scRNAseq_analysis

Abinet Joan

2023-05-31 16:36:14 +0200

Loading packages

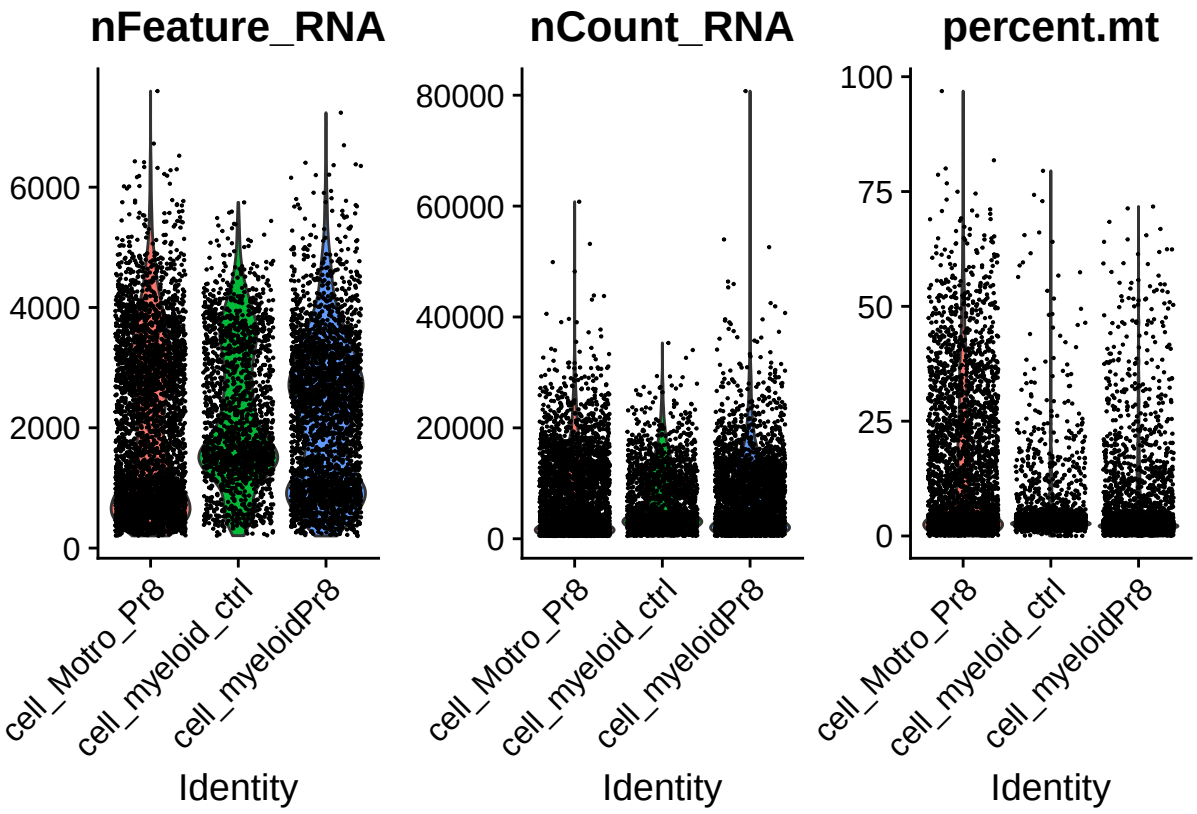
```
suppressMessages(library(dplyr))
suppressMessages(library(Seurat))
suppressMessages(library(patchwork))
suppressMessages(library(ggplot2))
suppressMessages(library(SingleR))
suppressMessages(library(formatR))
```

