

# SPEAQeasy\_example

Josh Stolz

9/18/2020

## Contents

### 1 Reproducibility

6

```
library("pheatmap")
library("tidyr")
library("jaffelab")
library("here")
library("VariantAnnotation")
library("SummarizedExperiment")
library("devtools")
library("BiocStyle")
```

In order to resolve the swaps to our best ability we need four data sets. Here we have load snpGeno\_example which is from our topmed imputed genotype data, a phenotype data sheet (pd\_example), a vcf file of the relevant speaqeasy output (Speaqeasy), and our current genotype sample sheet (brain\_sentrrix). This file is wrote in the directory listed below.

```
load(here("sample_selection", "snpsGeno_example.RData"), verbose = TRUE)
```

```
## Loading objects:
##   snpsGeno_example
```

```
load(here("sample_selection", "pd_example.Rdata"), verbose = TRUE)
```

```
## Loading objects:
##   pd_example
```

```
Speaqeasy<-readVcf(here("pipeline_outputs", "merged_variants", "mergedVariants.vcf.gz"), genome="hg38")
brain_sentrrix<- read.csv(here("brain_sentrrix_speaqeasy.csv"))
```

We can see that the genotype is represented in the form of 0s,1s, and 2s. The rare 2s are a result of multiallelic snps and we will drop those. 0 represent the reference allele with ones representing the alternate. We can see the distribution below.

```
Geno_speaqeasy<-geno(Speaqeasy)$GT
table(Geno_speaqeasy)
```

```
## Geno_speaqeasy
##   ./   0/1   0/2   1/1   2/2
## 14096 7803    4 6018    9
```

Given this we convert we convert the Genotype data from Speaqeasy to numeric data. The “./” were values that could not accurately be determined and are replaced with NA.

```
colnames_speaqeasy<- as.data.frame(colnames(Geno_speaqeasy))
colnames(colnames_speaqeasy)<-c("a")
samples<-separate(colnames_speaqeasy,a,into = c("a","b","c"),sep = "_")
```

```
## Warning: Expected 3 pieces. Additional pieces discarded in 42 rows [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
samples<-paste0(samples$a,"_",samples$b)
samples<-as.data.frame(samples)
colnames(Geno_speaqeasy)<-samples$samples
Geno_speaqeasy[Geno_speaqeasy == "./."] = NA
Geno_speaqeasy[Geno_speaqeasy == "0/0"] = 0
Geno_speaqeasy[Geno_speaqeasy == "0/1"] = 1
Geno_speaqeasy[Geno_speaqeasy == "1/1"] = 2
class(Geno_speaqeasy) = "numeric"
```

```
## Warning in class(Geno_speaqeasy) = "numeric": NAs introduced by coercion
```

```
corner(Geno_speaqeasy)
```

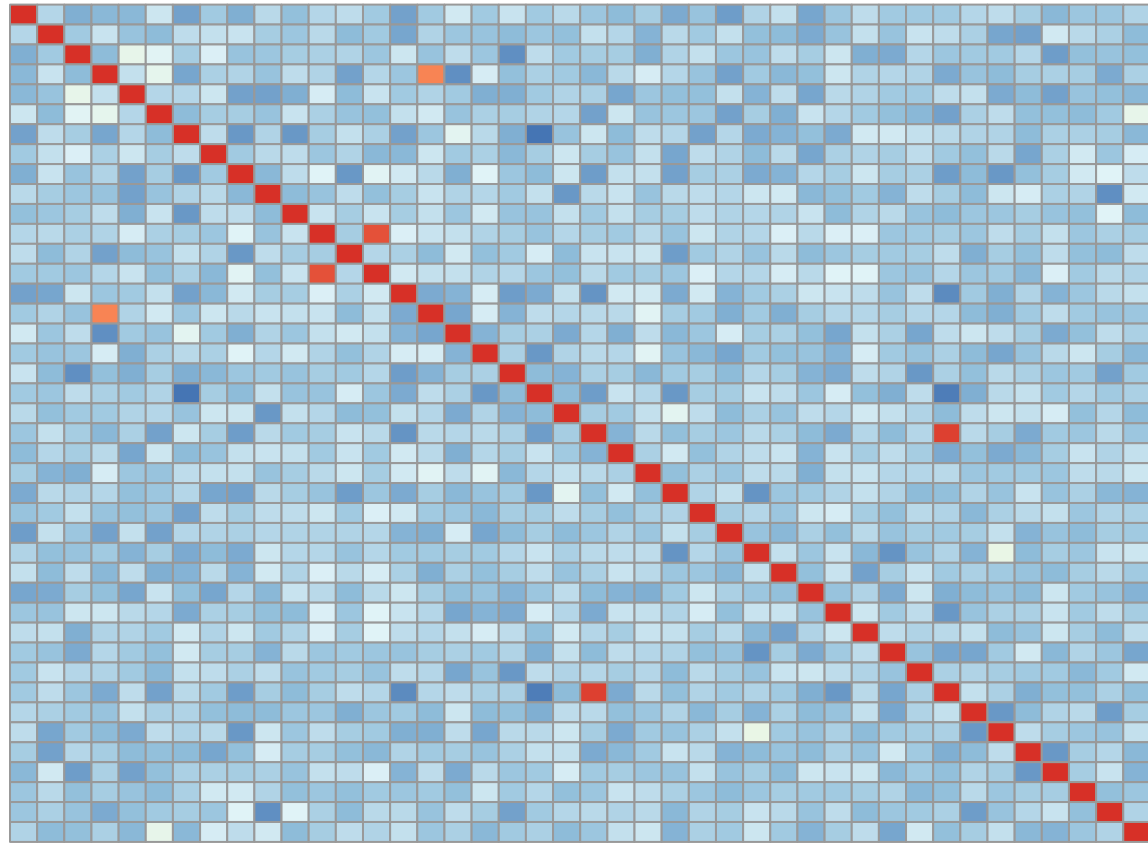
```
##
## R14030_H7K5NBBXX R14184_H7K5NBBXX R13904_H7K5NBBXX R14296_H7JLCBBXX R14247_HF3JYBBXX
## chr1:4712657_G/A      2      1      2      1
## chr1:7853370_G/A      NA     NA     NA     NA
## chr1:9263851_G/A      NA     NA     1     1
## chr1:13475857_T/C     NA     NA     NA     NA
## chr1:15289643_G/A     NA     NA     NA     NA
## chr1:15341780_G/A     NA     NA     NA     NA
```

