

# Cell Type Anotation

Cell type annotation with clustering

Ximing Ran

2025-05-01

## Contents

1. Read the merged Data	2
2. Culstering	3

```
library(tibble)
library(tidyr)
library(dplyr)
library(rtracklayer)
library(dplyr)
library(Seurat)
library(Signac)
library(EnsDb.Hsapiens.v86)
library(ggplot2)
library(cowplot)
library(simspec)
library(cowplot)
library(AnnotationHub)
library(Seurat)
library(tidyverse)
library(openxlsx)
library(dplyr)
library(conflicted)
library(harmony)
library(data.table)
library(pheatmap)
library(ggplot2)
library(patchwork)

conflict_prefer("filter", "dplyr") # Prefer dplyr's filter()
conflict_prefer("lag", "dplyr")    # Prefer dplyr's lag()

# load function from local files
source(here::here("source", "sc_functions.R"))
```

## 1. Read the merged Data

```
seurat_obj <- readRDS(here::here("data", "processed-data", "merged_2025-05-01.rds"))
seurat_obj<- JoinLayers(seurat_obj)
```

## 2. Culstering

```
# create a directory to save the results
dir.create("result/02-Clustering-v2/umap_plot", recursive = TRUE, showWarnings = FALSE)

ct_label <- read.csv(here::here("data", "processed-data", "ct_mannual_2025-04-09.csv"), row.names = 1)
gene_list_clean <- unlist(strsplit(ct_label$marker, ",")) # Split by comma
gene_list_clean <- unique(trimws(gene_list_clean)) # Trim whitespace, remove duplicates
gene_list_clean <- gene_list_clean[gene_list_clean != ""]

#####
## Run clustering res from 0.1 to 0.5 #####
for (res in seq(0.1, 0.5, by = 0.1)) {
  seurat_obj <- FindNeighbors(seurat_obj, reduction = "harmony", dims = 1:50)
  seurat_obj <- FindClusters(seurat_obj, resolution = res)

