

# 5-Scoring Endothelial cells

Joan Abinet

2026-01-09 16:35:23 +0100

## Contents

<b>Introduction</b>	<b>2</b>
<b>Loading packages</b>	<b>2</b>
<b>Loading Annotated cell</b>	<b>2</b>
<b>Calculate gene signature for the PBS and Treated gCap</b>	<b>3</b>
<b>data from reference atlas</b>	<b>3</b>
<b>gCap</b>	<b>3</b>
Score Treated . . . . .	3
Score PBS . . . . .	4
<b>aCap</b>	<b>5</b>
Calculate gene signature . . . . .	5
Score Treated . . . . .	6
Score PBS . . . . .	6

## Introduction

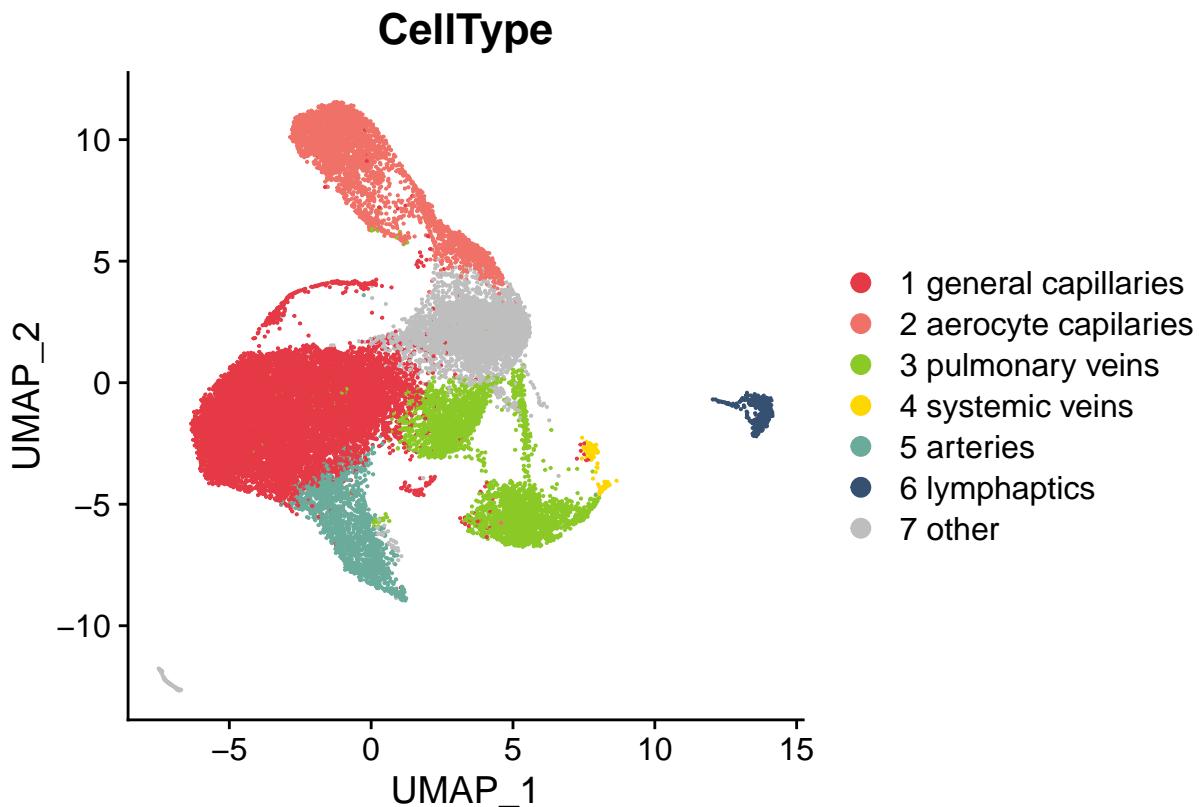
Gene signatures specific of transcriptional change in gCap were obtained by selecting the top 20 genes with the highest absolute log fold-change in each condition (control and MarNeu-targeted). The 2 signatures were then tested on Murine endothelial cells from different tissues. using the AddModuleScore function.