We then make a correlation matrix to find the possible mismatches between samples.

```
speaqeasy_Cor = cor(Geno_speaqeasy,use="pairwise.comp")
corner(speaqeasy_Cor)
```

```
##
## R14030_H7K5NBBXX R14184_H7K5NBBXX R13904_H7K5NBBXX R14296_H7JLCBBXX R14247_HF3JYBBXX
## R14030_H7K5NBBXX      1.00000000      0.13507812      0.01657484      0.03106472      0.03384912
## R14184_H7K5NBBXX      0.13507812      1.00000000      0.09347004      0.18748043      0.07080491
## R13904_H7K5NBBXX      0.01657484      0.09347004      1.00000000      0.04958385      0.28798955
## R14296_H7JLCBBXX      0.03106472      0.18748043      0.04958385      1.00000000      0.16653808
## R14247_HF3JYBBXX      0.03384912      0.07080491      0.28798955      0.16653808      1.00000000
## R15093_HFY2MBBXX      0.18938568      0.04720309      0.25477682      0.27251010      0.14097335
```

Here in the heatmap below we can see that several points do not correlate with themselves in a semetrical matrix. This could be mismatches, but it also could be a result of a brain being sequenced twice. We will dig



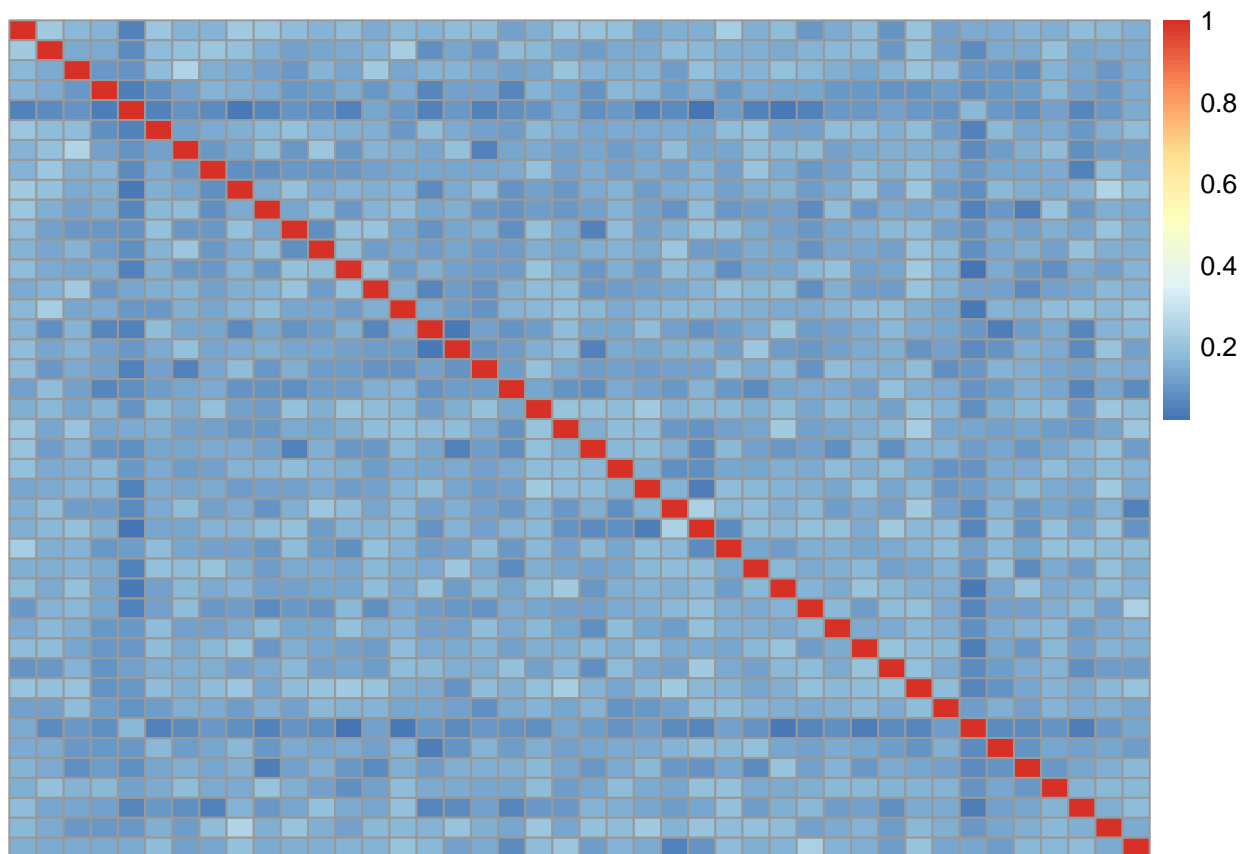
more into this later on.

