Wk3 Microbiome Analysis - Data Management and Transformations

Christine V. Hawkes

1/26/2022

Contents

Coding exercise answers

1. Use the transform_sample_counts() function in phyloseq for custom data transformation
Examples with relative abundance and log2 - choose something different!

```
## relative (proportional) abundance  - example with function definition:
ps_ra <- phyloseq::transform_sample_counts(ps, function(x){x / sum(x)})

## log2 - no function definition
ps_log2 <- phyloseq::transform_sample_counts(ps, log2)
```

```
  ## repeat with a transform of your choice example - could be anything
```

2. Some transformations result in negative values that make other analyses difficult

```
# check min in clr transform
min(phyloseq::otu_table(ps_clr))
```

```
## [1] -2.746116
```

```
# add a constant to each cell
ps_clr_pos <- phyloseq::transform_sample_counts(ps_clr, function(x) {x+2.75})

#check that the transform worked
min(phyloseq::otu_table(ps_clr_pos))
```

```
## [1] 0.003884303
```

3. Rerun the DeSeq2 dispersion model with local and mean fits and compare to parametric fit dispersion plots - which is better?

```
# data to DeSeq
ps_ds <- phyloseq::phyloseq_to_deseq2(ps, ~Trt1 + Trt2)
```

```
## Loading required namespace: DESeq2
```

```
## converting counts to integer mode
```

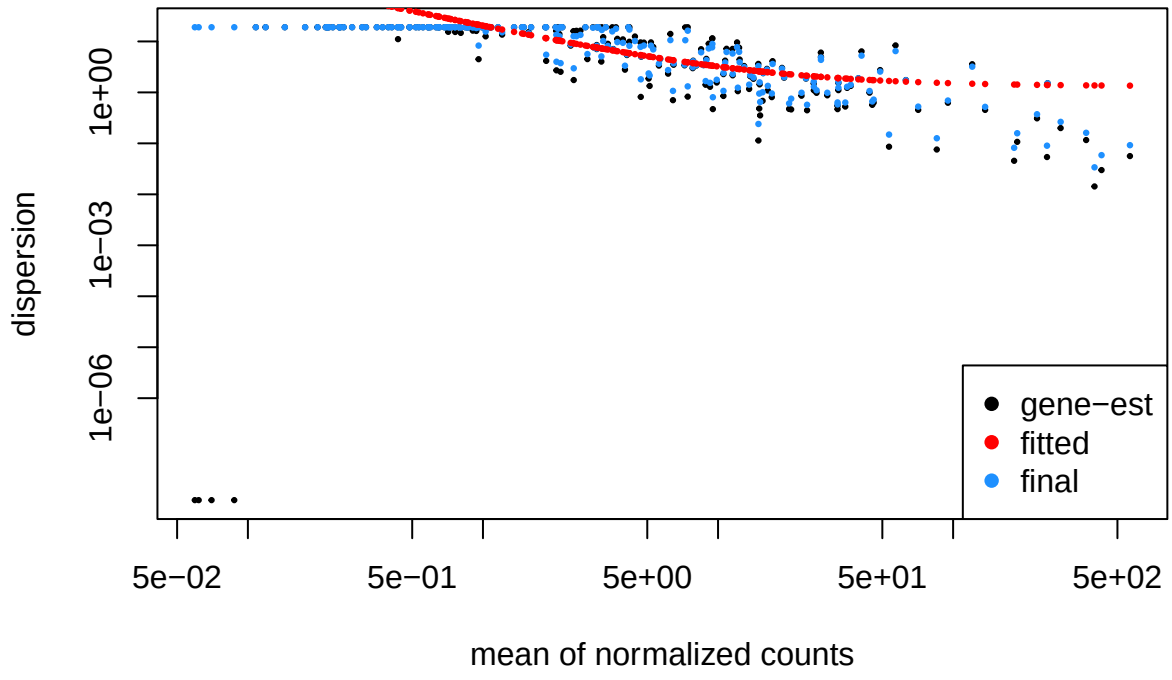
```
# parametric
ps_ds = DESeq2::estimateSizeFactors(ps_ds)
ps_ds = DESeq2::estimateDispersions(ps_ds, fitType = "parametric")
```

```
## gene-wise dispersion estimates
```

```
## mean-dispersion relationship
```

```
## final dispersion estimates
```

```
DESeq2::plotDispEsts(ps_ds)
```




```
# local
ps_ds = DESeq2::estimateSizeFactors(ps_ds)
ps_ds = DESeq2::estimateDispersions(ps_ds, fitType = "local")
```

