

# Scoring human Macrophages

Abinet Joan

2023-10-04 16:46:35 +0200

## Contents

Loading packages	1
Loading data	1
Quality control	2
Pre-processing Workflow	2
Adding cell type to metadata	5
Heatmap clusters	6
Score Ly6G Macs	7

## Loading packages

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

## Loading data

```
all_dirs <- list.dirs(path = "Data", full.names = TRUE, recursive = F)

list_sample_name <- c("Sample_1", "Sample_2", "Sample_3", "Sample_4")

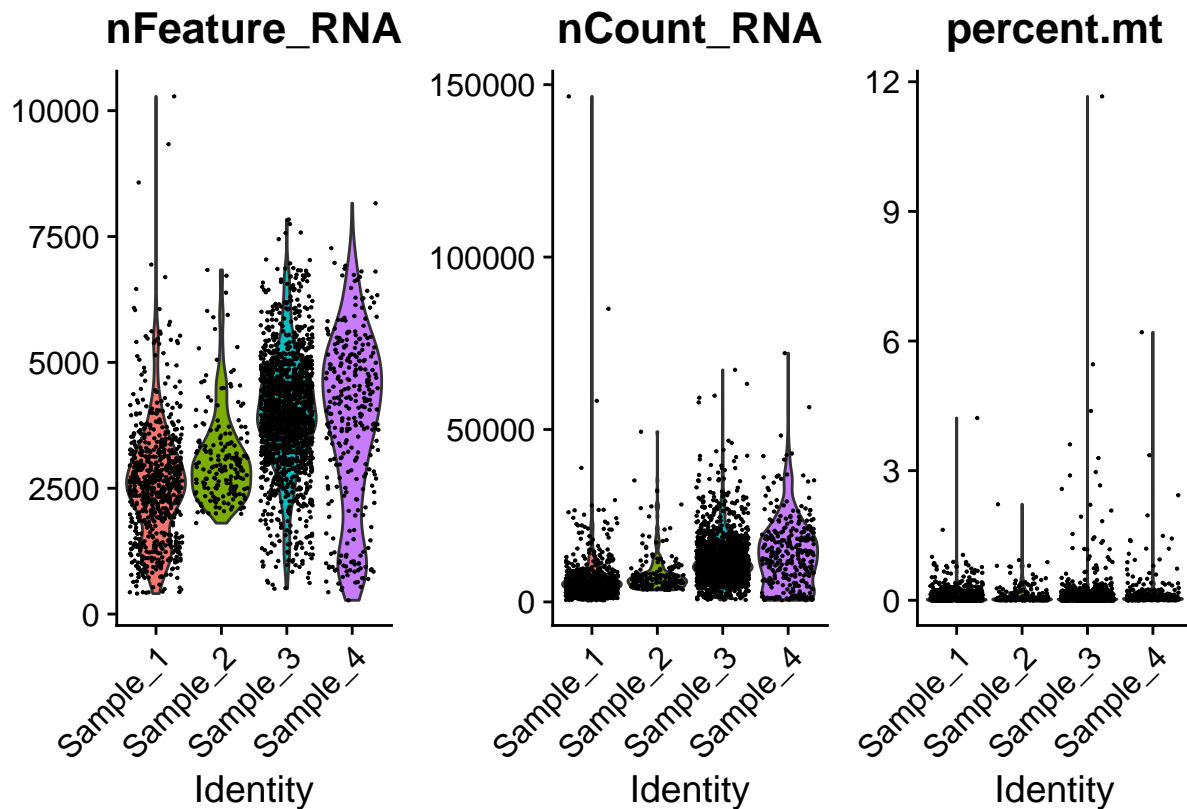
list_sample <- list()
for (i in 1:length(all_dirs)) {

  Seq_raw_file <- Read10X(data.dir = all_dirs[i])
  Seurat_file <- CreateSeuratObject(counts = Seq_raw_file, project = list_sample_name[i],
min.cells = 3, min.features = 200)
  list_sample <- append(list_sample, Seurat_file)
}
list_sample
```

```
lavage_cells <- merge(list_sample[[1]], y = list_sample[-1], add.cell.ids =
c("1","2","3", "4"), project = "bronchoalveolar_lavage")
```

