User’s guide to the “pubMR” package

周晓北，黄鹏，崔雷

2019 年 9 月 12 日

目录

List of Examples

1 Preliminaries

pubMR是 R 平台下一个高效的 PubMed 文本挖掘工具，集合了：检索下载、解析抽取、基本统计、多维矩阵、论文相似、热点分析、概念识别和网络分析等多种功能。

This guide provides an overview of the **pubMR** package and detailed information on how to use it across the different types of plots and features to customize the visualizations.

pubMR version: 0.001

Citation:

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

2 Abbreviations

These abbreviations are used through the whole user guide:

Abbreviations:

- 共现矩阵 (co-occurrence matrix): XXXX
- PubMed: a free resource developed and maintained by the National Center for Biotechnology Information (NCBI)
- MeSH: Medical Subject Headings
- 摘要 (Abstract): XXXX
- PubTator: a Web-based tool for accelerating manual literature curation through the use of advanced text-mining techniques

3 A quick start

In the simplest case, **pubMR** would start with a set of MeSH key works, as follows:

```
## 目前无法运行！ ##
library(pubMR)
m <- 'Neoplasms[mesh] AND Serine/metabolism[mesh]'
doc <- AB(query=m,output='xml')
```

Example 1: A quick start

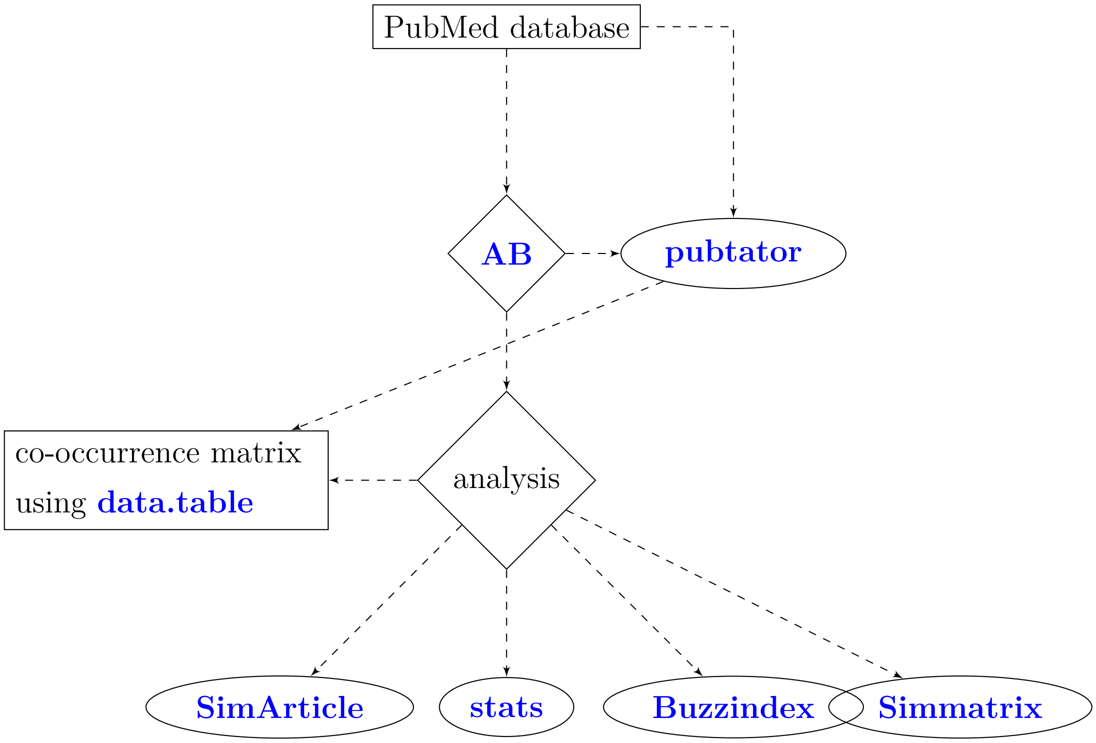
4 A short summary of pubMR

pubMR is an R package designed for text mining of PubMed abstracts. Additionally, it provide some highly customized metics to evaluate and visualize results for downstream analysis. Generally speaking, the **pubMR**

package can be divided into several two parts:

