Cell type deconvolution in colitis-associated colorectal cancer mouse model

Carlos Gallardo

12 December, 2024

```
# source and library import
source('code/helper_functions.R')

library(tidyverse)
library(magrittr)
library(Seurat)
library(SeuratDisk)
library(biomaRt)
library(DESeq2)
library(patchwork)
```

Bulk RNA-seq data

```
RNAseq <- list()

RNAseq[['counts']] <- read.table('data/bulkRNAseq/counts_15.csv',
                                 header = T,
                                 sep = ',') %>%
  tibble::column_to_rownames(var = 'X') %>%
  dplyr::rename(!!!setNames(colnames(.),
                            nm = gsub('^(.+)Aligned.+$','\\1', colnames(.))))

RNAseq[['sample_meta']] <- read.table('data/bulkRNAseq/sample_information_15.csv',
                                      header = T,
                                      sep = ';')

RNAseq[['gene_meta']] <- read.table('data/bulkRNAseq/ensembl_mmus_dec2017_annotation.tsv',
                                    header = T,
                                    sep = '\t',
                                    quote = "")
```

QC and filtering

```
# Preprocessing - Filtering zero-count genes
paste("Raw gene count:", nrow(RNAseq$counts))
```

```
## [1] "Raw gene count: 46078"
```

```
tokeep <- rowSums(RNAseq$counts) > 0
paste("Non-zero gene count:", sum(tokeep))
```

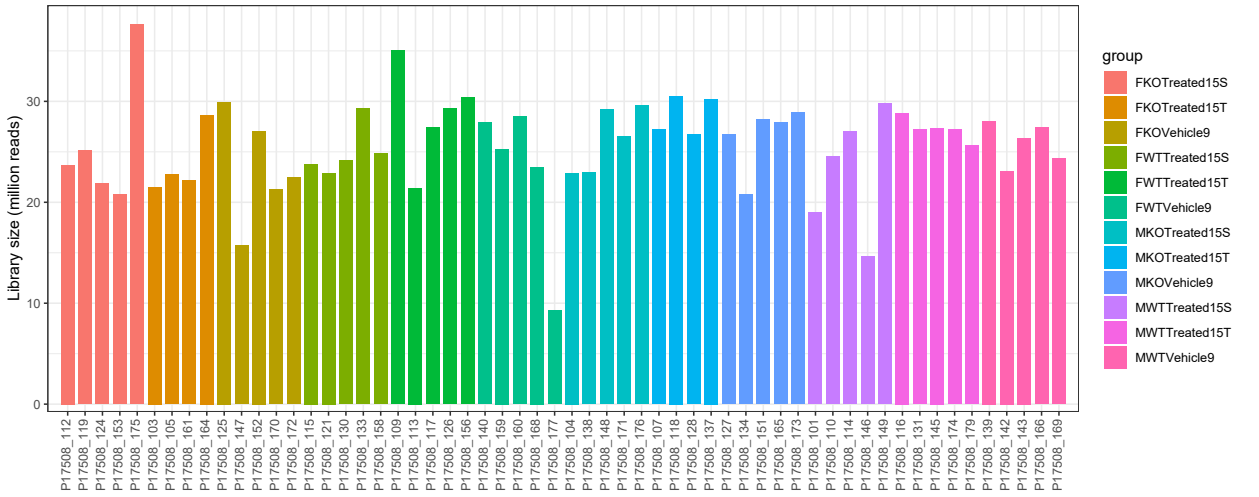
```
## [1] "Non-zero gene count: 30583"
```

```
RNAseq$counts <- RNAseq$counts[tokeep,]
RNAseq$gene_meta <- RNAseq$gene_meta[RNAseq$gene_meta$ensembl_gene_id %in% rownames(RNAseq$counts),]

rm(tokeep)
```

Only sample *P17508_177* has less than 10M counts. Belongs to vehicle wt females.

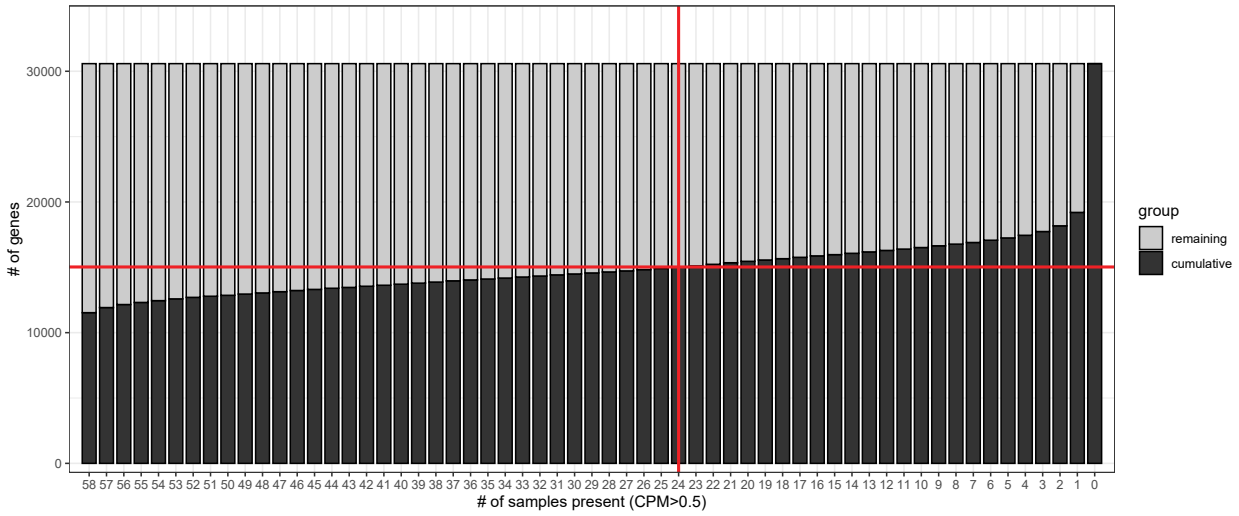
```
# library sizes
df <- RNAseq$sample_meta %>%
  mutate(lib.size = colSums(RNAseq$counts)) %>%
  mutate(sample = factor(.$sample, levels = .$sample))

ggplot(df, aes(x=sample, y=lib.size/1e6, fill=group)) +
```

```
  geom_bar(stat = "identity", width = 0.8) +
  xlab("") +
  ylab("Library size (million reads)") +
  scale_x_discrete(expand =expansion(mult = c(.02, .02))) +
  scale_y_continuous(expand =expansion(mult = c(.02, .05))) +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 90, hjust = 1, vjust = 0.3))
```