## Quality control

```
lavage_cells[["percent.mt"]] <- PercentageFeatureSet(lavage_cells, pattern = "^MT-")# MT
: human cells
VlnPlot(lavage_cells, features = c("nFeature_RNA", "nCount_RNA", "percent.mt"), ncol = 3,
pt.size = 0.1)
```



## Pre-processing Workflow

```
# removing low quality cells
lavage_cells <- subset(lavage_cells, subset = nFeature_RNA > 200 & nFeature_RNA < 6000 &
percent.mt < 5)

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

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

```

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

plot1 <- VariableFeaturePlot(lavage_cells)
plot1 <- LabelPoints(plot = plot1, points = 10, repel = T)

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

# Linear dimensional reduction
lavage_cells <- RunPCA(lavage_cells, features = VariableFeatures(object = lavage_cells))
plot2 <- DimPlot(lavage_cells, reduction = "pca")

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

lavage_cells <- FindNeighbors(lavage_cells, dims = 1:15)
lavage_cells <- FindClusters(lavage_cells, resolution = 0.5)
lavage_cells <- RunUMAP(lavage_cells, dims = 1:15)

```

```

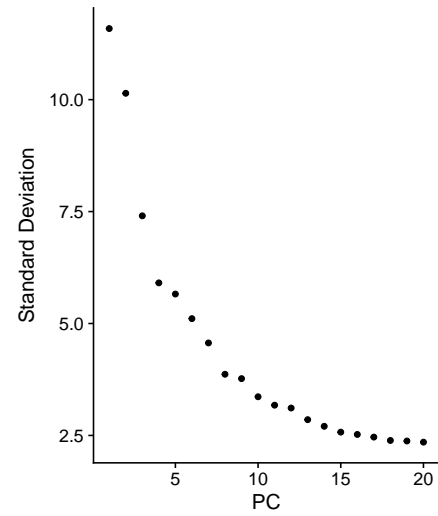
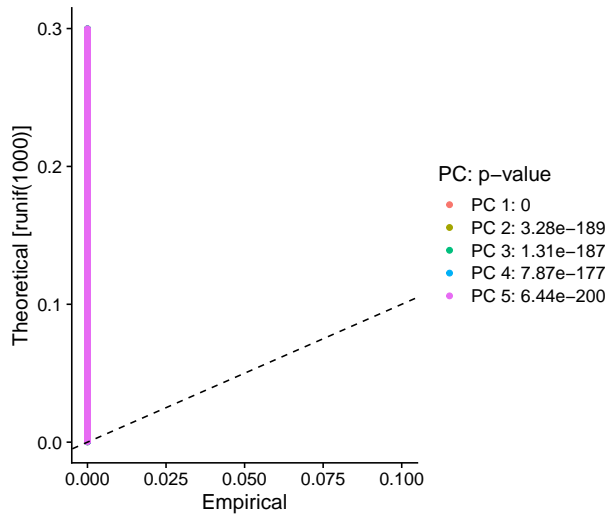
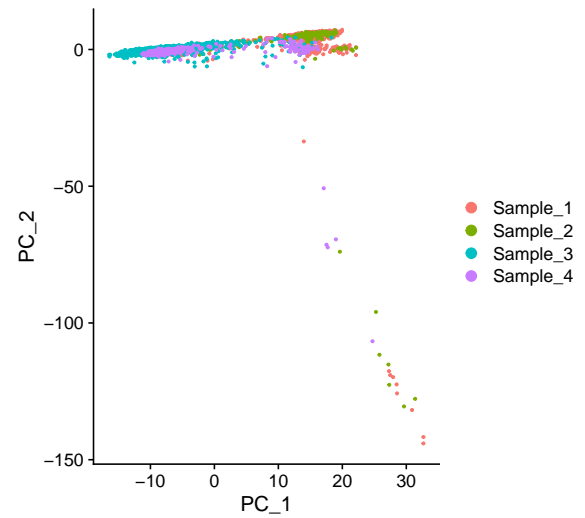
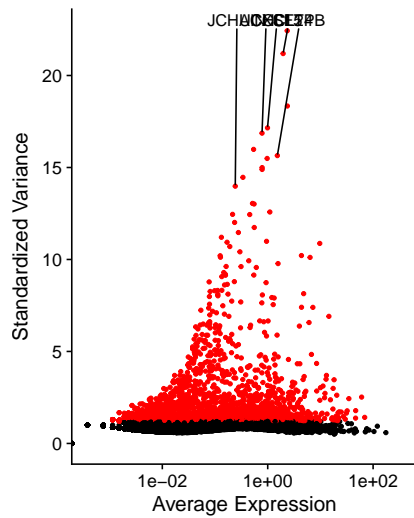
plot1 + plot2 + plot3 + plot4

```