  # plot the UMAP with the cluster labels
  # seurat_obj$seurat_clusters<- seurat_obj[[sprintf("RNA_snn_res.%s", res)]]
  p <- DimPlot(seurat_obj,
                reduction = "umap",
                group.by = "seurat_clusters", label = TRUE) +
    ggtitle(paste("Resolution:", res)) +
    theme(legend.position = "none")

  print(p)

  dir.create(sprintf("result/02-Clustering-v2/umap_plot/res_%s", res), recursive = TRUE, showWarnings = FALSE)

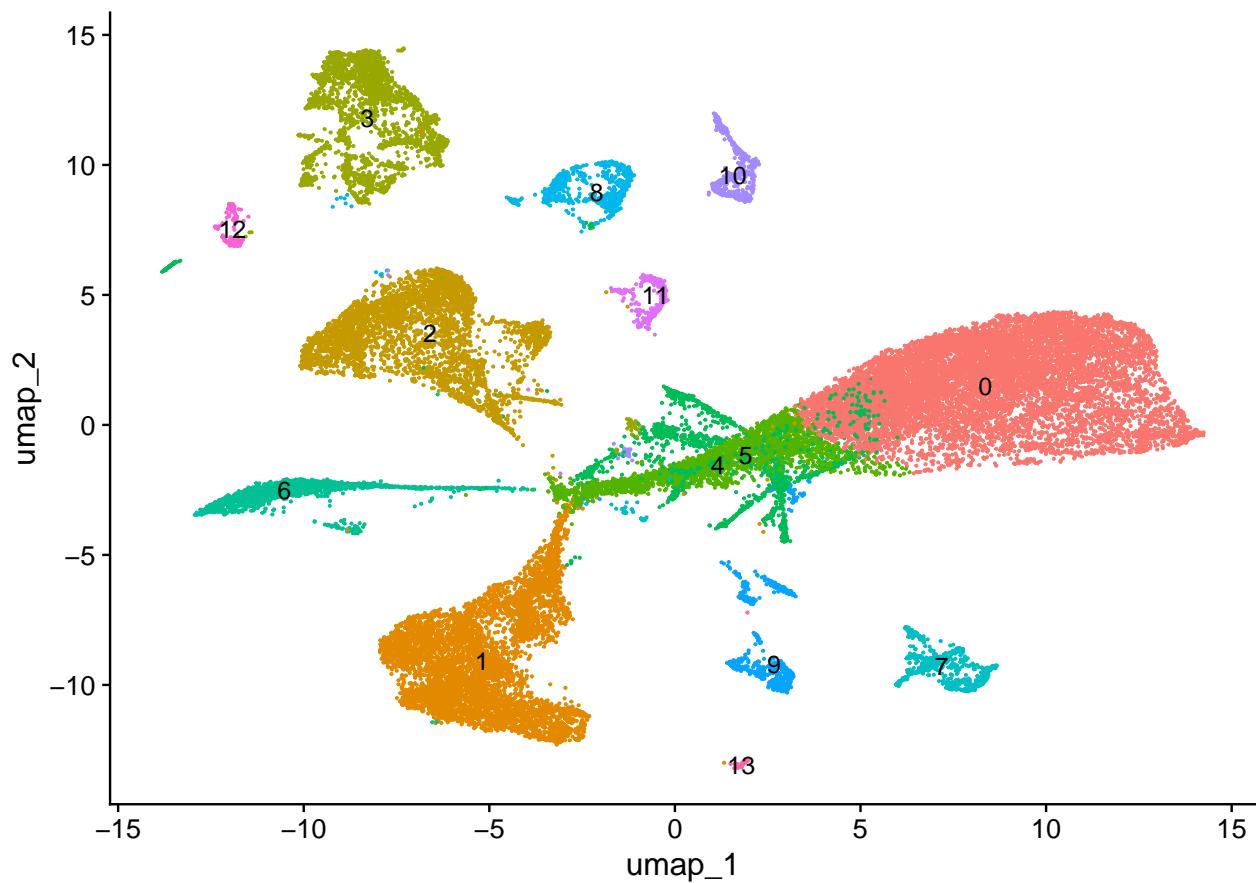
  ggsave(paste0("result/02-Clustering-v2/umap_plot/umap_harmony_res_", res, ".pdf"), plot = p, width = 8, height = 6)
  ggsave(paste0("result/02-Clustering-v2/umap_plot/umap_harmony_res_", res, ".png"), plot = p, width = 8, height = 6)

  dir.create(sprintf("result/02-Clustering-v2/umap_plot/res_%s", res), recursive = TRUE, showWarnings = FALSE)
  # for(index in 1:length(gene_list_clean )){ 
  #   gene <- gene_list_clean[index]
  #   file_path <- sprintf("result/02-Clustering-v2/umap_plot/res_%s/%02d_%s", res, index, gene)
  #   p <- plot_sc_feature(seurat_obj, res = res, features = gene, save = TRUE,
  #                         group_by = "seurat_clusters",
  #                         group_by_label = sprintf("seurat_clusters_%s", res),
  #                         reduction_label = "umap",
  #                         output_prefix = file_path)
  # }

}

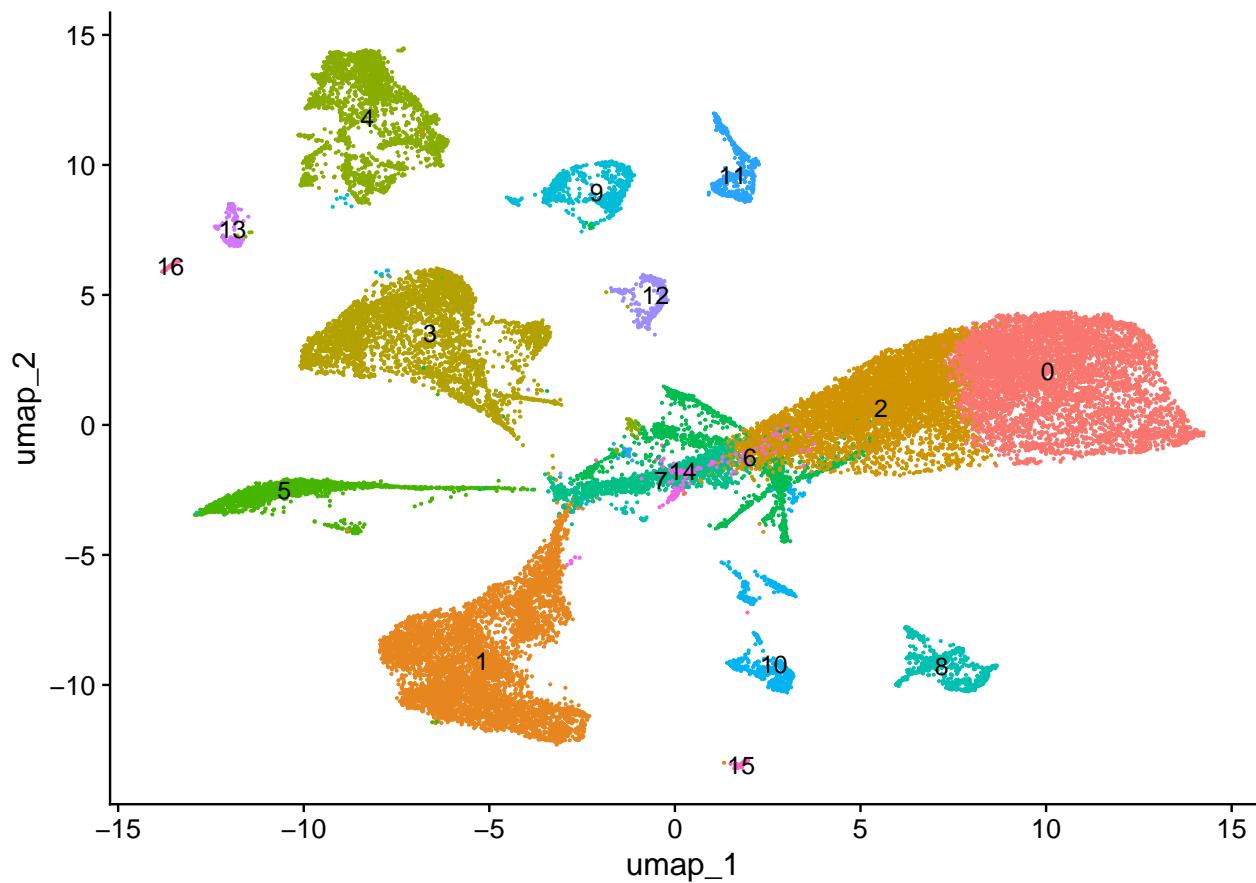
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 31403
## Number of edges: 1378633
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9758
## Number of communities: 14
## Elapsed time: 4 seconds
```