1. **AB**: a S4 class object to store abstracts and related information including PubMed ID, titles, authors, ISSN and so on
2. analysis: downstream analysis part for further analysis including **stats**, **SimArticle**, **Buzzindex**, **Simmatrix**, **pubtator** and co-occurrence matrix pipeline using **data.table**

Its whole structure is shown below:



The individual methods, **Statisticor**, **SimArticle**, **Buzzindex** and **Simmatrix** can be called individually and can be highly customized.

Customized methods:

- **stats**: providing basic statistics of information gererated from PubMed
- **SimArticle**: providing simliar articles related to traget article
- **Buzzindex**: providing 爆发词打分
- **Simmatrix**: providing 论文相似性

5 Data preparation (前期准备)

在提取 PubMed 信息前，一些前期工作是必需的，包括 JCR.csv、mtrees.bin 和 JCR.csv。JCR.csv 文件来自 Web of Science，包括了 JCR Abbreviated Title, ISSN and Journal Impact Factor。mtrees.bin 是 mesh 的树形结构图。在读取这两个文件后，我们需要载入 pubMR 源代码。所有工作展示如下：

```r
jourinfo <- read.csv("2018JCR.csv")
meshtree <- readLines("mtrees2018.bin")
meshtree <- strsplit(meshtree,";")
meshtree <- do.call("rbind",meshtree)
meshtree <- as.data.frame(meshtree)
colnames(meshtree) <- c("mesh","vname")
meshtree[,"class"] <- substr(meshtree[,"vname"],1,1)
rownames(meshtree) <- meshtree[,"vname"]
source("pubMR_v2.R")
meshtree <- as.data.table(meshtree)
```

Example 2: A practical example – 前期准备

6 文献信息提取与分析

6.1 文献信息提取

An example is shown below:

PubMed 检索途径:
Neoplasms[mesh] AND Serine/metabolism[mesh]
2019.3.27 检索到 1215 篇论文

```
## m <- 'Neoplasms[mesh] AND Serine/metabolism[mesh]'
## obj <- AB(query=m,output='ABprofile')
obj

## An object of class "ABprofile"
## with slot names: PMID,TI,AB,TA,PDAT,ISSN,MH,SH,MAJR,AU,MS.
```

Example 3: 文献信息提取

The function **AB** can also import XML file from local directory:

```
obj <- AB(input="obj.xml")
obj

## An object of class "ABprofile"
## with slot names: PMID,TI,AB,TA,PDAT,ISSN,MH,SH,MAJR,AU,MS.
```

Example 4: 读取 xml 文件

The xml file extracted by the function of **AB** can be exported into local directory as the following command:

```
m <- 'Neoplasms[mesh] AND Serine/metabolism[mesh]'
obj <- AB(query=m,output="xml")
saveXML(obj,file="yourResult.xml")
```

Example 5: 保存 xml 文件

In this case the results (**obj**) contains several components including **PMID** (PMIDList), **TI** (Title), **AB** (Abstracts), **PDAT** (Year published), **TA** (Journal information)and **ISSN**, **MH** (Mesh), **MAJR** (Major words), **SH** (Sub-mesh) and **AU** (Authors). To extract one of components, such as **PMID**, just do the following commands:

```
## obj@PMID
obj@PMID[1:5]

## [1] "31221965" "31209254" "31186416" "30801864" "30744688"
```

Example 6: 提取 PMID

6.1.1 文献信息提取 (某年内)

```
id <- which(obj@PDAT>2000&obj@PDAT<2017)
objs <- obj[id]
objs@PDAT[1:10]

##  [1] 2016 2016 2016 2016 2016 2016 2016 2016 2016 2016
```

