

scRNA-Seq RTM WT vs KO - Cluster assignment

Domien Vanneste

2024-12-10 15:27:42 +0100

Contents

Introduction	1
Load packages	1
Load Seurat object	2
Data visualization and Cluster assignment	3
Global UMAP	3
Genotype specific UMAP	4
Dotplot RTM subset specific genes	6
Dotplot CM subset specific signature	9
Validation MafB KO	12

Introduction

scRNA-Seq data of RTM from Mafbfl/fl (WT) and Lyz2Cre Mafbfl/fl (KO) mice was projected in global and genotype-specific uniform manifold approximation and projection (UMAP) plots. Based on the expression of hashtag barcodes and known RTM-specific genes, we identified mac populations corresponding to SPMs (Cd226, Retnla) and LPMs (Gata6, Icam2) in the peritoneal lavage; Kupffer cells (KCs) (Clec4f, Cdh5) in the liver; IMs (Mgl2, Ccl2) in the lung; and microglia (MG) (Tmem119, Sall1) in the brain (Figure 3C). In the colon, we identify two clusters of colonic macrophages (CMs) characterized by the expression of Plac8 and Ly6c2, and Cd4 and H2-M2, respectively. By mapping signature scores for Ly6C+ CMs and MHC-II+ CMs using DEGs between Ly6C+ cells and MHC-II+ cells in the monocyte waterfall, we found that Plac8/Ly6c2+ and Cd4/H2-M2+ CMs largely corresponded to Ly6C+ CM and MHC-II+ CM, respectively.

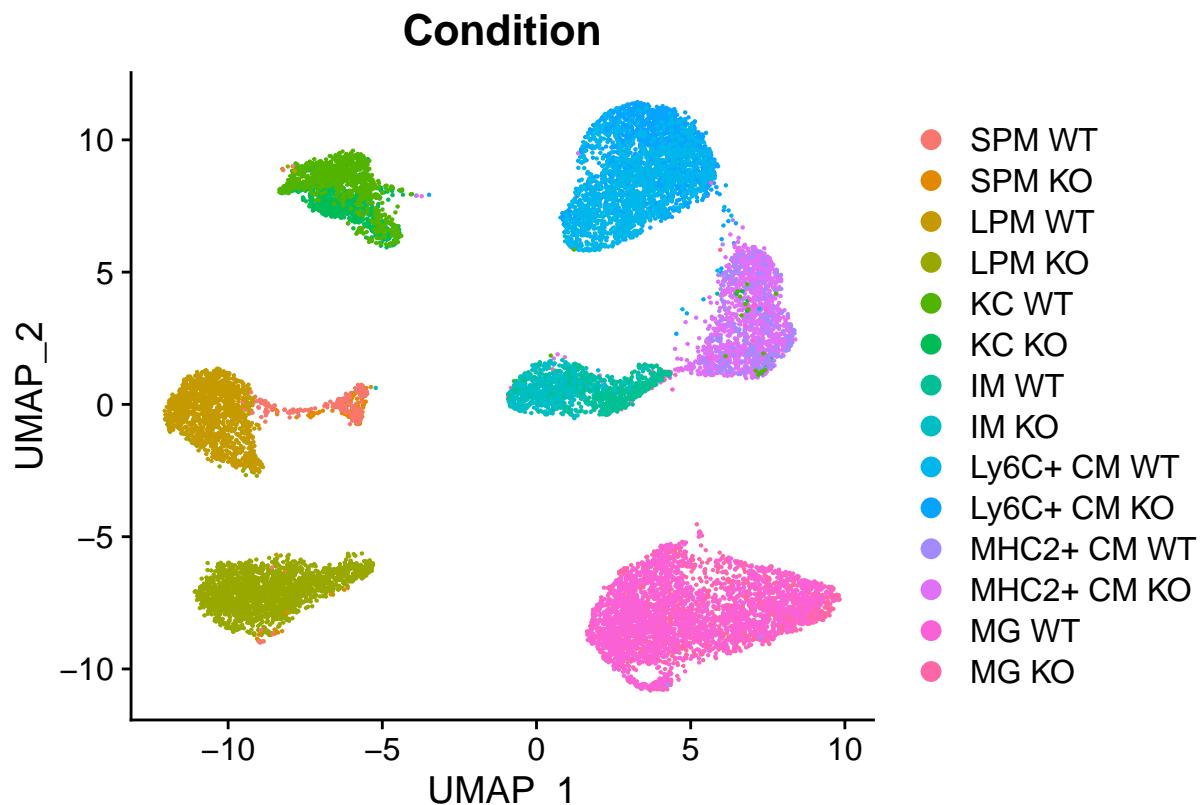
Efficient Mafb depletion was confirmed in all RTM subsets from Lyz2Cre Mafbfl/fl mice

