

# HistoMapST

Roman Laddach

14-Aug-2025

## Setup

### Libraries

```
library("ggplot2") # visualizations
library("dplyr") # data manipulation

## 
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

library("reshape2") # data manipulation
library("rdist") # matrix-matrix distance calculation

## Warning: package 'rdist' was built under R version 4.5.1
```

Optional

```
library("Seurat") # optional - spot coordinates can be retrieved from the Seurat object

## Loading required package: SeuratObject

## Loading required package: sp

## 
## Attaching package: 'SeuratObject'

## The following objects are masked from 'package:base':
## 
##     intersect, t
```

## Functions for the pipeline

```
source("HistoMapST_functions.R")
```

## Input

Output from QuPath, which should contain at least x/y coordinates of cell centroids. Additional metadata can be used for visualizations.

```
QuPath_full <- read_output_from_QuPath(path_to_file = "../HistoMapST_input/",
                                         file_name = "QuPath_output.csv")
QuPath_full[1:6, 1:10]

##           X          Object.ID      Name      Class
## hist_1 1 57ede494-14d7-43e6-9bb9-3cee62ed36a1 Immune cells Immune cells
## hist_2 2 97e29bc6-caac-4b5d-8a88-ca2c12187423 Immune cells Immune cells
## hist_3 3 7a2120cc-e704-4725-b7d0-4c2e109a6caa Immune cells Immune cells
## hist_4 4 b79289e6-57e8-4194-94c7-665a9ed154d7 Immune cells Immune cells
## hist_5 5 c3e7a60f-2c47-4a31-98c4-dd2cf8127705 Immune cells Immune cells
## hist_6 6 09fe661e-b565-420b-b4ad-1c5f68e326f6 Immune cells Immune cells
##           Parent      ROI Centroid.X.px Centroid.Y.px Nucleus..Area
## hist_1 Whole Slide Polygon     2882.8     4568.8        40
## hist_2 Whole Slide Polygon     2870.6     4574.4        74
## hist_3 Whole Slide Polygon     2889.2     4578.8        50
## hist_4 Whole Slide Polygon     2859.2     4577.6        22
## hist_5 Whole Slide Polygon     2831.5     4579.7        15
## hist_6 Whole Slide Polygon     2845.0     4578.2        23
##           Nucleus..Perimeter
## hist_1            23.8977
## hist_2            38.0077
## hist_3            36.6109
## hist_4            19.3400
## hist_5            16.2674
## hist_6            20.5386
```

Spot coordinates from the spaceranger output contained within the *tissue\_positions\_list.csv* file. The function requires the file location. After reading it extracts the px coordinates and converts the y coordinates to match the Cartesian coordinate system.

```
SR_coordinates <- read_coordinates_from_SR(path_to_file = "../HistoMapST_input/")
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## header and 'col.names' are of different lengths

head(SR_coordinates)
```

```
##           x_coord y_coord
## AAACCGTTCGTCCAGG-1    3332   -6336
## AACCGGTTGCGAACTG-1    4358   -7950
```

```

## AAACTTGCAAACGTAT-1    1928   -5573
## AAAGGCCCTATAATAC-1   2088   -7818
## AAAGGCTACGGACCAT-1   4057   -7412
## AAAGGCTCTCGGCCG-1    4126   -6665

```

## Optional input

The spot coordinates can also be retrieved using Seurat object in case the tissue\_positions\_list.csv is not available. In the example below the Seurat object is loaded from .RDS and subsequently the coordinates are retrieved and formatted.

```

# loading from RDS
ST_object <- readRDS("../output/new_slices/L257B_A1/L257B_A1_Seurat_object_post_QC.RDS")
# retrieving coordinates from Seurat
coordinates_from_Seurat <- ST_object@images$slice1@coordinates
coordinates_from_Seurat <- coordinates_from_Seurat[,c("imagecol", "imagerow")]
coordinates_from_Seurat$imagerow <- -coordinates_from_Seurat$imagerow
colnames(coordinates_from_Seurat) <- c("x_coord", "y_coord")
head(coordinates_from_Seurat)

##           x_coord y_coord
## AAACGGGTAGGTACC-1    2484   -5259
## AAACCGTTCGTCCAGG-1   3332   -6336
## AAACGGTTGCGAACTG-1   4358   -7950
## AAACTTGCAAACGTAT-1   1928   -5573
## AAAGGCCCTATAATAC-1   2088   -7818
## AAAGGCTACGGACCAT-1   4057   -7412

```

## Analysis

The coordinate system between QuPath and spaceranger output should match as well as the orientation of the image. This can be easily compared when tissue is plotted.