```

## Warning: Transformation introduced infinite values in continuous x-axis
## Transformation introduced infinite values in continuous x-axis
## Warning: Removed 7000 rows containing missing values (`geom_point()`).
## Warning: ggrepel: 13435 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

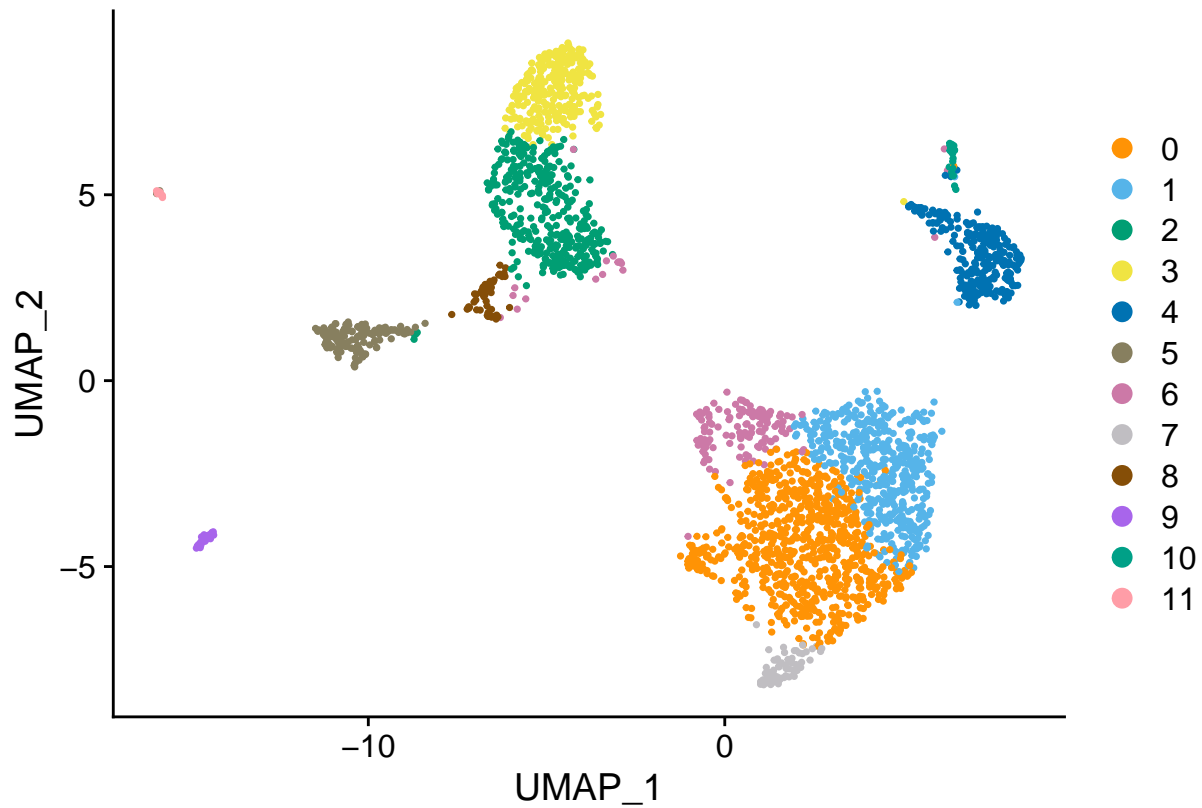
```



```
lavage_cells@reductions$umap@cell.embeddings[,2] <-  
-lavage_cells@reductions$umap@cell.embeddings[,2]  
lavage_cells@reductions$umap@cell.embeddings[,1] <-  
-lavage_cells@reductions$umap@cell.embeddings[,1]
```

