

scRNA-Seq RTM WT vs KO - DEG

Domien Vanneste

2024-12-11 09:35:58 +0100

Contents

Introduction	1
Load packages	1
Load Seurat object	2
Differential gene expression analysis	2
SPM	2
LPM	5
KC	8
IM	11
Ly6C+ CM	14
MHC-II+ CM	17
MG	20

Introduction

We conducted differential gene expression analysis between RTMs from Lyz2CreMafbfl/fl (KO) mice and Mafbfl/fl (WT) littermate controls and identified 92 DEGs in SPM, 535 DEGs in LPM, 262 DEGs in KC, 287 DEGs in IM, 723 DEGs in Ly6C+ CM, 150 DEGs in MHC-II+ CM and 299 DEGs in MG. Of note, known RTM-specific identity genes such as Retnla for SPM, Vsig4 for LPM, CD163 for KC, and Fcrls for MG were significantly lower expressed in Mafb-deficient cells.

Load packages

```
suppressMessages({  
    library(Seurat)  
    library(SeuratObject)  
    library(ggplot2)  
    library(RColorBrewer)
```

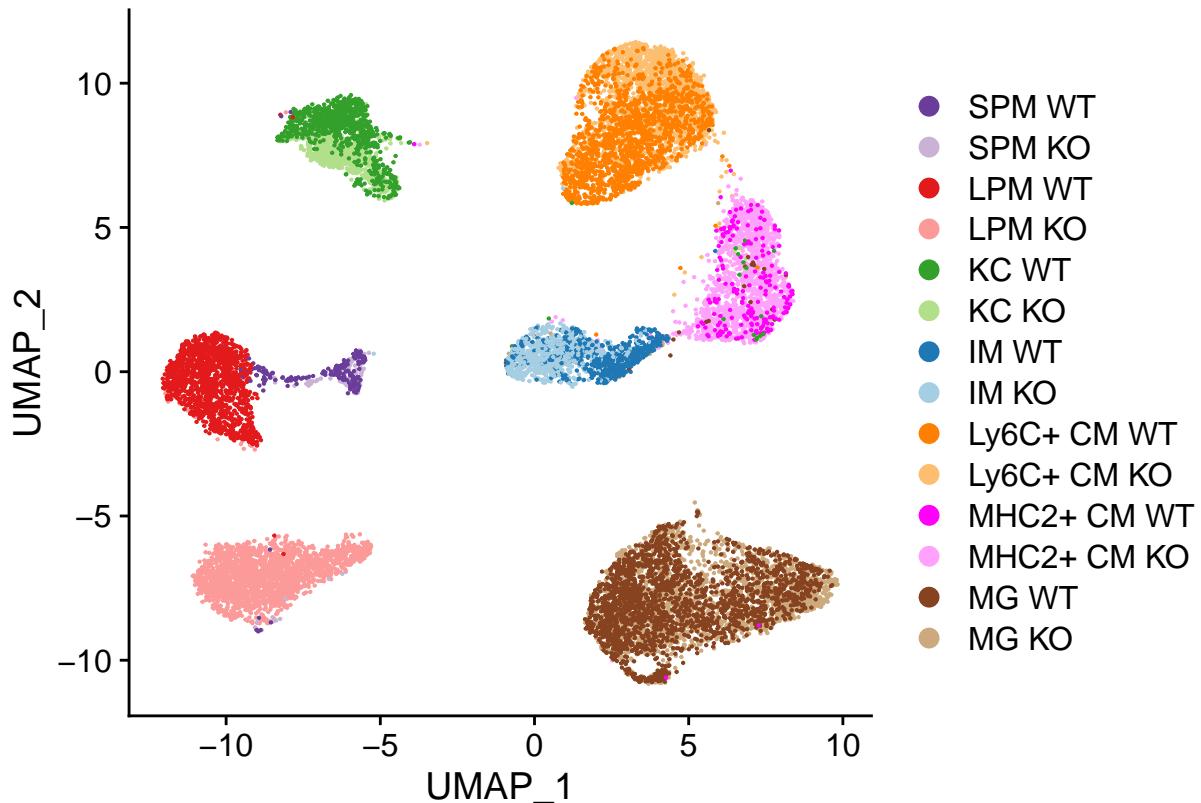
```
library(dplyr)
library(ggrepel)
})
```

Load Seurat object

```
sc <- readRDS("sc.rds")

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

DimPlot(sc, cols = pal)
```



Differential gene expression analysis

SPM

```

SPM_WTvsKO <- FindMarkers(sc, ident.1 = "SPM KO", ident.2 = "SPM WT",
                           logfc.threshold = 0)
write.csv(SPM_WTvsKO, file = "SPM_WTvsKO.csv")

SPM_WTvsKO <- read.csv("SPM_WTvsKO.csv", header = T, row.names = 1)
SPM_WTvsKO.markers <- SPM_WTvsKO[SPM_WTvsKO$p_val_adj < 0.05 &
  abs(SPM_WTvsKO$avg_log2FC) > 0.25, ]
SPM_WTvsKO.markers <- SPM_WTvsKO.markers[order(SPM_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(SPM_WTvsKO.markers)

## [1] 92

nrow(SPM_WTvsKO.markers[SPM_WTvsKO.markers$avg_log2FC > 0, ])

## [1] 44

nrow(SPM_WTvsKO.markers[SPM_WTvsKO.markers$avg_log2FC < 0, ])

## [1] 48

write.csv(SPM_WTvsKO.markers, file = "SPM_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
SPM_WTvsKO.volcano = mutate(SPM_WTvsKO, Sig = ifelse((abs(SPM_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (SPM_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
SPM_WTvsKO.volcano$Sig[SPM_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  SPM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"
SPM_WTvsKO.volcano$Sig[SPM_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  SPM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

SPM_WTvsKO.volcano$Gene <- rownames(SPM_WTvsKO.volcano)
Gene.to.show.ValcanoPlot <- rownames(SPM_WTvsKO.volcano[SPM_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(SPM_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#6a3d9a",
  n.s. = "grey", KO = "#cab2d6")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(SPM_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 3, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

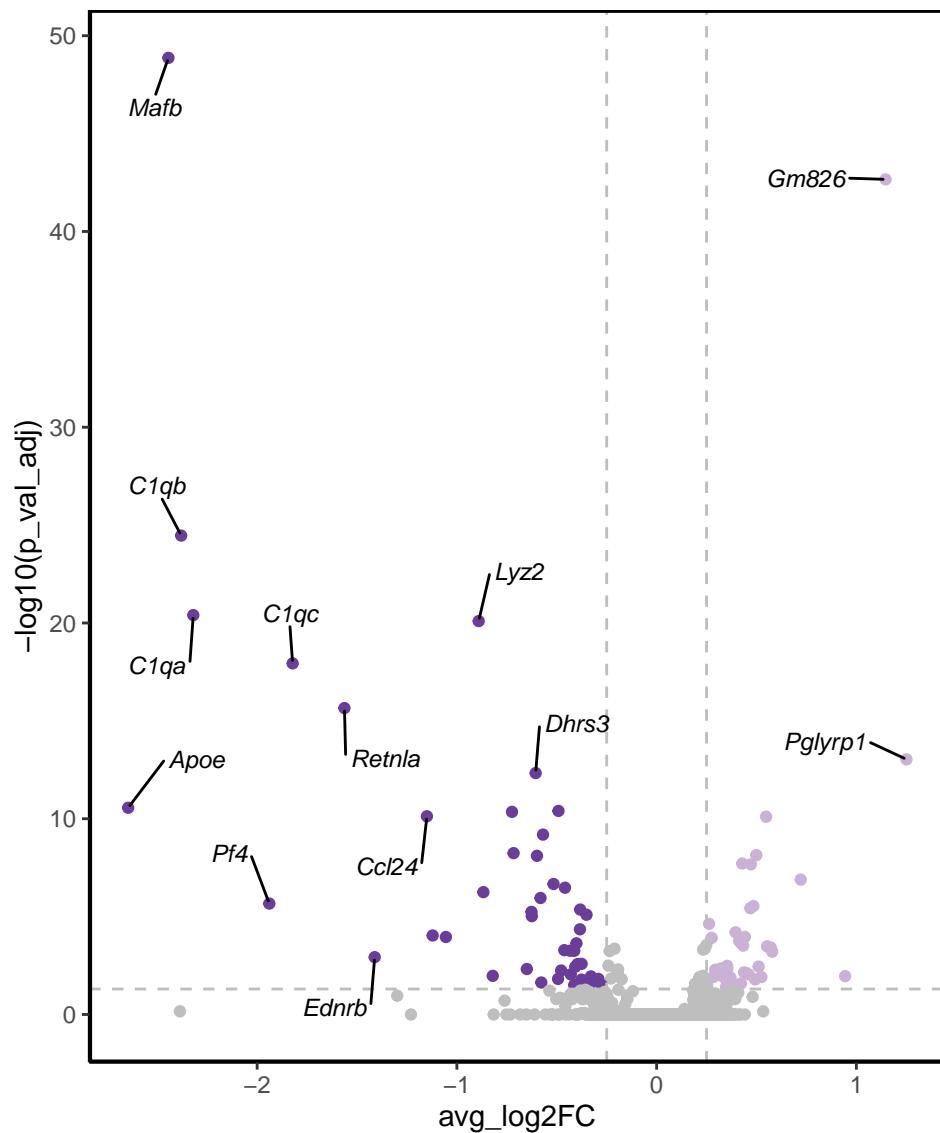
```

```

## Warning: The 'size' argument of 'element_rect()' is deprecated as of ggplot2 3.4.0.
## i Please use the 'linewidth' argument instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.

## Warning: ggrepel: 79 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

SPM <- subset(sc, idents = c("SPM WT", "SPM KO"))
all.genes <- rownames(SPM)
SPM <- ScaleData(SPM, features = all.genes)