**Resolution: 0.1**



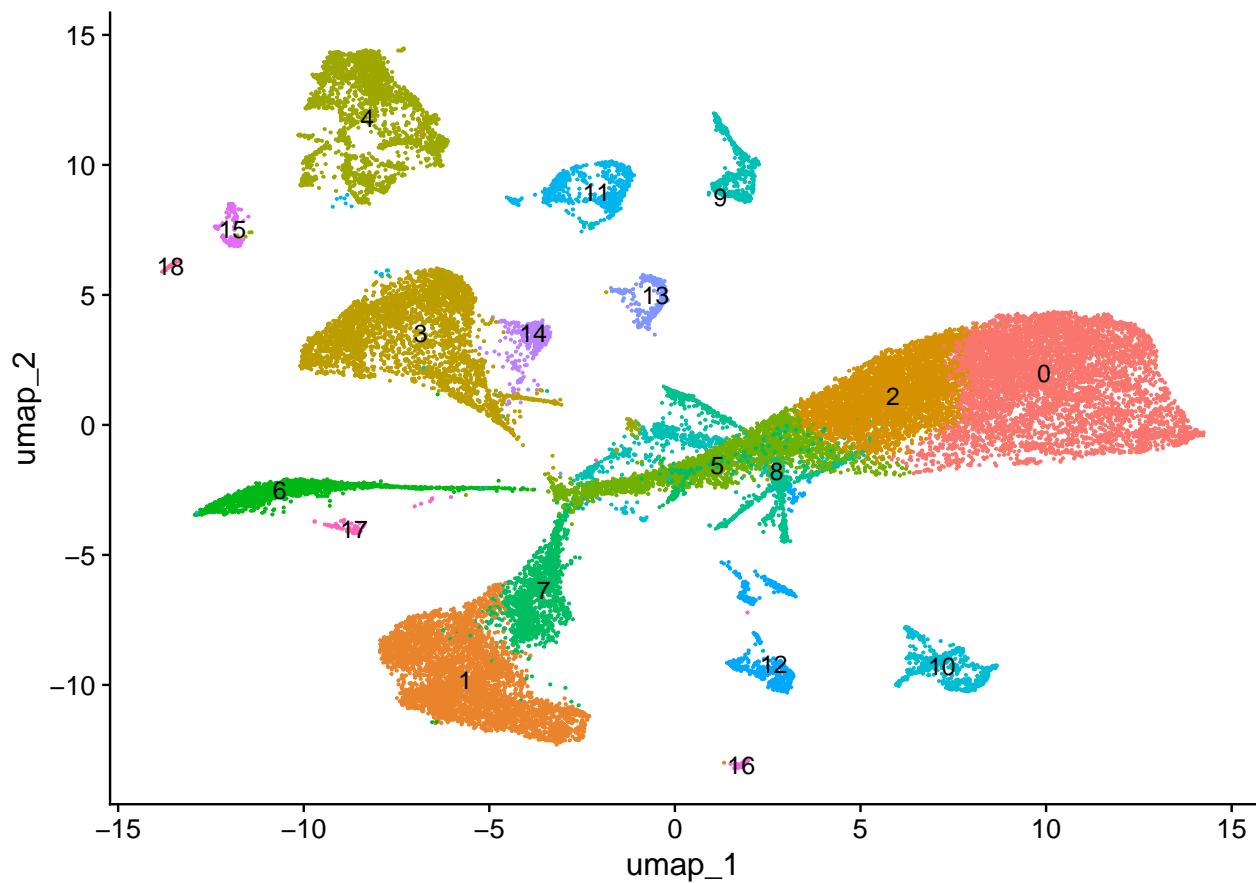
```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 31403
## Number of edges: 1378633
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9614
## Number of communities: 17
## Elapsed time: 4 seconds
```