Chose expression cutoff to filter low-abundance genes. Those with more than 0.5 cpm in at least 25 samples are kept.

```
RNAseq[["cpm"]] <- normalizeData(RNAseq$counts,
                                 method = "CPM")

abovethresh <- RNAseq$cpm > 0.5

df <- data.frame(samples = factor(seq(0,nrow(RNAseq$sample_meta)),
                                  levels = rev(seq(0,nrow(RNAseq$sample_meta)))),
                 genes = c(table(rowSums(abovethresh)))) %>%
  mutate(cumulative = rev(cumsum(rev(genes)))) %>%
  mutate(remaining = sum(genes)-cumulative) %>%
  pivot_longer(cols = c("cumulative","remaining"),
               names_to = "group") %>%
  mutate(group = factor(group, levels = c("remaining", "cumulative")))

ggplot(data=df, aes(x=samples, y=value, fill=group)) +
  geom_bar(color="black", size=0.5, width=0.8, position="stack", stat="identity") +
  # geom_line(data=df, aes(x=samples, y=cumsum(genes), group=1),  size=1) +
  # geom_point(data=df, aes(x=samples, y=cumsum(genes)), shape=21, size=2, stroke=1, color="black", fill="white") +
  # geom_hline(yintercept = 20000, linetype="solid", size=1, color="grey50") +
  geom_hline(yintercept = unlist(df[df$samples == "24" & df$group == "cumulative", "value"]),
             linetype="solid", size=1, color="#EF2126") +
  geom_vline(xintercept = "24", linetype="solid", size=1, color="#EF2126") +
  xlab("# of samples present (CPM>0.5)") +
  ylab("# of genes") +
  scale_x_discrete(expand =expansion(mult = c(.02, .02))) +
  scale_y_continuous(expand =expansion(mult = c(.02, .00)),
                     limits = c(0,35000), breaks = seq(0,40000,10000)) +
  scale_fill_manual(values = c("grey80", "grey20")) +
  theme_bw()
```




```
tokeep <- rowSums(abovethresh) >= 24
paste("Pre-filtering gene count:", length(tokeep))
```

```
## [1] "Pre-filtering gene count: 30583"
```

```
paste("Genes below abundance threshold:", length(tokeep)-sum(tokeep))
```

```
## [1] "Genes below abundance threshold: 15557"
```

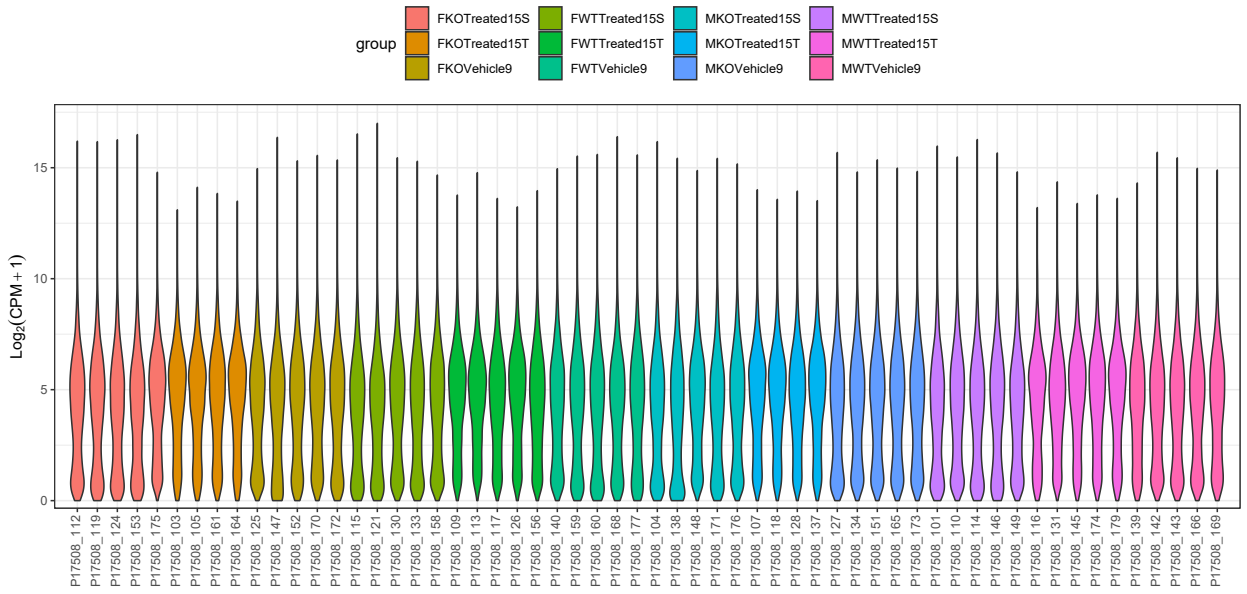
```
paste("Remaining genes:", sum(tokeep))
```

```
## [1] "Remaining genes: 15026"
```

```
# Filter genes
RNAseq$counts <- RNAseq$counts[tokeep,]
RNAseq$gene_meta <- RNAseq$gene_meta[RNAseq$gene_meta$ensembl_gene_id %in% rownames(RNAseq$counts),]

# Normalize
RNAseq$cpm <- normalizeData(RNAseq$counts,
                            method = "CPM")
```