## Loading packages

```
suppressMessages({  
  library(dplyr)  
  library(Seurat)  
  library(patchwork)  
  library(ggplot2)  
  library(formatR)  
  library(dittoSeq)  
  library(ggrepel)  
  library(xlsx)  
})
```

## Loading Annotated cell

```
endothelial_cells <- readRDS("../02-Clustering_Endo/endothelial_cells_annotated.rds")  
  
color <- c("#E63946", "#F07167", "#8AC926", "#FFD700", "#6AAB9C",  
  "#355070", "gray")  
DimPlot(endothelial_cells, group.by = "CellType", cols = color)
```



## Calculate gene signature for the PBS and Treated gCap

```
endothelial_cells_subset <- subset(endothelial_cells, CellType %in%  
  "1 general capillaries")  
  
Markers.PBSTreated <- FindMarkers(endothelial_cells_subset, ident.1 = "PBS",  
  ident.2 = c("aCXCR2-D14", "aLy6G-D14"), min.pct = 0.25, group.by = "Condition")  
  
Markers.PBSTreated <- Markers.PBSTreated[order(Markers.PBSTreated$avg_log2FC,  
  decreasing = T), ]  
  
signature_PBS <- rownames(Markers.PBSTreated)[1:20]  
  
signature_treated <- tail(rownames(Markers.PBSTreated), 20)
```

## data from reference atlas

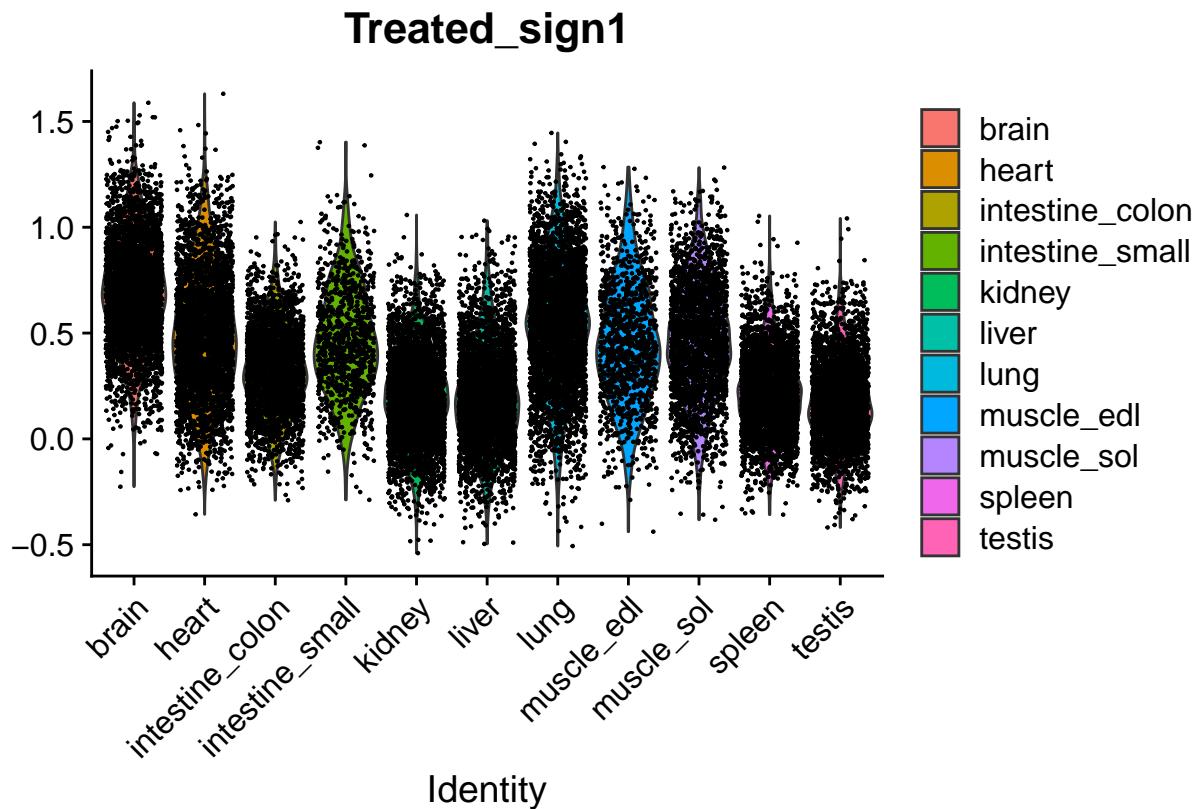
The raw counts and the metadata can be downloaded in [https://endotheliomics.shinyapps.io/ec\\_atlas/](https://endotheliomics.shinyapps.io/ec_atlas/)

```
Count_matrix <- read.csv("Data_endoMuris/Data.csv")  
rownames(Count_matrix) <- Count_matrix[, 1]  
Count_matrix <- Count_matrix[, -1]  
  
metadata <- read.csv("Data_endoMuris/Metadata.csv")  
rownames(metadata) <- metadata$Observation  
  
endo_cells_atlas <- CreateSeuratObject(Count_matrix, project = "Murine Endothelial Cells",  
  meta.data = metadata)  
endo_cells_atlas <- AddMetaData(endo_cells_atlas, metadata)
```