First the QuPath output is prepared and relevant columns containing x/y coordinates are selected.

```

QuPath_coordinates <- select_coordinates_from_QuPath(object_name = QuPath_full,
                                                       x_coord = "Centroid.X.px",
                                                       y_coord = "Centroid.Y.px")
head(QuPath_coordinates)

##           x_coord y_coord
## hist_1    2882.8 -4568.8
## hist_2    2870.6 -4574.4
## hist_3    2889.2 -4578.8
## hist_4    2859.2 -4577.6
## hist_5    2831.5 -4579.7
## hist_6    2845.0 -4578.2

```

## Visualizations

Comparison between tissue based on cell centroids from QuPath and spot coordinates from spaceranger.

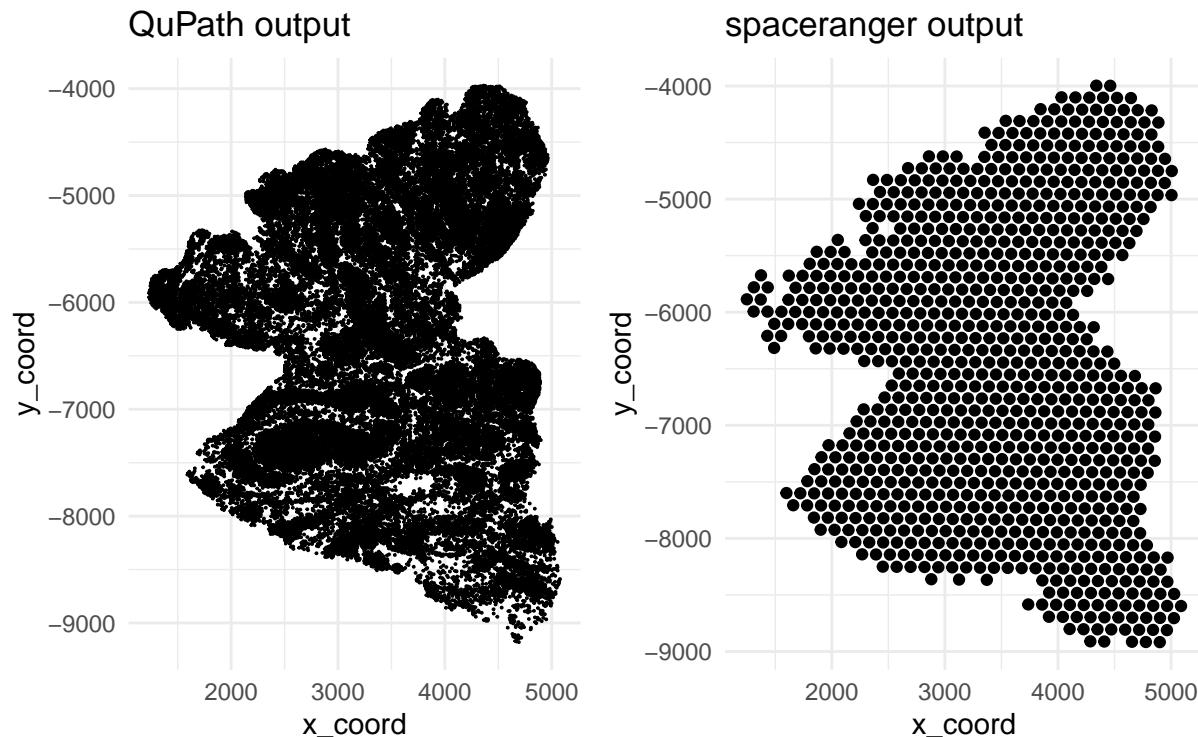
```

# QuPath output
p1 <- ggplot(QuPath_coordinates, aes(x=x_coord, y=y_coord)) +
  geom_point(size=0.01) +
  coord_fixed() +
  theme_minimal() +
  ggtitle("QuPath output")

# Spaceranger output
p2 <- ggplot(SR_coordinates, aes(x=x_coord, y=y_coord)) +
  geom_point() +
  coord_fixed() +
  theme_minimal() +
  ggtitle("spaceranger output")

p1 + p2