```

```

## Centering and scaling data matrix

```

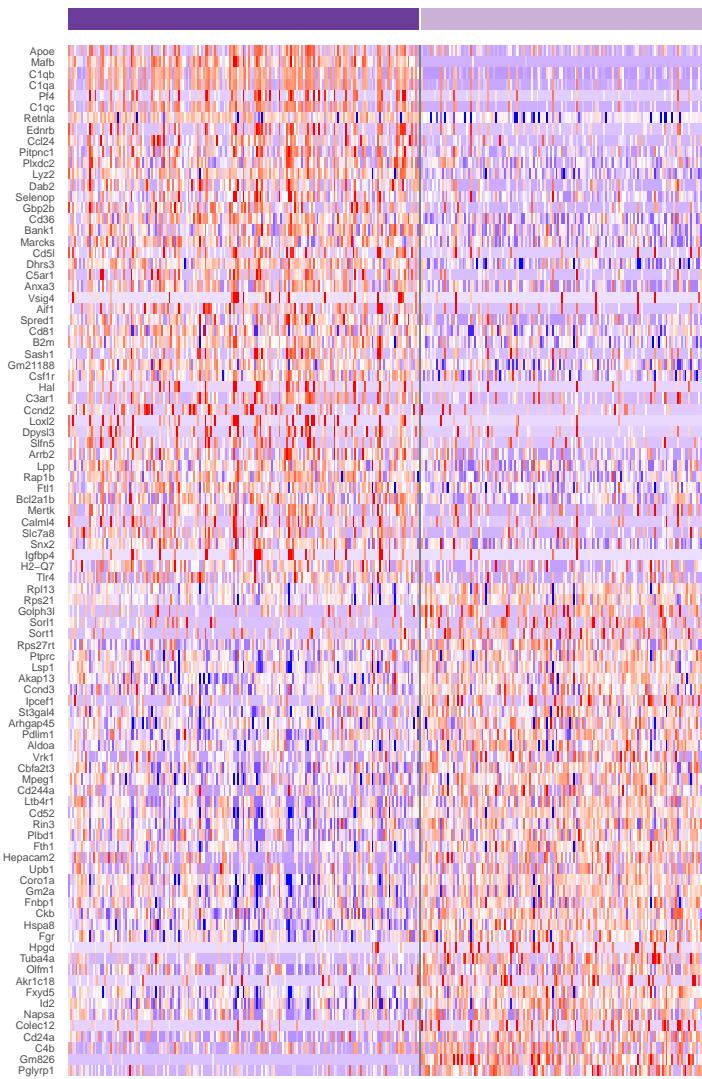
```

pal_SPM <- c("#6a3d9a", "#cab2d6")

DoHeatmap(SPM, features = rev(rownames(SPM_WTvsKO.markers)),
  group.colors = pal_SPM, size = 0) + scale_fill_gradientn(colors = c("blue",
  "white", "red")) + theme(axis.text.y = element_text(size = 4)) +
  theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



LPM

```

LPM_WTvsKO <- FindMarkers(sc, ident.1 = "LPM KO", ident.2 = "LPM WT",
  logfc.threshold = 0)
write.csv(LPM_WTvsKO, file = "LPM_WTvsKO.csv")

LPM_WTvsKO <- read.csv("LPM_WTvsKO.csv", header = T, row.names = 1)
LPM_WTvsKO.markers <- LPM_WTvsKO[LPM_WTvsKO$p_val_adj < 0.05 &
  abs(LPM_WTvsKO$avg_log2FC) > 0.25, ]
LPM_WTvsKO.markers <- LPM_WTvsKO.markers[order(LPM_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(LPM_WTvsKO.markers)

## [1] 535

nrow(LPM_WTvsKO.markers[LPM_WTvsKO.markers$avg_log2FC > 0, ])

## [1] 327

nrow(LPM_WTvsKO.markers[LPM_WTvsKO.markers$avg_log2FC < 0, ])

## [1] 208

write.csv(LPM_WTvsKO.markers, file = "LPM_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
LPM_WTvsKO.volcano = mutate(LPM_WTvsKO, Sig = ifelse((abs(LPM_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (LPM_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
LPM_WTvsKO.volcano$Sig[LPM_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  LPM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"

LPM_WTvsKO.volcano$Sig[LPM_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  LPM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

LPM_WTvsKO.volcano$Gene <- rownames(LPM_WTvsKO.volcano)

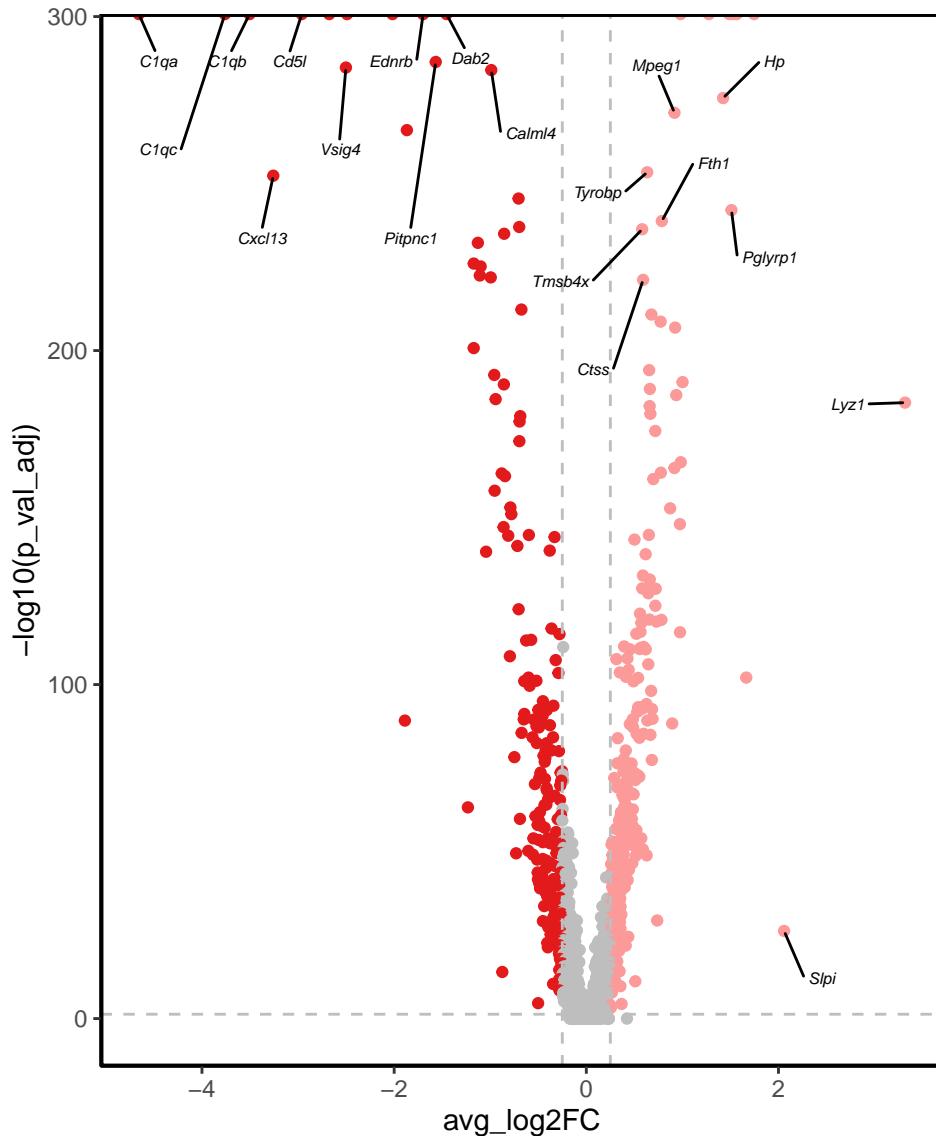
Gene.to.show.ValcanoPlot <- rownames(LPM_WTvsKO.volcano[LPM_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(LPM_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#e31a1c",
  n.s. = "grey", KO = "#fb9a99")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(LPM_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 2, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

```