**Resolution: 0.2**



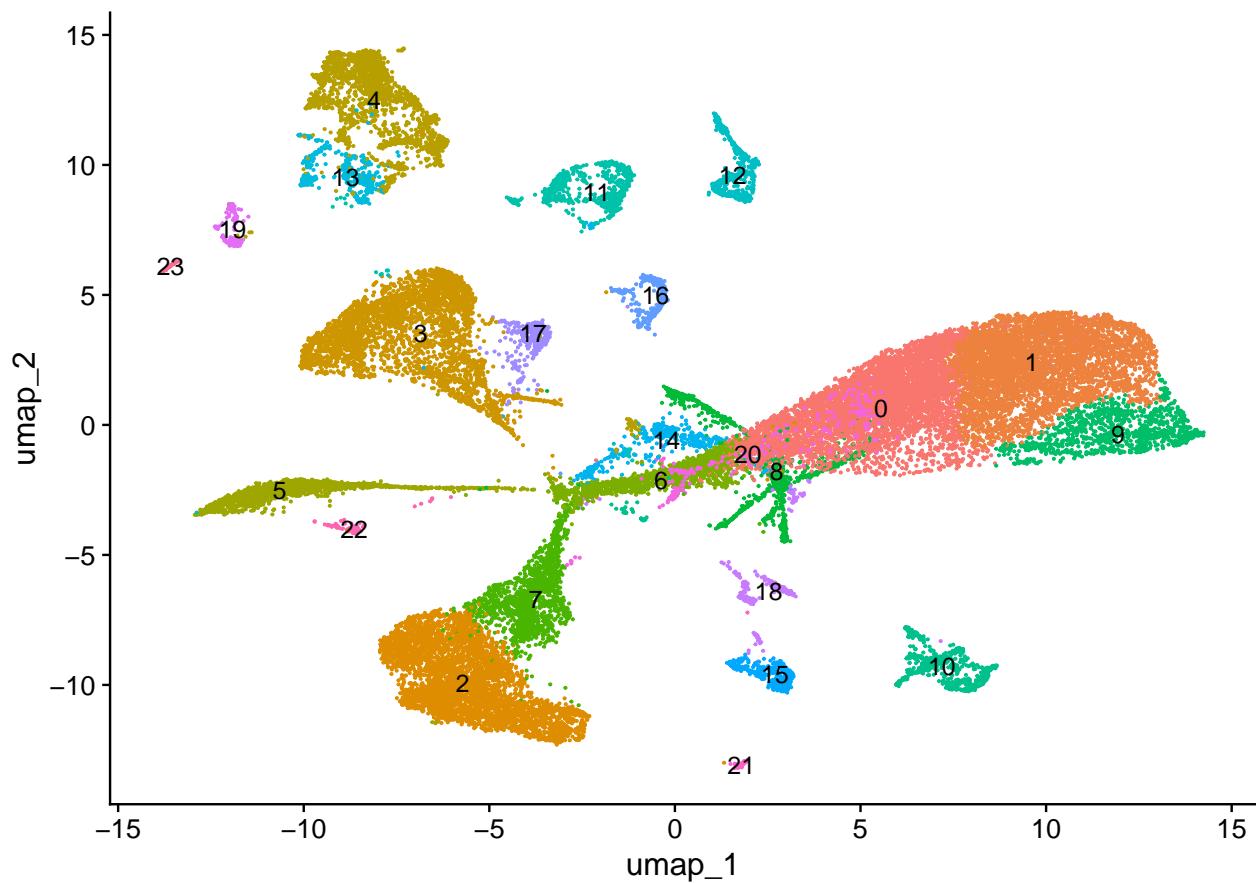
```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 31403
## Number of edges: 1378633
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9502
## Number of communities: 19
## Elapsed time: 4 seconds
```