```



### Optional input visualization

```

# QuPath output
p1 <- ggplot(QuPath_coordinates, aes(x=x_coord, y=y_coord)) +
  geom_point(size=0.01) +
  coord_fixed() +
  theme_minimal() +

```

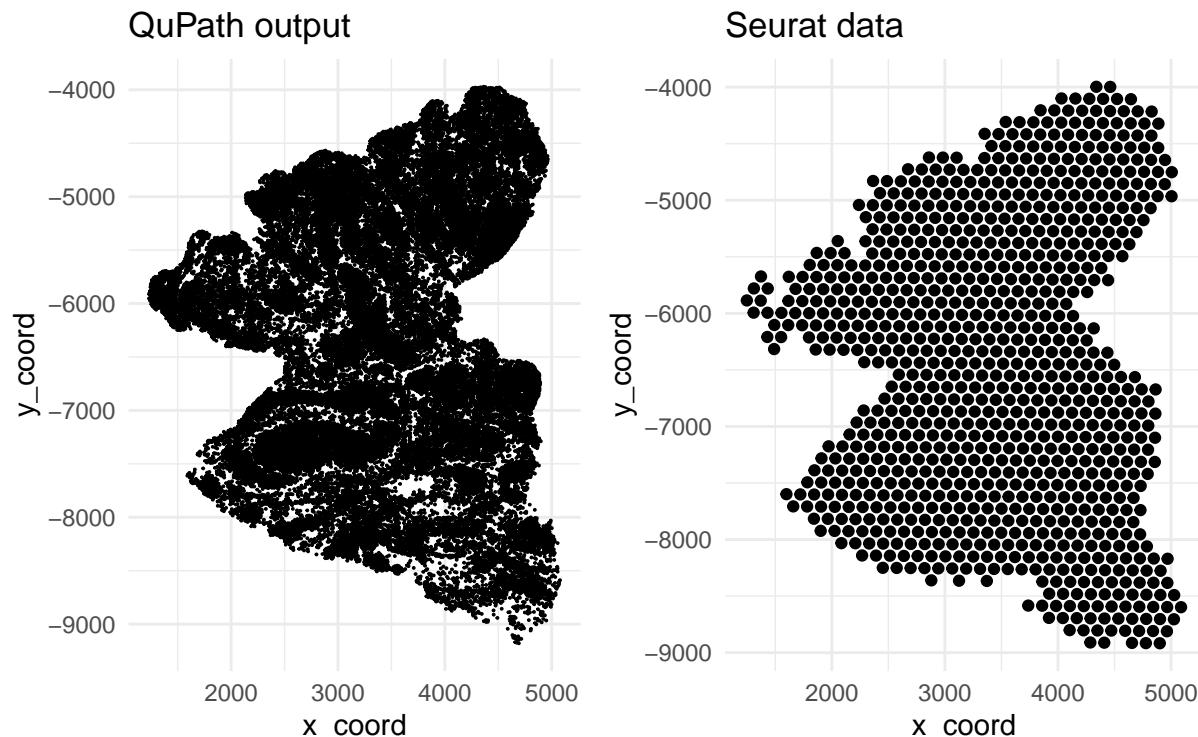
```

ggtitle("QuPath output")

# Seurat data
p3 <- ggplot(coordinates_from_Seurat, aes(x=x_coord, y=y_coord)) +
  geom_point() +
  coord_fixed() +
  theme_minimal() +
  ggtitle("Seurat data")

p1 + p3