Load packages

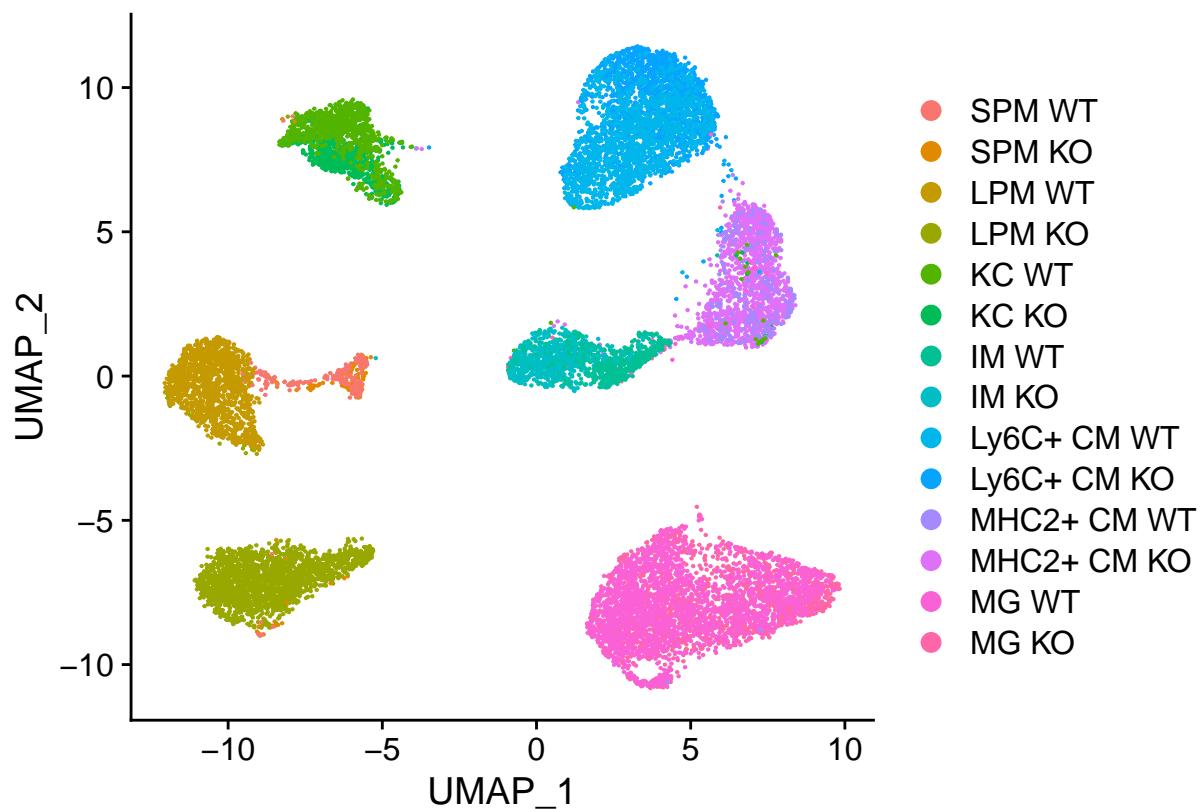
```
suppressMessages({  
  library(Seurat)  
  library(ggplot2)  
  library(RColorBrewer)  
  library(readxl)  
})
```

Load Seurat object

```
sc <- readRDS(file = "Sc_Macro_filter.rds")  
  
DimPlot(sc, group.by = "Condition")
```



```
Idents(sc) <- "Condition"  
  
DimPlot(sc)
```

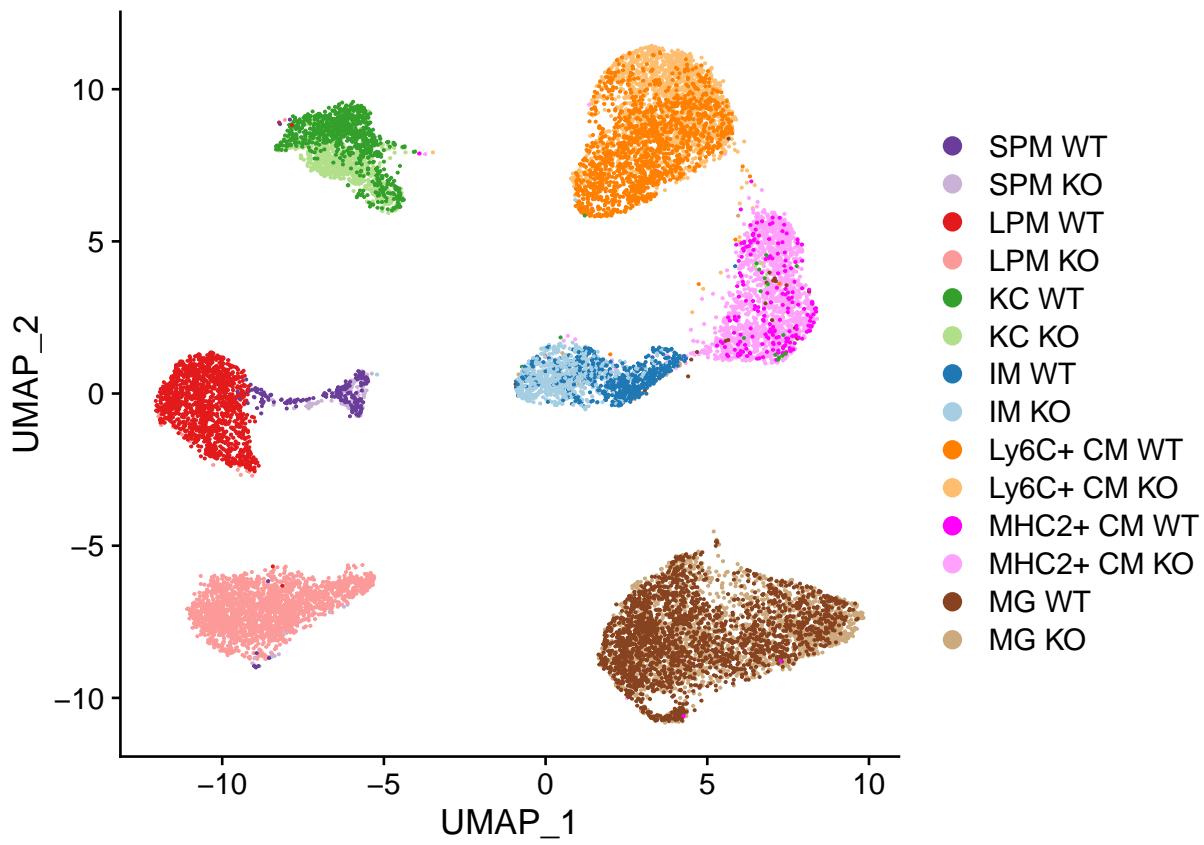


Data visualization and Cluster assignment

Global UMAP

```
pal <- c("#6a3d9a", "#cab2d6", "#e31a1c", "#fb9a99", "#33a02c",
  "#b2df8a", "#1f78b4", "#a6cee3", "#ff7f00", "#fdbf6f", "#ff00fa",
  "#ffa1fd", "#87421f", "#cdaa7d")
```

```
DimPlot(sc, cols = pal)
```



Genotype specific UMAP

Mafbfl/fl (WT)