**Resolution: 0.3**



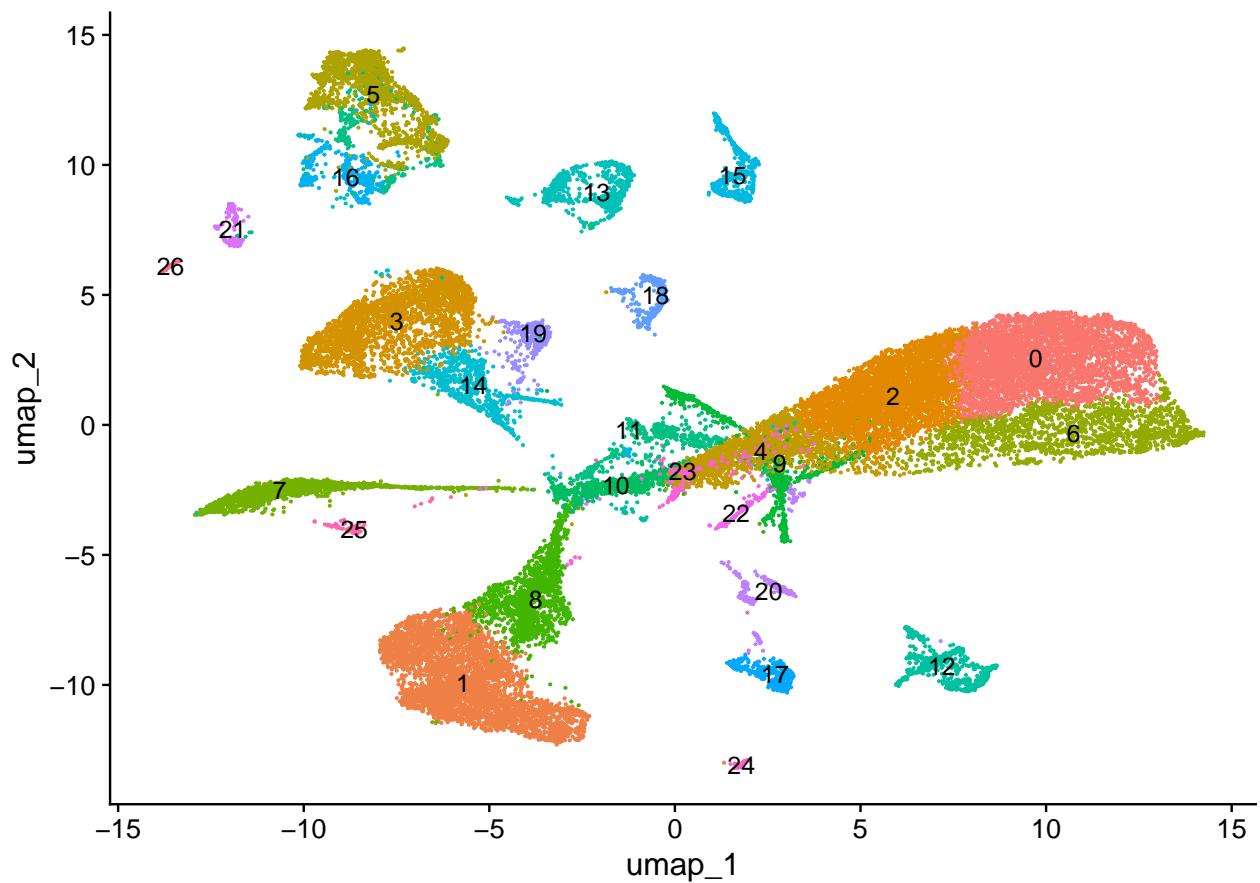
```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 31403
## Number of edges: 1378633
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9412
## Number of communities: 24
## Elapsed time: 4 seconds
```