```



## Cells and spot coordinates colocalization

The function below takes as an input formatted data from QuPath and spaceranger. It calculates the distance between two matrices (QuPath and spaceranger) using `cdist()` function. Additionally it calculates the spot radius in pixels by converting the distance between spots from um to pixels, based on the assumption that spot centroids are 100um apart (source - 10x Genomics). The shortest non-zero distance is calculated and converted to pixels per um, which allows to convert the 55um radius. Finally, cells which are outside of the calculated distance are filtered out and a dataframe is returned containing only cells within spots.

```

cells_within_spots <- identify_cells_within_spots(SR_coordinates_df = SR_coordinates, QuPath_coordinates
## Using spot_ID as id variables

```

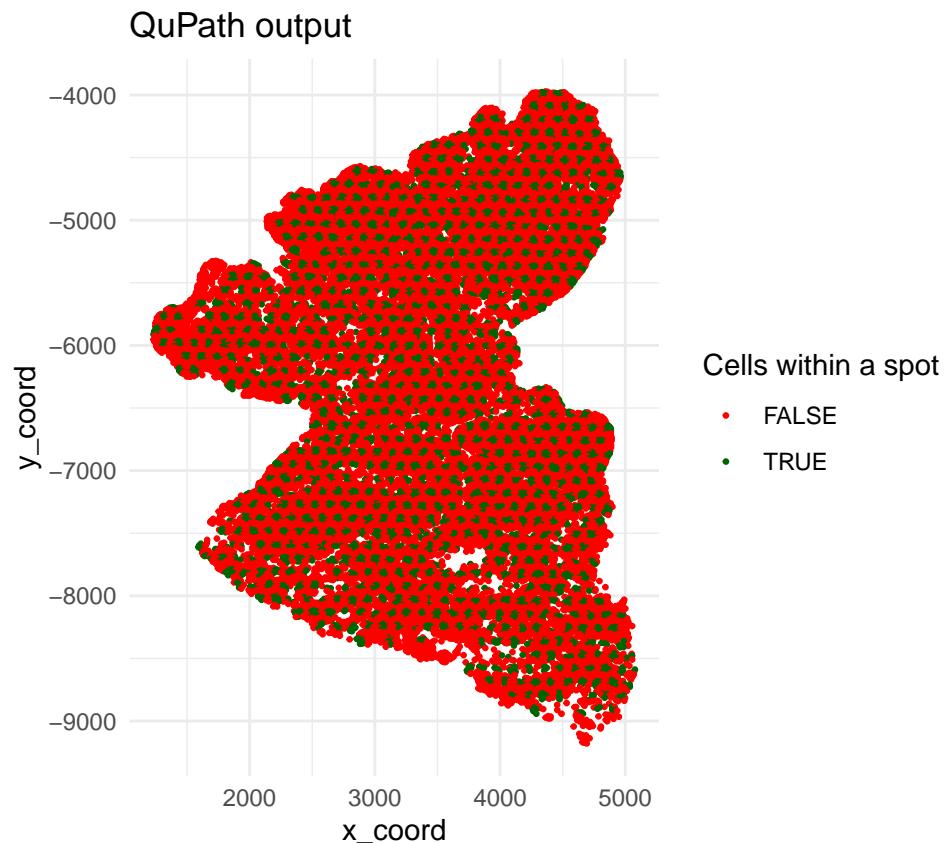
```
head(cells_within_spots)
```

```
##           spot_ID cell_ID distance
## 10748 TTGCCCTGATCACGGG-1 hist_12 32.48707
## 13844 CTAAAGAATGCCTACT-1 hist_16 32.35954
## 16130 TTGCCCTGATCACGGG-1 hist_18 28.75448
## 16535 CTAAAGAATGCCTACT-1 hist_19 23.09827
## 22814 CTAAAGAATGCCTACT-1 hist_26 24.84854
## 26894 TTGCCCTGATCACGGG-1 hist_30 21.24735
```

## Use cases

The initial data from QuPath can be coloured by cells membership to spots from Visium.

```
ggplot(QuPath_coordinates, aes(x=x_coord, y=y_coord, colour=rownames(QuPath_coordinates) %in% cells_within_spots)) +
  geom_point(size=0.5) +
  theme_minimal() +
  scale_colour_manual(values=c("red", "darkgreen")) +
  coord_fixed() +
  labs(color="Cells within a spot") +
  ggtitle("QuPath output")
```