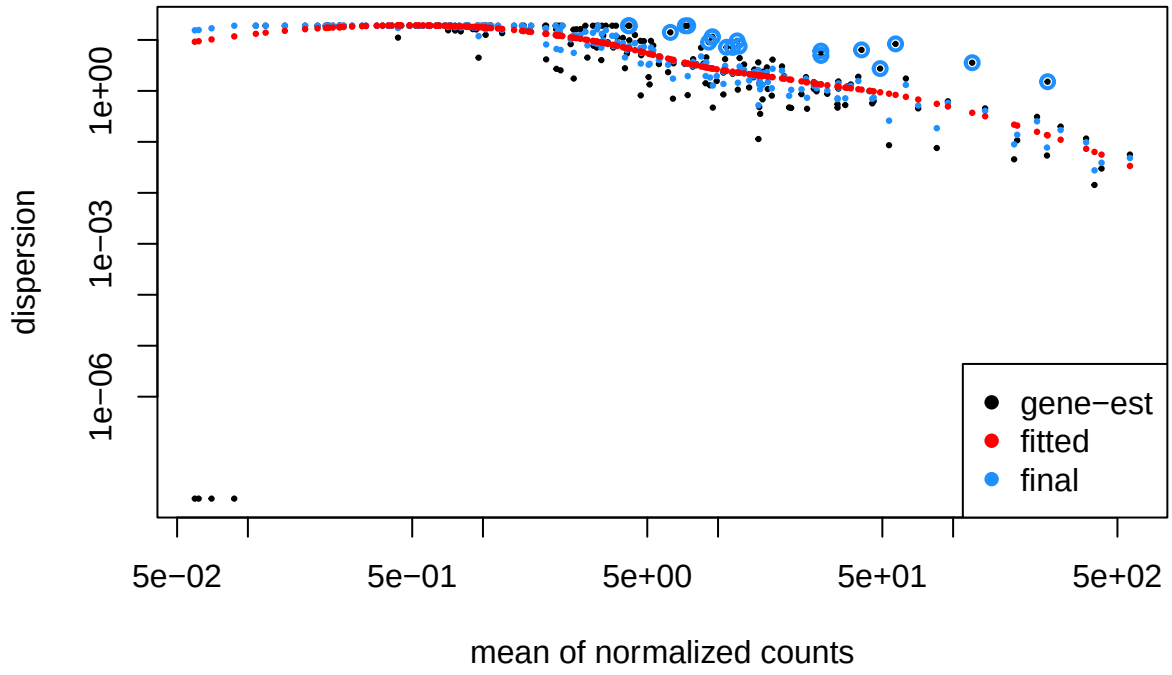
```
## found already estimated dispersions, replacing these

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates
```

```
DESeq2::plotDispEsts(ps_ds)
```




```
# mean
ps_ds = DESeq2::estimateSizeFactors(ps_ds)
ps_ds = DESeq2::estimateDispersions(ps_ds, fitType = "mean")
```