We repeat the process for the genotype data from topmed. First creating our numeric data for the genotypes.

```
## Geno_example
## 0|0 0|1 1|0 1|1
## 9656 6956 7185 7535
```

```
##          4463344375_R01C01 4463344375_R01C02 4572348848_R01C02 4572348855_R01C02 5535549043_R01C01
## 4463344375_R01C01          1.00000000          0.21736687          0.17615291          0.16590842          0.05367081
## 4463344375_R01C02          0.21736687          1.00000000          0.13803954          0.14670605          0.07697879
## 4572348848_R01C02          0.17615291          0.13803954          1.00000000          0.10600440          0.09356496
## 4572348855_R01C02          0.16590842          0.14670605          0.10600440          1.00000000          0.04874508
## 5535549043_R01C01          0.05367081          0.07697879          0.09356496          0.04874508          1.00000000
## 5535506145_R01C01          0.21189898          0.18114194          0.17829111          0.08186958          0.05331113
##          5535506145_R01C01
## 4463344375_R01C01          0.21189898
## 4463344375_R01C02          0.18114194
## 4572348848_R01C02          0.17829111
## 4572348855_R01C02          0.08186958
## 5535549043_R01C01          0.05331113
## 5535506145_R01C01          1.00000000
```

In this case the data only appears to have samples that match themselves. However there is the potential for a second kind of error where a brain has two samples, however the do not match each other.



In order to dig into this further we will collapse the correlation matrices into a data table shown below

```
corLong = data.frame(cor = signif(as.numeric(correlation_genotype),3))
corLong$rowSample = rep(colnames(snpsGeno_example), times = ncol(snpsGeno_example))
corLong$colSample = rep(colnames(snpsGeno_example), each = ncol(snpsGeno_example))
corLong = corLong[!is.na(corLong$cor),]
head(corLong)
```

```
##      cor      rowSample      colSample
## 1 1.0000 4463344375_R01C01 4463344375_R01C01
## 2 0.2170 4463344375_R01C02 4463344375_R01C01
## 3 0.1760 4572348848_R01C02 4463344375_R01C01
## 4 0.1660 4572348855_R01C02 4463344375_R01C01
## 5 0.0537 5535549043_R01C01 4463344375_R01C01
## 6 0.2120 5535506145_R01C01 4463344375_R01C01
```

```
corLong2 = data.frame(cor = signif(as.numeric(speaqeasy_Cor),3))
corLong2$rowSample = rep(colnames(Geno_speaqeasy), times = ncol(Geno_speaqeasy))
corLong2$colSample = rep(colnames(Geno_speaqeasy), each = ncol(Geno_speaqeasy))
corLong2 = corLong2[!is.na(corLong2$cor),]
head(corLong2)
```

```
##      cor      rowSample      colSample
## 1 1.0000 R14030_H7K5NBBXX R14030_H7K5NBBXX
## 2 0.1350 R14184_H7K5NBBXX R14030_H7K5NBBXX
```

```
## 3 0.0166 R13904_H7K5NBBXX R14030_H7K5NBBXX
## 4 0.0311 R14296_H7JLCBBXX R14030_H7K5NBBXX
## 5 0.0338 R14247_HF3JYBBXX R14030_H7K5NBBXX
## 6 0.1890 R15093_HFY2MBBXX R14030_H7K5NBBXX
```