**Resolution: 0.4**



```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 31403
## Number of edges: 1378633
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9338
## Number of communities: 27
## Elapsed time: 4 seconds
```

**Resolution: 0.5**



```
# save the seurat object
saveRDS(seurat_obj, file = here::here("data", "processed-data", "seurat-cluster_2025-05-01.rds"))
```

## Session Information

```
sessionInfo()

## R version 4.4.0 (2024-04-24)
## Platform: aarch64-apple-darwin20
## Running under: macOS 15.4
##
## Matrix products: default
## BLAS:    /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
## LAPACK:  /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRlapack.dylib;  LAPACK v
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/New_York
## tzcode source: internal
##
## attached base packages:
## [1] stats4      stats       graphics   grDevices  utils      datasets   methods
## [8] base
##
## other attached packages:
##  [1] future_1.40.0           patchwork_1.3.0
##  [3] pheatmap_1.0.12         data.table_1.17.0
##  [5] harmony_1.2.3          Rcpp_1.0.14
##  [7] conflicted_1.2.0        openxlsx_4.2.8
##  [9] AnnotationHub_3.12.0   BiocFileCache_2.12.0
## [11] dbplyr_2.5.0            simspec_0.0.0.9000
## [13] cowplot_1.1.3           EnsDb.Hsapiens.v86_2.99.0
## [15] ensemblDb_2.28.1        AnnotationFilter_1.28.0
## [17] GenomicFeatures_1.56.0  AnnotationDbi_1.66.0
## [19] Biobase_2.64.0           Signac_1.14.0
## [21] Seurat_5.2.1            SeuratObject_5.0.2
## [23] sp_2.2-0                rtracklayer_1.64.0
## [25] GenomicRanges_1.56.2   GenomeInfoDb_1.40.1
## [27] IRanges_2.38.1          S4Vectors_0.42.1
## [29] BiocGenerics_0.50.0   knitr_1.50
## [31] lubridate_1.9.4          forcats_1.0.0
## [33] stringr_1.5.1           dplyr_1.1.4
## [35] purrrr_1.0.4             readr_2.1.5
## [37] tidyR_1.3.1              tibble_3.2.1
## [39] ggplot2_3.5.2            tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] RcppAnnoy_0.0.22        splines_4.4.0
##  [3] later_1.4.2             BiocIO_1.14.0
##  [5] filelock_1.0.3           bitops_1.0-9
##  [7] polyclip_1.10-7          XML_3.99-0.18
##  [9] fastDummies_1.7.5        lifecycle_1.0.4
## [11] rprojroot_2.0.4           globals_0.17.0
## [13] lattice_0.22-7           MASS_7.3-65
## [15] magrittr_2.0.3            plotly_4.10.4
## [17] rmarkdown_2.29             yaml_2.3.10
## [19] httpuv_1.6.16            sctransform_0.4.1
```

```