Example 7: 文献信息提取 2000-2017

6.1.2 文献信息导出

保存摘要到本地:

```
write.csv(obj@AB,file="ab.csv")
```

Example 8: 文献信息导出

6.2 基本统计

```
stat <- stats(obj, "PDAT")
```

Example 9: 基本统计

6.3 论文相似

```
sm <- Simmatrix(obj, type="node")
```

Example 10: 论文相似性—基于信息量

```
sa <- SimArticle("30801864", score = T)
```

Example 11: 论文相似性—PubMed related articles

6.4 爆发词

```
b1 <- Buzzindex("Neoplasms", 3, obj)
b2 <- Buzzindex("Serine", 3, obj)
```

Example 12: 爆发词打分

6.5 共现矩阵

6.5.1 Mesh-Mesh 共现矩阵

第一个例子以 Mesh words 为选值（PMID 为单位，Mesh words 最少出现 5 次），借助**data.table**产生共线象矩阵：

```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,MH=obj@MH)
obj1 = obj1 %>% unnest(MH)
obj1[,n:=.N,by=.(MH)]
obj1 <- obj1[n>=5,]
V <- crossprod(table(obj1[,1:2]))
V[1:2,1:2]

##                  MH
## MH                14-3-3 Proteins 3T3 Cells
##   14-3-3 Proteins              17         0
##   3T3 Cells                     0        19
```

Example 13: A co-occurrence-matrix of Mesh words (counting frequencies of PMID)

6.5.2 (Sub)Mesh-(Sub)Mesh 共现矩阵

主副题词和在一起形成"主副题词-主副题词"共现矩阵:

```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,MH_SH=obj@MH)
obj2=data.table(PMID=obj@PMID,MH_SH=obj@SH)
obj3 <- rbind(obj1,obj2)
obj3 = obj3 %>% unnest(MH_SH)
obj3[,n:=.N,by=.(MH_SH)]
obj3 <- obj3[n>=5,]
V <- crossprod(table(obj3[,1:2]))
V[1:2,1:2]

##                  MH_SH
## MH_SH             14-3-3 Proteins 3T3 Cells
##   14-3-3 Proteins              17         0
##   3T3 Cells                     0        19
```

Example 14: (Sub)Mesh-(Sub)Mesh 共现矩阵

6.5.3 疾病-物质名共现矩阵

接下来，形成疾病和物质名（化学物质、药物、蛋白）的共现矩阵:

```
idr <- which(rownames(V) %in% meshtree[class=="D",mesh])
idc <- which(rownames(V) %in% meshtree[class=="C",mesh])
V1 <- V[idr,idc]
V1[1:2,1:2]

##                     MH_SH
## MH_SH               Acute Disease Adenocarcinoma
##   14-3-3 Proteins               0              1
##   Acetylglucosamine             0              0
```

Example 15: A Chemical-Disease co-occurrence matrix

6.5.4 作者-作者共现矩阵

实现 authors 共现象矩阵:

```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,AU=obj@AU)
obj1 = obj1 %>% unnest(AU)
obj1[,n:=.N,by=.(AU)]
obj1 <- obj1[n>=5,]
V <- crossprod(table(obj1[,1:2]))
V[1:3,1:3]

##                AU
## AU             Adachi Seiji Ali Simak Asara John M
##   Adachi Seiji            5         0            0
##   Ali Simak               0         5            0
##   Asara John M            0         0            5
```

Example 16: A co-occurrence-matrix of authors

6.5.5 MAJR-PMID 共现矩阵

接下来，实现 major-word-PMID 共现象矩阵:

```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,MAJR=obj@MAJR)
obj1 = obj1 %>% unnest(MAJR)
obj1[,n:=.N,by=.(MAJR)]
obj1 <- obj1[n>=5,]
obj1[,n:=1]
V=data.table::dcast(obj1,MAJR~PMID,fill=0,value.var="n")
nms <- V[,MAJR]
V[,MAJR:=NULL]
V <- as.matrix(V)
rownames(V) <- nms
V[1:3,1:3]
```

```
##                                      10047794 10082548 10092628
## Acetylglucosamine                           0        0        0
## Adaptor Proteins, Signal Transducing        0        0        0
## Adenocarcinoma                              0        0        0
```