```
## Warning: ggrepel: 516 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```



```
LPM <- subset(sc, idents = c("LPM WT", "LPM KO"))
all.genes <- rownames(LPM)
LPM <- ScaleData(LPM, features = all.genes)
```

```
## Centering and scaling data matrix
```

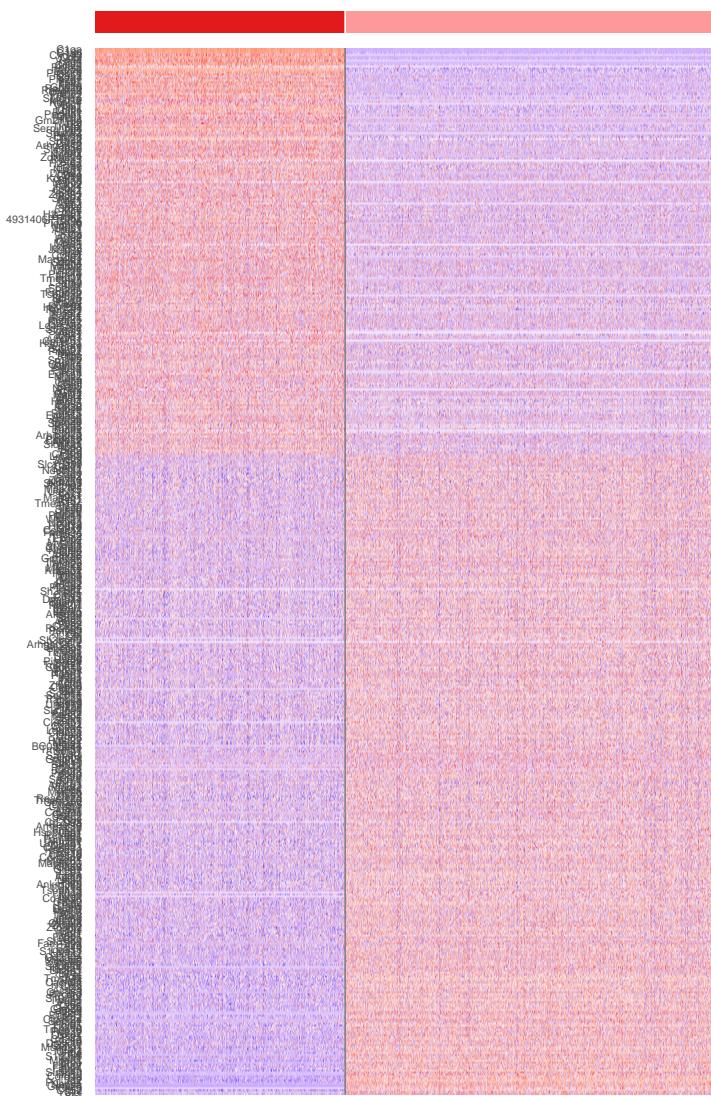
```
pal_LPM <- c("#e31a1c", "#fb9a99")

DoHeatmap(LPM, features = rev(rownames(LPM_WTvsKO.markers)),
  group.colors = pal_LPM, size = 0) + scale_fill_gradientn(colors = c("blue",
  "white", "red")) + theme(axis.text.y = element_text(size = 4)) +
  theme(legend.position = "none")
```

```

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



KC

```

KC_WTvsKO <- FindMarkers(sc, ident.1 = "KC KO", ident.2 = "KC WT",
  logfc.threshold = 0)
write.csv(KC_WTvsKO, file = "KC_WTvsKO.csv")

KC_WTvsKO <- read.csv("KC_WTvsKO.csv", header = T, row.names = 1)
KC_WTvsKO.markers <- KC_WTvsKO[KC_WTvsKO$p_val_adj < 0.05 & abs(KC_WTvsKO$avg_log2FC) >
  0.25, ]
KC_WTvsKO.markers <- KC_WTvsKO.markers[order(KC_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(KC_WTvsKO.markers)

```

```

## [1] 262

nrow(KC_WTvsKO.markers[KC_WTvsKO.markers$avg_log2FC > 0, ])

## [1] 128

nrow(KC_WTvsKO.markers[KC_WTvsKO.markers$avg_log2FC < 0, ])

## [1] 134

write.csv(KC_WTvsKO.markers, file = "KC_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
KC_WTvsKO.volcano = mutate(KC_WTvsKO, Sig = ifelse((abs(KC_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (KC_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
KC_WTvsKO.volcano$Sig[KC_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  KC_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"
KC_WTvsKO.volcano$Sig[KC_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  KC_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

KC_WTvsKO.volcano$Gene <- rownames(KC_WTvsKO.volcano)