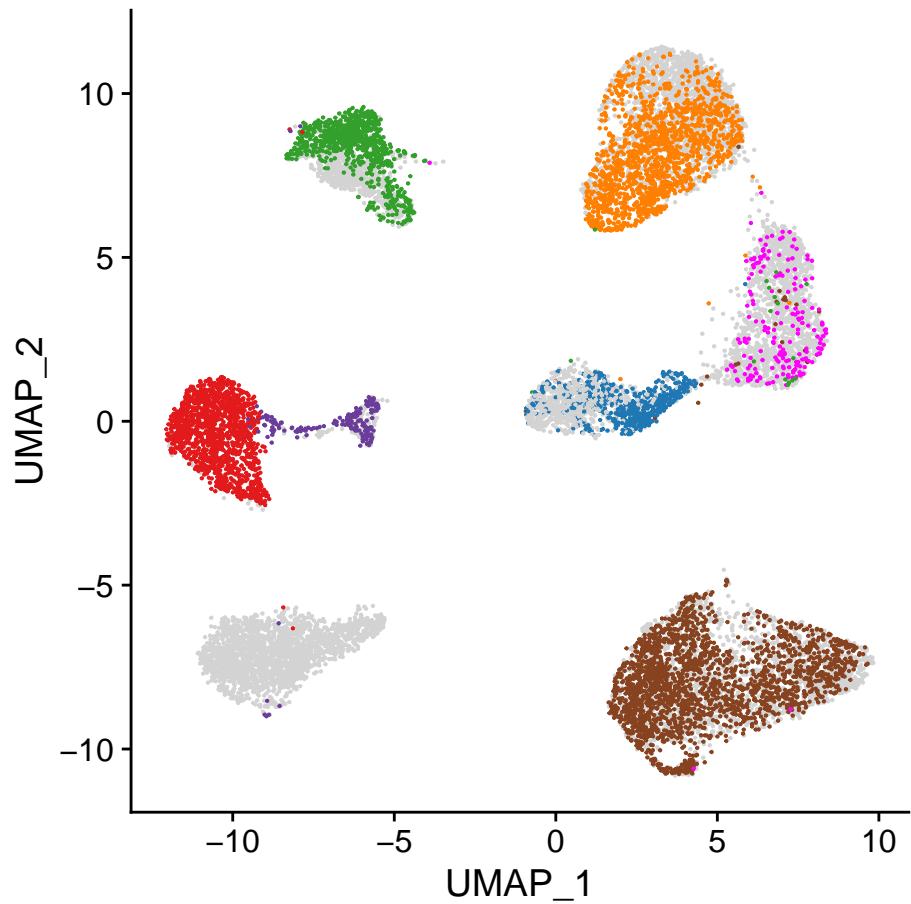
```
sc$cell.type.control <- as.character(sc@active.ident)

sc$cell.type.control[WhichCells(sc, idents = c("SPM KO", "LPM KO",
    "KC KO", "IM KO", "Ly6C+ CM KO", "MHC2+ CM KO", "MG KO"))] <- "MafB-KO"

sc$cell.type.control <- factor(sc$cell.type.control, levels = c("SPM WT",
    "LPM WT", "KC WT", "IM WT", "Ly6C+ CM WT", "MHC2+ CM WT",
    "MG WT", "MafB-KO"))
```

```
pal_wt <- c("#6a3d9a", "#e31a1c", "#33a02c", "#1f78b4", "#ff7f00",
    "#ff00fa", "#87421f", "lightgrey")
```

```
DimPlot(sc, cols = pal_wt, group.by = "cell.type.control") +
    theme(legend.position = "none", plot.title = element_blank())
```



Lyz2Cre Mafbfl/fl (KO)

```

sc$cell.type.mafbk0 <- as.character(sc@active.ident)

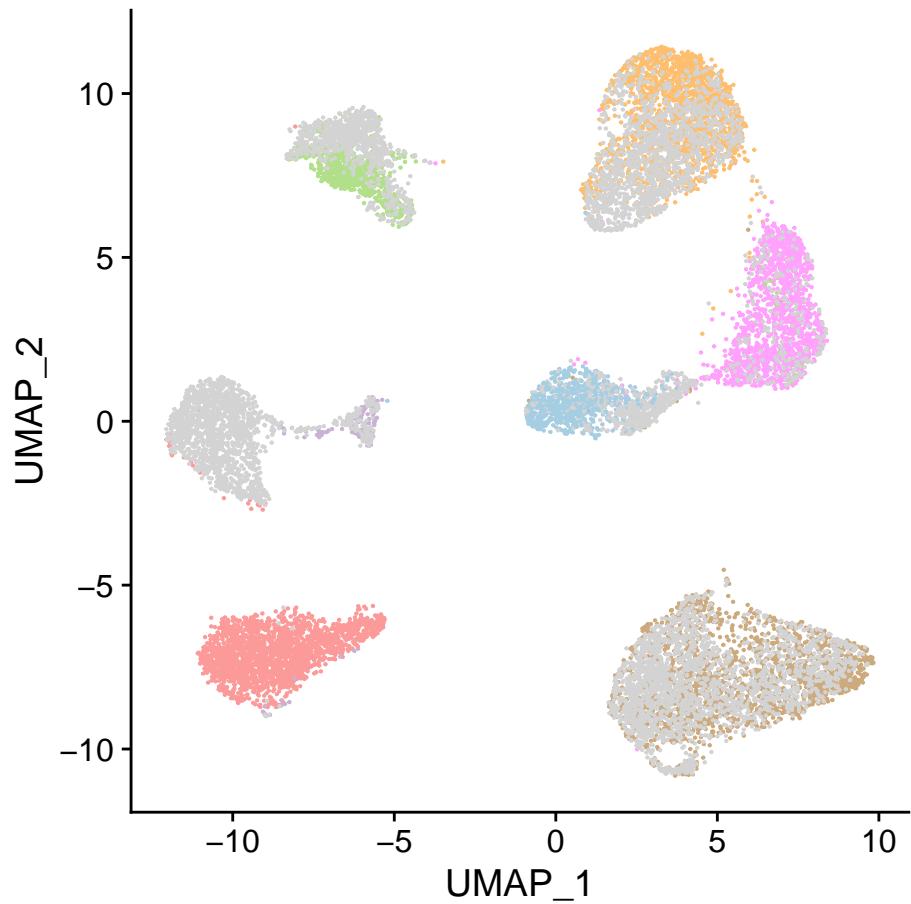
sc$cell.type.mafbk0[WhichCells(sc, idents = c("SPM WT", "LPM WT",
    "KC WT", "IM WT", "Ly6C+ CM WT", "MHC2+ CM WT", "MG WT"))] <- "MafB-WT"

sc$cell.type.mafbk0 <- factor(sc$cell.type.mafbk0, levels = c("SPM KO",
    "LPM KO", "KC KO", "IM KO", "Ly6C+ CM KO", "MHC2+ CM KO",
    "MG KO", "MafB-WT"))

pal_ko <- c("#cab2d6", "#fb9a99", "#b2df8a", "#a6cee3", "#fdbf6f",
    "#ffa1fd", "#cdaa7d", "lightgrey")

DimPlot(sc, cols = pal_ko, group.by = "cell.type.mafbk0") + theme(legend.position = "none",
    plot.title = element_blank())