We can check these tables for columns where different brains are strongly correlated and where the same brain fails to match itself. Below is the output of those analysis for the topmed genotypes.

```
##          cor          rowSample          colSample rowBrain colBrain  rowBatch  colBatch
## 180 0.0825 6017049081_R01C02 5535549043_R01C01 Br1652 Br1652 1M 1M
## 196 0.0501 201398400130_R04C01 5535549043_R01C01 Br1652 Br1652 2-5-8-v1-3 1M
## 467 0.0825 5535549043_R01C01 6017049081_R01C02 Br1652 Br1652 1M 1M
## 490 0.1440 201398400130_R04C01 6017049081_R01C02 Br1652 Br1652 2-5-8-v1-3 1M
## 1139 0.0501 5535549043_R01C01 201398400130_R04C01 Br1652 Br1652 1M 2-5-8-v1-3
## 1146 0.1440 6017049081_R01C02 201398400130_R04C01 Br1652 Br1652 1M 2-5-8-v1-3
## 1212 0.0379 9373406026_R02C01 201398400130_R06C01 Br2275 Br2275 5M 2-5-8-v1-3
## 1499 0.0379 201398400130_R06C01 9373406026_R02C01 Br2275 Br2275 2-5-8-v1-3 5M
```

```
## [1] cor          rowSample colSample rowBrain colBrain rowBatch colBatch
## <0 rows> (or 0-length row.names)
```

And we do this again for the speaqeasy data.

```
## [1] cor          rowSample colSample rowBrain colBrain
## <0 rows> (or 0-length row.names)
```

```
## [1] cor          rowSample colSample rowBrain colBrain
## <0 rows> (or 0-length row.names)
```

We will next compare the correlation between the speaqeasy samples and the topmed samples. In order to do this we need to subset the genotypes for only SNPs that are common between the two. We can see that we have 656 snps common between the 42 samples.

```
## [1] 662 42
```

```
## [1] 662 42
```

As we did before we create a correlation matrix this time between the two data sets.

Check to correlation between Speaqeasy and Genotype for mismatches and swaps.

```
##          cor          rowSample          colSample colBrain rowBrain
## 165 0.00962 9373408026_R01C01 R14296_H7JLCBBXX Br2473 Br2473
## 425 0.11600 5535549043_R01C01 R14129_HCTYLBBXX Br1652 Br1652
## 432 0.11800 6017049081_R01C02 R14129_HCTYLBBXX Br1652 Br1652
## 498 -0.04270 9373406026_R02C01 R14222_H7JHNBBXX Br2275 Br2275
## 582 -0.03790 9373406026_R02C01 R13997_H7JHNBBXX Br2275 Br2275
## 669 0.08540 9373408026_R01C01 R14077_HCTYLBBXX Br2473 Br2473
## 917 0.04020 9373406026_R01C01 R14290_H7L3FBBXX Br2260 Br2260
## 1463 0.05050 9373406026_R01C01 R14071_HF3JYBBXX Br2260 Br2260
## 1697 0.76500 3998646040_R06C01 R14017_HF5JNBBXX Br5190 Br5190
```

##	cor	rowSample	colSample	colBrain	rowBrain
## 161	0.865	9373406026_R01C01	R14296_H7JLCBBXX	Br2473	Br2260
## 665	0.815	9373406026_R01C01	R14077_HCTYLBXX	Br2473	Br2260
## 921	0.866	9373408026_R01C01	R14290_H7L3FBBXX	Br2260	Br2473
## 1467	0.915	9373408026_R01C01	R14071_HF3JYBBXX	Br2260	Br2473

We can see from this from this analysis there are a few swaps present between RNA and DNA samples here. We can categorize them as simple and complex sample swaps. Because the two Br2275 do not match each other and also match nothing else we will be forced to consider this a complex swap and drop the sample. In the case of Br2473 it is a simple swap with Br2260 in both cases. This can be ammended by swapping with in the phenotype data sheet manually. Now we have our accurate data outputs and will need to fix our ranged summarized experiment object for our Speaqeasy data.

```
load(here('pipeline_outputs', 'count_objects', 'rse_gene_Jlab_experiment_n42.Rdata'))

## drop sample from rse with speaqeasy data
ids<-pd_example$SAMPLE_ID[pd_example$BrNum == "Br2275"]
rse_gene<-rse_gene[!rse_gene$SAMPLE_ID ==ids[1]]
rse_gene<- rse_gene[!rse_gene$SAMPLE_ID ==ids[2]]

# resolve swaps and drops in pd_example
pd_example<-pd_example[!pd_example$SAMPLE_ID ==ids[1],]
pd_example<-pd_example[!pd_example$SAMPLE_ID ==ids[2],]
ids2<-pd_example$SAMPLE_ID[pd_example$BrNum == "Br2260"]
ids3<-pd_example$SAMPLE_ID[pd_example$BrNum == "Br2473"]
pd_example$SAMPLE_ID[pd_example$Sample_ID == ids2]<- "Br2473"
pd_example$SAMPLE_ID[pd_example$Sample_ID == ids3]<- "Br2260"

# reorder phenotype data by the sample order present in the 'rse_gene' object
pd_example = pd_example[match(rse_gene$SAMPLE_ID, pd_example$SAMPLE_ID),]

# add important colData to 'rse_gene'
rse_gene$BrainRegion = pd_example$BrainRegion
rse_gene$Race = pd_example$Race
rse_gene$PrimaryDx = pd_example$PrimaryDx
rse_gene$Sex = pd_example$Sex
rse_gene$AgeDeath = pd_example$AgeDeath

#add correct BrNum to colData for rse_gene
colData(rse_gene)$BrNum <- pd_example$BrNum

save(rse_gene, file = "rse_speaqeasy.RData")
```