Zooming closer to a specific set of coordinates

```

ggplot(QuPath_coordinates, aes(x=x_coord, y=y_coord, colour=rownames(QuPath_coordinates) %in% cells_within_spots)) +
  geom_point(size=0.75) + theme_minimal() +
  coord_fixed() +
  labs(color="Cells within a spot") +
  scale_colour_manual(values=c("red", "darkgreen")) +
  xlim(c(1500,3000)) +
  ylim(c(-6500, -8000))

## Warning: Removed 27540 rows containing missing values or values outside the scale range
## ('geom_point()').

```



The number of cells per spot can be calculated helping with deconvolution and understanding how densely packed some of the tissue areas are.

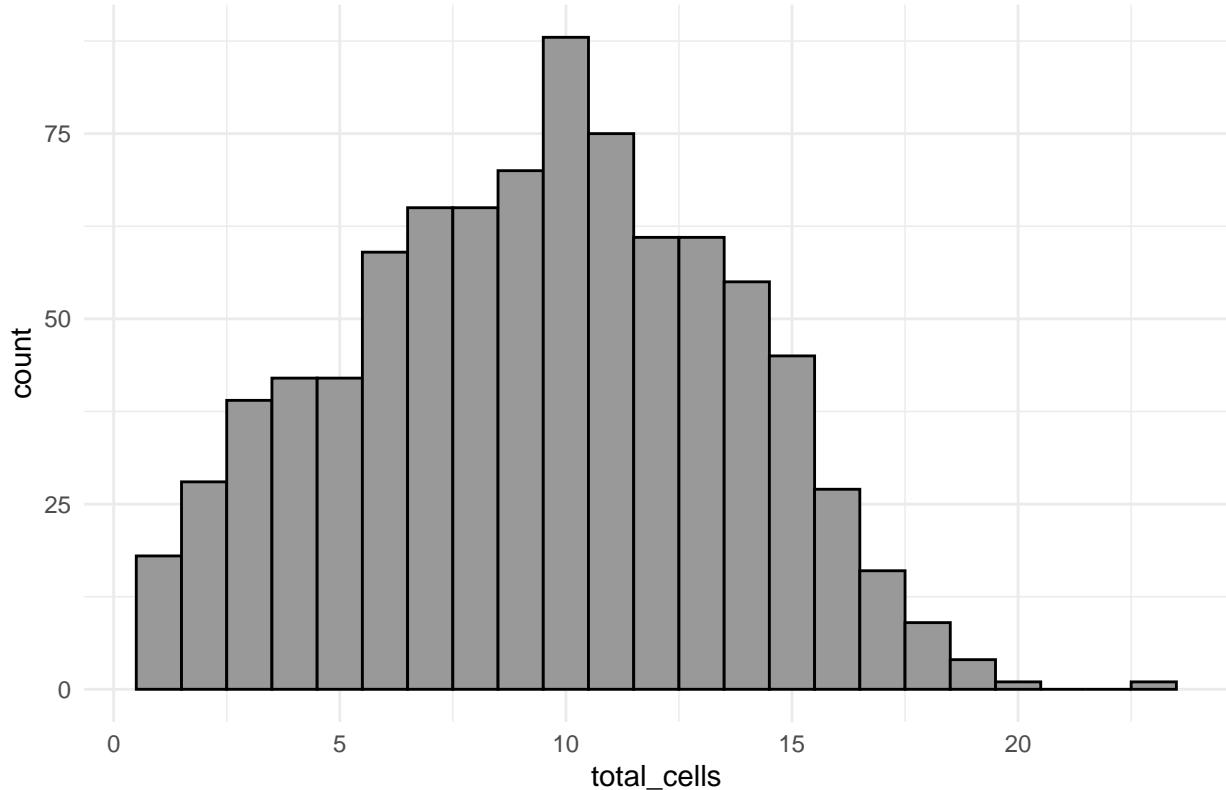
```

cells_per_spot <- as.data.frame(table(cells_within_spots$spot_ID))
colnames(cells_per_spot) <- c("spot_ID", "total_cells")

ggplot(cells_per_spot, aes(x=total_cells)) +
  geom_histogram(color="black", fill="grey60", binwidth=1) +
  theme_minimal() +
  ggtitle("Histogram of cells per spot")