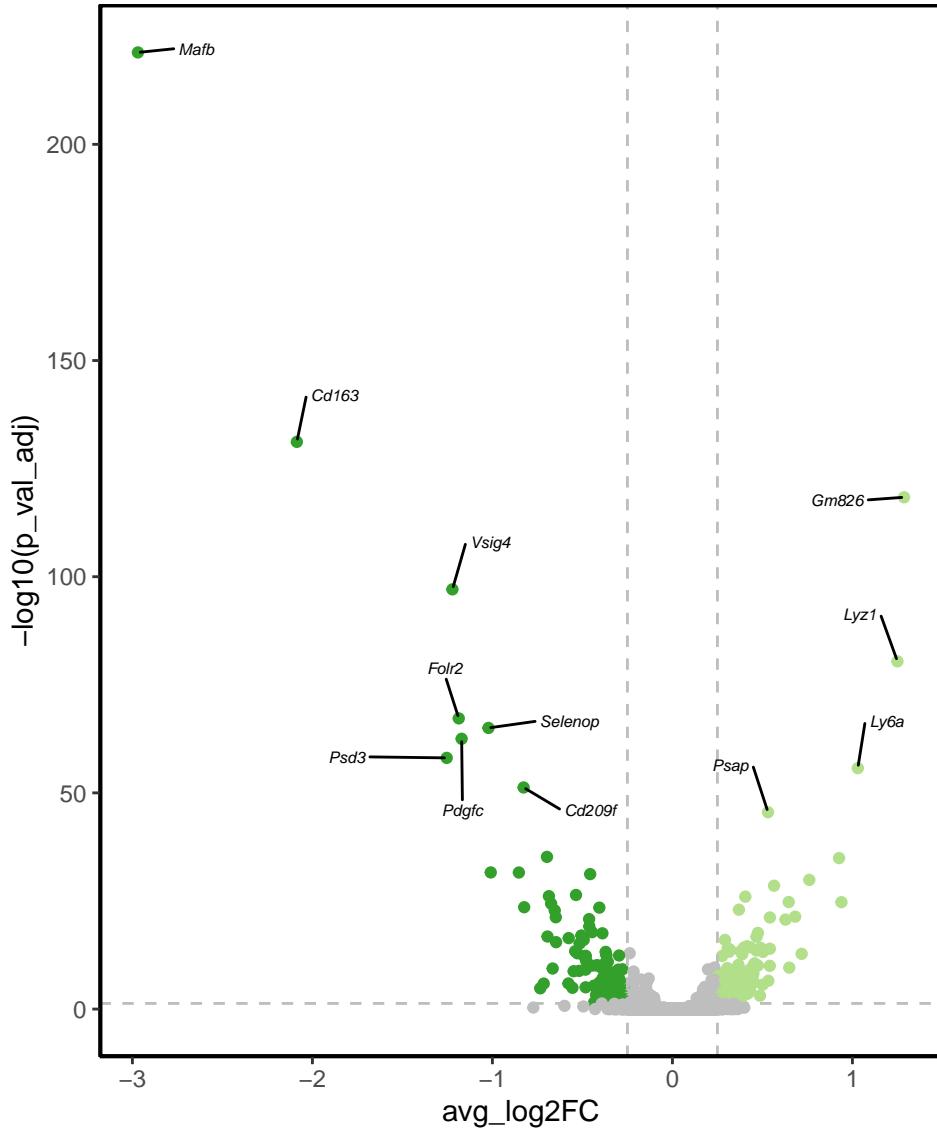
Gene.to.show.ValcanoPlot <- rownames(KC_WTvsKO.volcano[KC_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(KC_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#33a02c",
  n.s. = "grey", KO = "#b2df8a")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(KC_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 2, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

## Warning: ggrepel: 250 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

KC <- subset(sc, idents = c("KC WT", "KC KO"))
all.genes <- rownames(KC)
KC <- ScaleData(KC, features = all.genes)

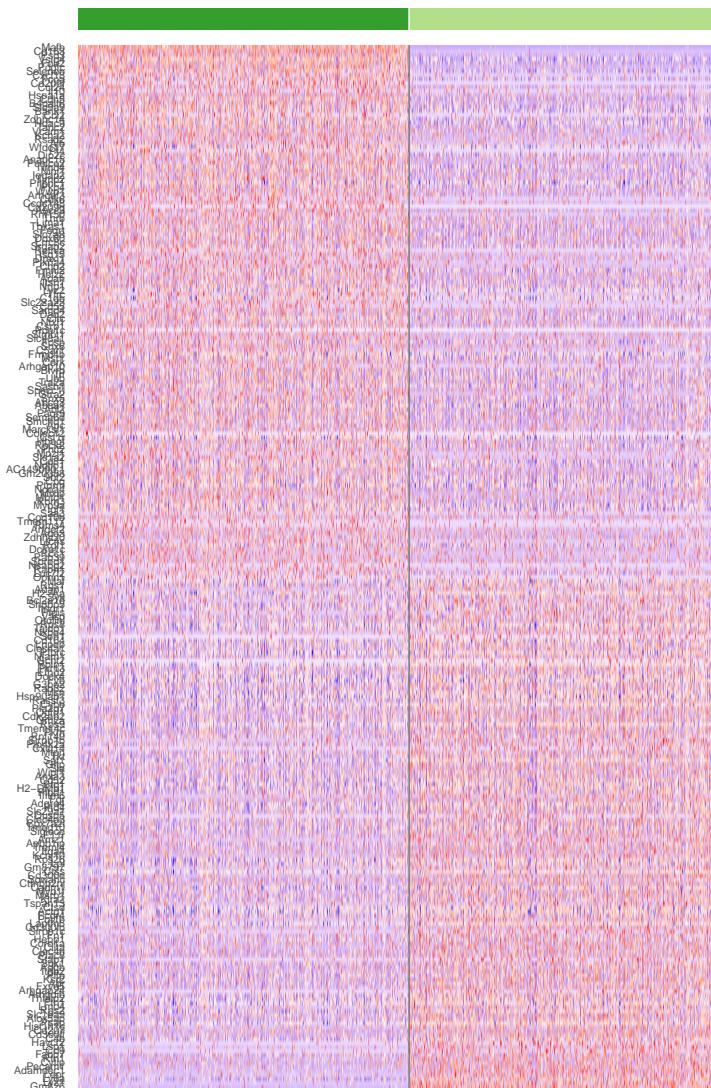
## Centering and scaling data matrix

pal_KC <- c("#33a02c", "#b2df8a")

DoHeatmap(KC, features = rev(rownames(KC_WTvsKO.markers)), group.colors = pal_KC,
          size = 0) + scale_fill_gradientn(colors = c("blue", "white",
          "red")) + theme(axis.text.y = element_text(size = 4)) + theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



IM

```

IM_WTvsKO <- FindMarkers(sc, ident.1 = "IM KO", ident.2 = "IM WT",
  logfc.threshold = 0)
write.csv(IM_WTvsKO, file = "IM_WTvsKO.csv")

IM_WTvsKO <- read.csv("IM_WTvsKO.csv", header = T, row.names = 1)
IM_WTvsKO.markers <- IM_WTvsKO[IM_WTvsKO$p_val_adj < 0.05 & abs(IM_WTvsKO$avg_log2FC) >
  0.25, ]
IM_WTvsKO.markers <- IM_WTvsKO.markers[order(IM_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(IM_WTvsKO.markers)

## [1] 287