## gCap

### Score Treated

```
endo_cells_atlas <- AddModuleScore(endo_cells_atlas, features = list(signature_treated),  
  name = "Treated_sign")  
  
VlnPlot(endo_cells_atlas, features = "Treated_sign1", group.by = "Tissue")
```



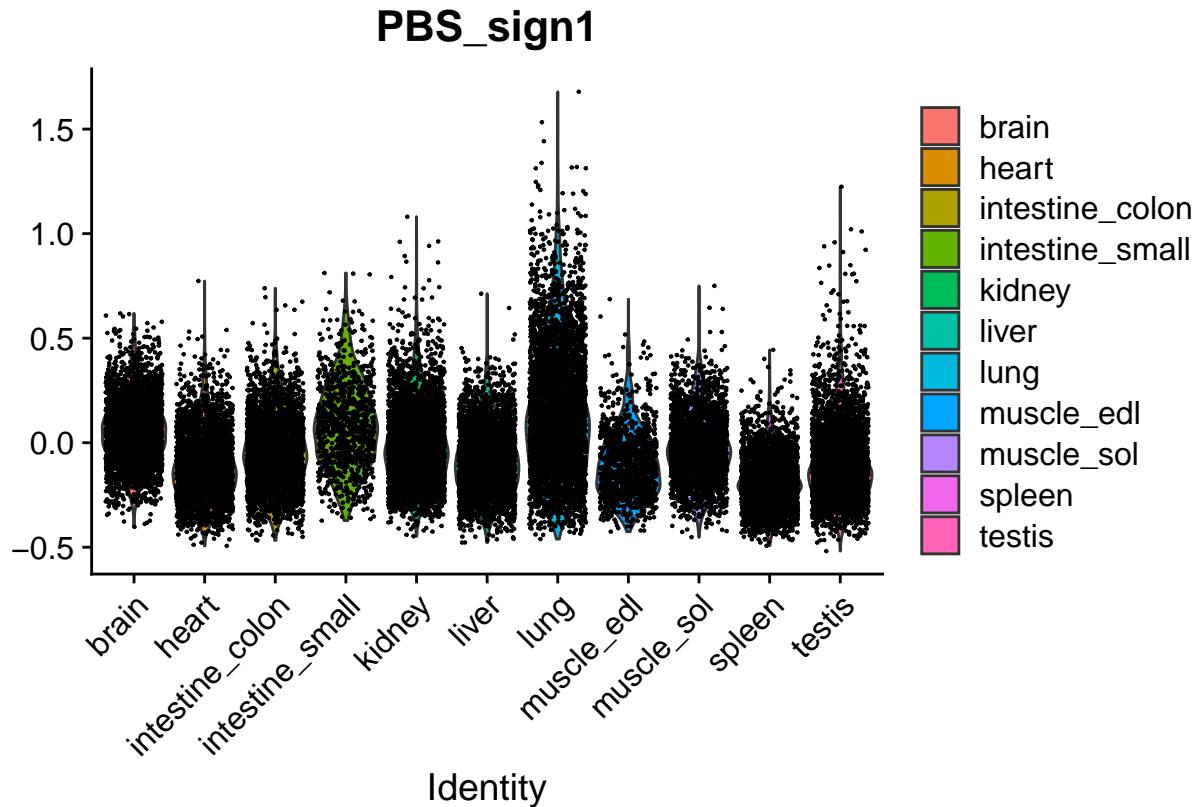
```
# ggsave('endo_atlas_treated.png', width = 12, height = 8)

Treated_Score <- FetchData(endo_cells_atlas, vars = c("Treated_sign1",
  "Tissue"))
# write.xlsx(Treated_Score, 'Scoring_endo_treated.xlsx',
# rowNames = T)
```