```
my_palette <- c("#ff9305", "#56B4E9", "#009E73", "#F0E442", "#0072B2",  
               "#877f5f", "#CC79A7", "#c0bec2", "#854e07", "#a865eb",  
               "#00A087", "#FF9DA7", "#A55AFF", "#8C4B4B", "#00A08E")
```

```
DimPlot(lavage_cells, reduction = "umap", cols = my_palette)
```

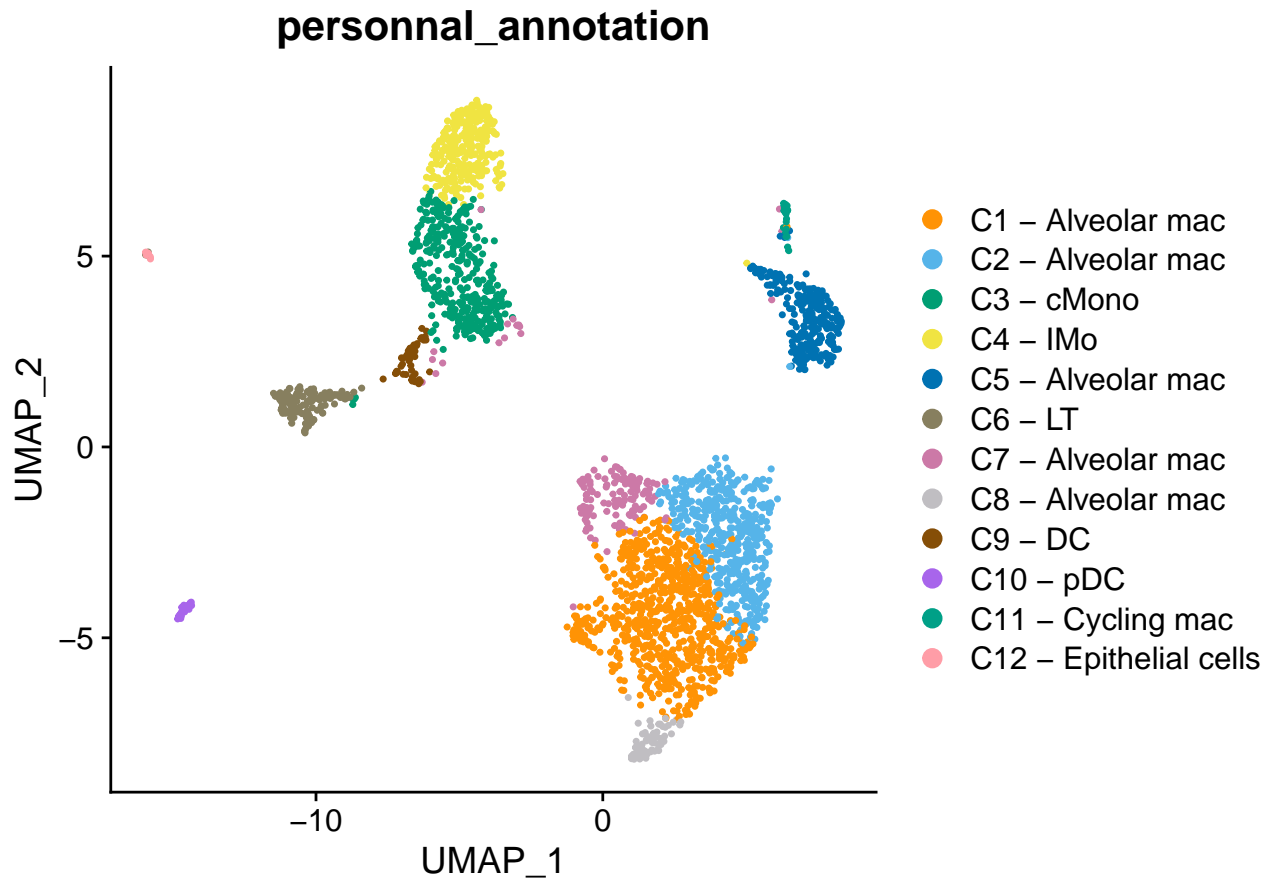


## Adding cell type to metadata