Distributions of gene expression per sample look ok.

```
df <- RNAseq$cpm %>%
  t() %>%
  as.data.frame() %>%
  tibble::rownames_to_column(var = 'sample') %>%
  left_join(RNAseq$sample_meta, by = 'sample') %>%
  tidyr::pivot_longer(cols = c(2:(nrow(RNAseq$cpm)+1)),
                      names_to = 'geneID',
                      values_to = 'cpm') %>%
  mutate(sample = factor(.$sample, levels = RNAseq$sample_meta$sample)) %>%
  mutate(counts = RNAseq$counts %>%
           t() %>%
           as.data.frame() %>%
           tibble::rownames_to_column(var = 'sample') %>%
           left_join(RNAseq$sample_meta, by = 'sample') %>%
           tidyr::pivot_longer(cols = c(2:(nrow(RNAseq$counts)+1)),
                               names_to = 'geneID',
                               values_to = 'counts') %>%
           pull(counts))

ggplot(df, aes(x=sample, y=log2(cpm+1), fill=group)) +
  geom_violin(scale = "area") +
  xlab("") +
  ylab(expression(Log[2](CPM+1))) +
  scale_x_discrete(expand =expansion(mult = c(.02, .02))) +
  scale_y_continuous(expand =expansion(mult = c(.02, .05))) +
  theme_bw() +
  theme(
    axis.text.x.bottom = element_text(angle = 90, hjust = 1, vjust = 0.3),
    legend.position = "top")
```



Overall trascriptomes correlate well for samples from the same condition. Major differences are found between epithelial scrapes and tumor samples. However, some scrape samples match well the gene expression of tumor samples (e.g., P17508_175).