```

## Histogram of cells per spot



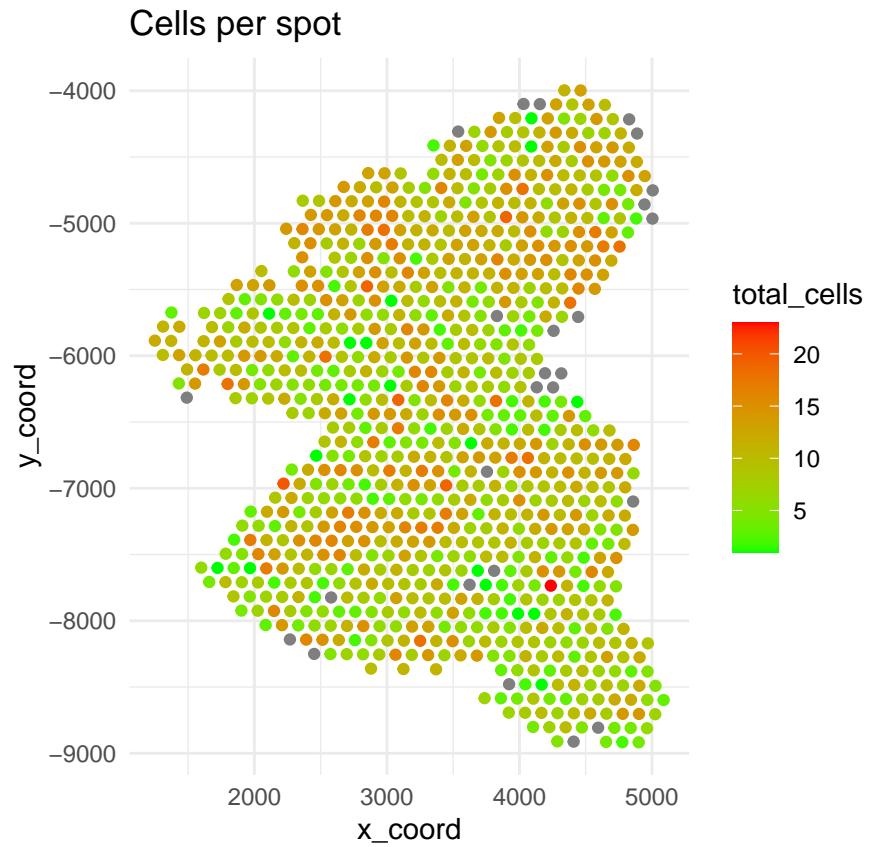
This can be projected back to the spaceranger output. In some cases there are no cells found within a spot based on the image, and this is represented as 'NA' values.

```
SR_coordinates$spot_ID = rownames(SR_coordinates)
cells_per_spot_with_coordinates <- left_join(SR_coordinates, cells_per_spot, by = "spot_ID")
# cells_per_spot_with_coordinates$total_cells[is.na(cells_per_spot_with_coordinates$total_cells)] = 0
head(cells_per_spot_with_coordinates)
```

	x_coord	y_coord	spot_ID	total_cells
## 1	3332	-6336	AAACCGTTCGTCCAGG-1	10
## 2	4358	-7950	AAACGGTTGCGAACTG-1	11
## 3	1928	-5573	AAACTTGCAAACGTAT-1	3
## 4	2088	-7818	AAAGGCCCTATAATAC-1	7
## 5	4057	-7412	AAAGGCTACGGACCAT-1	8
## 6	4126	-6665	AAAGGCTCTCGCGCCG-1	14

Visualization of the number of cells per spot.

```
ggplot(cells_per_spot_with_coordinates, aes(x=x_coord, y=y_coord, color=total_cells)) +
  geom_point() +
  coord_fixed() +
  theme_minimal() +
  scale_colour_gradient(low = "green",
                        high = "red",
                        na.value = "grey50") +
  ggtitle("Cells per spot")
```