Example 17: A major-PMID co-occurrence-matrix

6.5.6 Major-Major 共现矩阵

下面实现 major words 共现象矩阵:

```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,MAJR=obj@MAJR)
obj1 = obj1 %>% unnest(MAJR)
obj1[,n:=.N,by=.(MAJR)]
obj1 <- obj1[n>=5,]
V <- crossprod(table(obj1[,1:2]))
V[1:2,1:2]

##                                      MAJR
## MAJR                                  Acetylglucosamine
##   Acetylglucosamine                                   6
##   Adaptor Proteins, Signal Transducing                0
##                                      MAJR
## MAJR                                  Adaptor Proteins, Signal Transducing
##   Acetylglucosamine                                                      0
##   Adaptor Proteins, Signal Transducing                                  21
```

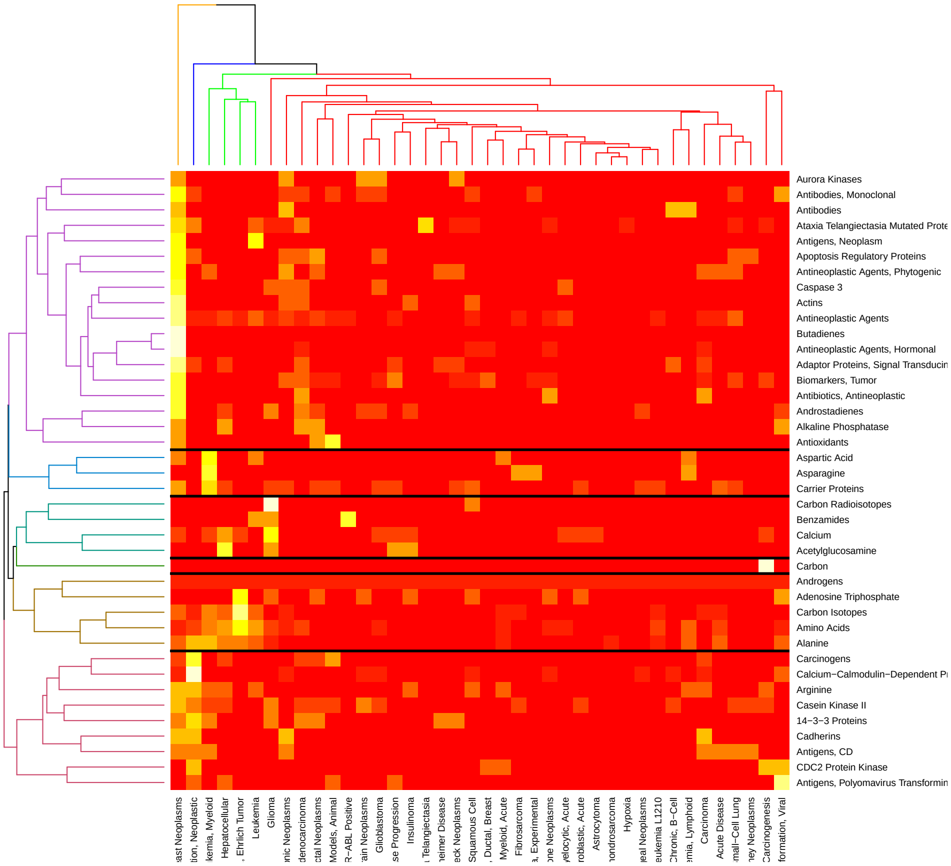
Example 18: A co-occurrence-matrix of major words