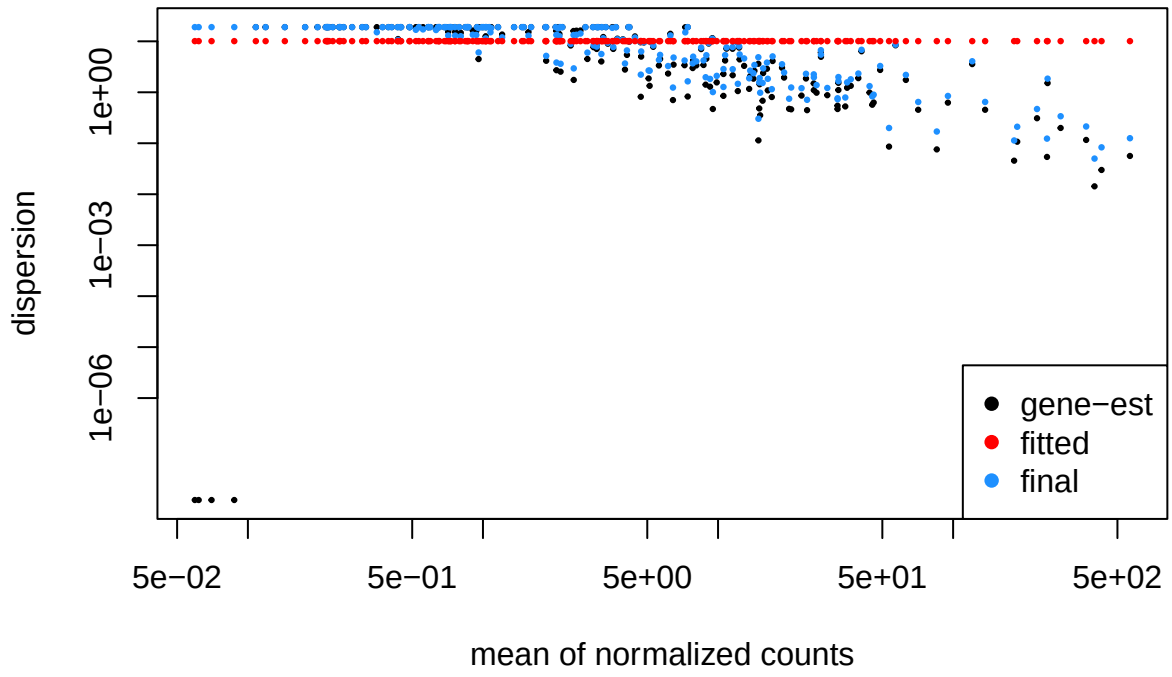
```
## found already estimated dispersions, replacing these

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates
```

```
DESeq2::plotDispEsts(ps_ds)
```