```



Dotplot RTM subset specific genes

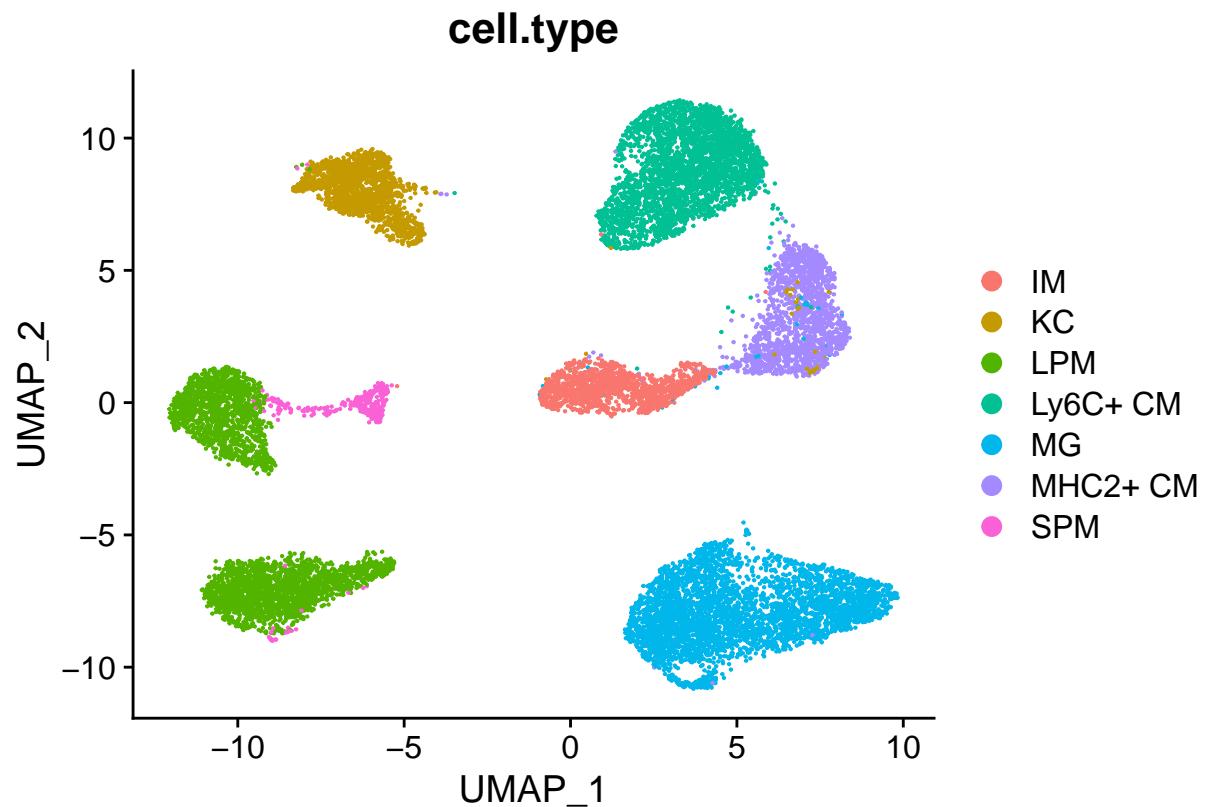
```

sc$cell.type <- as.character(sc@active.ident)

sc$cell.type[WhichCells(sc, idents = c("SPM WT", "SPM KO"))] <- "SPM"
sc$cell.type[WhichCells(sc, idents = c("LPM WT", "LPM KO"))] <- "LPM"
sc$cell.type[WhichCells(sc, idents = c("KC WT", "KC KO"))] <- "KC"
sc$cell.type[WhichCells(sc, idents = c("IM WT", "IM KO"))] <- "IM"
sc$cell.type[WhichCells(sc, idents = c("Ly6C+ CM WT", "Ly6C+ CM KO"))] <- "Ly6C+ CM"
sc$cell.type[WhichCells(sc, idents = c("MHC2+ CM WT", "MHC2+ CM KO"))] <- "MHC2+ CM"
sc$cell.type[WhichCells(sc, idents = c("MG WT", "MG KO"))] <- "MG"

DimPlot(sc, group.by = "cell.type")

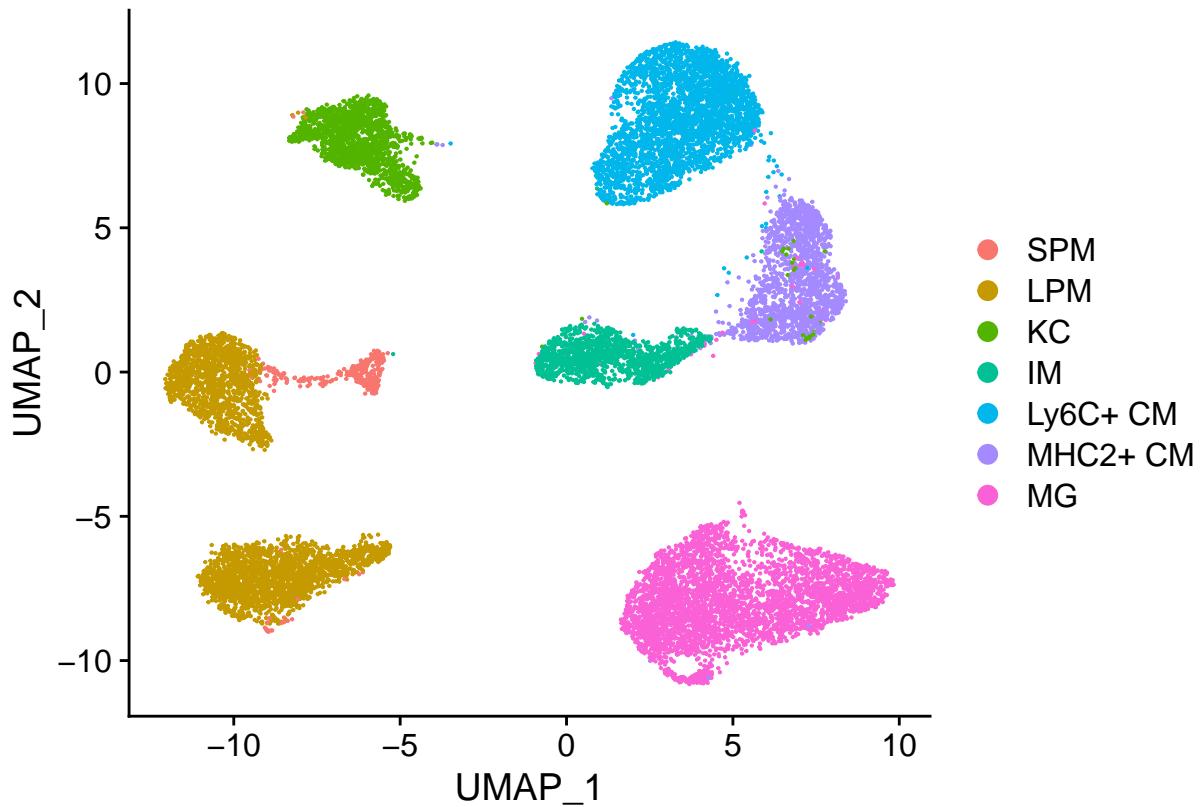
```



```
Idents(sc) <- "cell.type"

levels(sc) <- c("SPM", "LPM", "KC", "IM", "Ly6C+ CM", "MHC2+ CM",
               "MG")

DimPlot(sc)
```