```

```

nrow(IM_WTvsKO.markers[IM_WTvsKO.markers$avg_log2FC > 0, ])

## [1] 180

nrow(IM_WTvsKO.markers[IM_WTvsKO.markers$avg_log2FC < 0, ])

## [1] 107

write.csv(IM_WTvsKO.markers, file = "IM_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
IM_WTvsKO.volcano = mutate(IM_WTvsKO, Sig = ifelse((abs(IM_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (IM_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
IM_WTvsKO.volcano$Sig[IM_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  IM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"
IM_WTvsKO.volcano$Sig[IM_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  IM_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

IM_WTvsKO.volcano$Gene <- rownames(IM_WTvsKO.volcano)

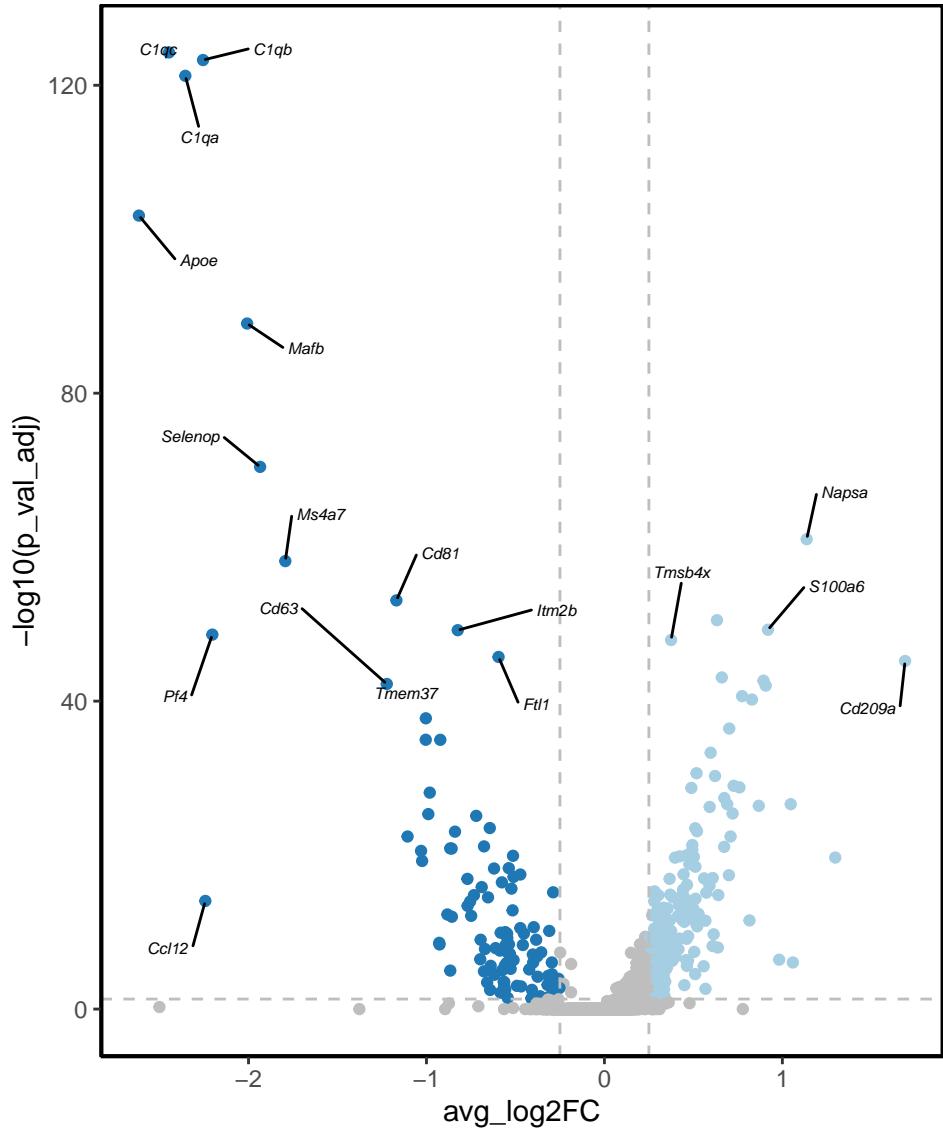
Gene.to.show.ValcanoPlot <- rownames(IM_WTvsKO.volcano[IM_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(IM_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#1f78b4",
  n.s. = "grey", KO = "#a6cee3")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(IM_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 2, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

## Warning: ggrepel: 269 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

IM <- subset(sc, idents = c("IM WT", "IM KO"))
all.genes <- rownames(IM)
IM <- ScaleData(IM, features = all.genes)

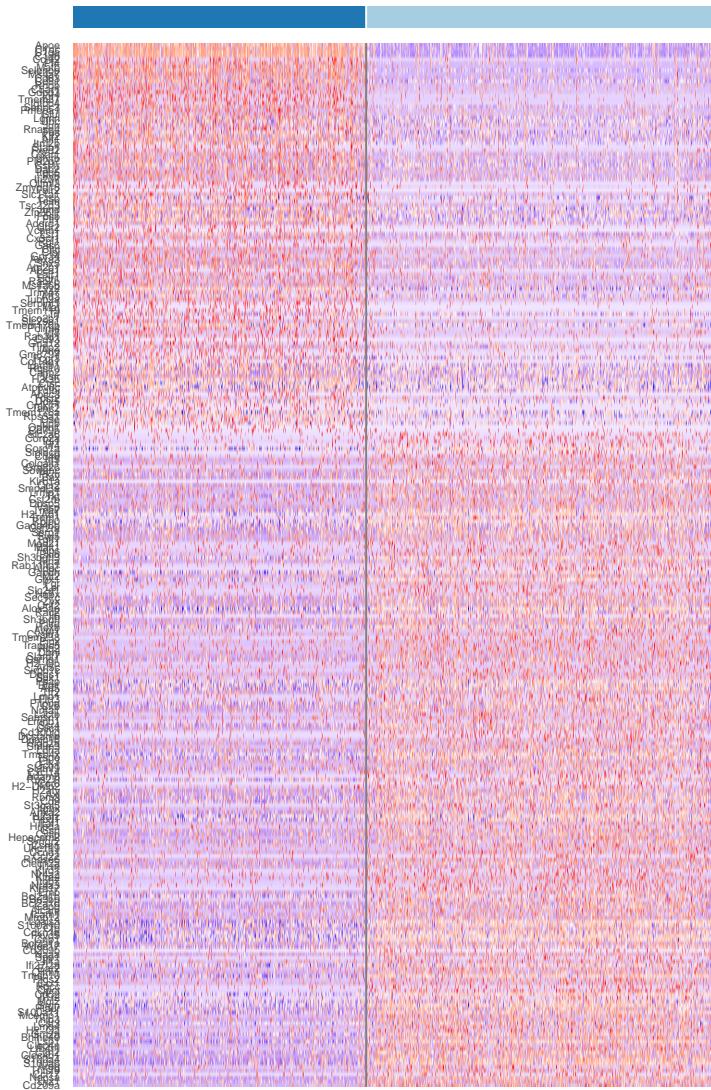
## Centering and scaling data matrix

pal_IM <- c("#1f78b4", "#a6cee3")

DoHeatmap(IM, features = rev(rownames(IM_WTvsKO.markers)), group.colors = pal_IM,
          size = 0) + scale_fill_gradientn(colors = c("blue", "white",
          "red")) + theme(axis.text.y = element_text(size = 4)) + theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



Ly6C+ CM

```
Ly6C_WTvsKO <- FindMarkers(sc, ident.1 = "Ly6C+ CM KO", ident.2 = "Ly6C+ CM WT",
  logfc.threshold = 0)
write.csv(Ly6C_WTvsKO, file = "Ly6C_WTvsKO.csv")
```

```
Ly6C_WTvsKO <- read.csv("Ly6C_WTvsKO.csv", header = T, row.names = 1)
Ly6C_WTvsKO.markers <- Ly6C_WTvsKO[Ly6C_WTvsKO$p_val_adj < 0.05 &
  abs(Ly6C_WTvsKO$avg_log2FC) > 0.25, ]
Ly6C_WTvsKO.markers <- Ly6C_WTvsKO.markers[order(Ly6C_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(Ly6C_WTvsKO.markers)
```

```
## [1] 723
```

```

nrow(Ly6C_WTvsKO.markers[Ly6C_WTvsKO.markers$avg_log2FC > 0,
  ])

## [1] 309

nrow(Ly6C_WTvsKO.markers[Ly6C_WTvsKO.markers$avg_log2FC < 0,
  ])

## [1] 414

write.csv(Ly6C_WTvsKO.markers, file = "Ly6C_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
Ly6C_WTvsKO.volcano = mutate(Ly6C_WTvsKO, Sig = ifelse((abs(Ly6C_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (Ly6C_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
Ly6C_WTvsKO.volcano$Sig[Ly6C_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  Ly6C_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"

Ly6C_WTvsKO.volcano$Sig[Ly6C_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  Ly6C_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

Ly6C_WTvsKO.volcano$Gene <- rownames(Ly6C_WTvsKO.volcano)