6.5.7 共现矩阵可视化

以疾病-物质名共现矩阵为例，共现矩阵可以实现双聚类热图：

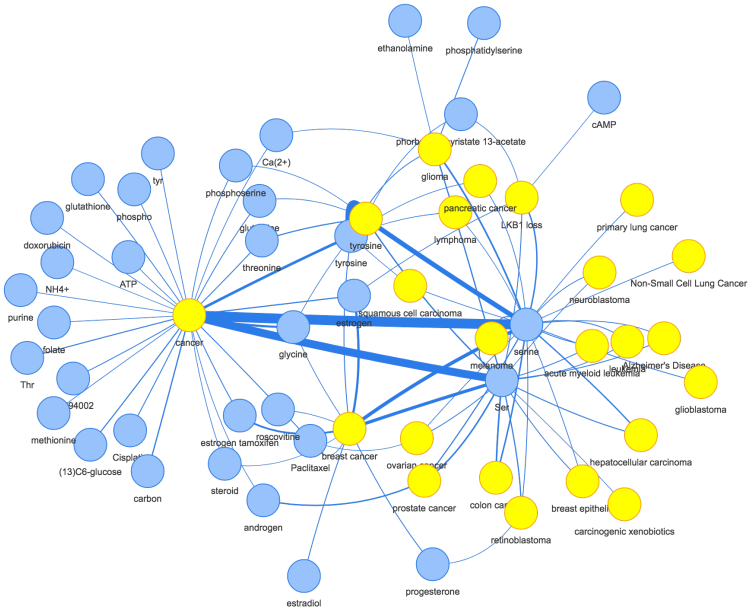
```
library(data.table)
library(tidyr)
options(datatable.prettyprint.char=10L)
obj1=data.table(PMID=obj@PMID,MH=obj@MH)
obj1 = obj1 %>% unnest(MH)
obj1[,n:=.N,by=.(MH)]
obj1 <- obj1[n>=5,]
V <- crossprod(table(obj1[,1:2]))
V <- crossprod(table(obj3[,1:2]))
V[1:2,1:2]
idr <- which(rownames(V) %in% meshtree[class=="D",mesh])
idc <- which(rownames(V) %in% meshtree[class=="C",mesh])
V1 <- V[idr,idc]

x <- V1[1:40,1:40]
x <- sweep(x,1L,rowMeans(x,na.rm=TRUE),check.margin=FALSE)
sx <- apply(x,1L,sd,na.rm=TRUE)
x <- sweep(x,1L,sx,"/",check.margin=FALSE)
x[is.na(x)] <- 0
library(dendextend)
kr <- 6;kc <- 4
Rowv <- x %>% dist %>% hclust %>% as.dendrogram %>%
   set("branches_k_color", k = kr) %>%
   set("branches_lwd", 1.2)
Colv <- t(x) %>% dist %>%
   hclust %>% as.dendrogram %>%
   set("branches_k_color",k=kc,value=c("orange","blue","green","red")) %>%
   set("branches_lwd",1.2)
cluster <- cutree(as.hclust(Rowv),k=kr)
clustab <- table(cluster)[unique(cluster[as.hclust(Rowv)$order])]
m <- cumsum(clustab)
m <- m[-length(m)]
heatmap(x, Rowv = Rowv,Colv=Colv,scale="none",add.expr=abline(h=m+0.5,lwd=3))
```

Example 19: 双聚类热图



以疾病-物质名共现矩阵为例，共现矩阵可以实现网络图：

```
## constnetwork(V1)
```



Example 20: 网络图

6.5.8 共现矩阵导出

共现矩阵保存到本地并方便使用其它工具：

```
write.csv(V1,file="co.csv")
```

Example 21: 共现矩阵导出

6.6 Pubtator

```
## res0 <- pubtator(obj@PMID)
res0

##            pmid     type     name           id
##     1: 27042806 Chemical   carbon MESH:D0022...
##     2: 27042806  Disease   cancer MESH:D0093...
##     3: 27042806  Species    human         9606
##     4: 27042806 Chemical   carbon MESH:D0022...
##     5: 27042806  Disease   cancer MESH:D0093...
##    ---
## 33751: 13270262  Disease   tumors MESH:D0093...
## 33752: 13374701 Chemical Glycerol MESH:D0059...
## 33753: 13374701  Disease    tumor MESH:D0093...
## 33754: 13374701 Chemical   serine MESH:C0479...
## 33755: 13374701 Chemical  glycine MESH:D0059...
```