### Score PBS

```
endo_cells_atlas <- AddModuleScore(endo_cells_atlas, features = list(signature_PBS),
  name = "PBS_sign")

## Warning: The following features are not present in the object: Scgb1a1, not
## searching for symbol synonyms
VlnPlot(endo_cells_atlas, features = "PBS_sign1", group.by = "Tissue")
```



```
# ggsave('endo_atlas_PBS.png', width = 12, height = 8)

PBS_Score <- FetchData(endo_cells_atlas, vars = c("PBS_sign1", "Tissue"))
# write.xlsx(PBS_Score, 'Scoring_endo_PBS.xlsx', rowNames = T)
```

## aCap

### Calculate gene signature

```
endothelial_cells_subset <- subset(endothelial_cells, CellType %in%
  "2 aerocyte capilaries")

Markers.PBSTreated <- FindMarkers(endothelial_cells_subset, ident.1 = "PBS",
  ident.2 = c("aCXCR2-D14", "aLy6G-D14"), min.pct = 0.1, group.by = "Condition")

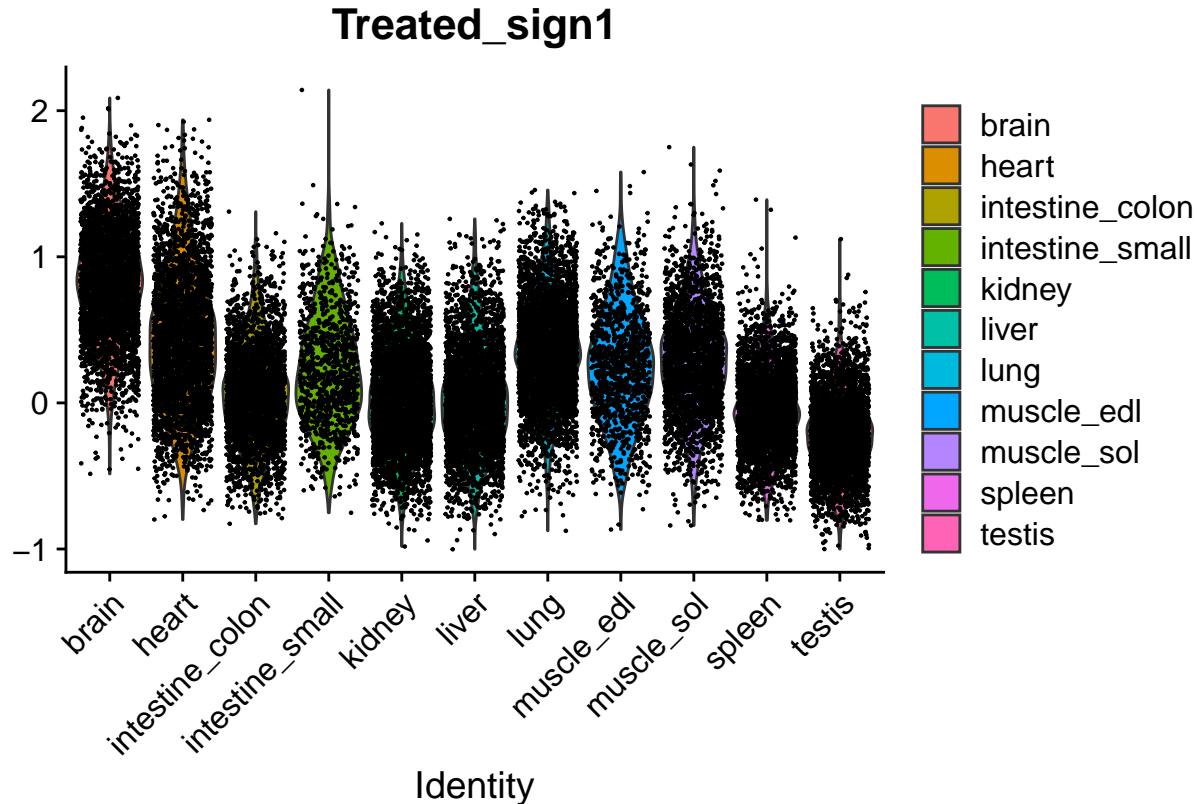
Markers.PBSTreated <- Markers.PBSTreated[order(Markers.PBSTreated$avg_log2FC,
  decreasing = T), ]
Markers.PBSTreated <- Markers.PBSTreated[Markers.PBSTreated$p_val_adj <
  0.05, ]

signature_PBS <- rownames(Markers.PBSTreated)[1:6]

signature_treated <- tail(rownames(Markers.PBSTreated), 10)
```

