**基因芯片数据的预处理**

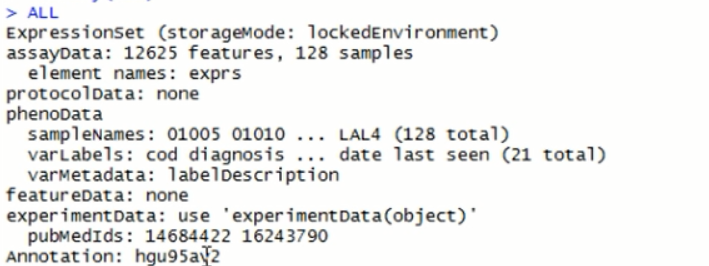
基地一班 刘利生 201730152009

基因芯片通过质量控制剔除不合格的样品，剩下的数据要经过背景校正（通过MM的数据去除背景噪音）、标准化（使不同样品间的数据能够互相比较，消除测量时的非实验差异）、汇总（用一定的统计方法将各个探针数据汇总到基因水平）这三步处理才能的到后续分析需要的表达矩阵。

**在R中的实现**

使用Affy包的expresso函数可以一步完成，通过参数控制使用不同的方法进行背景校正、标准化、汇总三步处理。





library(ALL)

ALL

#导入ALL数据包，可查看得出其共有12625组基因信息，芯片平台为hgu95av2。





library(affy)

library(gcrma)

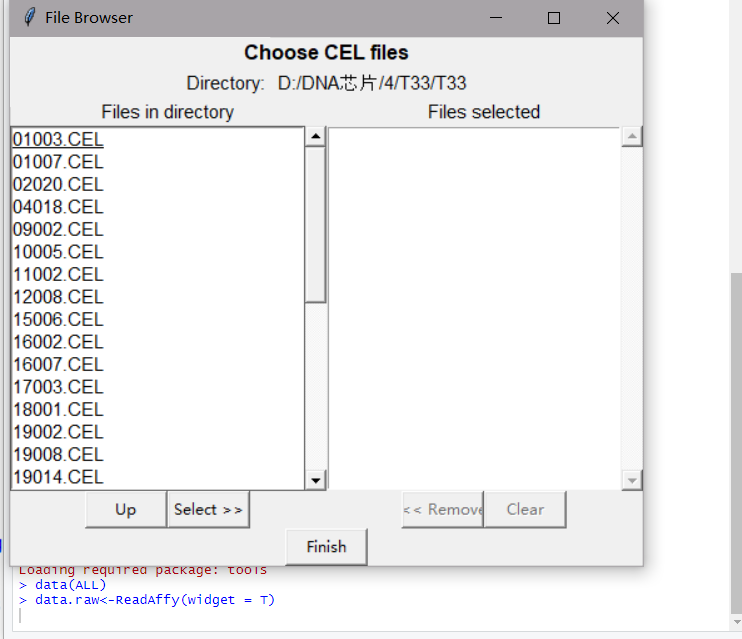
library(hgu95av2cdf)

library(hgu95av2probe)

#载入affy、gcrma、hgu95av2cdf、hgu95av2probe相关数据包

# hgu95av2cdf 中cdf文件提供基因芯片的探针排布信息，最终对应到基因表达的数据，hgu95av2probe对应的probe数据提供了探针的序列信息





data(ALL)

data.raw<-ReadAffy(widget = T)

#导入cel原始数据，命名为data.raw

**预处理**：



ALLMAS5<-mas5(data.raw)

#使用MAS5算法处理原始数据，命名为ALLMAS5

ALLRMA<-rma(data.raw)