4. Practice creating a ps object

```
# open wk3_Wagner files
ASVw <- read.csv("Wk3_Wagner_sm_ASV_data.csv",  row.names = 1)
SAMw <- read.csv("Wk3_Wagner_sm_SAM_data.csv", row.names=1)
TAXw <- read.csv("Wk3_Wagner_sm_TAX_data.csv", row.names=1)

colnames(ASVw) # Are ASVs in rows or cols? Need to specify for taxa_are_rows
```

```
##   [1] "M1551P81"   "M1551P29"   "M1551P90"   "M1551P48"   "M1551P52"
##   [6] "M1551P31"   "M1551P77"   "M1551P37"   "M1551P40"   "M1551P72"
##  [11] "M1551P67"   "M1551P10"   "M1551P73"   "M1551P92"   "M1551P21"
##  [16] "M1551P68"   "M1551P8"    "M1551P83"   "M1551P80"   "M1551P56"
##  [21] "M1554P130"  "M1554P182"  "M1554P102"  "M1554P103"  "M1554P117"
##  [26] "M1554P119"  "M1554P126"  "M1554P140"  "M1554P144"  "M1554P157"
##  [31] "M1554P171"  "M1554P184"  "M1554P153"  "M1554P138"  "M1554P132"
##  [36] "M1554P179"  "M1554P101"  "M1554P160"  "M1554P128"  "M1554P175"
##  [41] "M1554P161"  "M1555P214"  "M1555P216"  "M1555P221"  "M1555P228"
##  [46] "M1555P231"  "M1555P250"  "M1555P254"  "M1555P255"  "M1555P265"
##  [51] "M1555P212"  "M1555P268"  "M1555P232"  "M1555P203"  "M1555P188"
##  [56] "M1555P272"  "M1555P238"  "M1555P225"  "M1555P245"  "M1555P207"
##  [61] "M1555P256"  "M1555P204"  "M1555P219"  "M1555P276"  "M1555P205"
##  [66] "M1555P271"  "M1555P190"  "M1555P239"  "M1555P195"  "M1555P240"
```

```
##  [71] "M1955P739"  "M1955P741"  "M1955P745"  "M1955P750"  "M1955P764"
##  [76] "M1955P767"  "M1955P814"  "M1956P873"  "M1957P1002" "M1958P1060"
##  [81] "M1958P1020" "M1959P1171" "M1960P1224" "M1960P1254" "M1961P1298"
##  [86] "M1961P1321" "M1961P1333" "M1961P1359" "M1961P1364" "M1977P1696"
##  [91] "M1977P1698" "M1977P1705" "M1977P1706" "M1977P1711" "M1977P1717"
##  [96] "M1977P1719" "M1977P1737" "M1977P1690" "M1978P290"  "M1978P293"
## [101] "M1978P296"  "M1978P300"  "M1978P306"  "M1978P307"  "M1978P308"
## [106] "M1978P314"  "M1978P316"  "M1978P317"  "M1978P318"  "M1978P327"
## [111] "M1978P350"  "M1978P353"  "M1978P356"  "M1978P359"  "M1978P361"
## [116] "M1979P422"  "M1979P426"  "M1979P430"  "M1979P442"  "M1979P447"
## [121] "M1979P449"  "M1979P454"  "M1979P458"  "M1979P439"  "M1979P420"
## [126] "M1979P383"  "M1979P394"  "M1979P440"  "M1979P369"  "M1979P378"
## [131] "M1979P381"  "M1979P441"  "M1979P377"  "M1979P417"  "M1979P398"
## [136] "M1979P404"  "M1979P370"  "M1979P451"  "M1980P461"  "M1980P462"
## [141] "M1980P464"  "M1980P469"  "M1980P470"  "M1980P476"  "M1980P477"
## [146] "M1980P481"  "M1980P484"  "M1980P487"  "M1980P489"  "M1980P491"
## [151] "M1980P496"  "M1980P499"  "M1980P501"  "M1980P503"  "M1980P513"
## [156] "M1980P519"  "M1980P520"  "M1980P532"  "M1980P533"  "M1980P534"
## [161] "M1980P535"  "M1980P539"  "M1980P540"  "M1980P541"  "M1980P543"
## [166] "M1980P547"  "M1980P507"  "M1980P510"  "M1980P494"  "M1980P538"
## [171] "M1980P483"  "M1980P505"  "M1980P506"  "M1980P486"  "M1980P471"
## [176] "M1980P502"  "M1981P555"  "M1981P556"  "M1981P560"  "M1981P561"
## [181] "M1981P563"  "M1981P564"  "M1981P565"  "M1981P569"  "M1981P573"
## [186] "M1981P575"  "M1981P577"  "M1981P578"  "M1981P582"  "M1981P584"
## [191] "M1981P586"  "M1981P587"  "M1981P588"  "M1981P589"  "M1981P591"
## [196] "M1981P593"  "M1981P594"  "M1981P597"  "M1981P598"  "M1981P599"
## [201] "M1981P606"  "M1981P609"  "M1981P611"  "M1981P614"  "M1981P616"
## [206] "M1981P618"  "M1981P619"  "M1981P623"  "M1981P626"  "M1981P631"
## [211] "M1981P632"  "M1981P636"  "M1981P638"  "M1981P639"  "M1981P643"
## [216] "M1982P701"  "M1982P729"  "M1982P728"  "M1982P663"  "M1982P696"
## [221] "M1982P691"  "M1982P730"  "M1982P705"  "M1982P667"  "M1982P670"
## [226] "M1982P661"  "M1982P651"  "M1982P664"  "M1982P711"  "M1982P645"
## [231] "M1982P662"  "M1982P679"  "M1982P720"  "M1982P710"  "M1982P646"
## [236] "M1982P724"  "M1982P683"  "M1982P703"  "M1982P655"  "M1982P707"
## [241] "M1982P700"  "M1024P1782" "M1024P1791" "M1024P1805" "M1024P1775"
## [246] "M1024P1790" "M1024P1794" "M1024P1814" "M1024P1799" "M1024P1779"
## [251] "M1024P1773" "M1024P1788" "M1024P1819" "M1024P1783" "M1551P7"
## [256] "M1551P65"   "M1551P20"   "M1551P35"   "M1551P16"   "M1551P70"
## [261] "M1551P51"   "M1554P112"  "M1554P124"  "M1554P163"  "M1690P1397"
## [266] "M1690P1399" "M1955P738"  "M1955P749"  "M1955P757"  "M1955P766"
## [271] "M1955P817"  "M1955P819"  "M1956P915"  "M1956P853"  "M1956P899"
## [276] "M1957P982"  "M1959P1106" "M1959P1121" "M1960P1219" "M1960P1238"
## [281] "M1961P1308" "M1961P1338" "M1961P1349" "M1961P1369" "M1977P1709"
## [286] "M1977P1714" "M1978P303"  "M1978P324"  "M1978P339"  "M1980P475"
## [291] "M1982P709"  "M1955P759"
```

```
str(TAXw) # Matrix? If not, convert below
```

```
## 'data.frame':    10919 obs. of  6 variables:
##  $ Kingdom   : chr  "Bacteria" "Bacteria" "Bacteria" "Bacteria" ...
##  $ Phylum    : chr  "Proteobacteria" "Proteobacteria" "Proteobacteria" "Proteobacteria" ...
##  $ Class     : chr  "Alphaproteobacteria" "Alphaproteobacteria" "Alphaproteobacteria" "Alphaproteoba
##  $ Order     : chr  "Rickettsiales" "Sphingomonadales" "Rickettsiales" "Rhizobiales" ...
##  $ Family    : chr  "mitochondria" "Sphingomonadaceae" "mitochondria" "Rhizobiaceae" ...
```

```
##  $ Confidence: num  1 1 1 1 1 1 1 1 1 1 ...
```

```
# make ps object
ASV <- phyloseq::otu_table(ASVw, taxa_are_rows = TRUE)
SAM <- phyloseq::sample_data(SAMw)
TAX <- phyloseq::tax_table(as.matrix(TAXw))

ps_wag <- phyloseq::phyloseq(ASV, SAM, TAX)
ps_wag
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 10919 taxa and 292 samples ]
## sample_data() Sample Data:       [ 292 samples by 12 sample variables ]
## tax_table()   Taxonomy Table:    [ 10919 taxa by 6 taxonomic ranks ]
```