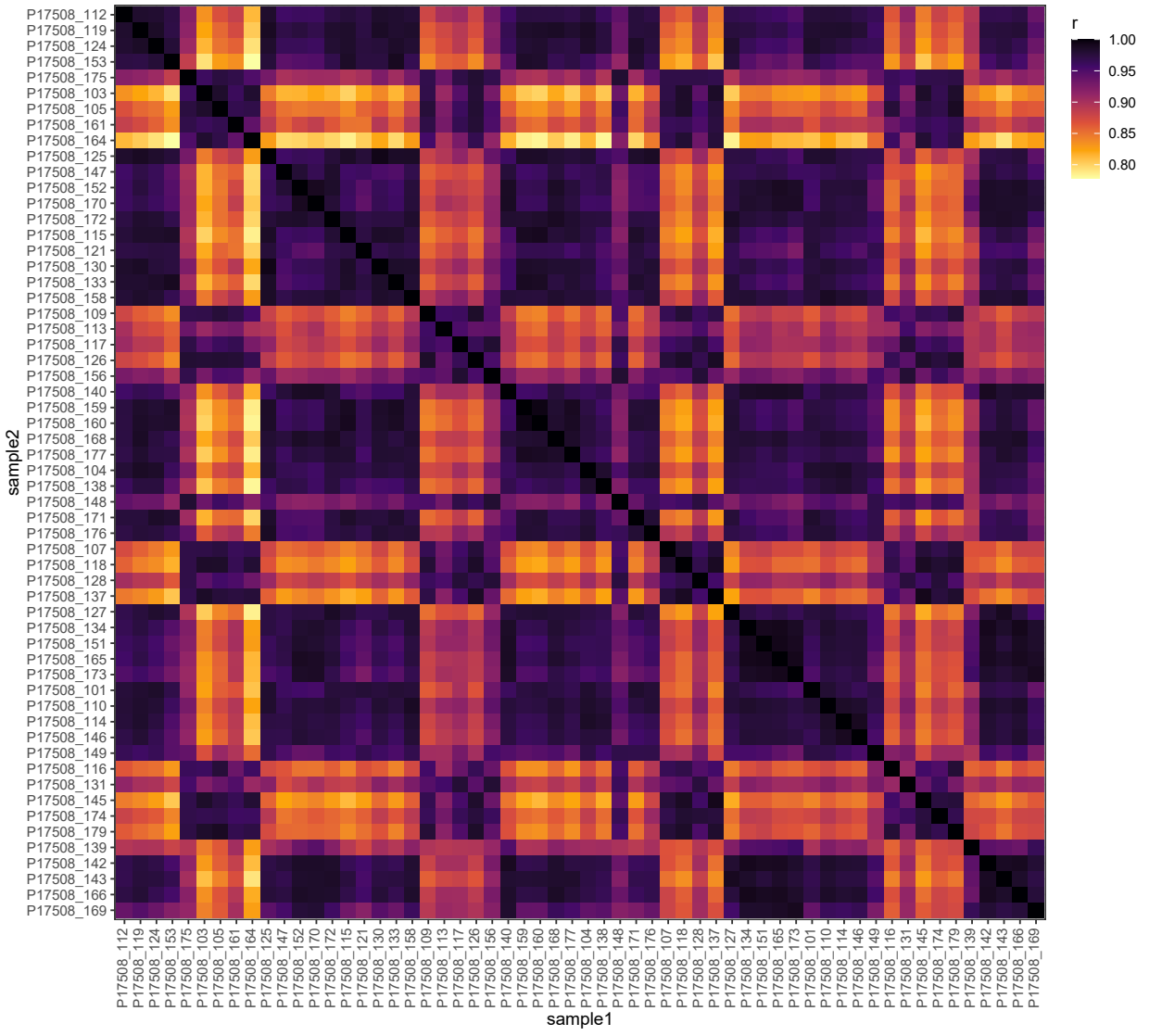
```
RNAseq$sample_meta
```

```
##        sample  animal_id         group treatment genotype    sex
## 1  P17508_112 17CW17:5 S FKOTreated15S   aom_dss       ko female
## 2  P17508_119 17CW30:2 S FKOTreated15S   aom_dss       ko female
## 3  P17508_124 17CW30:3 S FKOTreated15S   aom_dss       ko female
## 4  P17508_153  17CW9:2 S FKOTreated15S   aom_dss       ko female
## 5  P17508_175 17CW30:4 S FKOTreated15S   aom_dss       ko female
## 6  P17508_103 17CW30:3 T FKOTreated15T   aom_dss       ko female
## 7  P17508_105  17CW9:2 T FKOTreated15T   aom_dss       ko female
## 8  P17508_161 17CW17:5 T FKOTreated15T   aom_dss       ko female
## 9  P17508_164 17CW30:4 T FKOTreated15T   aom_dss       ko female
## 10 P17508_125   18CW28:1   FKOVehicle9   vehicle       ko female
## 11 P17508_147   18CW28:3   FKOVehicle9   vehicle       ko female
## 12 P17508_152   17CW52:1   FKOVehicle9   vehicle       ko female
## 13 P17508_170   17CW52:3   FKOVehicle9   vehicle       ko female
## 14 P17508_172    18CW5:4   FKOVehicle9   vehicle       ko female
## 15 P17508_115 17CW28:4 S FWTTreated15S   aom_dss       wt female
## 16 P17508_121  17CW9:1 S FWTTreated15S   aom_dss       wt female
## 17 P17508_130 17CW28:1 S FWTTreated15S   aom_dss       wt female
## 18 P17508_133  17CW9:3 S FWTTreated15S   aom_dss       wt female
## 19 P17508_158 17CW19:1 S FWTTreated15S   aom_dss       wt female
## 20 P17508_109  17CW9:1 T FWTTreated15T   aom_dss       wt female
## 21 P17508_113  17CW9:3 T FWTTreated15T   aom_dss       wt female
## 22 P17508_117 17CW28:1 T FWTTreated15T   aom_dss       wt female
## 23 P17508_126 17CW19:1 T FWTTreated15T   aom_dss       wt female
## 24 P17508_156 17CW28:4 T FWTTreated15T   aom_dss       wt female
## 25 P17508_140   17CW52:4   FWTVehicle9   vehicle       wt female
## 26 P17508_159   17CW21:4   FWTVehicle9   vehicle       wt female
## 27 P17508_160   17CW20:2   FWTVehicle9   vehicle       wt female
## 28 P17508_168    18CW5:1   FWTVehicle9   vehicle       wt female
## 29 P17508_177   17CW15:1   FWTVehicle9   vehicle       wt female
## 30 P17508_104 17CW21:7 S MKOTreated15S   aom_dss       ko   male
## 31 P17508_138 17CW19:8 S MKOTreated15S   aom_dss       ko   male
## 32 P17508_148 17CW14:4 S MKOTreated15S   aom_dss       ko   male
## 33 P17508_171 17CW14:7 S MKOTreated15S   aom_dss       ko   male
## 34 P17508_176 17CW19:7 S MKOTreated15S   aom_dss       ko   male
## 35 P17508_107 17CW14:7 T MKOTreated15T   aom_dss       ko   male
## 36 P17508_118 17CW14:4 T MKOTreated15T   aom_dss       ko   male
## 37 P17508_128 17CW19:8 T MKOTreated15T   aom_dss       ko   male
## 38 P17508_137 17CW19:7 T MKOTreated15T   aom_dss       ko   male
## 39 P17508_127   18CW12:6   MKOVehicle9   vehicle       ko   male
## 40 P17508_134   18CW21:2   MKOVehicle9   vehicle       ko   male
## 41 P17508_151   17CW48:4   MKOVehicle9   vehicle       ko   male
## 42 P17508_165   17CW48:2   MKOVehicle9   vehicle       ko   male
```

```
## 43 P17508_173   17CW47:7   MKOVehicle9   vehicle       ko   male
## 44 P17508_101 17CW11:9 S MWTTreated15S   aom_dss       wt   male
## 45 P17508_110 17CW11:7 S MWTTreated15S   aom_dss       wt   male
## 46 P17508_114 17CW21:6 S MWTTreated15S   aom_dss       wt   male
## 47 P17508_146 17CW21:8 S MWTTreated15S   aom_dss       wt   male
## 48 P17508_149 17CW14:6 S MWTTreated15S   aom_dss       wt   male
## 49 P17508_116 17CW11:7 T MWTTreated15T   aom_dss       wt   male
## 50 P17508_131 17CW21:8 T MWTTreated15T   aom_dss       wt   male
## 51 P17508_145 17CW11:9 T MWTTreated15T   aom_dss       wt   male
## 52 P17508_174 17CW21:6 T MWTTreated15T   aom_dss       wt   male
## 53 P17508_179 17CW14:6 T MWTTreated15T   aom_dss       wt   male
## 54 P17508_139   18CW28:7   MWTVehicle9   vehicle       wt   male
## 55 P17508_142   18CW28:4   MWTVehicle9   vehicle       wt   male
## 56 P17508_143   18CW12:4   MWTVehicle9   vehicle       wt   male
## 57 P17508_166   18CW25:6   MWTVehicle9   vehicle       wt   male
## 58 P17508_169    18CW7:7   MWTVehicle9   vehicle       wt   male
##               tissue
## 1  epithelial_scrape
## 2  epithelial_scrape
## 3  epithelial_scrape
## 4  epithelial_scrape
## 5  epithelial_scrape
## 6              tumor
## 7              tumor
## 8              tumor
## 9              tumor
## 10 epithelial_scrape
## 11 epithelial_scrape
## 12 epithelial_scrape
## 13 epithelial_scrape
## 14 epithelial_scrape
## 15 epithelial_scrape
## 16 epithelial_scrape
## 17 epithelial_scrape
## 18 epithelial_scrape
## 19 epithelial_scrape
## 20             tumor
## 21             tumor
## 22             tumor
## 23             tumor
## 24             tumor
## 25 epithelial_scrape
## 26 epithelial_scrape
## 27 epithelial_scrape
## 28 epithelial_scrape
## 29 epithelial_scrape
## 30 epithelial_scrape
## 31 epithelial_scrape
## 32 epithelial_scrape
## 33 epithelial_scrape
## 34 epithelial_scrape
## 35             tumor
## 36             tumor
## 37             tumor
## 38             tumor
## 39 epithelial_scrape
## 40 epithelial_scrape
## 41 epithelial_scrape
## 42 epithelial_scrape
## 43 epithelial_scrape
## 44 epithelial_scrape
## 45 epithelial_scrape
## 46 epithelial_scrape
## 47 epithelial_scrape
## 48 epithelial_scrape
## 49             tumor
## 50             tumor
## 51             tumor
## 52             tumor
## 53             tumor
## 54 epithelial_scrape
## 55 epithelial_scrape
## 56 epithelial_scrape
## 57 epithelial_scrape
## 58 epithelial_scrape
```

```
df <- cor(RNAseq$cpm, method = "spearman") %>%
  as.data.frame() %>%
  rownames_to_column(var = "sample1") %>%
  mutate(across(everything(), as.character)) %>%
  pivot_longer(cols = c(2:length(.)),
```

```
              names_to = "sample2") %>%
  dplyr::rename(r = value) %>%
  mutate(sample1 = factor(sample1, levels = names(RNAseq$counts)),
         sample2 = factor(sample2, levels = names(RNAseq$counts)),
         r = as.numeric(r))

ggplot(df, aes(x=sample1, y=sample2, fill= r)) +
  geom_tile() +
  scale_y_discrete(limits=rev) +
  scale_fill_gradientn(colours = alpha(rev(c("#000004", "#420A68", "#932667", "#DD513A", "#FCA50A", "#FCFFA4")), 1)) +
  theme_bw() +
  theme(
    axis.text.x.bottom = element_text(angle = 90, hjust = 1, vjust = 0.3),
    legend.position = "right",
    legend.justification = "top"
  )
```