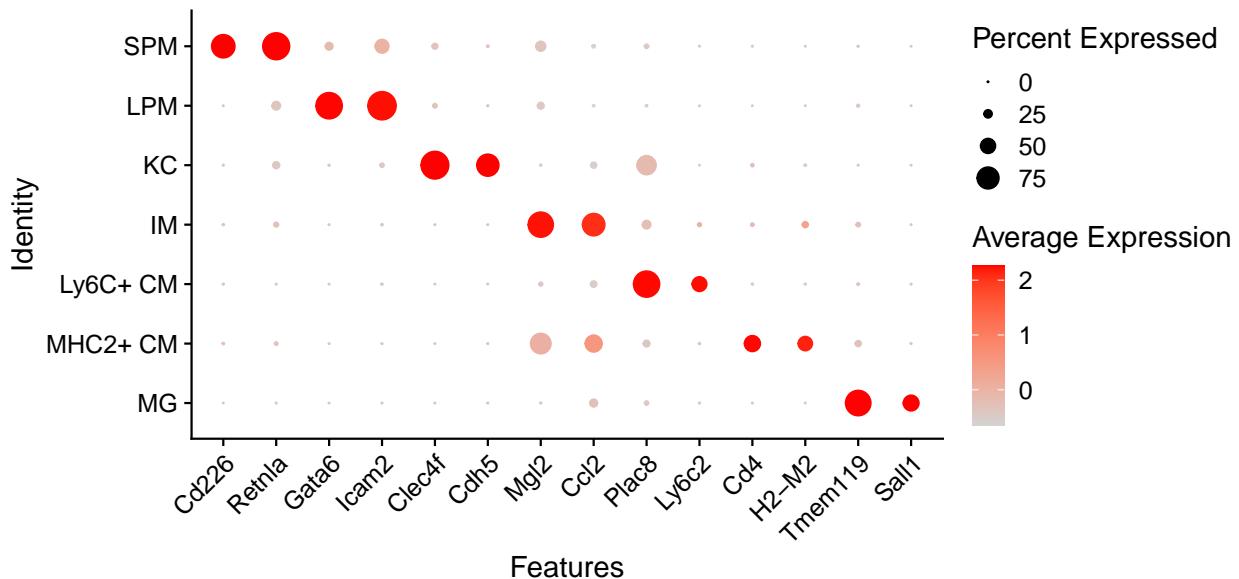


```

levels(sc) <- c("MG", "MHC2+ CM", "Ly6C+ CM", "IM", "KC", "LPM",
"SPM")

DotPlot(sc, cols = c("lightgrey", "red"), features = c("Cd226",
"Retnla", "Gata6", "Icam2", "Clec4f", "Cdh5", "Mgl2", "Ccl2",
"Plac8", "Ly6c2", "Cd4", "H2-M2", "Tmem119", "Sall1")) +
theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))

```



```
levels(sc) <- c("SPM", "LPM", "KC", "IM", "Ly6C+ CM", "MHC2+ CM",
  "MG")
```