## 1 Reproducibility

This analysis report was made possible thanks to:

- R (Müller, 2017)
- *devtools* (Allaire, Xie, McPherson, Luraschi, et al., 2020)
- *here* (Obenchain, Lawrence, Carey, Gogarten, et al., 2014)
- *jaffelab* (Boettiger, 2019)
- *knitcitations* (Collado-Torres, Jaffe, and Burke, 2019)

- *pheatmap* (Kolde, 2019)
- *rmarkdown*
- *SummarizedExperiment* (R Core Team, 2020)
- *tidyr* (Morgan, Obenchain, Hester, and Pagès, 2020)
- *voom*
- *VariantAnnotation* (Xie, Allaire, and Golemund, 2018)

#### Bibliography file

- [1] J. Allaire, Y. Xie, J. McPherson, J. Luraschi, et al. *rmarkdown: Dynamic Documents for R*. R package version 2.3. 2020. <URL: <https://github.com/rstudio/rmarkdown>>.
- [2] C. Boettiger. *knitcitations: Citations for ‘Knitr’ Markdown Files*. R package version 1.0.10. 2019. <URL: <https://github.com/cboettig/knitcitations>>.
- [3] L. Collado-Torres, A. E. Jaffe, and E. E. Burke. *jaffelab: Commonly used functions by the Jaffe lab*. R package version 0.99.30. 2019. <URL: <https://github.com/LieberInstitute/jaffelab>>.
- [4] R. Kolde. *pheatmap: Pretty Heatmaps*. R package version 1.0.12. 2019.
- [5] M. Morgan, V. Obenchain, J. Hester, and H. Pagès. *SummarizedExperiment: SummarizedExperiment container*. R package version 1.19.6. 2020.
- [6] K. Müller. *here: A Simpler Way to Find Your Files*. <https://github.com/krlmlr/here>, <http://krlmlr.github.io/here>. 2017.
- [7] V. Obenchain, M. Lawrence, V. Carey, S. Gogarten, et al. “VariantAnnotation: a Bioconductor package for exploration and annotation of genetic variants”. In: *Bioinformatics* 30.14 (2014), pp. 2076-2078. DOI: 10.1093/bioinformatics/btu168.
- [8] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2020. <URL: <https://www.R-project.org/>>.
- [9] Y. Xie, J. Allaire, and G. Golemund. *R Markdown: The Definitive Guide*. ISBN 9781138359338. Boca Raton, Florida: Chapman and Hall/CRC, 2018. <URL: <https://bookdown.org/yihui/rmarkdown>>.

```
# Time spent creating this report:
diff(c(timestart, Sys.time()))
```

```
## Time difference of 1.317441 secs
```

```
# Date this report was generated
message(Sys.time())
```

```
## 2020-09-25 17:00:20
```

```
# Reproducibility info
options(width = 120)
devtools::session_info()
```

```
## - Session info -----
## setting value
## version R version 4.0.2 Patched (2020-09-17 r79226)
## os      Ubuntu 20.04.1 LTS
## system x86_64, linux-gnu
## ui      X11
```

```

## language (EN)
## collate en_US.UTF-8
## ctype en_US.UTF-8
## tz UTC
## date 2020-09-25
##
## - Packages -----
## package * version date lib source
## AnnotationDbi * 1.51.3 2020-07-25 [1] Bioconductor
## askpass 1.1 2019-01-13 [2] RSPM (R 4.0.0)
## assertthat 0.2.1 2019-03-21 [2] RSPM (R 4.0.0)
## backports 1.1.10 2020-09-15 [2] RSPM (R 4.0.2)
## base64enc 0.1-3 2015-07-28 [2] RSPM (R 4.0.0)
## bibtex 0.4.2.2 2020-01-02 [1] RSPM (R 4.0.0)
## Biobase * 2.49.1 2020-09-03 [1] Bioconductor
## BiocFileCache 1.13.1 2020-08-04 [1] Bioconductor
## BiocGenerics * 0.35.4 2020-06-04 [1] Bioconductor
## BiocManager * 1.30.10 2019-11-16 [2] CRAN (R 4.0.2)
## BiocParallel 1.23.2 2020-07-06 [1] Bioconductor
## BiocStyle * 2.17.0 2020-04-27 [1] Bioconductor
## biomaRt 2.45.2 2020-07-05 [1] Bioconductor
## Biostrings * 2.57.2 2020-06-09 [1] Bioconductor
## bit 4.0.4 2020-08-04 [1] RSPM (R 4.0.2)
## bit64 4.0.5 2020-08-30 [1] RSPM (R 4.0.2)
## bitops 1.0-6 2013-08-17 [1] RSPM (R 4.0.0)
## blob 1.2.1 2020-01-20 [1] RSPM (R 4.0.0)
## bookdown 0.20 2020-06-23 [1] RSPM (R 4.0.2)
## BSgenome 1.57.6 2020-09-02 [1] Bioconductor
## bumphunter 1.31.0 2020-04-28 [1] Bioconductor
## callr 3.4.4 2020-09-07 [2] RSPM (R 4.0.2)
## checkmate 2.0.0 2020-02-06 [1] RSPM (R 4.0.0)
## cli 2.0.2 2020-02-28 [2] RSPM (R 4.0.0)
## cluster 2.1.0 2019-06-19 [3] CRAN (R 4.0.2)
## clusterProfiler * 3.17.3 2020-09-10 [1] Bioconductor
## codetools 0.2-16 2018-12-24 [3] CRAN (R 4.0.2)
## colorspace 1.4-1 2019-03-18 [1] RSPM (R 4.0.0)
## cowplot 1.1.0 2020-09-08 [1] RSPM (R 4.0.2)
## crayon 1.3.4 2017-09-16 [2] RSPM (R 4.0.0)
## curl 4.3 2019-12-02 [2] RSPM (R 4.0.0)
## data.table 1.13.0 2020-07-24 [1] RSPM (R 4.0.2)
## DBI 1.1.0 2019-12-15 [1] RSPM (R 4.0.0)
## dbplyr 1.4.4 2020-05-27 [1] RSPM (R 4.0.0)
## DelayedArray * 0.15.7 2020-07-14 [1] Bioconductor
## derfinder 1.23.1 2020-05-10 [1] Bioconductor
## derfinderHelper 1.23.1 2020-05-10 [1] Bioconductor
## desc 1.2.0 2018-05-01 [2] RSPM (R 4.0.0)
## devtools * 2.3.1 2020-07-21 [2] RSPM (R 4.0.2)
## digest 0.6.25 2020-02-23 [2] RSPM (R 4.0.0)
## D0.db 2.9 2020-09-10 [1] Bioconductor
## doRNG 1.8.2 2020-01-27 [1] RSPM (R 4.0.0)
## DOSE 3.15.0 2020-04-27 [1] Bioconductor
## downloader 0.4 2015-07-09 [1] RSPM (R 4.0.0)
## dplyr 1.0.2 2020-08-18 [1] RSPM (R 4.0.2)
## edgeR * 3.31.4 2020-06-10 [1] Bioconductor