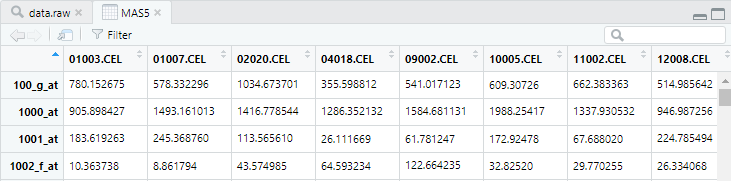
#使用RMA算法处理原始数据，命名为ALLRMA

**背景噪音的处理，标准化、表达量的计算**

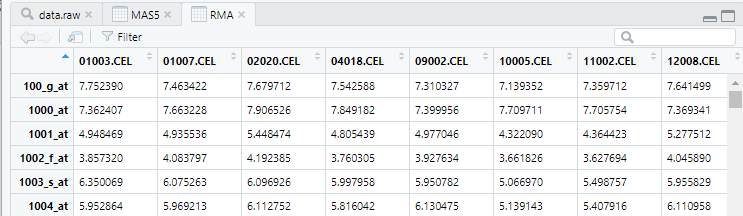
**提取表达矩阵：**







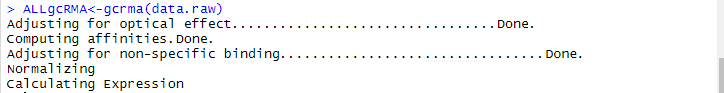




#利用exprs()函数提取相关表达矩阵信息，view()查看矩阵信息

列名代表原文件名称（对应各位病人），行名为探针名称（对应相关基因），数据代表表达量，如在01003号文件（即病人1）中100\_g\_at探针对应的基因表达量为7.752390

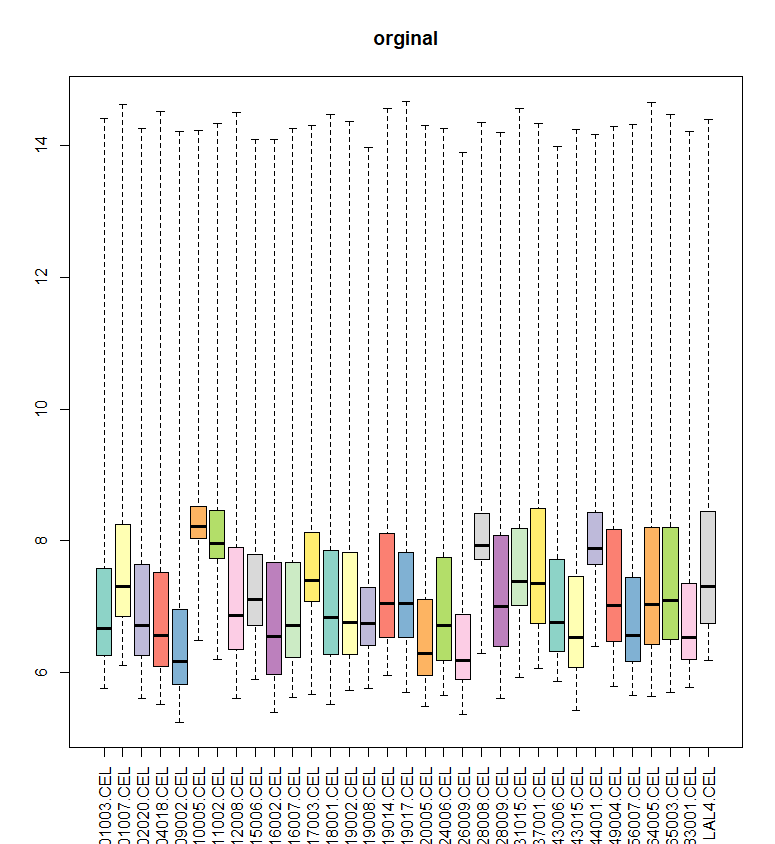
#对比两个矩阵，可见RMA算法得出的数据信息明显小于MAS5，这是由于RMA算法所处理后的数据是通过以2为底的对数转化的。而MAS5不是。