Dotplot CM subset specific signature

```
Ly6C_CM_sign <- read_excel("Ly6C_CM_sign.xlsx")

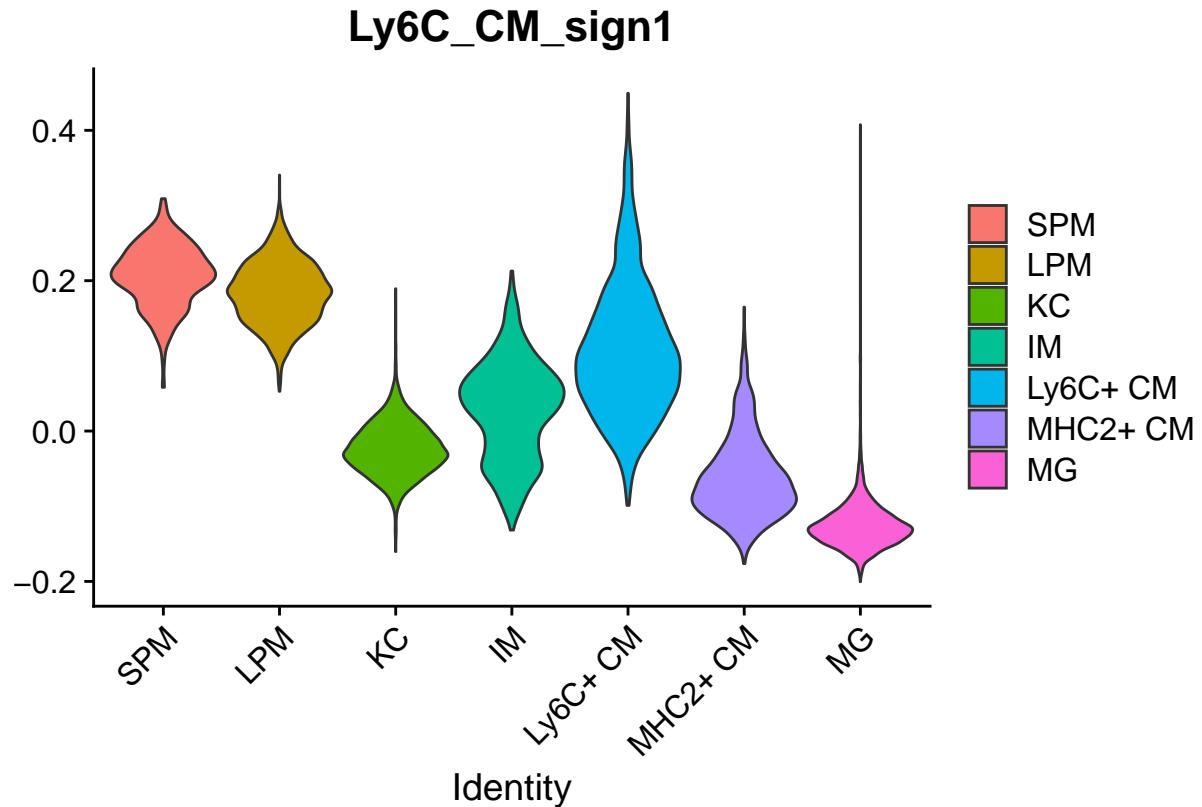
Ly6C_CM_sign <- Ly6C_CM_sign$Ly6C_CM_sign

Ly6C_CM_sign <- list(Ly6C_CM_sign)

sc <- AddModuleScore(sc, features = Ly6C_CM_sign, name = "Ly6C_CM_sign")
```

```
## Warning: The following features are not present in the object: Chi3l3,
## I830127L07Rik, Serpinb10, 1810033B17Rik, E430024C06Rik, Chi3l1, E230008N13Rik,
## Sepx1, 5730408K05Rik, Emr4, Snord33, Snora52, Gm14005, Pyhin1, 4930562F07Rik,
## Aim1, Agpat9, Fam65b, Mki67ip, Irg1, Gm4955, Diap1, Mx1, Gyk, Diap2,
## LOC100505155, A530023014Rik, n-R5s180, Pira11, Gm1966, 2410091C18Rik, Hmha1,
## Pira1, Slc24a6, Gnb211, 2610018G03Rik, Chi3l4, 38961, Gm4876, 1110007C09Rik,
## Gm10268, 40057, A630033H20Rik, not searching for symbol synonyms
```

```
VlnPlot(sc, features = "Ly6C_CM_sign1", pt.size = 0)
```



```

MHC2_CM_sign <- read_excel("MHC2_CM_sign.xlsx")
MHC2_CM_sign <- MHC2_CM_sign$MHC2_CM_sign
MHC2_CM_sign <- list(MHC2_CM_sign)
sc <- AddModuleScore(sc, features = MHC2_CM_sign, name = "MHC2_CM_sign")

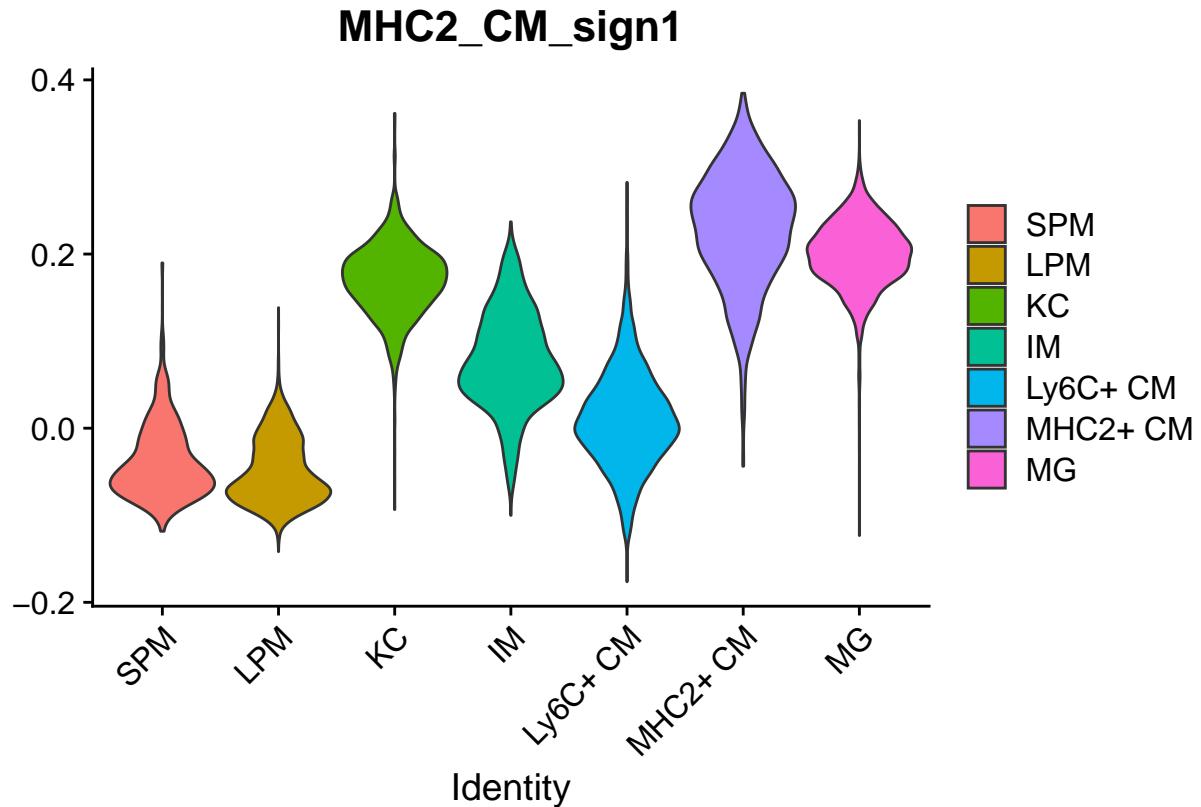
```

```

## Warning: The following features are not present in the object: Igk-V19-14,
## Gbp1, Gpr114, F630111L10Rik, Gm10134, Leprel1, Amica1, Igkv19-93, LOC435333,
## BC055004, LOC672291, AI324046, AW112010, LOC100046496, 8430419L09Rik, Igk-V28,
## 4930506M07Rik, G530011006Rik, D8Ertd82e, 1810011H11Rik, Igkv4-71, Gm11711,
## Igh-VJ558, LOC100504914, 6720401G13Rik, Ccdc23, D230041D01Rik, Gm16848,
## 1700047G07Rik, Igkv1-72, Olfr161, Igk-V2, Tdgf1, Fam213a, Gm9001, Lphn3,
## Igk-V21-2, Mir680-2, Igkv4-55, Wbp5, Igj, 1300002K09Rik, Gm10307, D17H6S56E-5,
## 1110001A16Rik, AI427809, Ifi27l1, Gm5129, Fam46c, Mpa2l, Pisd-ps1, Epb4.113,
## Sepp1, 1700009P17Rik, Gm10522, vipr1, Cxx1c, Plac9, Fam213b, Mir1956, Selm,
## 5430435G22Rik, Emr1, Pvt1, Fam178a, Ngfrap1, Mir99a, 8430410A17Rik,
## 1500011K16Rik, A730008H23Rik, Trdn, 1110058L19Rik, Hbb-b2, Mir212, Fam212a,
## Gm7120, not searching for symbol synonyms