## Session info

```
sessionInfo()

## R version 4.5.0 (2025-04-11 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## Matrix products: default
## LAPACK version 3.12.1
##
## locale:
## [1] LC_COLLATE=English_United Kingdom.utf8
## [2] LC_CTYPE=English_United Kingdom.utf8
## [3] LC_MONETARY=English_United Kingdom.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United Kingdom.utf8
##
## time zone: Europe/London
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics   grDevices  utils      datasets   methods    base
```

```

##
## other attached packages:
## [1] Seurat_5.3.0      SeuratObject_5.1.0  sp_2.2-0          rdist_0.0.5
## [5] reshape2_1.4.4    dplyr_1.1.4     ggplot2_3.5.2
##
## loaded via a namespace (and not attached):
##   [1] deldir_2.0-4        pbapply_1.7-2       gridExtra_2.3
##   [4] rlang_1.1.6         magrittr_2.0.3      RcppAnnoy_0.0.22
##   [7] spatstat.geom_3.3-6 matrixStats_1.5.0   ggridges_0.5.6
##  [10] compiler_4.5.0      png_0.1-8          vctrs_0.6.5
##  [13] stringr_1.5.1      pkgconfig_2.0.3    fastmap_1.2.0
##  [16] labeling_0.4.3     promises_1.3.2    rmarkdown_2.29
##  [19] purrrr_1.0.4       xfun_0.52         jsonlite_2.0.0
##  [22] goftest_1.2-3     later_1.4.2       spatstat.utils_3.1-3
##  [25] irlba_2.3.5.1    parallel_4.5.0   cluster_2.1.8.1
##  [28] R6_2.6.1          ica_1.0-3         stringi_1.8.7
##  [31] RColorBrewer_1.1-3 spatstat.data_3.1-6 reticulate_1.42.0
##  [34] parallelly_1.43.0  spatstat.univar_3.1-2 lmtest_0.9-40
##  [37] scattermore_1.2    Rcpp_1.0.14        knitr_1.50
##  [40] tensor_1.5         future.apply_1.11.3 zoo_1.8-14
##  [43] sctransform_0.4.2 httpuv_1.6.16      Matrix_1.7-3
##  [46] splines_4.5.0      igraph_2.1.4      tidyselect_1.2.1
##  [49] rstudioapi_0.17.1 abind_1.4-8       yaml_2.3.10
##  [52] spatstat.random_3.3-3 codetools_0.2-20 miniUI_0.1.2
##  [55] spatstat.explore_3.4-2 listenv_0.9.1   lattice_0.22-6
##  [58] tibble_3.2.1       plyr_1.8.9        shiny_1.10.0
##  [61] withr_3.0.2       ROCR_1.0-11       evaluate_1.0.3
##  [64] Rtsne_0.17         future_1.40.0    fastDummies_1.7.5
##  [67] survival_3.8-3    polyclip_1.10-7 fitdistrplus_1.2-2
##  [70] pillar_1.10.2     KernSmooth_2.23-26 plotly_4.10.4
##  [73] generics_0.1.3     RcppHNSW_0.6.0    scales_1.4.0
##  [76] globals_0.17.0    xtable_1.8-4     glue_1.8.0
##  [79] lazyeval_0.2.2     tools_4.5.0      data.table_1.17.0
##  [82] RSpectra_0.16-2    RANN_2.6.2       dotCall64_1.2
##  [85] cowplot_1.1.3      grid_4.5.0       tidyverse_1.3.1
##  [88] nlme_3.1-168      patchwork_1.3.0 cli_3.6.5
##  [91] spatstat.sparse_3.1-0 spam_2.11-1    viridisLite_0.4.2
##  [94] uwot_0.2.3        gtable_0.3.6     digest_0.6.37
##  [97] progressr_0.15.1  ggrepel_0.9.6    htmlwidgets_1.6.4
## [100] farver_2.1.2      htmtools_0.5.8.1 lifecycle_1.0.4
## [103] httr_1.4.7        mime_0.13       MASS_7.3-65

```