## Score Treated

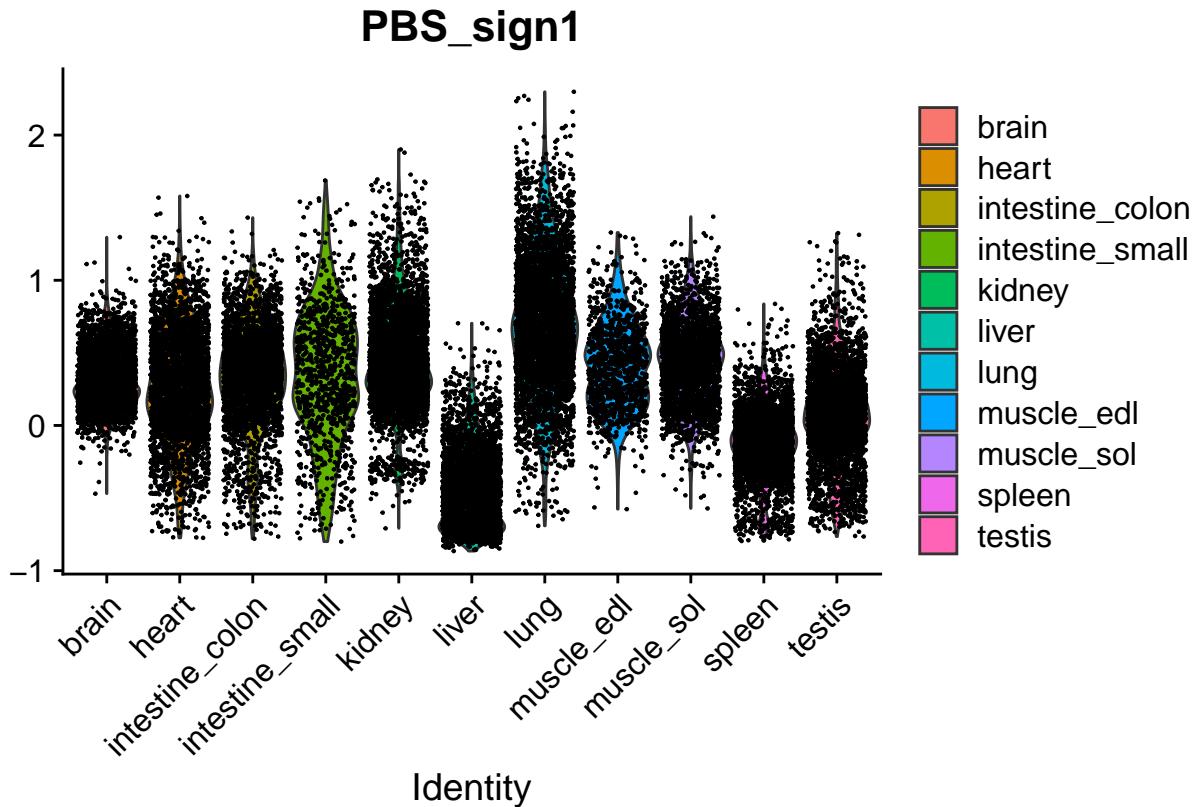
```
endo_cells_atlas <- AddModuleScore(endo_cells_atlas, features = list(signature_treated),  
                                    name = "Treated_sign")  
  
VlnPlot(endo_cells_atlas, features = "Treated_sign1", group.by = "Tissue")
```



```
ggsave("atlas_acap_treated.png", width = 12, height = 8)  
  
Treated_Score <- FetchData(endo_cells_atlas, vars = c("Treated_sign1",  
                                         "Tissue"))  
# write.xlsx(Treated_Score, 'Scoring_acap_treated.xlsx',  
# row.names = T)
```

