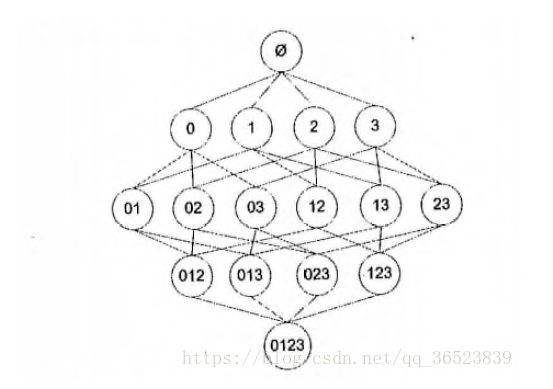
1. （必填）自己提出的问题的理解（罗列全部）：
2. 提出的问题1：P18 怎么得到Apriori算法是指数级的？

讨论后的理解：Apriori扫描物品之间的所有可能的组合，如下图所示，从上往下一个集合是，表示不包含任何物品的空集，物品集之间的连线表明两个或者更多集合可以组合形成一个更大的集合。所以如果项目集I中项目的数目是m，则整个项目集合空间的规模将达到，为指数级。



1. 提出的问题2：P30最后一段，如果一个规则项具有100%的置信度，那么再往这个规则项里扩展更多的前件，置信度依然保持在100%？如果是分类关联规则挖掘，加入前件之后可能会直接改变该属性的分类，置信度还会是100%吗？

讨论后的理解：因为该算法仅将具有相同分类的规则项通过合并他们的条件集进行合并，所以不存在加入前件之后改变属性分类的情况，则置信度依然保持在100%。

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3：假定各个项目的MIS都由用户指定，那么参数φ应该怎么考虑呢？为什么要设置该参数呢（即在满足各自MIS的情况下，支持度相差很大的事物为什么要过滤掉）？P25

自己的理解：引入支持度差别限制，就是为了不把同时含有频繁项目和稀有项目的项集作为频繁项目集生成，这个限制可以将最终生成的频繁项目集数量大大减少，而且还不会影响向下封闭属性。因为如果不同项目之间的出现频率相差很大，则为了同时包含出现频率高和低的项目，最小支持度不得不设置的很小，这样的话会导致组合爆炸，因为符合要求的频繁项目集和关联规则将以指数级的速度增长，从而使得挖掘过程根本没有办法进行。而且同时含有频繁项目和稀有项目的项集没有实际的应用意义。

1. 问题4：我感觉Apriori太暴力了，有没有一些其他的优于Apriori的算法？

自己的理解：有，比如FP-growth算法。这种算法比Apriori算法要快，基于Apriori构建，但在完成相同任务时采用了一些不同的技术。不同于Apriori算法的”产生－测试”，FP-growth算法的任务是将数据集存储在一个特定的称做FP树的结构之后发现频繁项集或者频繁项对，即常在一块出现的元素项的集合FP树,这种做法使得算法的执行速度要快于apriori，通常性能要好两个数量级以上。

1. （必填）读书计划

1、本周完成的内容章节：如2.1-2.5

2、下周计划：2.6-2.9

四、（选做）读书摘要及理解或伪代码的具体实现（读书摘要、伪代码的具体实现代码等可以写到这个部分）

1、读书摘要及理解（选做）