Example 22: PubTator API

通过 PubTatorAPI 提取的主题词有大量重复现象，例如：serine，Serine，D-serine 和 serines。这些词都对应同一 Concept ID (MESH:C047902)。以下方法可以处理重复主题词：

```
tmp <-  unique(res0,by="id")
res1 <- merge(res0[,!"name"],tmp[,.(name,id)],all.x=TRUE,sort=FALSE)
```

Example 23: 去除重复主题词

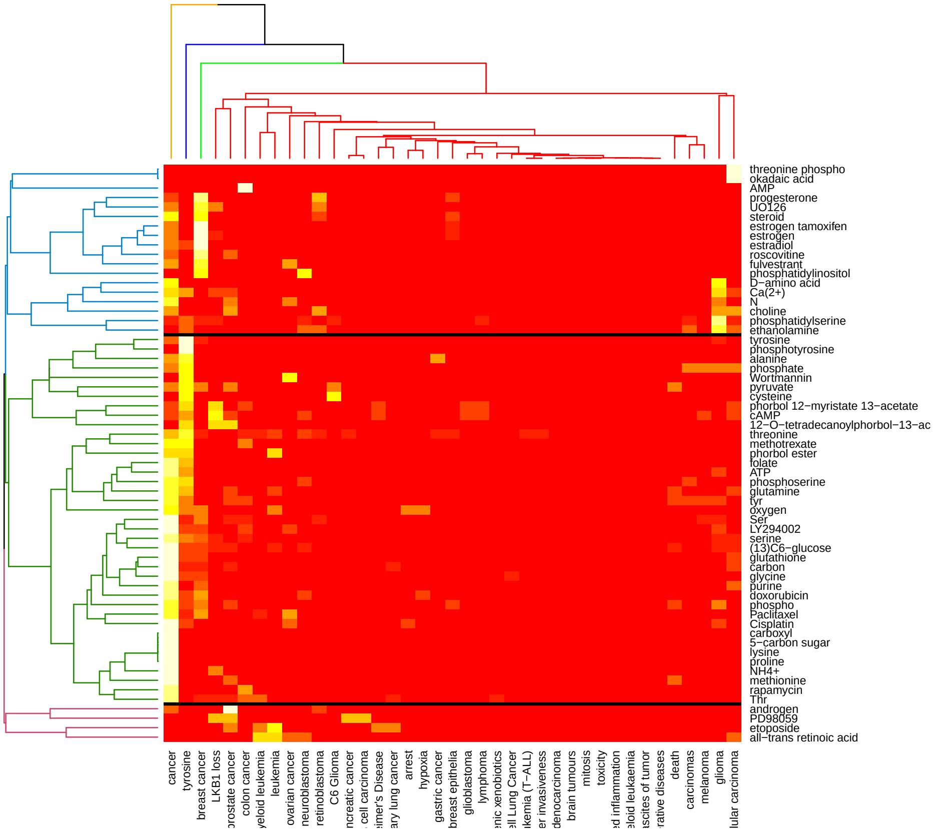
接下来，可以实现提取共现象矩阵 (疾病-物质名)：

```r
library(data.table)
library(reshape2)
res2 <- data.table(res1[,2:4])
a=res2[,n:=.N,by=.(name,pmid)]
a = a[,global:=length(unique(pmid)),by=.(name)]
b = a[n>=2&global>=10,]
res2 <- unique(b)
V <- crossprod(table(res2[,c(1,3)]))
idr <- which(rownames(V) %in% res2[type=="Chemical",name])
idc <- which(rownames(V) %in% res2[type=="Disease",name])
V1 <- V[idr,idc]
```