```



##	ellipsis	0.3.1	2020-05-15	[2]	RSPM (R 4.0.0)
##	enrichplot	1.9.1	2020-04-29	[1]	Bioconductor
##	europemc	0.4	2020-05-31	[1]	RSPM (R 4.0.0)
##	evaluate	0.14	2019-05-28	[2]	RSPM (R 4.0.0)
##	fansi	0.4.1	2020-01-08	[2]	RSPM (R 4.0.0)
##	farver	2.0.3	2020-01-16	[1]	RSPM (R 4.0.0)
##	fastmatch	1.1-0	2017-01-28	[1]	RSPM (R 4.0.0)
##	fgsea	1.15.0	2020-04-27	[1]	Bioconductor
##	foreach	1.5.0	2020-03-30	[1]	RSPM (R 4.0.0)
##	foreign	0.8-80	2020-05-24	[3]	CRAN (R 4.0.2)
##	Formula	1.2-3	2018-05-03	[1]	RSPM (R 4.0.0)
##	fs	1.5.0	2020-07-31	[2]	RSPM (R 4.0.2)
##	generics	0.0.2	2018-11-29	[1]	RSPM (R 4.0.0)
##	GenomeInfoDb	* 1.25.11	2020-09-03	[1]	Bioconductor
##	GenomeInfoDbData	1.2.3	2020-09-10	[1]	Bioconductor
##	GenomicAlignments	1.25.3	2020-06-09	[1]	Bioconductor
##	GenomicFeatures	1.41.3	2020-09-03	[1]	Bioconductor
##	GenomicFiles	1.25.0	2020-04-27	[1]	Bioconductor
##	GenomicRanges	* 1.41.6	2020-08-12	[1]	Bioconductor
##	GEOquery	2.57.0	2020-04-27	[1]	Bioconductor
##	ggforce	0.3.2	2020-06-23	[1]	RSPM (R 4.0.2)
##	ggplot2	3.3.2	2020-06-19	[1]	RSPM (R 4.0.1)
##	ggplotify	0.0.5	2020-03-12	[1]	RSPM (R 4.0.0)
##	ggraph	2.0.3	2020-05-20	[1]	RSPM (R 4.0.2)
##	ggrepel	0.8.2	2020-03-08	[1]	RSPM (R 4.0.2)
##	ggridges	0.5.2	2020-01-12	[1]	RSPM (R 4.0.0)
##	glue	1.4.2	2020-08-27	[2]	RSPM (R 4.0.2)
##	G0.db	3.11.4	2020-09-10	[1]	Bioconductor
##	googledrive	1.0.1	2020-05-05	[1]	RSPM (R 4.0.0)
##	GOSemSim	2.15.2	2020-09-04	[1]	Bioconductor
##	graphlayouts	0.7.0	2020-04-25	[1]	RSPM (R 4.0.2)
##	gridExtra	2.3	2017-09-09	[1]	RSPM (R 4.0.0)
##	gridGraphics	0.5-0	2020-02-25	[1]	RSPM (R 4.0.0)
##	gtable	0.3.0	2019-03-25	[1]	RSPM (R 4.0.0)
##	here	* 0.1	2017-05-28	[1]	RSPM (R 4.0.0)
##	Hmisc	4.4-1	2020-08-10	[1]	RSPM (R 4.0.2)
##	hms	0.5.3	2020-01-08	[1]	RSPM (R 4.0.0)
##	htmlTable	2.0.1	2020-07-05	[1]	RSPM (R 4.0.2)
##	htmltools	0.5.0	2020-06-16	[2]	RSPM (R 4.0.1)
##	htmlwidgets	1.5.1	2019-10-08	[2]	RSPM (R 4.0.0)
##	httr	1.4.2	2020-07-20	[2]	RSPM (R 4.0.2)
##	igraph	1.2.5	2020-03-19	[1]	RSPM (R 4.0.0)
##	IRanges	* 2.23.10	2020-06-13	[1]	Bioconductor
##	iterators	1.0.12	2019-07-26	[1]	RSPM (R 4.0.0)
##	jaffelab	* 0.99.30	2020-09-10	[1]	Github (LieberInstitute/jaffelab@42637ff)
##	jpeg	0.1-8.1	2019-10-24	[1]	RSPM (R 4.0.0)
##	jsonlite	1.7.1	2020-09-07	[1]	RSPM (R 4.0.2)
##	knitcitations	* 1.0.10	2019-09-15	[1]	RSPM (R 4.0.0)
##	knitr	1.29	2020-06-23	[2]	RSPM (R 4.0.2)
##	lattice	0.20-41	2020-04-02	[3]	CRAN (R 4.0.2)
##	latticeExtra	0.6-29	2019-12-19	[1]	RSPM (R 4.0.0)
##	lifecycle	0.2.0	2020-03-06	[2]	RSPM (R 4.0.0)
##	limma	* 3.45.14	2020-09-12	[1]	Bioconductor
##	locfit	1.5-9.4	2020-03-25	[1]	RSPM (R 4.0.0)