#使用gcrma算法处理原始数据，命名为ALLgcrma

**三种预处理一体化算法的效果：**



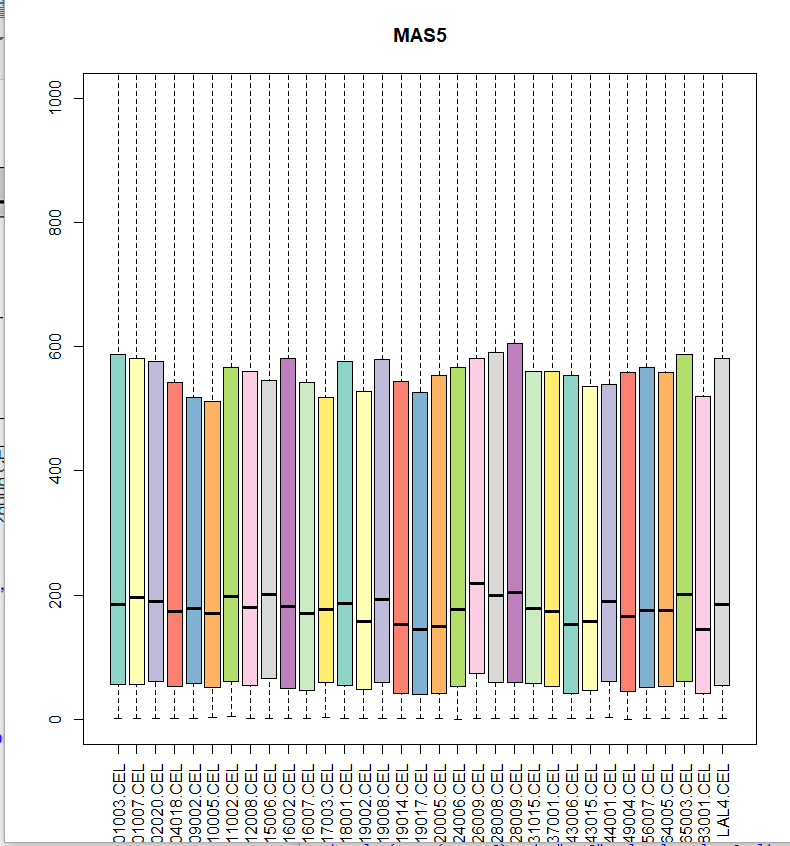


boxplot(data.raw,col=colors,main=”orginal”)