Loading data

```
myeloid_ctrl.data <- Read10X(data.dir = "Counts/1_sample_feature_bc_matrix/")

cell_MyeloidCtrl <- CreateSeuratObject(counts = myeloid_ctrl.data$`Gene Expression`,
    project = "cell_myeloid_ctrl", min.cells = 3, min.features = 200)

motro_pr8.data <- Read10X(data.dir = "Counts/2_sample_feature_bc_matrix/")
cell_MotroPr8 <- CreateSeuratObject(counts = motro_pr8.data$`Gene Expression`, project =
"cell_Motro_Pr8",
    min.cells = 3, min.features = 200)

myeloid_pr8.data <- Read10X(data.dir = "Counts/3_sample_feature_bc_matrix/")
cell_myeloidPr8 <- CreateSeuratObject(counts = myeloid_pr8.data$`Gene Expression`,
    project = "cell_myeloidPr8", min.cells = 3, min.features = 200)

# Merge the 3 seurat object
myeloid_cell <- merge(cell_MyeloidCtrl, y = c(cell_MotroPr8, cell_myeloidPr8),
add.cell.ids = c("1",
    "2", "3"), project = "Myeloid_cells")
```

Quality control

```
myeloid_cell[["percent.mt"]] <- PercentageFeatureSet(myeloid_cell, pattern = "^mt-")
VlnPlot(myeloid_cell, features = c("nFeature_RNA", "nCount_RNA", "percent.mt"), ncol = 3,
    pt.size = 0.1)
```



Pre-processing workflow

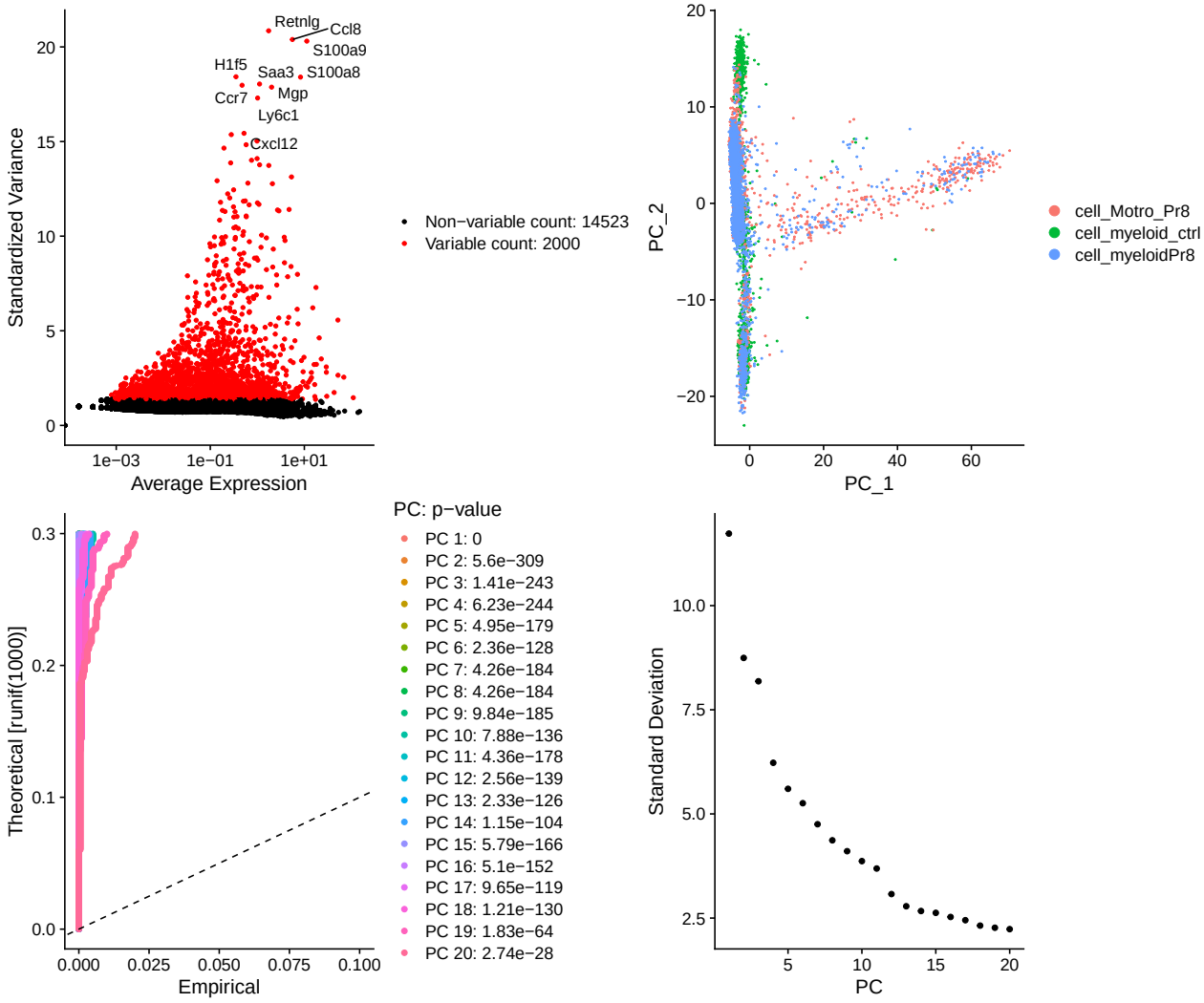
```
# removing low quality cells
myeloid_cell <- subset(myeloid_cell, subset = nFeature_RNA > 200 & nFeature_RNA <
    5000 & nCount_RNA < 20000 & percent.mt < 10)