```

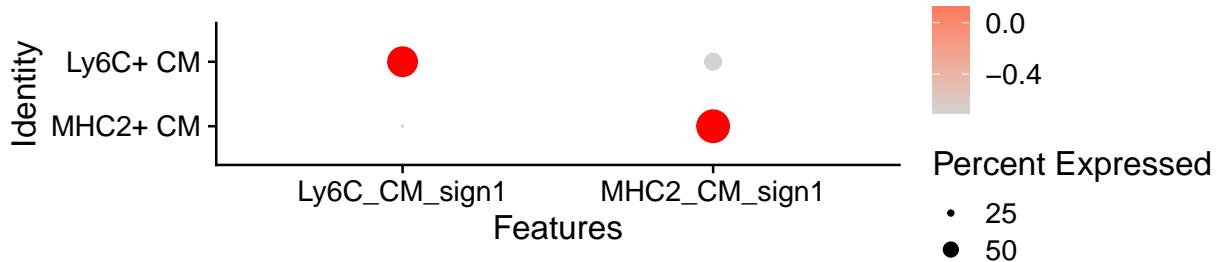
```
VlnPlot(sc, features = "MHC2_CM_sign1", pt.size = 0)
```



```
levels(sc) <- c("MG", "MHC2+ CM", "Ly6C+ CM", "IM", "KC", "LPM",
"SPM")

DotPlot(sc, cols = c("lightgrey", "red"), features = c("Ly6C_CM_sign1",
"Ly6C_CM_sign1"), idents = c("MHC2+ CM", "Ly6C+ CM"))
```

Warning: Scaling data with a low number of groups may produce misleading
results

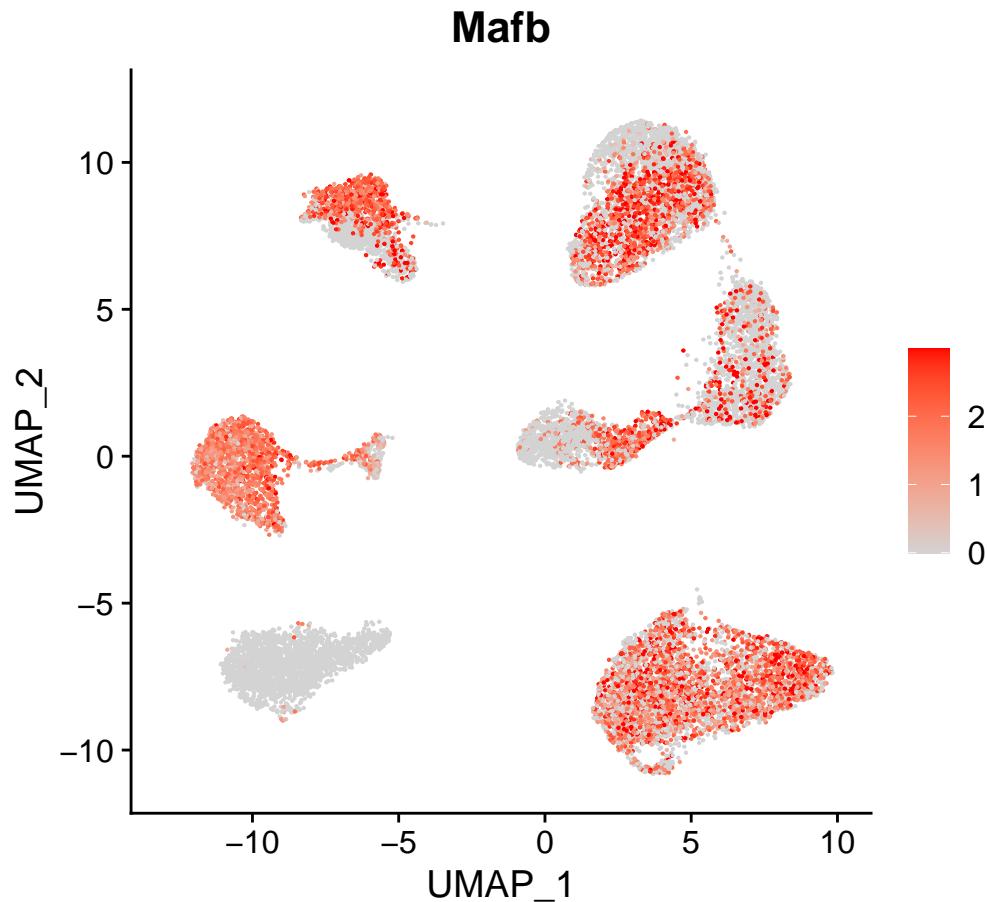


```
levels(sc) <- c("SPM", "LPM", "KC", "IM", "Ly6C+ CM", "MHC2+ CM",
"MG")
```