```
lavage_cells$personnal_annotation <- lavage_cells$seurat_clusters
levels(lavage_cells$personnal_annotation) <- c(
  "C1 - Alveolar mac",
  "C2 - Alveolar mac",
  "C3 - cMono",
  "C4 - IMo",
  "C5 - Alveolar mac",
  "C6 - LT",
  "C7 - Alveolar mac",
  "C8 - Alveolar mac",
  "C9 - DC",
  "C10 - pDC",
  "C11 - Cycling mac",
  "C12 - Epithelial cells"
)

my_palette <- c("#ff9305", "#56B4E9", "#009E73", "#F0E442", "#0072B2",
  "#877f5f", "#CC79A7", "#c0bec2", "#854e07", "#a865eb",
  "#00A087", "#FF9DA7", "#A55AFF", "#8C4B4B", "#00A08E")

DimPlot(lavage_cells, reduction = "umap", group.by = "personnal_annotation", cols =
my_palette)
```



```
#ggsave("Umap_annotated.pdf", width = 7, height = 5)
```

## Heatmap clusters

```
lavage.markers <- FindAllMarkers(lavage_cells, only.pos = TRUE, min.pct = 0.25)
```

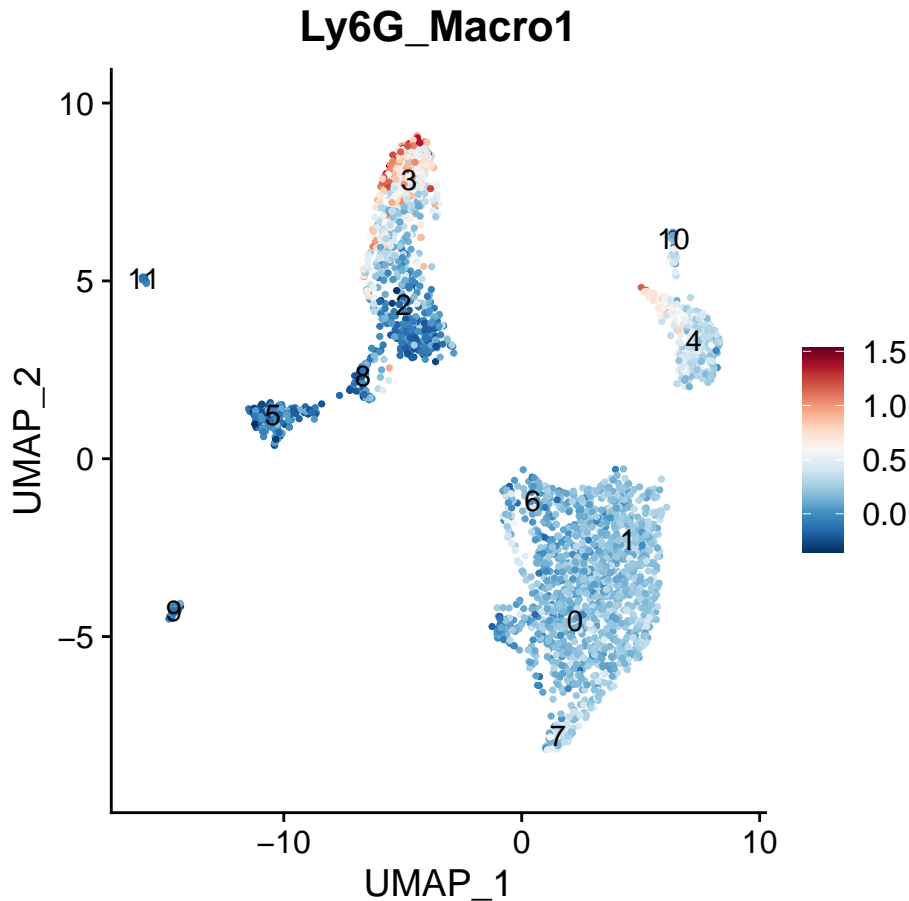
```
lavage.markers %>%
  group_by(cluster) %>%
  top_n(n = 10, wt = avg_log2FC) -> top10
```

```
DoHeatmap(lavage_cells, features = top10$gene, group.colors = my_palette, group.by =
"personnal_annotation") + NoLegend() + scale_fill_gradientn(colors = c("blue", "white",
"red"))+
  theme(plot.margin = margin(t = 1, r = 4, b = 1, l = 3, unit = "cm"))
```



```
lavage_cells <- AddModuleScore(lavage_cells,
                               features = list(top_motro_human),
                               name="Ly6G_Macro")