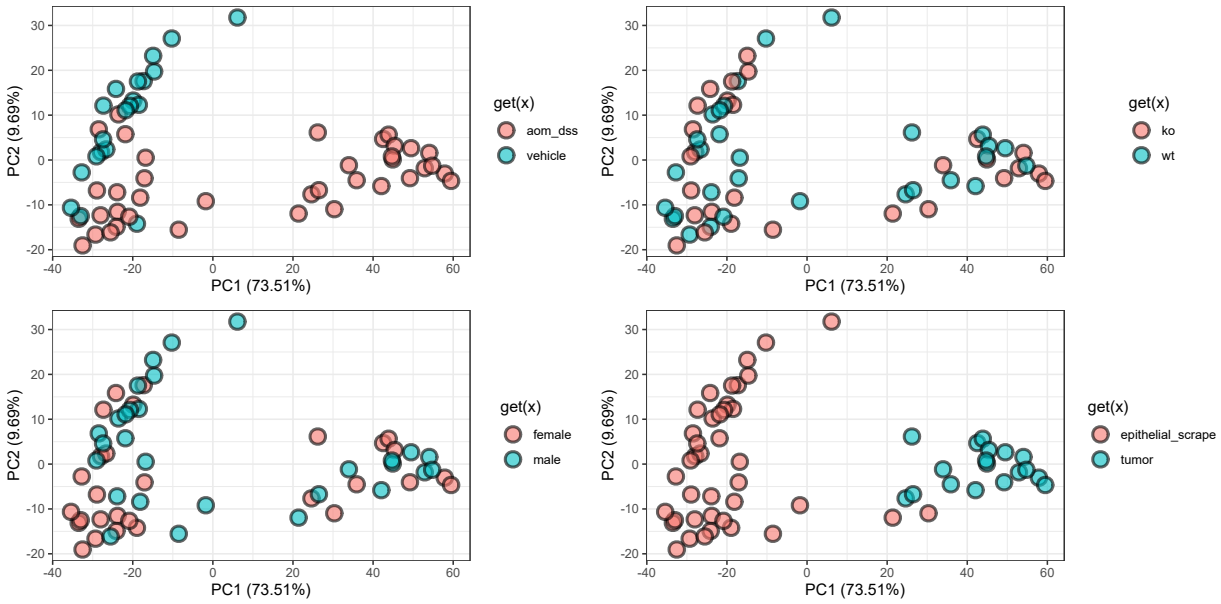
PCA plots

```
dsdata <- DESeqDataSetFromMatrix(countData = RNAseq$counts,
                                 colData = RNAseq$sample_meta,
                                 design = ~treatment + genotype + sex + tissue)
dsdata <- estimateSizeFactors(dsdata)
```

```
RNAseq[["DESeq_norm"]] <- counts(dsdata, normalized=TRUE) %>% as.data.frame()
RNAseq[["DESeq_vst"]] <- assay(vst(dsdata, blind=FALSE)) %>% as.data.frame()
RNAseq[["DESeq_rlog"]] <- assay(rlog(dsdata, blind=FALSE)) %>% as.data.frame()
```