Example 24: a Chemical-Disease co-occurrence matrix using PubTator API

```r
x <- V1
x <- sweep(x,1L,rowMeans(x,na.rm=TRUE),check.margin=FALSE)
sx <- apply(x,1L,sd,na.rm=TRUE)
x <- sweep(x,1L,sx,"/",check.margin=FALSE)
library(dendextend)
kr <- 3;kc <- 4
Rowv <- x %>% dist %>% hclust %>% as.dendrogram %>%
   set("branches_k_color", k = kr) %>%
   set("branches_lwd", 1.2)
Colv <- t(x) %>% dist %>%
   hclust %>% as.dendrogram %>%
   set("branches_k_color",k=kc,value=c("orange","blue","green","red")) %>%
   set("branches_lwd",1.2)
cluster <- cutree(as.hclust(Rowv),k=kr)
```

```
clustab <- table(cluster)[unique(cluster[as.hclust(Rowv)$order])]
m <- cumsum(clustab)
m <- m[-length(m)]
heatmap(x, Rowv = Rowv,Colv=Colv,scale="none",add.expr=abline(h=m+0.5,lwd=3))
```



7 Session info

- R version 3.5.3 (2019-03-11), `x86_64-apple-darwin15.6.0`
- Locale: `C/UTF-8/C/C/C/C`
- Running under: `macOS Sierra 10.12.6`

- Matrix products: default
- BLAS: `/Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib`
- LAPACK: `/Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: Biobase 2.42.0, BiocGenerics 0.28.0, BiocParallel 1.16.6, DelayedArray 0.8.0, EnrichmentBrowser 2.12.1, GenomeInfoDb 1.18.2, GenomicRanges 1.34.0, IRanges 2.16.0, Matrix 1.2-17, RCurl 1.95-4.12, S4Vectors 0.20.1, SummarizedExperiment 1.12.0, XML 3.98-1.19, bitops 1.0-6, data.table 1.12.2, dendextend 1.12.0, dplyr 0.8.1, ggplot2 3.1.1, graph 1.60.0, httr 1.4.0, igraph 1.2.4.1, knitr 1.23, matrixStats 0.54.0, plotly 4.9.0, plyr 1.8.4, reshape2 1.4.3, stringr 1.4.0, tidyr 0.8.3, visNetwork 2.0.7, wordcloud2 0.2.1
- Loaded via a namespace (and not attached): AnnotationDbi 1.44.0, BiocManager 1.30.4, DBI 1.0.0, GSEABase 1.44.0, GenomeInfoDbData 1.2.0, KEGGgraph 1.42.0, R6 2.4.0, RSQLite 2.1.1, Rcpp 1.0.1.3, SparseM 1.77, XVector 0.22.0, annotate 1.60.1, assertthat 0.2.1, bit 1.1-14, bit64 0.9-7, blob 1.1.1, colorspace 1.4-1, compiler 3.5.3, crayon 1.3.4, digest 0.6.19, evaluate 0.13, glue 1.3.1, grid 3.5.3, gridExtra 2.3, gtable 0.3.0, highr 0.8, htmltools 0.3.6, htmlwidgets 1.3, jsonlite 1.6, lattice 0.20-38, lazyeval 0.2.2, limma 3.38.3, magrittr 1.5, memoise 1.1.0, munsell 0.5.0, pillar 1.4.0, pkgconfig 2.0.2, purrr 0.3.2, rappdirs 0.3.1, rlang 0.3.4, safe 3.22.0, scales 1.0.0, stringi 1.4.3,

tcltk 3.5.3, tibble 2.1.2, tidyselect 0.2.5, tools 3.5.3, viridis 0.5.1, viridisLite 0.3.0, withr 2.1.2, xfun 0.7, xtable 1.8-4, zlibbioc 1.28.0