## [21] zip_2.3.2
## [23] spatstat.sparse_3.1-0
## [25] pbapply_1.7-2
## [27] RColorBrewer_1.1-3
## [29] zlibbioc_1.50.0
## [31] RCurl_1.98-1.17
## [33] GenomeInfoDbData_1.2.12
## [35] irlba_2.3.5.1
## [37] spatstat.utils_3.1-3
## [39] RSpectra_0.16-2
## [41] fitdistrplus_1.2-2
## [43] codetools_0.2-20
## [45] RcppRoll_0.3.1
## [47] UCSC.utils_1.0.0
## [49] matrixStats_1.5.0
## [51] GenomicAlignments_1.40.0
## [53] progressr_0.15.1
## [55] survival_3.8-3
## [57] tools_4.4.0
## [59] ica_1.0-3
## [61] gridExtra_2.3
## [63] here_1.0.1
## [65] MatrixGenerics_1.16.0
## [67] BiocManager_1.30.25
## [69] digest_0.6.37
## [71] R6_2.6.1
## [73] textshaping_1.0.0
## [75] scattermore_1.2
## [77] spatstat.data_3.1-6
## [79] generics_0.1.3
## [81] htmlwidgets_1.6.4
## [83] uwot_0.2.3
## [85] gtable_0.3.6
## [87] lmtest_0.9-40
## [89] htmltools_0.5.8.1
## [91] ProtGenerics_1.36.0
## [93] png_0.1-8
## [95] rstudioapi_0.17.1
## [97] reshape2_1.4.4
## [99] nlme_3.1-168
## [101] cachem_1.1.0
## [103] BiocVersion_3.19.1
## [105] parallel_4.4.0
## [107] restfulr_0.0.15
## [109] grid_4.4.0
## [111] RANN_2.6.2
## [113] xtable_1.8-4
## [115] evaluate_1.0.3
## [117] cli_3.6.4
## [119] Rsamtools_2.20.0
## [121] crayon_1.5.3
## [123] labeling_0.4.3
## [125] stringi_1.8.7
## [127] deldir_2.0-4
spam_2.11-1
reticulate_1.42.0
DBI_1.2.3
abind_1.4-8
Rtsne_0.17
rappdirs_0.3.3
ggrepel_0.9.6
listenv_0.9.1
goftest_1.2-3
spatstat.random_3.3-3
parallelly_1.43.0
DelayedArray_0.30.1
tidyselect_1.2.1
farver_2.1.2
spatstat.explore_3.4-2
jsonlite_2.0.0
ggridges_0.5.6
systemfonts_1.2.2
ragg_1.4.0
glue_1.8.0
SparseArray_1.4.8
xfun_0.52
withr_3.0.2
fastmap_1.2.0
timechange_0.3.0
mime_0.13
colorspace_2.1-1
tensor_1.5
RSQLite_2.3.9
httr_1.4.7
S4Arrays_1.4.1
pkgconfig_2.0.3
blob_1.2.4
XVector_0.44.0
dotCall164_1.2
scales_1.3.0
spatstat.univar_3.1-2
tzdb_0.5.0
rjson_0.2.23
curl_6.2.2
zoo_1.8-14
KernSmooth_2.23-26
miniUI_0.1.2
pillar_1.10.2
vctrs_0.6.5
promises_1.3.2
cluster_2.1.8.1
tinytex_0.57
compiler_4.4.0
rlang_1.1.6
future.apply_1.11.3
plyr_1.8.9
viridisLite_0.4.2
BiocParallel_1.38.0

```

```
## [129] munsell_0.5.1           Biostrings_2.72.1
## [131] lazyeval_0.2.2            spatstat.geom_3.3-6
## [133] Matrix_1.7-3              RcppHNSW_0.6.0
## [135] hms_1.1.3                 bit64_4.6.0-1
## [137] KEGGREST_1.44.1           shiny_1.10.0
## [139] SummarizedExperiment_1.34.0 ROCR_1.0-11
## [141] memoise_2.0.1              igraph_2.1.4
## [143] fastmatch_1.1-6             bit_4.6.0
```