From PCA plots below we see that PC1 (~74% variation) seppparates samples well into two groups based on if they are from tumors or epithelial scrapes. PC2 (~10% variation) can separate vehicle from AOM-DSS treated conditions.

```
pca_res <- doPCA(RNAseq$DESeq_vst, top_var = 500)
df <- pca_res$pcs %>%
  dplyr::bind_cols(RNAseq$sample_meta)

p <- setNames(c('treatment','genotype','sex','tissue'),
              c('treatment','genotype','sex','tissue')) %>%
  as.list()

p <- lapply(p, function(x){
  ggplot(df, aes(x=PC1, y=PC2, fill=get(x))) +
    geom_point(color="black", shape=21, size=4, stroke=1.5, alpha=0.6) +
    xlab(paste0('PC1 (',round(pca_res$perc_var[[1]],2),'%)')) +
    ylab(paste0('PC2 (',round(pca_res$perc_var[[2]],2),'%)')) +
    theme_bw()
})

patchwork::wrap_plots(p, ncol = 2)
```



Genes associated to PC1:

```
# reports that Cyp2c55 protects in colon cancer
# Mmp7, Nkd1 are tumor markers
df <- as.data.frame(pca_res$rotation) %>%
  rownames_to_column(var = "GeneSymbol") %>%
  dplyr::mutate(GeneSymbol = dplyr::recode(GeneSymbol,
                                           !!!setNames(RNAseq$gene_meta$external_gene_name,
                                                       RNAseq$gene_meta$ensembl_gene_id))) %>%
  dplyr::select(GeneSymbol, PC1) %>%
  mutate(sign = sign(PC1)) %>%
  mutate(PC1 = abs(PC1)) %>%
  arrange(desc(PC1))

head(df, 20)
```

```
##       GeneSymbol        PC1 sign
## 1          Spock2 0.09910813    1
## 2         Cyp2c55 0.09199182   -1
## 3            Lyz1 0.08635332    1
## 4            Mmp7 0.08380172    1
```

```
## 5           Nkd1 0.08194418    1
## 6           Lcn2 0.08089800    1
## 7          Notum 0.07858547    1
## 8           Wif1 0.07793118    1
## 9          Mmp10 0.07769841    1
## 10         Stra6 0.07688544    1
## 11          Plat 0.07665341    1
## 12       Aldh1a3 0.07446336    1
## 13 1810065E05Rik 0.07444636   -1
## 14        Clca4b 0.07440138    1
## 15           Mgp 0.07432466    1
## 16         Mmp13 0.07284021    1
## 17         Cxcl5 0.07278473    1
## 18        Apcdd1 0.07250168    1
## 19        Igfbp5 0.07249857    1
## 20         Prox1 0.07180640    1
```

Genes associated to PC2:

```
# Reg3b promotes proliferation
# Tgm3 is a tumor supressor (epithelial-to-mesenchymal transition and PI3K/AKT signaling)
df <- as.data.frame(pca_res$rotation) %>%
  rownames_to_column(var = "GeneSymbol") %>%
  dplyr::mutate(GeneSymbol = dplyr::recode(GeneSymbol,
                                           !!!setNames(RNAseq$gene_meta$external_gene_name,
                                                       RNAseq$gene_meta$ensembl_gene_id))) %>%
  dplyr::select(GeneSymbol, PC2) %>%
  mutate(sign = sign(PC2)) %>%
  mutate(PC2 = abs(PC2)) %>%
  arrange(desc(PC2))

head(df, 20)
```

```
##       GeneSymbol       PC2 sign
## 1           Reg3b 0.2054179   -1
## 2         Slc37a2 0.1620160    1
## 3           Reg3g 0.1605219   -1
## 4            Tgm3 0.1391366    1
## 5          Hoxd13 0.1317297    1
## 6          Slc51a 0.1263808   -1
## 7          Atp12a 0.1239607    1
## 8            Osr2 0.1172595   -1
## 9        AI854703 0.1161414    1
## 10          Plet1 0.1124364   -1
## 11          Mptx1 0.1114202    1
## 12          Fxyd4 0.1107029    1
## 13            Nts 0.1101046   -1
## 14 1810065E05Rik 0.1076971   -1
## 15         B3gnt7 0.1071323    1
## 16         Abca12 0.1064869   -1
## 17           Sycn 0.1056855    1
## 18          Prap1 0.1041179   -1
## 19           Xist 0.1040288   -1
## 20         Clca4b 0.1027284   -1
```

Single cell RNA-seq data

Matching single cell data from Vega et al. 2022 with annotations obtained from authors. Sample conditions include:

- Wild type
- APC tumor ($Lrig1^{CreERT2/+}; Apc^{fl/+}$)
- APC adjacent
- AOM/DSS tumor (inflammation-driven)

Whole dataset contains 12804 cells across the different conditions.