# Normalizing the data
myeloid_cell <- NormalizeData(myeloid_cell, normalization.method = "LogNormalize",
    scale.factor = 10000)

# Feature selection
myeloid_cell <- FindVariableFeatures(myeloid_cell, selection.method = "vst", nfeatures =
2000)

# Identify the 10 most highly variable genes
top10 <- head(VariableFeatures(myeloid_cell), 10)

# plot variable features without labels
plot1 <- VariableFeaturePlot(myeloid_cell)
plot1 <- LabelPoints(plot = plot1, points = top10, repel = TRUE)

# Scaling the data
all.genes <- rownames(myeloid_cell)
myeloid_cell <- ScaleData(myeloid_cell, features = all.genes)

# Linear dimensional reduction
```

```
myeloid_cell <- RunPCA(myeloid_cell, features = VariableFeatures(object = myeloid_cell))
plot2 <- DimPlot(myeloid_cell, reduction = "pca")

# Determine the 'dimensionality' of the dataset
myeloid_cell <- JackStraw(myeloid_cell, num.replicate = 100)
myeloid_cell <- ScoreJackStraw(myeloid_cell, dims = 1:20)
plot3 <- JackStrawPlot(myeloid_cell, dims = 1:20)
plot4 <- ElbowPlot(myeloid_cell, ndims = 20)

# Cluster cells in umap
myeloid_cell <- FindNeighbors(myeloid_cell, dims = 1:15)
myeloid_cell <- FindClusters(myeloid_cell, resolution = 0.25)
myeloid_cell <- RunUMAP(myeloid_cell, dims = 1:15)
```

```
plot1 + plot2 + plot3 + plot4
```