5. Practice subsetting and filtering data

```
colnames(SAMw)
```

```
##  [1] "Name"       "Plant_ID"   "Type"       "Experiment" "Cohort"
##  [6] "Harvested"  "Age"        "Site"       "Treatment"  "Line"
## [11] "Genotype"   "Block"
```

```
# subset by sample data for genotype or site
ps_wag_g <- phyloseq::subset_samples(ps_wag, Genotype=="MIL")
ps_wag_g
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 10919 taxa and 65 samples ]
## sample_data() Sample Data:       [ 65 samples by 12 sample variables ]
## tax_table()   Taxonomy Table:    [ 10919 taxa by 6 taxonomic ranks ]
```

```
ps_wag_s <- phyloseq::subset_samples(ps_wag, Site=="Jam")
ps_wag_s
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 10919 taxa and 60 samples ]
## sample_data() Sample Data:       [ 60 samples by 12 sample variables ]
## tax_table()   Taxonomy Table:    [ 10919 taxa by 6 taxonomic ranks ]
```

```
# subset by phylum
phyloseq::get_taxa_unique(ps_wag, "Phylum") # note 40 different phyla - pick one
```

```
##  [1] "Proteobacteria"   "Actinobacteria"   "Cyanobacteria"    "Bacteroidetes"
##  [5] "Firmicutes"       "Verrucomicrobia"  "Chloroflexi"      "Gemmatimonadetes"
##  [9] "Acidobacteria"    "Planctomycetes"   "FBP"              "Armatimonadetes"
## [13] "Chlamydiae"       "Fibrobacteres"    "FCPU426"          "Spirochaetes"
## [17] "Chlorobi"         "TM6"              "TM7"              "Tenericutes"
## [21] "WPS-2"            "AD3"              "Elusimicrobia"    "BHI80-139"
## [25] "Nitrospirae"      "BRC1"             NA                 "OP11"
## [29] "Thermi"           "OD1"              "WS3"              "SR1"
## [33] "WS2"              "OP3"              "Aquificae"        "OP8"
## [37] "NKB19"            "MVP-21"           "GN02"             "Fusobacteria"
```

```r
ps_wag_p <- phyloseq::subset_taxa(ps_wag, Phylum=="Proteobacteria")
phyloseq::get_taxa_unique(ps_wag_p, "Phylum") # note only Proteobacteria
```

```
## [1] "Proteobacteria"
```

```r
# subset by read abundance
ps_wag_a <- phyloseq::prune_taxa(phyloseq::taxa_sums(ps_wag) > 1000, ps_wag)
ps_wag_a # note change from 10,919 in ps_wag to 1,931 ASVs in ps_wag_a
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 1931 taxa and 292 samples ]
## sample_data() Sample Data:       [ 292 samples by 12 sample variables ]
## tax_table()   Taxonomy Table:    [ 1931 taxa by 6 taxonomic ranks ]
```