#原始cel数据的镶嵌图，

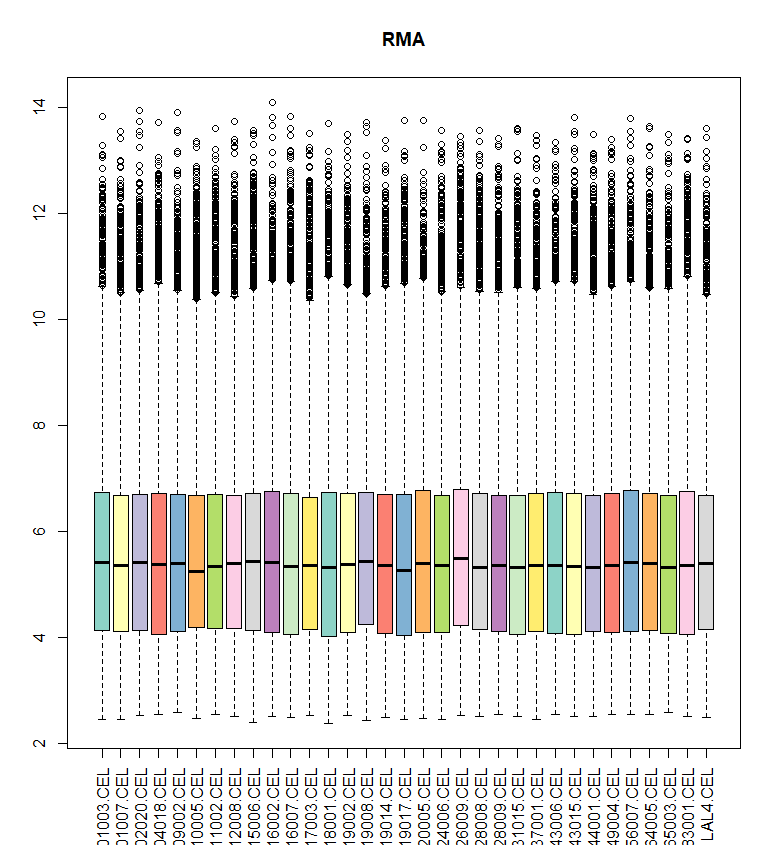
可见其表现得很不整齐，中心值偏差大，数据分布分散。





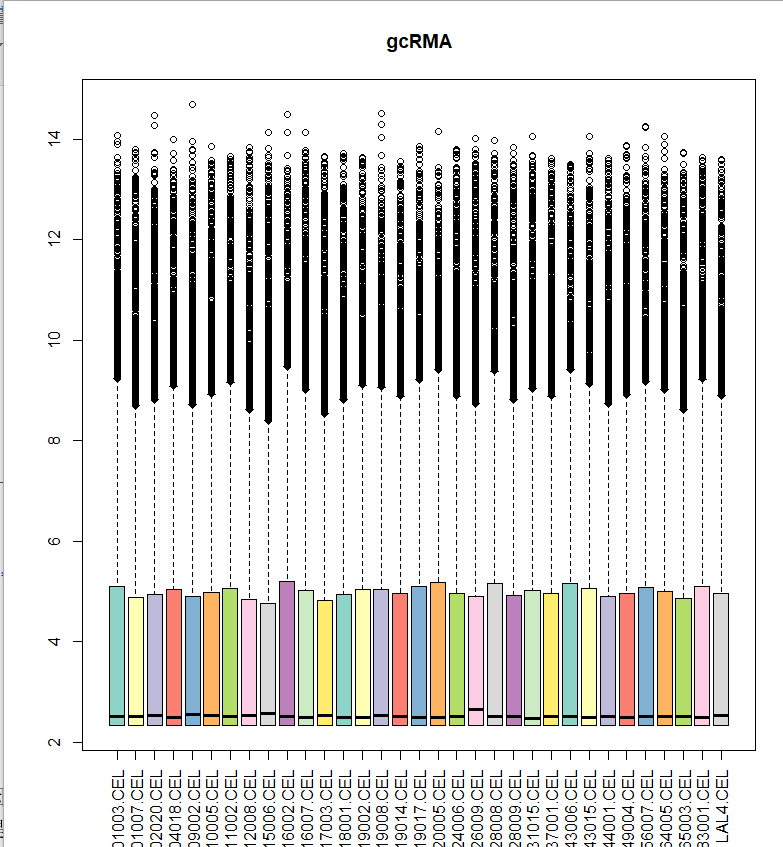
#使用MAS5处理后的图表信息较为整齐，但其分布范围存在拖尾现象





#使用RMA方法处理后，数据图表非常整齐，中位数大致在同一水平内，且分布范围很接近





#使用gcRMA算法处理后，与RMA算法类似，数据图表非常整齐，中位数大致在同一水平内，分布范围接近，但存在拖尾现象。

综上比较得知，采取RMA算法处理后的数据最为理想，故后续操作中使用RMA数据。

**基因的表达差异分析**

**T检验：**

**非特异性筛选**：剔除各个芯片之间表达量相近的那些基因，一般有三种策略：

1. nsFilter筛选
2. 剔除芯片上所有低表达或者不表达的基因数据
3. 根据标准差筛选

**nsFilter筛选**：nsFilter函数由genefilter包提供

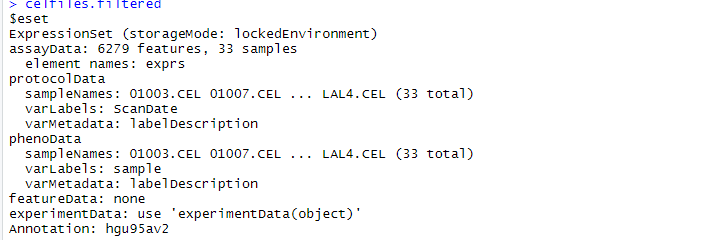


Library(genefilter)

#载入genefilter程辑包

celfiles.filtered<-nsFilter(ALLRMA,require.entrez=FALSE,remove.dupEntrez=FALSE)

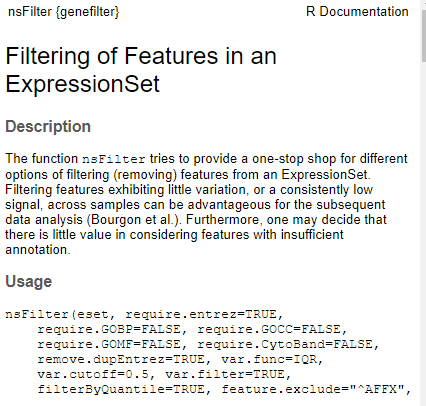
#使用函数处理RMA算法得到的数据，过滤后命名为celfiles.filtered