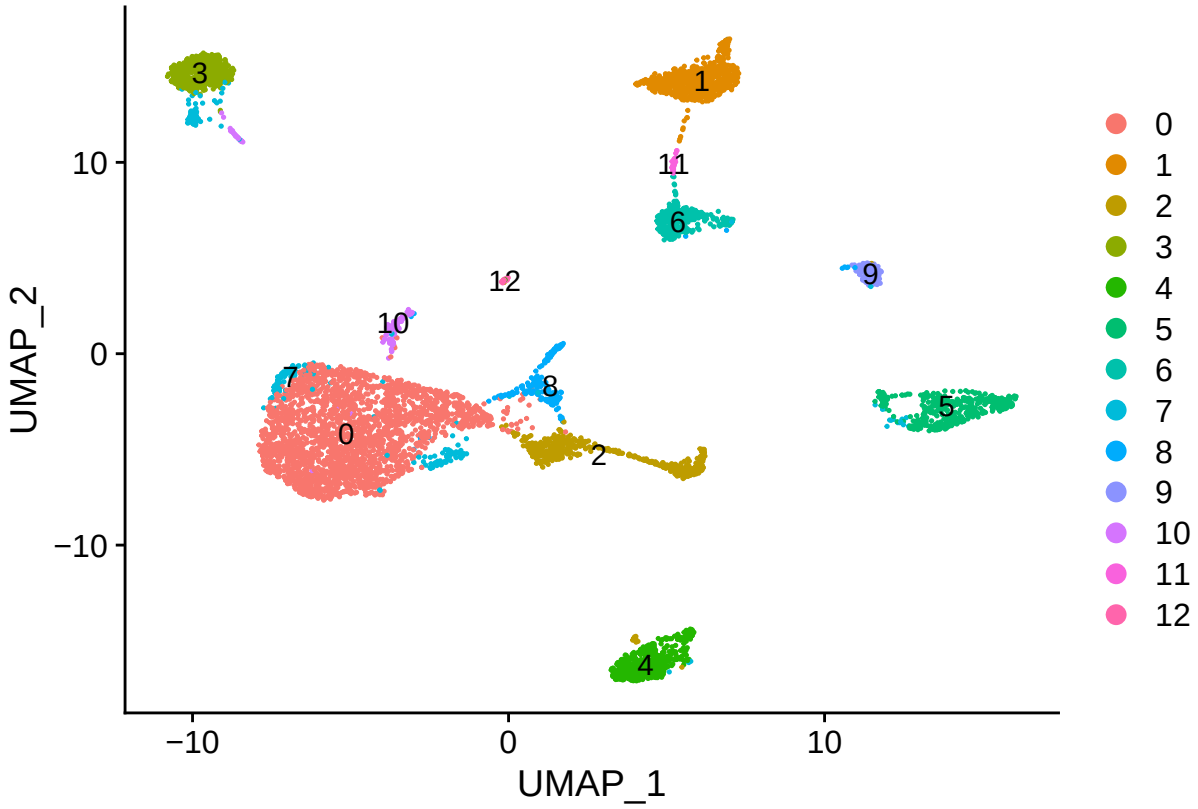
```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 28000 rows containing missing values (`geom_point()`).
```