## Score PBS

```
endo_cells_atlas <- AddModuleScore(endo_cells_atlas, features = list(signature_PBS),  
                                    name = "PBS_sign")  
  
## Warning: The following features are not present in the object: Scgb1a1, not  
## searching for symbol synonyms  
VlnPlot(endo_cells_atlas, features = "PBS_sign1", group.by = "Tissue")
```



```
ggsave("atlas_acap_PBS.png", width = 12, height = 8)

PBS_Score <- FetchData(endo_cells_atlas, vars = c("PBS_sign1", "Tissue"))
# write.xlsx(PBS_Score, 'Scoring_acap_PBS.xlsx', row.names = T)
```

Endothelial cell scores were subsequently normalized and expressed relative to brain endothelial cells. The results were plotted into violin plot with prism.

```
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 24.04.3 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/libopenblas-p0.3.26.so;  LAPACK version 3.12.0
##
## 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=fr_BE.UTF-8
## [9] LC_ADDRESS=fr_BE.UTF-8        LC_TELEPHONE=fr_BE.UTF-8
## [11] LC_MEASUREMENT=fr_BE.UTF-8   LC_IDENTIFICATION=fr_BE.UTF-8
##
## time zone: Europe/Brussels
## tzcode source: system (glibc)
##
```

```

## attached base packages:
## [1] stats      graphics   grDevices utils     datasets   methods    base
##
## other attached packages:
## [1] xlsx_0.6.5        ggrepel_0.9.3      dittoSeq_1.12.0    formatR_1.14
## [5] ggplot2_3.4.2     patchwork_1.1.2    SeuratObject_4.1.3 Seurat_4.3.0
## [9] dplyr_1.1.2
##
## loaded via a namespace (and not attached):
##  [1] RColorBrewer_1.1-3          rstudioapi_0.14
##  [3] jsonlite_1.8.7              magrittr_2.0.3
##  [5] ggbeeswarm_0.7.2           spatstat.utils_3.0-3
##  [7] farver_2.1.1                rmarkdown_2.23
##  [9] ragg_1.2.5                 zlibbioc_1.46.0
## [11] vctrs_0.6.3                ROCOCR_1.0-11
## [13] spatstat.explore_3.2-1     RCurl_1.98-1.12
## [15] S4Arrays_1.2.1             htmltools_0.5.5
## [17] sctransform_0.3.5          parallelly_1.36.0
## [19] KernSmooth_2.23-22         htmlwidgets_1.6.2
## [21] ica_1.0-3                  plyr_1.8.8
## [23] plotly_4.10.2              zoo_1.8-12
## [25] igraph_1.5.0.1            mime_0.12
## [27] lifecycle_1.0.3            pkgconfig_2.0.3
## [29] Matrix_1.6-1               R6_2.5.1
## [31] fastmap_1.1.1              GenomeInfoDbData_1.2.10
## [33] MatrixGenerics_1.12.2       fitdistrplus_1.1-11
## [35] future_1.33.0              shiny_1.7.4.1
## [37] digest_0.6.33              colorspace_2.1-0
## [39] S4Vectors_0.38.1           tensor_1.5