##	lubridate	1.7.9	2020-06-08	[1]	RSPM (R 4.0.2)
##	magrittr	1.5	2014-11-22	[2]	RSPM (R 4.0.0)
##	MASS	7.3-53	2020-09-09	[3]	CRAN (R 4.0.2)
##	Matrix	* 1.2-18	2019-11-27	[3]	CRAN (R 4.0.2)
##	matrixStats	* 0.56.0	2020-03-13	[1]	RSPM (R 4.0.0)
##	memoise	1.1.0	2017-04-21	[2]	RSPM (R 4.0.0)
##	munsell	0.5.0	2018-06-12	[1]	RSPM (R 4.0.0)
##	nnet	7.3-14	2020-04-26	[3]	CRAN (R 4.0.2)
##	openssl	1.4.2	2020-06-27	[2]	RSPM (R 4.0.2)
##	org.Hs.eg.db	* 3.11.4	2020-09-25	[1]	Bioconductor
##	pheatmap	* 1.0.12	2019-01-04	[1]	RSPM (R 4.0.0)
##	pillar	1.4.6	2020-07-10	[2]	RSPM (R 4.0.2)
##	pkgbuild	1.1.0	2020-07-13	[2]	RSPM (R 4.0.2)
##	pkgconfig	2.0.3	2019-09-22	[2]	RSPM (R 4.0.0)
##	pkgload	1.1.0	2020-05-29	[2]	RSPM (R 4.0.0)
##	plyr	1.8.6	2020-03-03	[1]	RSPM (R 4.0.2)
##	png	0.1-7	2013-12-03	[1]	RSPM (R 4.0.0)
##	polyclip	1.10-0	2019-03-14	[1]	RSPM (R 4.0.0)
##	prettyunits	1.1.1	2020-01-24	[2]	RSPM (R 4.0.0)
##	processx	3.4.4	2020-09-03	[2]	RSPM (R 4.0.2)
##	progress	1.2.2	2019-05-16	[1]	RSPM (R 4.0.0)
##	ps	1.3.4	2020-08-11	[2]	RSPM (R 4.0.2)
##	purrr	0.3.4	2020-04-17	[2]	RSPM (R 4.0.0)
##	qvalue	2.21.0	2020-04-27	[1]	Bioconductor
##	R6	2.4.1	2019-11-12	[2]	RSPM (R 4.0.0)
##	rafalib	* 1.0.0	2015-08-09	[1]	RSPM (R 4.0.0)
##	rappdirs	0.3.1	2016-03-28	[1]	RSPM (R 4.0.0)
##	RColorBrewer	* 1.1-2	2014-12-07	[1]	RSPM (R 4.0.0)
##	Rcpp	1.0.5	2020-07-06	[2]	RSPM (R 4.0.2)
##	RCurl	1.98-1.2	2020-04-18	[1]	RSPM (R 4.0.0)
##	readr	1.3.1	2018-12-21	[1]	RSPM (R 4.0.2)
##	recount	* 1.15.1	2020-05-12	[1]	Bioconductor
##	RefManageR	1.2.12	2019-04-03	[1]	RSPM (R 4.0.0)
##	remotes	2.2.0	2020-07-21	[1]	RSPM (R 4.0.2)
##	rentrez	1.2.2	2019-05-02	[1]	RSPM (R 4.0.0)
##	reshape2	1.4.4	2020-04-09	[1]	RSPM (R 4.0.2)
##	rlang	0.4.7	2020-07-09	[2]	RSPM (R 4.0.2)
##	rmarkdown	2.3	2020-06-18	[1]	RSPM (R 4.0.1)
##	rngtools	1.5	2020-01-23	[1]	RSPM (R 4.0.0)
##	rpart	4.1-15	2019-04-12	[3]	CRAN (R 4.0.2)
##	