就比较大。然而较长的maxspan也可能捕获不真实的模式，因为增加这两个不相关的事件成为时间相关事件的可能性，此外，模式也可能涉及陈旧事件。2.最小间隔和最大间隔约束时限约束也可以通过限制序列中两个相继元素之间的时间差来指定，使用最大间隔约束的一个旁效就是可能违反先验原理。可能会出现序列中事件数增加时，支持度也增加，这就违背了先验原理。

1. 问题3：prefixspan算法的核心思想是什么？

自己的理解： PrefixSpan算法的目标是挖掘出满足最小支持度的频繁序列。那么怎么去挖掘出所有满足要求的频繁序列呢。回忆Aprior算法，它是从频繁1项集出发，一步步的挖掘2项集，直到最大的K项集。PrefixSpan算法也类似，它从长度为1的前缀开始挖掘序列模式，搜索对应的投影数据库得到长度为1的前缀对应的频繁序列，然后递归的挖掘长度为2的前缀所对应的频繁序列……以此类推，一直递归到不能挖掘到更长的前缀挖掘为止。

1. 问题4：P41最后一行，经过算法计算之后得到的序列模式集合为包括{<{30}，{40}>,<{40}，{30}>}这样的序列，算不算是矛盾呢？需不需要考虑删除呢？

自己的理解：一个项在项集中最多出现一次，但是一个时间可以在序列中出现多次。

而且次序在序列中很重要，但是在项集中不重要。如{1, 2}和{2, 1}表示同一个项集，但是<i1i2><i1i2>和<i2i1><i2i1>表示不同序列。所以{<{30}，{40}>,<{40}，{30}>}是不算矛盾的，不需要删除。

1. （必填）读书计划

1、本周完成的内容章节：如2.6-2.9

2、下周计划：5.1

四、读书小结

2.6序列模式的基本概念

给定一个由不同序列组成的集合，其中，每个序列由不同的元素按顺序有序排列，每个元素(交易)由不同项目组成，同时给定一个用户指定的最小支持度阈值，序列模式挖掘就是找出所有的频繁子序列，即该子序列在序列集中的出现频率不低于用户指定的最小支持度阈值。

2.7基于GSP挖掘序列模式

GPS算法是类Apriori算法。用于从候选项集中发现具有时序先后性的频繁项集。两个步骤：进行自连接、进行剪枝。缺点：每次计算支持度，都需要扫描全部数据集；对序列模式很长的情况，由于其对应的短的序列模式规模太大，算法很难处理。可以使用SPADE算法把扫描的结果存入列表。

2.8基于 prefixspan算法的序列模式挖掘

prefixspan算法流程：

1）找出所有长度为1的前缀和对应的投影数据库

2）对长度为1的前缀进行计数，将支持度低于阈值α的前缀对应的项从数据集S删除，同时得到所有的频繁1项序列，i=1.

3）对于每个长度为i满足支持度要求的前缀进行递归挖掘

PrefixSpan算法由于不用产生候选序列，且投影数据库缩小的很快，内存消耗比较稳定，作频繁序列模式挖掘的时候效果很高。比起其他的序列挖掘算法比如GSP,FreeSpan有较大优势，因此是在生产环境常用的算法。

2.9从序列模式中生成规则

序列规则（Sequential Rules）：可表述为蕴含式形式：X→ Y，X是Y的一个子序列，支持度和置信度类似先前定义。给定minsuo和minconf，可以从序列模式中得到序列规则。

标签序列规则（Label Sequential Rules）：可表述为蕴含式形式：X → Y ，X是将Y中部分项目替换成通配符得到的序列。替换掉的项目称作标签（Label）。

分类序列规则（Class Sequential Rules）：类似于CAR，可表述为蕴含式形式：X→ y，y为Y元素，X为序列，Y为分类标记集合。