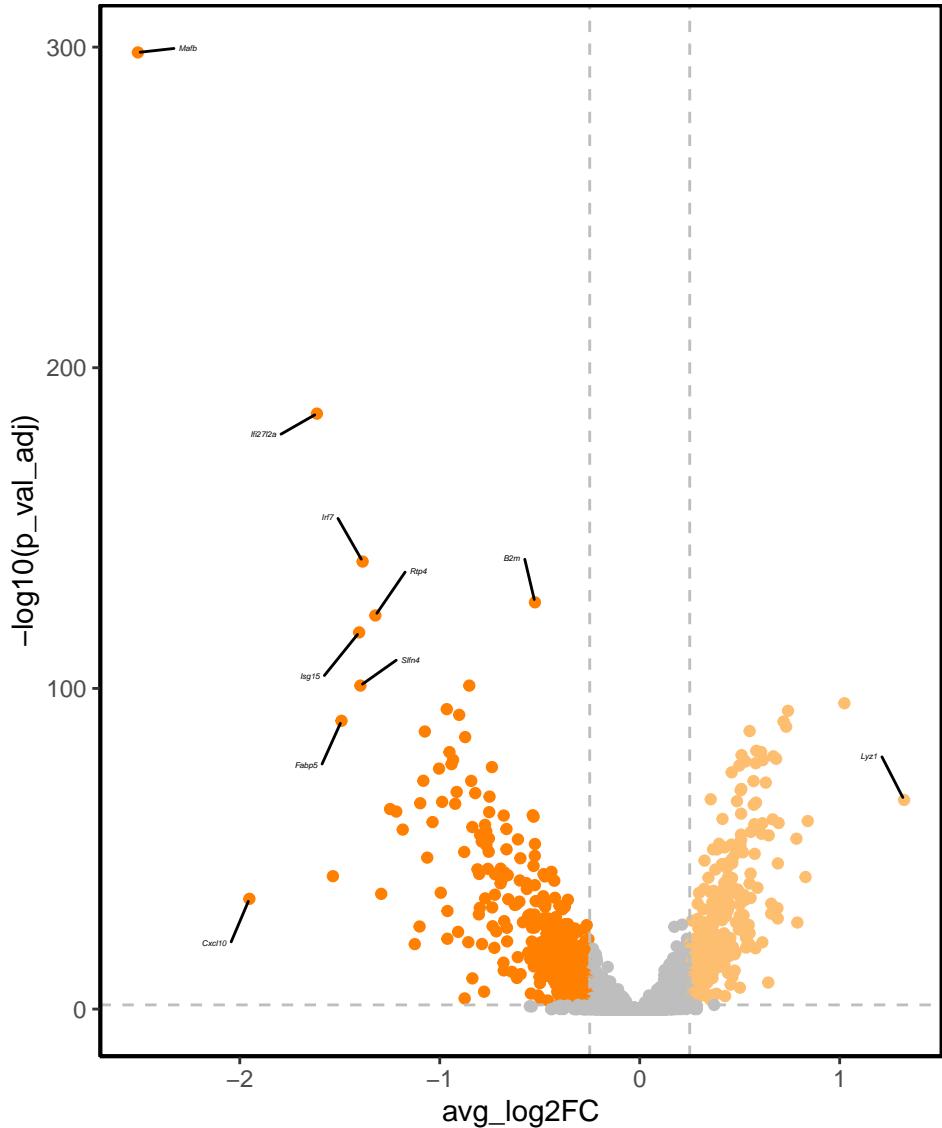
Gene.to.show.ValcanoPlot <- rownames(Ly6C_WTvsKO.volcano[Ly6C_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(Ly6C_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#ff7f00",
  n.s. = "grey", KO = "#fdbf6f")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(Ly6C_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 1, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

## Warning: ggrepel: 713 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

Ly6C_colon <- subset(sc, idents = c("Ly6C+ CM WT", "Ly6C+ CM KO"))
all.genes <- rownames(Ly6C_colon)
Ly6C_colon <- ScaleData(Ly6C_colon, features = all.genes)

## Centering and scaling data matrix

pal_Ly6C_colon <- c("#ff7f00", "#fdbf6f")

DoHeatmap(Ly6C_colon, features = rev(rownames(Ly6C_WTvsKO.markers)),
  group.colors = pal_Ly6C_colon, size = 0) + scale_fill_gradientn(colors = c("blue",
  "white", "red")) + theme(axis.text.y = element_text(size = 4)) +
  theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



MHC-II+ CM

```
MHC2_WTvsKO <- FindMarkers(sc, ident.1 = "MHC2+ CM KO", ident.2 = "MHC2+ CM WT",
  logfc.threshold = 0)
write.csv(MHC2_WTvsKO, file = "MHC2_WTvsKO.csv")
```

```
MHC2_WTvsKO <- read.csv("MHC2_WTvsKO.csv", header = T, row.names = 1)
MHC2_WTvsKO.markers <- MHC2_WTvsKO[MHC2_WTvsKO$p_val_adj < 0.05 &
  abs(MHC2_WTvsKO$avg_log2FC) > 0.25, ]
MHC2_WTvsKO.markers <- MHC2_WTvsKO.markers[order(MHC2_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(MHC2_WTvsKO.markers)
```

```
## [1] 150
```

```

nrow(MHC2_WTvsKO.markers[MHC2_WTvsKO.markers$avg_log2FC > 0,
  ])

## [1] 82

nrow(MHC2_WTvsKO.markers[MHC2_WTvsKO.markers$avg_log2FC < 0,
  ])

## [1] 68

write.csv(MHC2_WTvsKO.markers, file = "MHC2_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
MHC2_WTvsKO.volcano = mutate(MHC2_WTvsKO, Sig = ifelse((abs(MHC2_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (MHC2_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
MHC2_WTvsKO.volcano$Sig[MHC2_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  MHC2_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"
MHC2_WTvsKO.volcano$Sig[MHC2_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  MHC2_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

MHC2_WTvsKO.volcano$Gene <- rownames(MHC2_WTvsKO.volcano)

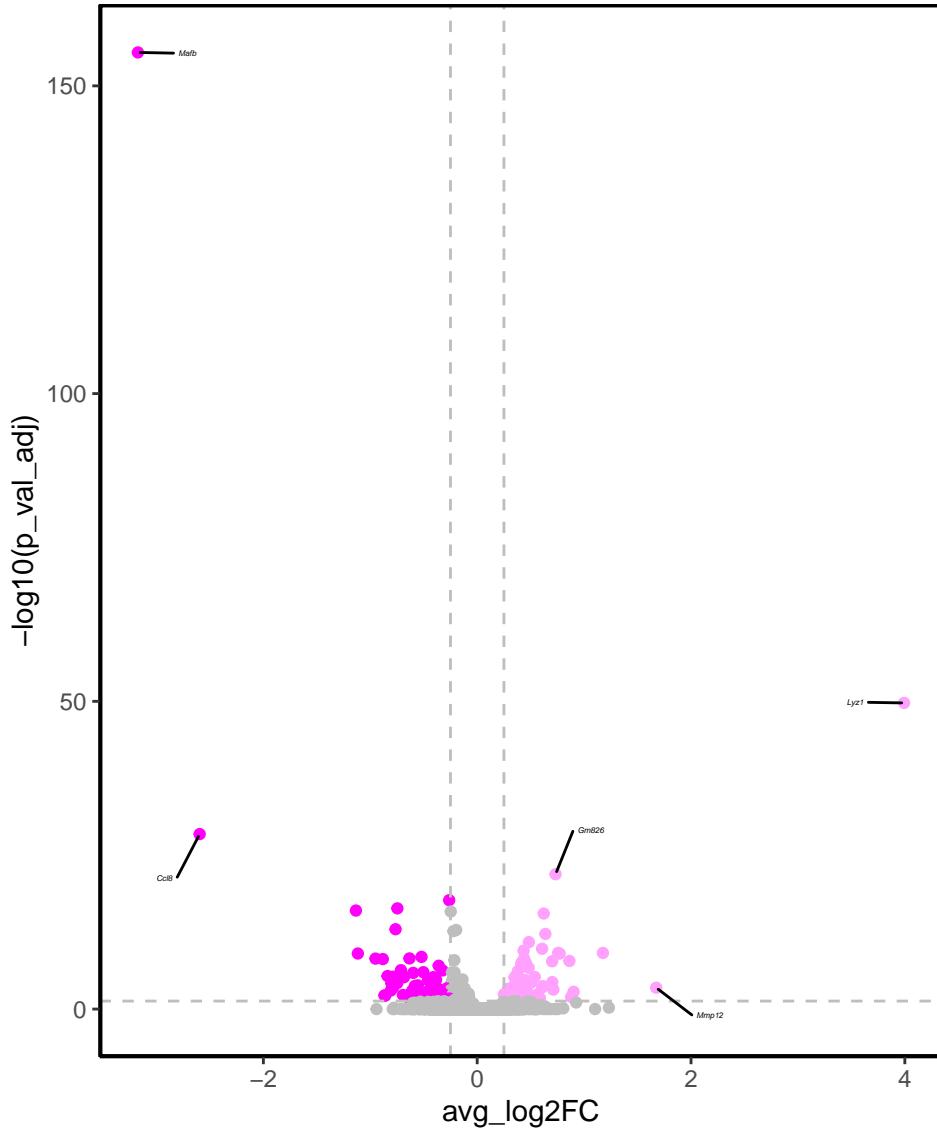
Gene.to.show.ValcanoPlot <- rownames(MHC2_WTvsKO.volcano[MHC2_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(SPM_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(MHC2_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#ff00fa",
  n.s. = "grey", KO = "#ffa1fd")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(MHC2_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 1, aes(label = Gene,
  fontface = "italic"), box.padding = 1, max.overlaps = 30) +
  theme_classic() + theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

## Warning: ggrepel: 145 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

MHC2_colon <- subset(sc, idents = c("MHC2+ CM WT", "MHC2+ CM KO"))
all.genes <- rownames(MHC2_colon)
MHC2_colon <- ScaleData(MHC2_colon, features = all.genes)

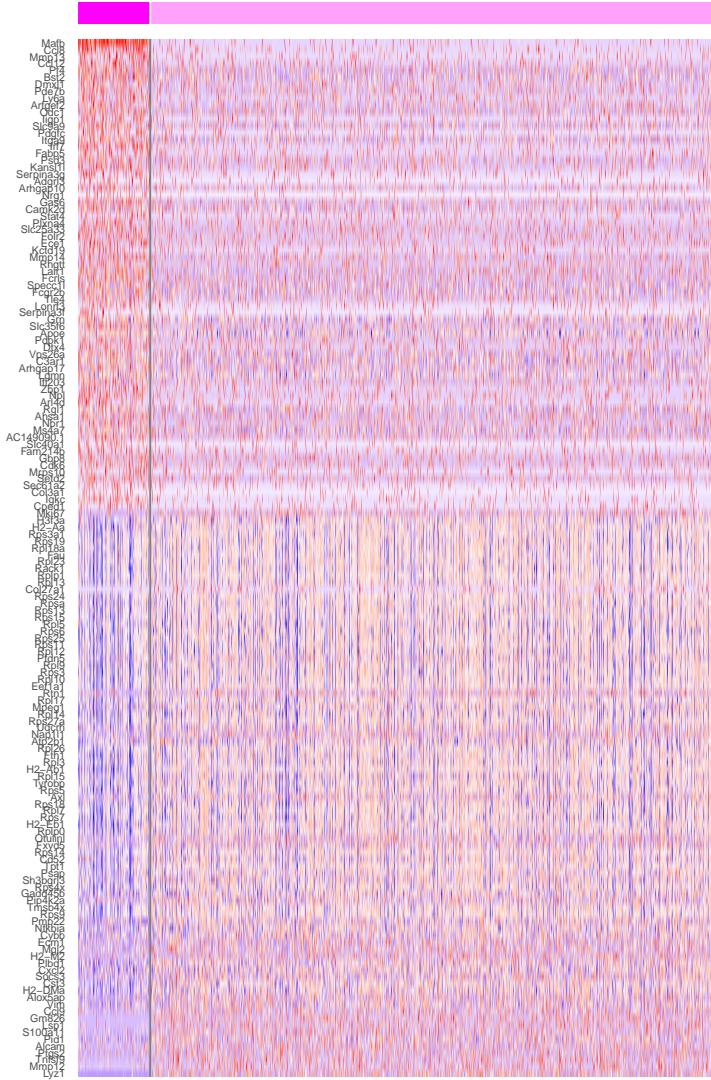
## Centering and scaling data matrix

pal_MHC2_colon <- c("#ff00fa", "#ffa1fd")

DoHeatmap(MHC2_colon, features = rev(rownames(MHC2_WTvsKO.markers)),
  group.colors = pal_MHC2_colon, size = 0) + scale_fill_gradientn(colors = c("blue",
  "white", "red")) + theme(axis.text.y = element_text(size = 4)) +
  theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



MG