## [41] irlba_2.3.5.1              textshaping_0.3.6
## [43] GenomicRanges_1.52.0        labeling_0.4.2
## [45] progressr_0.13.0            fansi_1.0.4
## [47] spatstat.sparse_3.0-2       httr_1.4.6
## [49] polyclip_1.10-4            abind_1.4-5
## [51] compiler_4.3.3              withr_2.5.0
## [53] highr_0.10                 MASS_7.3-60.0.1
## [55] DelayedArray_0.26.3         tools_4.3.3
## [57] viper_0.4.5                lmtest_0.9-40
## [59] beeswarm_0.4.0              httpuv_1.6.11
## [61] future.apply_1.11.0          goftest_1.2-3
## [63] glue_1.6.2                  nlme_3.1-164
## [65] promises_1.2.0.1            grid_4.3.3
## [67] Rtsne_0.16                  cluster_2.1.6
## [69] reshape2_1.4.4              generics_0.1.3
## [71] gtable_0.3.3                spatstat.data_3.0-1
## [73] tidyR_1.3.0                 data.table_1.14.8
## [75] XVector_0.40.0              sp_2.2-0
## [77] utf8_1.2.3                 BiocGenerics_0.46.0
## [79] spatstat.geom_3.2-4          RcppAnnoy_0.0.21
## [81] RANN_2.6.1                  pillar_1.9.0
## [83] stringr_1.5.0
## [85] spam_2.9-1
## [87] rJava_1.0-6
## [89] lattice_0.22-5             limma_3.56.2
##                                later_1.3.1
##                                splines_4.3.3
##                                survival_3.5-8

```

```
## [91] deldir_1.0-9                  tidyselect_1.2.0
## [93] SingleCellExperiment_1.22.0    miniUI_0.1.1.1
## [95] pbapply_1.7-2                 knitr_1.43
## [97] gridExtra_2.3                  IRanges_2.34.0
## [99] SummarizedExperiment_1.30.2   scattermore_1.2
## [101] stats4_4.3.3                 xfun_0.39
## [103] Biobase_2.60.0                matrixStats_1.0.0
## [105] pheatmap_1.0.12               stringi_1.8.4
## [107] lazyeval_0.2.2                yaml_2.3.7
## [109] xlsxjars_0.6.1                evaluate_0.21
## [111] codetools_0.2-19              tibble_3.2.1
## [113] cli_3.6.1                   uwot_0.1.16
## [115] systemfonts_1.0.4             xtable_1.8-4
## [117] reticulate_1.30               munsell_0.5.0
## [119] Rcpp_1.0.11                  GenomeInfoDb_1.36.0
## [121] globals_0.16.2                spatstat.random_3.1-5
## [123] png_0.1-8                   ggrastr_1.0.2
## [125] parallel_4.3.3               ellipsis_0.3.2
## [127] dotCall64_1.0-2              bitops_1.0-7
## [129] listenv_0.9.0                viridisLite_0.4.2
## [131] scales_1.2.1                 ggridges_0.5.4
## [133] crayon_1.5.2                 leiden_0.4.3
## [135] purrrr_1.0.1                 rlang_1.1.1
## [137] cowplot_1.1.1
```