 celfiles.filtered

#查看celfiles.filtered相关信息，可见其也为ExpressionSet类型数据

筛选后还具有6279个基因数据、筛选掉了6279个没有表达差异性的基因，删除了67个内控基因



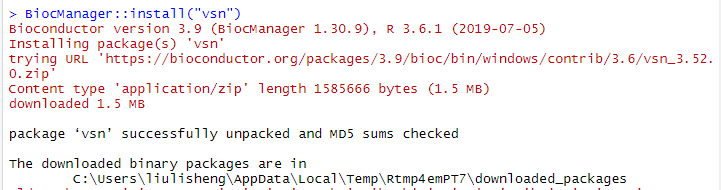


关于函数的功能以及参数意义，输入？nsFilter查询，如图显示其功能为为剔除变化较少的基因与公司检测时加入的自控基因

**根据标准差筛选：**

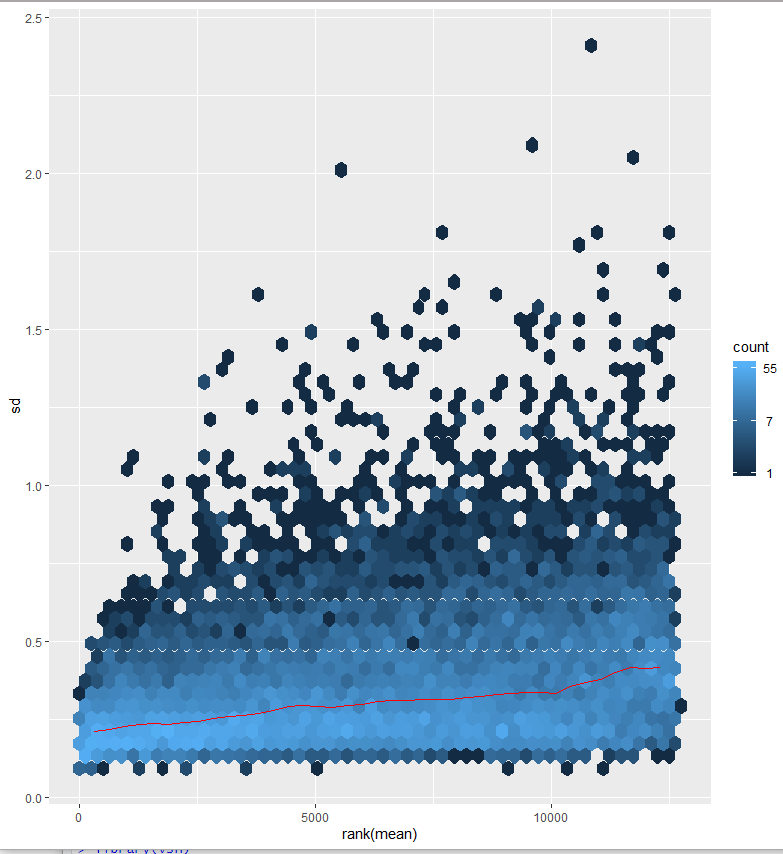
在进行非特异性筛选之前，我们要确定标准差与基因表达量没有关系，因为我们要剔除的是基因表达量标准差很小的基因，这里的方法是vsn包中的meanSdPlot函数。

下载并载入vsn



#下载vsn程辑包





library(vsn)

#载入vsn程辑包

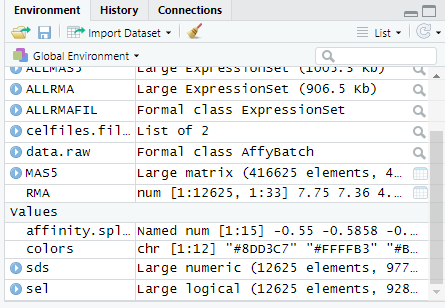
meanSdPlot(ALLRMA)

#mean代表平均值， Sd为标准差， Plot进行作图作图

横坐标代表基因表达量的平均值，纵坐标代表该基因在33个样本中的标准差，我们可以看到图中下方的红线比较平，代表标准差与基因平均表达值的存在些微相关性，但相关性很弱，因此芯片数据可以进行下一步的筛选剔除

**筛选**：







#Apply函数：对ALLRMA这个对象，逐行计算表达矩阵的标准差



即把每一个基因在33个样本里表达的数据的标准差给逐行计算出来，numeric代表其为数值型对象，有12625行

**查看数据类型：**



quantile：分位数，把这些标准差的数值从小到大排列，最小的数值为0％，最大数值为100％，处于0.8的即80％；sds表示将上一步计算的所有的标准差与这个位置的数字大小做一个比较。



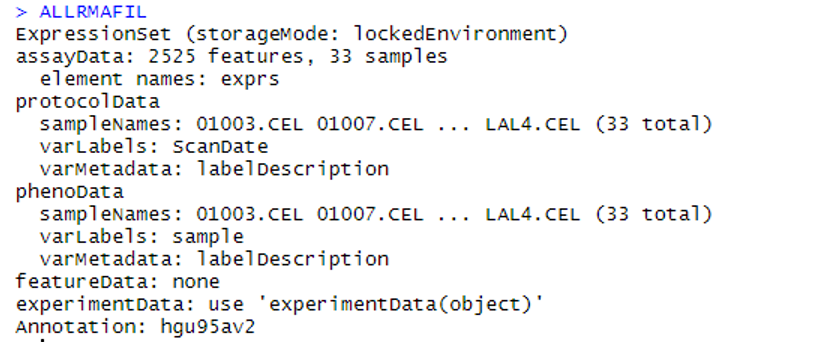
logical代表sel中元素的类型是逻辑型，意思是如果这个标准差比这个位置的数字大的话，表示TRUE，比它小的话表示的是FALSE。

**选执行筛**



剔除比80%对应的数字小的数据

**查看筛选后情况**



筛选后剩下2525个基因，代表设置为80%的分离度比较大，剔除了大部分基因。