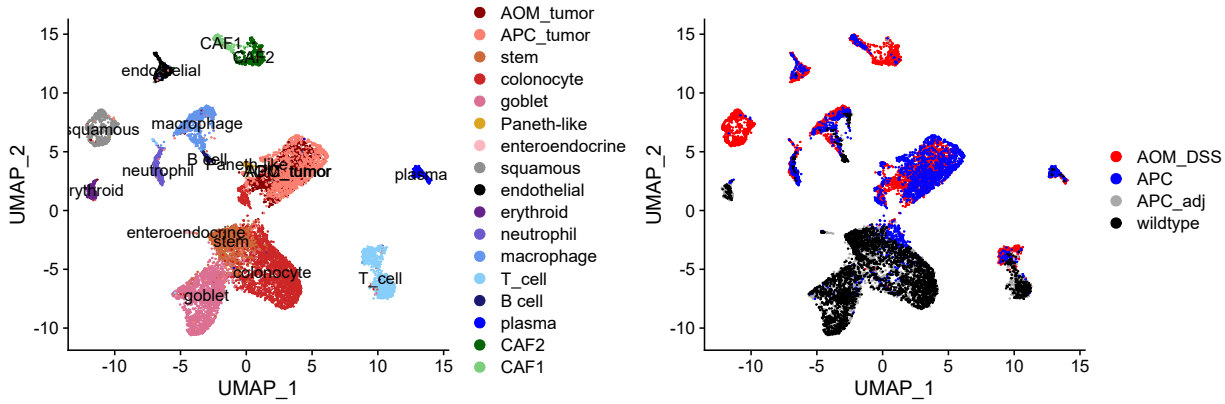
```
# load h5 object provided by authors
sc_dat <- LoadH5Seurat('data/scRNAseq/Vega_et_al_2022_CAF.h5Seurat')

# add cell type annotation as indicated by authors
cluster_number <- 1:17
cell_annotation <- c('macrophage', 'CAF2', 'CAF1', 'AOM_tumor',
                     'Paneth-like', 'squamous', 'stem', 'T_cell',
```

```
                    'APC_tumor', 'plasma', 'endothelial', 'neutrophil',
                    'erythroid', 'colonocyte', 'goblet', 'B cell',
                    'enteroendocrine')
sc_dat@meta.data <- sc_dat@meta.data %>%
  dplyr::mutate(cl_num = sc_dat@active.ident) %>%
  dplyr::mutate(cell_annot = dplyr::recode(cl_num,
                                           !!!setNames(cell_annotation,
                                                       cluster_number)))

sc_dat
```

```
## An object of class Seurat
## 38457 features across 12804 samples within 1 assay
## Active assay: RNA (38457 features, 2000 variable features)
##  2 dimensional reductions calculated: pca, umap
```

UMAP plots found in publication are reproducible.

```
p <- list()

# umap with clusters by color
p[[1]] <- DimPlot(object = sc_dat,
                  reduction = "umap",
                  cols = c( 'darkred', 'salmon', 'sienna3', 'firebrick3', 'palevioletred',
                            'goldenrod', "lightpink", 'grey56', 'black', 'darkorchid4',
                            'slateblue', 'cornflowerblue', 'lightskyblue','midnightblue',
                            'blue', 'darkgreen', 'palegreen3'),
                  group.by = "cell_annot",
                  label = TRUE) +
  labs(title = NULL)

# umap with sample type by color
p[[2]] <- DimPlot(object = sc_dat,
                  reduction = "umap",
                  group.by = "orig.ident",
                  cols = c("red", "blue", "darkgrey", "black")) +
  labs(title = NULL)

patchwork::wrap_plots(p, ncol = 2)
```