Session Info

```
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
##   [1] nlme_3.1-152                bitops_1.0-7
##   [3] matrixStats_0.61.0          phyloseq_1.36.0
##   [5] bit64_4.0.5                 RColorBrewer_1.1-2
##   [7] httr_1.4.2                  GenomeInfoDb_1.28.4
##   [9] tensorA_0.36.2              tools_4.1.1
##  [11] utf8_1.2.2                  R6_2.5.1
##  [13] vegan_2.5-7                 DBI_1.1.1
##  [15] BiocGenerics_0.38.0         mgcv_1.8-36
##  [17] colorspace_2.0-2            permute_0.9-5
##  [19] rhdf5filters_1.4.0          ade4_1.7-18
##  [21] tidyselect_1.1.1            DESeq2_1.32.0
##  [23] bit_4.0.4                   bayesm_3.1-4
##  [25] compiler_4.1.1              compositions_2.0-2
##  [27] Biobase_2.52.0              DelayedArray_0.18.0
##  [29] scales_1.1.1                DEoptimR_1.0-9
##  [31] robustbase_0.93-9           genefilter_1.74.1
##  [33] stringr_1.4.0               digest_0.6.28
##  [35] rmarkdown_2.11              XVector_0.32.0
##  [37] pkgconfig_2.0.3             htmltools_0.5.2
##  [39] MatrixGenerics_1.4.3        highr_0.9
##  [41] fastmap_1.1.0               rlang_0.4.11
##  [43] RSQLite_2.2.8               generics_0.1.1
##  [45] jsonlite_1.7.2              BiocParallel_1.26.2
##  [47] dplyr_1.0.7                 RCurl_1.98-1.5
##  [49] magrittr_2.0.1              GenomeInfoDbData_1.2.6
##  [51] biomformat_1.20.0           Matrix_1.3-4
##  [53] Rcpp_1.0.7                  munsell_0.5.0
##  [55] S4Vectors_0.30.2            Rhdf5lib_1.14.2
##  [57] fansi_0.5.0                 ape_5.5
##  [59] lifecycle_1.0.1             stringi_1.7.5
##  [61] yaml_2.2.1                  MASS_7.3-54
##  [63] SummarizedExperiment_1.22.0 zlibbioc_1.38.0
```

```
##  [65] rhdf5_2.36.0               plyr_1.8.6
##  [67] grid_4.1.1                 blob_1.2.2
##  [69] parallel_4.1.1             crayon_1.4.1
##  [71] lattice_0.20-44            Biostrings_2.60.2
##  [73] splines_4.1.1              multtest_2.48.0
##  [75] annotate_1.70.0            KEGGREST_1.32.0
##  [77] locfit_1.5-9.4             knitr_1.36
##  [79] pillar_1.6.4               igraph_1.2.7
##  [81] GenomicRanges_1.44.0       geneplotter_1.70.0
##  [83] reshape2_1.4.4             codetools_0.2-18
##  [85] stats4_4.1.1               XML_3.99-0.8
##  [87] glue_1.4.2                 evaluate_0.14
##  [89] data.table_1.14.2          png_0.1-7
##  [91] vctrs_0.3.8                foreach_1.5.1
##  [93] gtable_0.3.0               purrr_0.3.4
##  [95] assertthat_0.2.1           cachem_1.0.6
##  [97] ggplot2_3.3.5              xfun_0.26
##  [99] xtable_1.8-4               survival_3.2-11
## [101] tibble_3.1.5               iterators_1.0.13
## [103] AnnotationDbi_1.54.1       memoise_2.0.0
## [105] IRanges_2.26.0             cluster_2.1.2
## [107] ellipsis_0.3.2
```