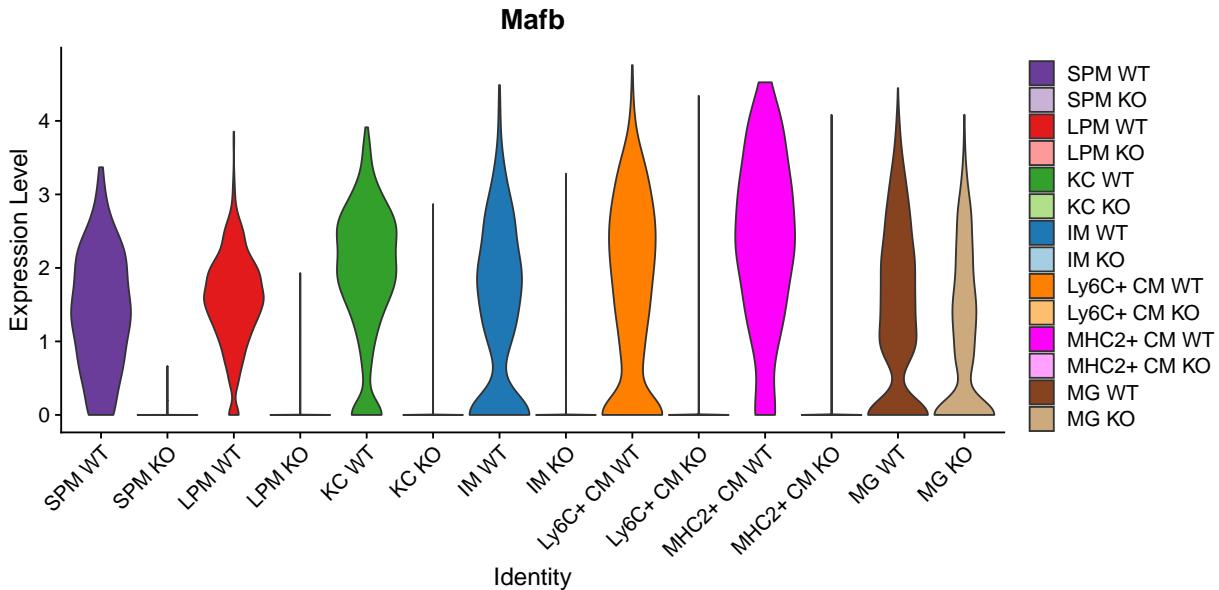
```
Idents(sc) <- "Condition"
```

Validation MafB KO

```
FeaturePlot(sc, features = "Mafb", cols = c("lightgrey", "red"),
max.cutoff = "q90")
```



```
VlnPlot(sc, features = "Mafb", cols = pal, pt.size = 0)
```



```
saveRDS(sc, "sc.rds")
```

```
sessionInfo()
```

```
## R version 4.2.3 (2023-03-15 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Dutch_Netherlands.utf8  LC_CTYPE=Dutch_Netherlands.utf8
## [3] LC_MONETARY=Dutch_Netherlands.utf8 LC_NUMERIC=C
## [5] LC_TIME=Dutch_Netherlands.utf8
##
## attached base packages:
## [1] stats      graphics   grDevices  utils       datasets   methods    base
##
## other attached packages:
## [1] readxl_1.4.3        RColorBrewer_1.1-3  ggplot2_3.4.4      SeuratObject_4.1.3
## [5] Seurat_4.3.0
##
## loaded via a namespace (and not attached):
## [1] Rtsne_0.17           colorspace_2.1-0     deldir_2.0-2
## [4] ellipsis_0.3.2       ggridges_0.5.5      rstudioapi_0.15.0
## [7] spatstat.data_3.0-3   farver_2.1.1       leiden_0.4.3.1
## [10] listenv_0.9.0        ggrepel_0.9.4      fansi_1.0.6
## [13] codetools_0.2-19     splines_4.2.3      knitr_1.45
## [16] polyclip_1.10-6      spam_2.10-0       jsonlite_1.8.8
## [19] ica_1.0-3           cluster_2.1.4     png_0.1-8
## [22] uwot_0.1.16          shiny_1.8.0       sctransform_0.4.1
## [25] spatstat.sparse_3.0-3 compiler_4.2.3    httr_1.4.7
## [28] Matrix_1.6-4         fastmap_1.1.1     lazyeval_0.2.2
```

```

## [31] cli_3.6.2           later_1.3.2          formatR_1.14
## [34] htmltools_0.5.7      tools_4.2.3           dotCall164_1.1-1
## [37] igraph_1.6.0         gtable_0.3.5          glue_1.6.2
## [40] RANN_2.6.1           reshape2_1.4.4        dplyr_1.1.4
## [43] Rcpp_1.0.11          scattermore_1.2       cellranger_1.1.0
## [46] vctrs_0.6.5          spatstat.explore_3.2-5 nlme_3.1-162
## [49] progressr_0.14.0     lmtest_0.9-40         spatstat.random_3.2-2
## [52] xfun_0.39            stringr_1.5.1         globals_0.16.2
## [55] mime_0.12             miniUI_0.1.1.1        lifecycle_1.0.4
## [58] irlba_2.3.5.1        goftest_1.2-3         future_1.33.1
## [61] MASS_7.3-58.2         zoo_1.8-12           scales_1.3.0
## [64] promises_1.2.1        spatstat.utils_3.0-4  parallel_4.2.3
## [67] yaml_2.3.7            reticulate_1.34.0     pbapply_1.7-2
## [70] gridExtra_2.3          stringi_1.8.3          highr_0.10
## [73] rlang_1.1.2            pkgconfig_2.0.3        matrixStats_1.2.0
## [76] evaluate_0.23          lattice_0.20-45       ROCR_1.0-11
## [79] purrrr_1.0.2           tensor_1.5            labeling_0.4.3
## [82] patchwork_1.1.3        htmlwidgets_1.6.4      cowplot_1.1.2
## [85] tidyselect_1.2.1        parallelly_1.36.0     RcppAnnoy_0.0.21
## [88] plyr_1.8.9              magrittr_2.0.3          R6_2.5.1
## [91] generics_0.1.3          withr_3.0.1            pillar_1.9.0
## [94] fitdistrplus_1.1-11    survival_3.5-3         abind_1.4-5
## [97] sp_2.1-2                tibble_3.2.1           future.apply_1.11.1
## [100] KernSmooth_2.23-20    utf8_1.2.4             spatstat.geom_3.2-7
## [103] plotly_4.10.3          rmarkdown_2.25          grid_4.2.3
## [106] data.table_1.14.10    digest_0.6.31          xtable_1.8-4
## [109] tidyverse_1.3.0         httpuv_1.6.13          munsell_0.5.1
## [112] viridisLite_0.4.2

```