Visualizing all clusters

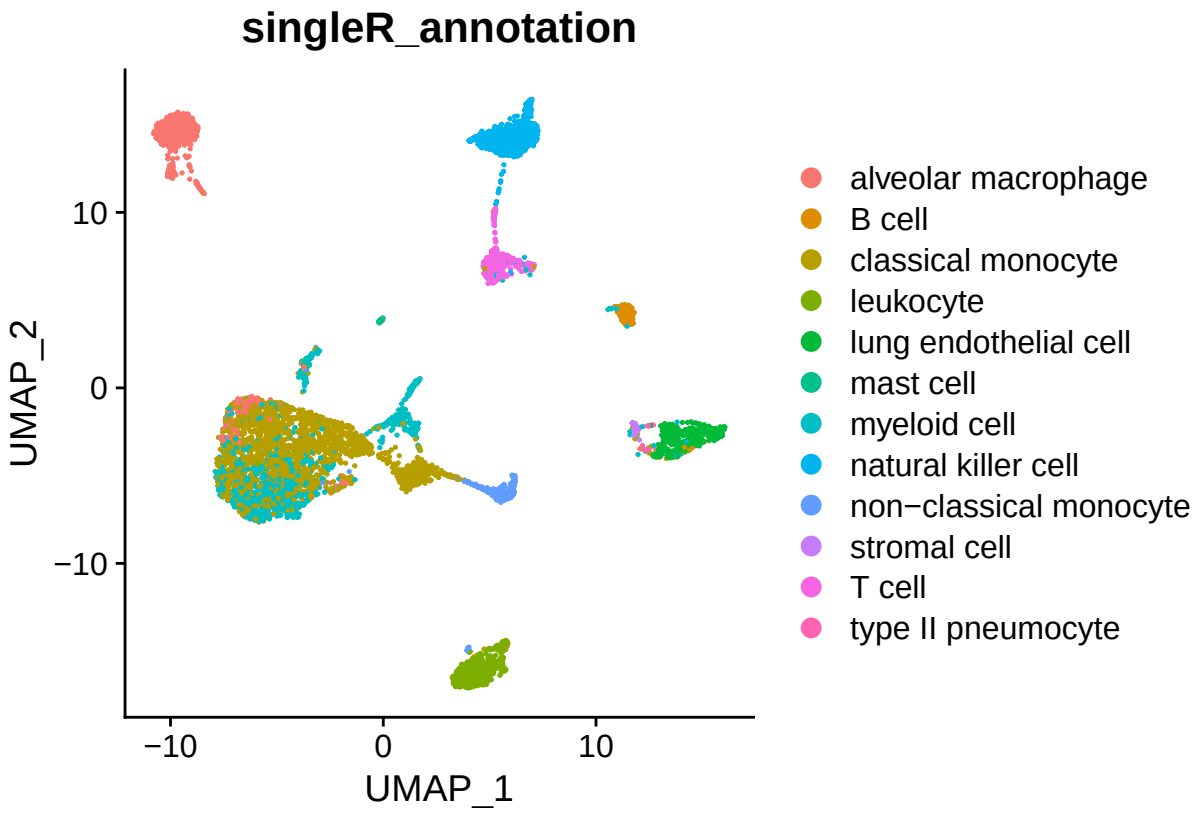
```
DimPlot(myeloid_cell, reduction = "umap", label = T)
```



