###仓库说明：

\*\*2120151040 王晶\*\*

`AssociationRule.R`是对数据集进行\*\*处理\*\*和\*\*关联规则挖掘\*\*的代码，\*\*处理完的数据和关联挖掘的结果文件，记录在分析和报告中\*\*。

`AssociationRuleMdoc.docx`是\*\*结果分析\*\*和\*\*实验报告\*\*文档，\*\*结果记录文件名\*\*和\*\*可视化图片文件名\*\*都记录在其中。

**本次实验的主要目的是对炎症的症状进行数据挖掘，找到症状和炎症间的关联规则。实验的要求如以下大标题。**

**另外一个数据集taitanic幸存者的关联规则挖掘参见：**<http://www.rdatamining.com/examples/association-rules>

## 对数据集进行处理，转换成适合关联规则挖掘的形式；

原始数据集文件是：diagnosis.data，数据说明在文件diagnosis.names中

对一组数据转化为适合关联规则挖掘的处理方法为：

对体温离散化为：正常（35.5-36.9）；低烧（37-37.9）；中等发烧（38-39.9）；高烧（40-42.0）

去掉症状为“no”的项，保留症状为“yes”的项，意义为保留出现的项，用于关联规则挖掘。

采用apriori算法进行关联挖掘，先决条件x（lhs）和关联结果y（rhs），算法中support（支持度）为support(x->y)=P(x,y)；confidence（置信度）为confidenct(x->y)=P(y|x)；lift（提升度）为lift(x->y)=P(y|x)/P(y)。

结果保存在data\_preprocess.csv中

## 找出频繁项集；

核心函数为apriori(data, parameter=NULL, appearance = NULL, control = NULL)

Parameter参数可以对support、confidence、maxlen/minlen（每个项集包含项数的最大最小值）、target（输出结果是什么类型：rules、frequent itemsets）进行设置。比如：parameter = list (support=0.1,confidence=0.5,maxlen=8,target = "frequent itemsets")

Appearance可以限制先决条件X（lhs）和关联结果Y（rhs），比如lhs=beer。

Control用来控制函数性能，如设定对项集进行排序（sort=1或sort=-1表示升序或降序）。

该算法就是对项集进行项数n=1，……8的组合，然后求出每一个组合的support。

结果保存在frequent\_items.txt中

## 导出关联规则，计算其支持度和置信度；

核心函数为apriori(data, parameter=NULL, appearance = NULL, control = NULL)，只不过targe="rules"。

该算法会按照confidence的定义和lift的定义求出X->Y的置信度和提升度，其中X为项集的组合，Y为两种炎症。

结果保存在rules.txt中

## 去除冗余的规则；

满足支持度阈值和置信度阈值的规则中有很多的冗余规则，冗余规则的定义是：如果rules2的lhs和rhs是包含rules1的，而且rules2的lift小于或者等于rules1，则称rules2是rules1的冗余规则，可以理解为rules2中的lhs存在了多余的信息，只需要rules1中的lhs就能判断出rhs的概率了。

结果保存在rules\_delete\_redundant.txt中

## 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；

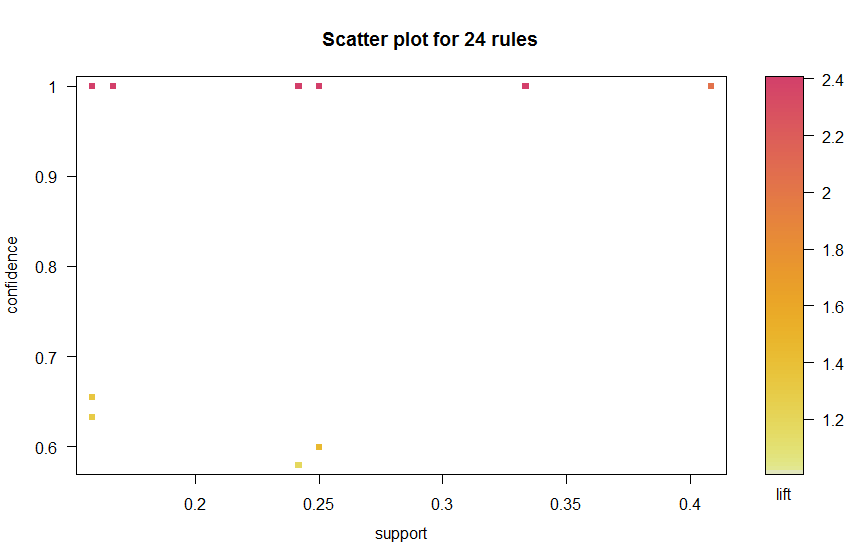
根据lift的值对规则进行排序，lift越高，lhs和rhs的关联度越大，该规则越有用。

结果保存在rules\_delete\_redundant\_sorted\_lift.txt中

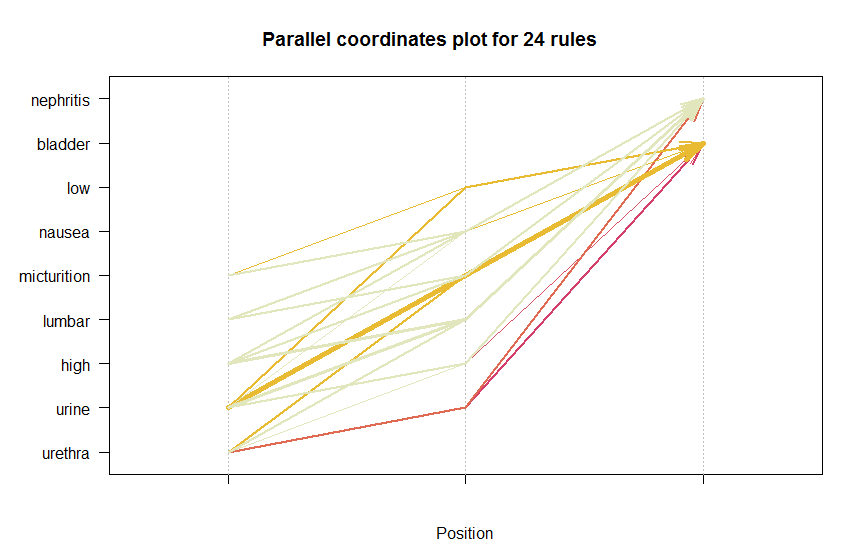
## 使用可视化技术，如散点图、平行坐标、泡泡图等，对规则进行展示。

用三种图对规则进行可视化：

散点图：横纵坐标分别为support和confidence，颜色深度越深，lift值越高，可见confidence和lift有较强的正相关性。



平行坐标图：横坐标为lhs中的项集第1个项和第2个项，以及rhs，纵坐标为项的名称。可以看出每条规则的趋势。



泡泡图：横轴为lhs，纵轴为rhs。面积大小为support的大小，颜色深度越深，关联性越强。