rprojroot	1.3-2	2018-01-03	[2]	RSPM (R 4.0.0)
##	Rsamtools	* 2.5.3	2020-06-26	[1]	Bioconductor
##	RSQLite	2.2.0	2020-01-07	[1]	RSPM (R 4.0.2)
##	rstudioapi	0.11	2020-02-07	[2]	RSPM (R 4.0.0)
##	rtracklayer	1.49.5	2020-08-07	[1]	Bioconductor
##	rvcheck	0.1.8	2020-03-01	[1]	RSPM (R 4.0.0)
##	S4Vectors	* 0.27.12	2020-06-09	[1]	Bioconductor
##	scales	1.1.1	2020-05-11	[1]	RSPM (R 4.0.0)
##	scatterpie	0.1.4	2019-11-08	[1]	RSPM (R 4.0.0)
##	segmented	1.2-0	2020-06-23	[1]	RSPM (R 4.0.2)
##	sessioninfo	1.1.1	2018-11-05	[1]	RSPM (R 4.0.0)
##	statmod	1.4.34	2020-02-17	[1]	RSPM (R 4.0.0)
##	stringi	1.5.3	2020-09-09	[2]	RSPM (R 4.0.2)
##	stringr	1.4.0	2019-02-10	[2]	RSPM (R 4.0.0)

```

## SummarizedExperiment * 1.19.6 2020-07-09 [1] Bioconductor
## survival 3.2-3 2020-06-13 [3] CRAN (R 4.0.2)
## testthat 2.3.2 2020-03-02 [2] RSPM (R 4.0.0)
## tibble 3.0.3 2020-07-10 [2] RSPM (R 4.0.2)
## tidygraph 1.2.0 2020-05-12 [1] RSPM (R 4.0.2)
## tidyr * 1.1.2 2020-08-27 [1] RSPM (R 4.0.2)
## tidyselect 1.1.0 2020-05-11 [1] RSPM (R 4.0.0)
## tinytex 0.25 2020-07-24 [1] RSPM (R 4.0.2)
## triebeard 0.3.0 2016-08-04 [1] RSPM (R 4.0.2)
## tweenr 1.0.1 2018-12-14 [1] RSPM (R 4.0.2)
## urltools 1.7.3 2019-04-14 [1] RSPM (R 4.0.2)
## usethis * 1.6.3 2020-09-17 [2] RSPM (R 4.0.2)
## VariantAnnotation * 1.35.3 2020-06-26 [1] Bioconductor
## vctrs 0.3.4 2020-08-29 [2] RSPM (R 4.0.2)
## viridis 0.5.1 2018-03-29 [1] RSPM (R 4.0.0)
## viridisLite 0.3.0 2018-02-01 [1] RSPM (R 4.0.0)
## withr 2.2.0 2020-04-20 [2] RSPM (R 4.0.0)
## xfun 0.17 2020-09-09 [2] RSPM (R 4.0.2)
## XML 3.99-0.5 2020-07-23 [1] RSPM (R 4.0.2)
## xml2 1.3.2 2020-04-23 [2] RSPM (R 4.0.0)
## XVector * 0.29.3 2020-06-25 [1] Bioconductor
## yaml 2.2.1 2020-02-01 [2] RSPM (R 4.0.0)
## zlibbioc 1.35.0 2020-04-27 [1] Bioconductor
##
## [1] /usr/local/lib/R/host-site-library
## [2] /usr/local/lib/R/site-library
## [3] /usr/local/lib/R/library

```