```

MG_WTvsKO <- FindMarkers(sc, ident.1 = "MG KO", ident.2 = "MG WT",
  logfc.threshold = 0)
write.csv(MG_WTvsKO, file = "MG_WTvsKO.csv")

MG_WTvsKO <- read.csv("MG_WTvsKO.csv", header = T, row.names = 1)
MG_WTvsKO.markers <- MG_WTvsKO[MG_WTvsKO$p_val_adj < 0.05 & abs(MG_WTvsKO$avg_log2FC) >
  0.25, ]
MG_WTvsKO.markers <- MG_WTvsKO.markers[order(MG_WTvsKO.markers$avg_log2FC,
  decreasing = TRUE), ]
nrow(MG_WTvsKO.markers)

## [1] 299

```

```

nrow(MG_WTvsKO.markers[MG_WTvsKO.markers$avg_log2FC > 0, ])

## [1] 129

nrow(MG_WTvsKO.markers[MG_WTvsKO.markers$avg_log2FC < 0, ])

## [1] 170

write.csv(MG_WTvsKO.markers, file = "MG_WTvsKO_markers.csv")

threshold.log2fc <- 0.25
threshold.adjp <- 0.05
MG_WTvsKO.volcano = mutate(MG_WTvsKO, Sig = ifelse((abs(MG_WTvsKO$avg_log2FC) >
  threshold.log2fc) & (MG_WTvsKO$p_val_adj < threshold.adjp),
  "Sig", "n.s."))
# add two colors to 2 sig lists
MG_WTvsKO.volcano$Sig[MG_WTvsKO.volcano$avg_log2FC < -threshold.log2fc &
  MG_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "WT"
MG_WTvsKO.volcano$Sig[MG_WTvsKO.volcano$avg_log2FC > threshold.log2fc &
  MG_WTvsKO.volcano$p_val_adj < threshold.adjp] <- "KO"

MG_WTvsKO.volcano$Gene <- rownames(MG_WTvsKO.volcano)

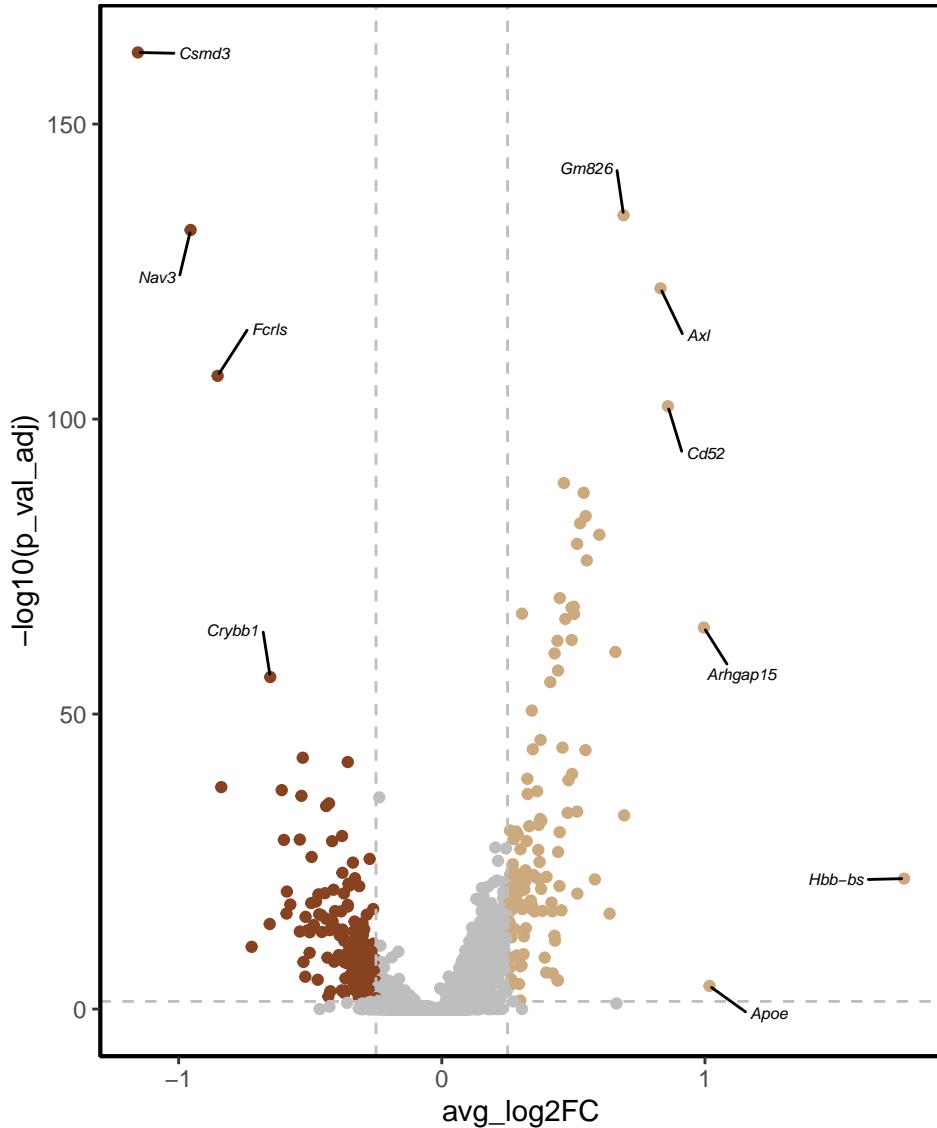
Gene.to.show.ValcanoPlot <- rownames(MG_WTvsKO.volcano[MG_WTvsKO.volcano$Sig != "n.s.", ])