Cell annotation

```
library(SingleR)
library(ExperimentHub)
library(scuttle)

eh <- ExperimentHub()
query(eh, "TabulaMurisData")

ref <- eh[["EH1617"]]
myeloid_ref <- ref[, !is.na(ref$cell_ontology_class)]
myeloid_ref <- myeloid_ref[, myeloid_ref$tissue == "Lung"]

myeloid_ref <- logNormCounts(myeloid_ref)

tested_data <- as.SingleCellExperiment(myeloid_cell)
tested_data <- logNormCounts(tested_data)

results <- SingleR(test = tested_data, ref = myeloid_ref, labels =
myeloid_ref$cell_ontology_class)
cell_annotations <- results

myeloid_cell[["singleR_annotation"]] <- cell_annotations[, c(4)]
```

```
DimPlot(myeloid_cell, reduction = "umap", group.by = "singleR_annotation")
```



Removing the variable used for annotation

```
rm(tested_data)
rm(myeloid_ref)
```

The next step is to subset the clusters of myeloid cells (Macropages, neutrophils, DCs). The others clusters will be removed.

Removing contamination

```
myeloid_cells <- subset(myeloid_cell, seurat_clusters %in% c(0, 2, 3, 4, 7, 8, 10))
```

Saving file

```
saveRDS(myeloid_cells, "Myeloid_cells_Part1.rds")
```

```
sessionInfo()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.6 LTS
##
```

```
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=fr_BE.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=fr_BE.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=fr_BE.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=fr_BE.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] TabulaMurisData_1.8.0       scuttle_1.0.4
##  [3] SingleCellExperiment_1.12.0 ExperimentHub_1.16.1
##  [5] AnnotationHub_2.22.1        BiocFileCache_1.14.0
##  [7] dbplyr_2.2.1                formatR_1.14
##  [9] SingleR_1.4.1               SummarizedExperiment_1.20.0
## [11] Biobase_2.50.0              GenomicRanges_1.42.0
## [13] GenomeInfoDb_1.26.7         IRanges_2.24.1
## [15] S4Vectors_0.28.1            BiocGenerics_0.36.1
## [17] MatrixGenerics_1.2.1        matrixStats_0.63.0
## [19] ggplot2_3.4.0               patchwork_1.1.2
## [21] SeuratObject_4.1.3          Seurat_4.3.0
## [23] dplyr_1.0.10
##
## loaded via a namespace (and not attached):
##   [1] utf8_1.2.3                   spatstat.explore_3.0-5
##   [3] reticulate_1.27              R.utils_2.12.2
##   [5] tidyselect_1.2.0             RSQLite_2.3.0
##   [7] AnnotationDbi_1.52.0         htmlwidgets_1.6.1
##   [9] grid_4.0.3                   BiocParallel_1.24.1
##  [11] Rtsne_0.16                   munsell_0.5.0
##  [13] codetools_0.2-19             ica_1.0-3
##  [15] future_1.32.0                miniUI_0.1.1.1
##  [17] withr_2.5.0                  spatstat.random_3.1-3
##  [19] colorspace_2.1-0             progressr_0.13.0
##  [21] highr_0.10                   knitr_1.42
##  [23] rstudioapi_0.14              ROCR_1.0-11
##  [25] tensor_1.5                   listenv_0.9.0
##  [27] labeling_0.4.2               GenomeInfoDbData_1.2.4
##  [29] polyclip_1.10-4              bit64_4.0.5
##  [31] farver_2.1.1                 parallelly_1.34.0
##  [33] vctrs_0.5.2                  generics_0.1.3
##  [35] xfun_0.37                    R6_2.5.1
##  [37] ggbeeswarm_0.7.1             rsvd_1.0.5
##  [39] bitops_1.0-7                 spatstat.utils_3.0-1
##  [41] cachem_1.0.7                 DelayedArray_0.16.3
##  [43] assertthat_0.2.1             promises_1.2.0.1
##  [45] scales_1.2.1                 beeswarm_0.4.0
```

```
##  [47] gtable_0.3.1                 beachmat_2.6.4
##  [49] globals_0.16.2               goftest_1.2-3
##  [51] rlang_1.0.6                  splines_4.0.3
##  [53] lazyeval_0.2.2               spatstat.geom_3.0-6
##  [55] BiocManager_1.30.20          yaml_2.3.7
##  [57] reshape2_1.4.4               abind_1.4-5
##  [59] httpuv_1.6.9                 tools_4.0.3
##  [61] ellipsis_0.3.2               RColorBrewer_1.1-3
##  [63] ggridges_0.5.4               Rcpp_1.0.10
##  [65] plyr_1.8.8                   sparseMatrixStats_1.2.1
##  [67] zlibbioc_1.36.0              purrr_1.0.1
##  [69] RCurl_1.98-1.10              deldir_1.0-6
##  [71] pbapply_1.7-0                cowplot_1.1.1
##  [73] zoo_1.8-11                   ggrepel_0.9.2
##  [75] cluster_2.1.0                magrittr_2.0.3
##  [77] data.table_1.14.8            scattermore_0.8
##  [79] lmtest_0.9-40                RANN_2.6.1
##  [81] fitdistrplus_1.1-8           mime_0.12
##  [83] evaluate_0.20                xtable_1.8-4
##  [85] gridExtra_2.3                compiler_4.0.3
##  [87] tibble_3.1.8                 KernSmooth_2.23-20
##  [89] crayon_1.5.2                 R.oo_1.25.0
##  [91] htmltools_0.5.4              later_1.3.0
##  [93] tidyr_1.2.1                  DBI_1.1.3
##  [95] MASS_7.3-53                  rappdirs_0.3.3
##  [97] Matrix_1.5-3                 cli_3.6.0
##  [99] R.methodsS3_1.8.2            igraph_1.4.1
## [101] pkgconfig_2.0.3              sp_1.6-0
## [103] plotly_4.10.1                spatstat.sparse_3.0-0
## [105] vipor_0.4.5                  XVector_0.30.0
## [107] stringr_1.5.0                digest_0.6.31
## [109] sctransform_0.3.5            RcppAnnoy_0.0.20
## [111] spatstat.data_3.0-0          rmarkdown_2.19
## [113] leiden_0.4.3                 uwot_0.1.14
## [115] DelayedMatrixStats_1.12.3    curl_5.0.0
## [117] shiny_1.7.4                  lifecycle_1.0.3
## [119] nlme_3.1-162                 jsonlite_1.8.4
## [121] BiocNeighbors_1.8.2          viridisLite_0.4.1
## [123] fansi_1.0.4                  pillar_1.8.1
## [125] lattice_0.20-41              ggrastr_1.0.1
## [127] fastmap_1.1.1                httr_1.4.5
## [129] survival_3.2-7               interactiveDisplayBase_1.28.0
## [131] glue_1.6.2                   png_0.1-8
## [133] BiocVersion_3.12.0           bit_4.0.5
## [135] stringi_1.7.12               blob_1.2.3
## [137] BiocSingular_1.6.0           memoise_2.0.1
## [139] irlba_2.3.5.1                future.apply_1.10.0
```