FeaturePlot(lavage_cells, features = "Ly6G_Macro1", label = T) +
  scale_colour_gradientn(colours = rev(brewer.pal(n = 11, name = "RdBu")))
```



```
#ggsave("ScoreMacro.pdf", width = 5.5, height = 5)
```

```
sessionInfo()
```

```
## R version 4.3.1 (2023-06-16)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 22.04.3 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-p0.3.20.so; LAPACK version 3.10.0
##
## 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=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
```



```

## [11] LC_MEASUREMENT=fr_BE.UTF-8 LC_IDENTIFICATION=C
##
## time zone: Europe/Brussels
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] RColorBrewer_1.1-3 formatR_1.14      ggplot2_3.4.2      patchwork_1.1.2
## [5] SeuratObject_4.1.3 Seurat_4.3.0      dplyr_1.1.2
##
## loaded via a namespace (and not attached):
## [1] deldir_1.0-9          pbapply_1.7-2          gridExtra_2.3
## [4] rlang_1.1.1           magrittr_2.0.3         RcppAnnoy_0.0.21
## [7] spatstat.geom_3.2-4   matrixStats_1.0.0      ggridges_0.5.4
## [10] compiler_4.3.1        png_0.1-8              vctrs_0.6.3
## [13] reshape2_1.4.4        stringr_1.5.0          pkgconfig_2.0.3
## [16] fastmap_1.1.1         ellipsis_0.3.2         labeling_0.4.2
## [19] utf8_1.2.3            promises_1.2.0.1       rmarkdown_2.23
## [22] purrr_1.0.1           xfun_0.39              jsonlite_1.8.7
## [25] goftest_1.2-3         highr_0.10             later_1.3.1
## [28] spatstat.utils_3.0-3  irlba_2.3.5.1          parallel_4.3.1
## [31] cluster_2.1.4         R6_2.5.1               ica_1.0-3
## [34] stringi_1.7.12        spatstat.data_3.0-1     limma_3.56.2
## [37] reticulate_1.30       parallelly_1.36.0      lmtest_0.9-40
## [40] scattermore_1.2       Rcpp_1.0.11            knitr_1.43
## [43] tensor_1.5            future.apply_1.11.0     zoo_1.8-12
## [46] R.utils_2.12.2        sctransform_0.3.5      httpuv_1.6.11
## [49] Matrix_1.6-0          splines_4.3.1          igraph_1.5.0.1
## [52] tidyselect_1.2.0      rstudioapi_0.14        abind_1.4-5
## [55] yaml_2.3.7            spatstat.random_3.1-5   codetools_0.2-19
## [58] miniUI_0.1.1.1        spatstat.explore_3.2-1 listenv_0.9.0
## [61] lattice_0.21-8        tibble_3.2.1           plyr_1.8.8
## [64] withr_2.5.0           shiny_1.7.4.1          ROCR_1.0-11
## [67] evaluate_0.21         Rtsne_0.16             future_1.33.0
## [70] survival_3.5-5        polyclip_1.10-4        fitdistrplus_1.1-11
## [73] pillar_1.9.0          KernSmooth_2.23-22     plotly_4.10.2
## [76] generics_0.1.3        sp_2.0-0               munsell_0.5.0
## [79] scales_1.2.1          globals_0.16.2         xtable_1.8-4
## [82] glue_1.6.2            lazyeval_0.2.2         tools_4.3.1
## [85] data.table_1.14.8     RANN_2.6.1             dotCall64_1.0-2
## [88] leiden_0.4.3          cowplot_1.1.1          grid_4.3.1
## [91] tidyr_1.3.0           colorspace_2.1-0       nlme_3.1-162
## [94] cli_3.6.1            spatstat.sparse_3.0-2  spam_2.9-1
## [97] fansi_1.0.4           viridisLite_0.4.2      uwot_0.1.16
## [100] gtable_0.3.3          R.methodsS3_1.8.2      digest_0.6.33
## [103] progressr_0.13.0      ggrepel_0.9.3          farver_2.1.1
## [106] htmlwidgets_1.6.2     R.oo_1.25.0            htmltools_0.5.5
## [109] lifecycle_1.0.3       httr_1.4.6             mime_0.12
## [112] MASS_7.3-60

```