axis.lim <- max(abs(MG_WTvsKO.volcano$avg_log2FC)) + 1

ggplot(MG_WTvsKO.volcano, aes(avg_log2FC, -log10(p_val_adj))) +
  geom_point(aes(col = Sig)) + scale_color_manual(values = c(WT = "#87421f",
  n.s. = "grey", KO = "#cdaa7d")) + geom_hline(yintercept = -log10(threshold.adjp),
  linetype = "dashed", col = "grey") + geom_vline(xintercept = c(-threshold.log2fc,
  threshold.log2fc), linetype = "dashed", col = "grey") + geom_text_repel(data = filter(MG_WTvsKO.volcano,
  Gene %in% Gene.to.show.ValcanoPlot), size = 2, aes(label = Gene,
  fontface = "italic"), box.padding = 1) + theme_classic() +
  theme(legend.position = "none") + theme(panel.border = element_rect(colour = "black",
  fill = NA, size = 1))

## Warning: ggrepel: 289 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



```

MG <- subset(sc, idents = c("MG WT", "MG KO"))
all.genes <- rownames(MG)
MG <- ScaleData(MG, features = all.genes)

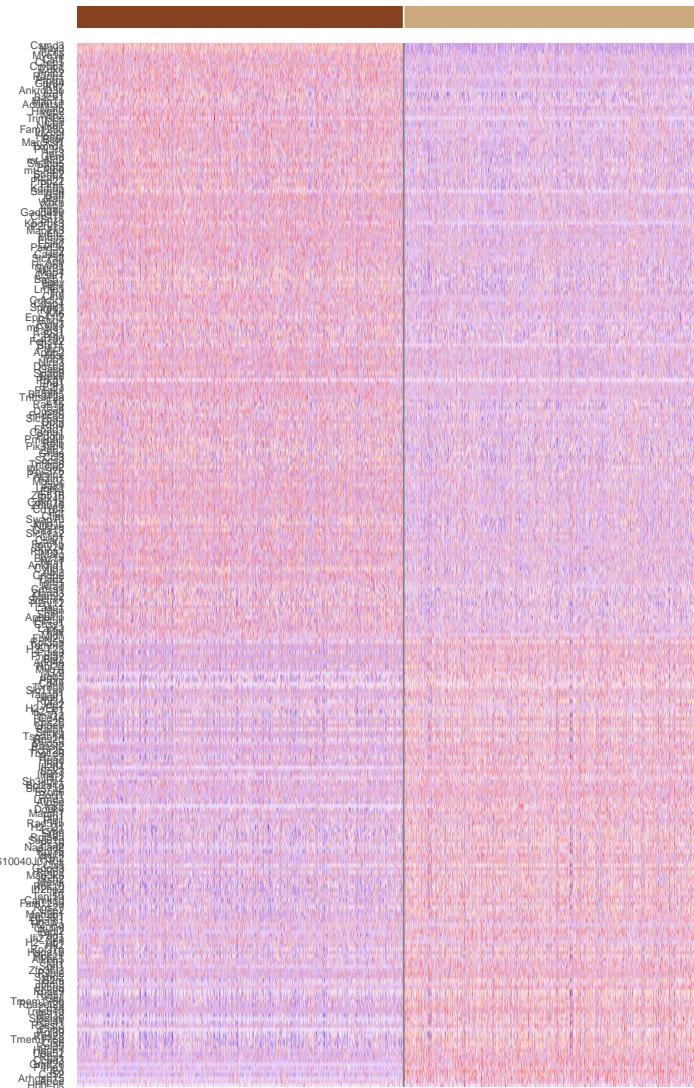
## Centering and scaling data matrix

pal_MG <- c("#87421f", "#cdaa7d")

DoHeatmap(MG, features = rev(rownames(MG_WTvsKO.markers)), group.colors = pal_MG,
  size = 0) + scale_fill_gradientn(colors = c("blue", "white",
  "red")) + theme(axis.text.y = element_text(size = 4)) + theme(legend.position = "none")

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



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] ggrepel_0.9.4      dplyr_1.1.4          RColorBrewer_1.1-3 ggplot2_3.4.4
## [5] SeuratObject_4.1.3 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        fansi_1.0.6       codetools_0.2-19
## [13] splines_4.2.3        knitr_1.45        polyclip_1.10-6
## [16] spam_2.10-0          jsonlite_1.8.8     ica_1.0-3
## [19] cluster_2.1.4        png_0.1-8         uwot_0.1.16
## [22] shiny_1.8.0          sctransform_0.4.1 spatstat.sparse_3.0-3
## [25] compiler_4.2.3       httr_1.4.7        Matrix_1.6-4
## [28] fastmap_1.1.1       lazyeval_0.2.2     cli_3.6.2
## [31] later_1.3.2          formatR_1.14       htmltools_0.5.7
## [34] tools_4.2.3          dotCall64_1.1-1     igraph_1.6.0
## [37] gtable_0.3.5         glue_1.6.2        RANN_2.6.1
## [40] reshape2_1.4.4        Rcpp_1.0.11       scattermore_1.2
## [43] vctrs_0.6.5          spatstat.explore_3.2-5 nlme_3.1-162
## [46] progressr_0.14.0     lmtest_0.9-40      spatstat.random_3.2-2
## [49] xfun_0.39            stringr_1.5.1      globals_0.16.2
## [52] mime_0.12             miniUI_0.1.1.1     lifecycle_1.0.4
## [55] irlba_2.3.5.1        goftest_1.2-3      future_1.33.1
## [58] MASS_7.3-58.2         zoo_1.8-12        scales_1.3.0
## [61] promises_1.2.1        spatstat.utils_3.0-4 parallel_4.2.3
## [64] yaml_2.3.7            reticulate_1.34.0 pbapply_1.7-2
## [67] gridExtra_2.3          stringi_1.8.3      highr_0.10
## [70] rlang_1.1.2           pkgconfig_2.0.3     matrixStats_1.2.0
## [73] evaluate_0.23         lattice_0.20-45    ROCR_1.0-11
## [76] purrrr_1.0.2          tensor_1.5        labeling_0.4.3
## [79] patchwork_1.1.3        htmlwidgets_1.6.4   cowplot_1.1.2
## [82] tidyselect_1.2.1       parallelly_1.36.0 RcppAnnoy_0.0.21
## [85] plyr_1.8.9             magrittr_2.0.3      R6_2.5.1
## [88] generics_0.1.3          withr_3.0.1        pillar_1.9.0
## [91] fitdistrplus_1.1-11    survival_3.5-3     abind_1.4-5
## [94] sp_2.1-2               tibble_3.2.1       future.apply_1.11.1
## [97] crayon_1.5.2           KernSmooth_2.23-20 utf8_1.2.4
## [100] spatstat.geom_3.2-7    plotly_4.10.3      rmarkdown_2.25
## [103] grid_4.2.3              data.table_1.14.10 digest_0.6.31
## [106] xtable_1.8-4           tidyverse_1.3.0     httpuv_1.6.13
## [109] munsell_0.5.1          viridisLite_0.4.2

```