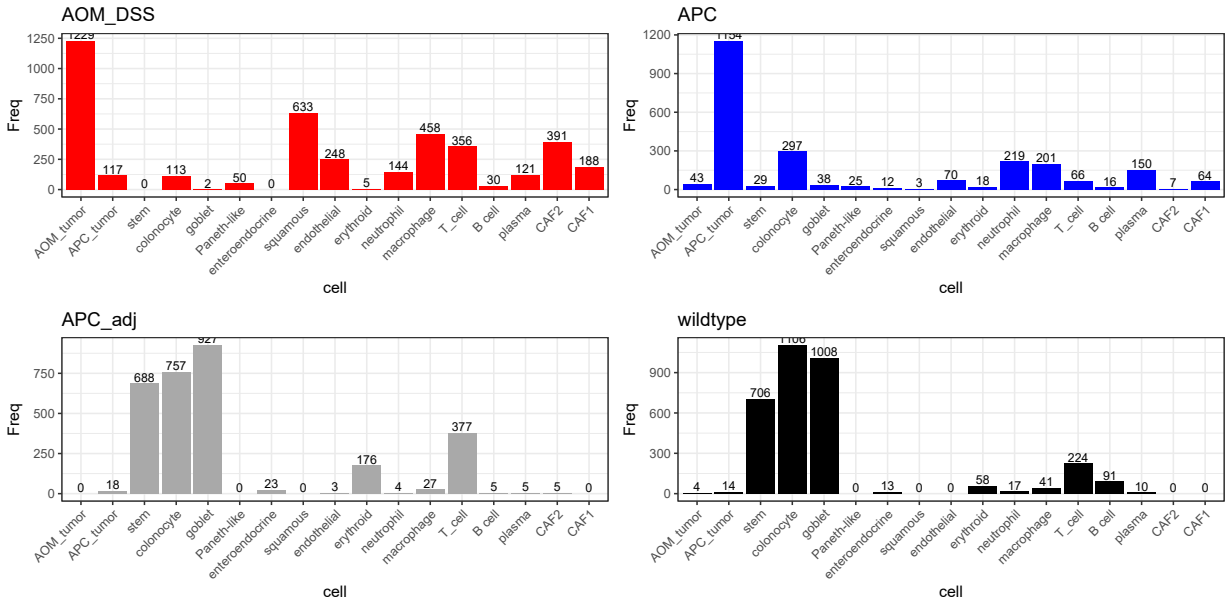


Cell distribution across conditions

From all available, cells from wildtype and AOM_DSS samples are relevant for us. Cells derived from the APC model will be excluded and will not used for deconvolution.

Ideally, we would build **gene expression signatures** that are specific to cell types in each condition (i.e., vehicle- and aom-dss-treated) and use these for the deconvulution of matching bulk RNA-seq samples. However, the abundance distribution of cell types looks very different in these two conditions (see top left and bottom right bar plots below). This would not be a problem if more cells were available in this dataset but it is here because some cell types end up with very low counts (affects building a robust signature) or none at all. We proceed to build the signature on the combined dataset for enhanced representation and robustness.

```r
p <- list()

df <- sc_dat@meta.data %>%
  filter(orig.ident == 'AOM_DSS') %>%
  pull(cell_annot) %>%
  table() %>%
  as.data.frame()

colnames(df) <- c('cell', 'Freq')

p[[1]] <- ggplot(df, aes(x=cell, y=Freq, label=Freq)) +
  geom_bar(stat='identity', fill='red') +
  geom_text(size=3, vjust=-0.2) +
  ggtitle('AOM_DSS') +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 45, hjust = 1, vjust = 1))

df <- sc_dat@meta.data %>%
  filter(orig.ident == 'APC') %>%
  pull(cell_annot) %>%
  table() %>%
  as.data.frame()

colnames(df) <- c('cell', 'Freq')

p[[2]] <- ggplot(df, aes(x=cell, y=Freq, label=Freq)) +
  geom_bar(stat='identity', fill='blue') +
  geom_text(size=3, vjust=-0.2) +
  ggtitle('APC') +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 45, hjust = 1, vjust = 1))

df <- sc_dat@meta.data %>%
  filter(orig.ident == 'APC_adj') %>%
  pull(cell_annot) %>%
  table() %>%
  as.data.frame()

colnames(df) <- c('cell', 'Freq')

p[[3]] <- ggplot(df, aes(x=cell, y=Freq, label=Freq)) +
  geom_bar(stat='identity', fill='darkgrey') +
  geom_text(size=3, vjust=-0.2) +
  ggtitle('APC_adj') +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 45, hjust = 1, vjust = 1))

df <- sc_dat@meta.data %>%
  filter(orig.ident == 'wildtype') %>%
  pull(cell_annot) %>%
  table() %>%
  as.data.frame()

colnames(df) <- c('cell', 'Freq')

p[[4]] <- ggplot(df, aes(x=cell, y=Freq, label=Freq)) +
  geom_bar(stat='identity', fill='black') +
  geom_text(size=3, vjust=-0.2) +
  ggtitle('wildtype') +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 45, hjust = 1, vjust = 1))

patchwork::wrap_plots(p, ncol = 2)
```



Cell type deconvolution

For the deconvolution, we apply the CIBERSORTx method (available at https://cibersortx.stanford.edu/). For information on the method check the publication Newman et al. 2019.

Export input data for CIBERSORTx

Single-cell count matrix (wildtype and AOM-DSS cells)

```
dat <- sc_dat@assays$RNA@counts
dat <- GetAssayData(sc_dat, assay = "RNA", slot = "counts")
df <- sc_dat@meta.data %>%
  tibble::rownames_to_column(var = 'cell_id') %>%
  dplyr::filter(orig.ident %in% c('AOM_DSS','wildtype')) %>%
  dplyr::filter(cell_annot != 'APC_tumor') %>%
  dplyr::mutate(cell_annot = dplyr::recode(cell_annot, 'CAF1' = 'CAF', 'CAF2' = 'CAF')) %>%
  dplyr::mutate(cell_annot = as.character(cell_annot))

features <- intersect(rownames(dat),
                      RNAseq$gene_meta$external_gene_name)

dat <- dat[features, df$cell_id] %>%
  as.matrix() %>%
  as.data.frame() %>%
  rownames_to_column(var = 'geneID')

# write.table(dat,
#             file="data/CIBERSORTx/CIBERSORTx_single_cell_Vega2022_counts.txt",
#             row.names=FALSE,
#             col.names=c('geneID', df$cell_annot),
#             sep="\t",
#             quote = FALSE)
```

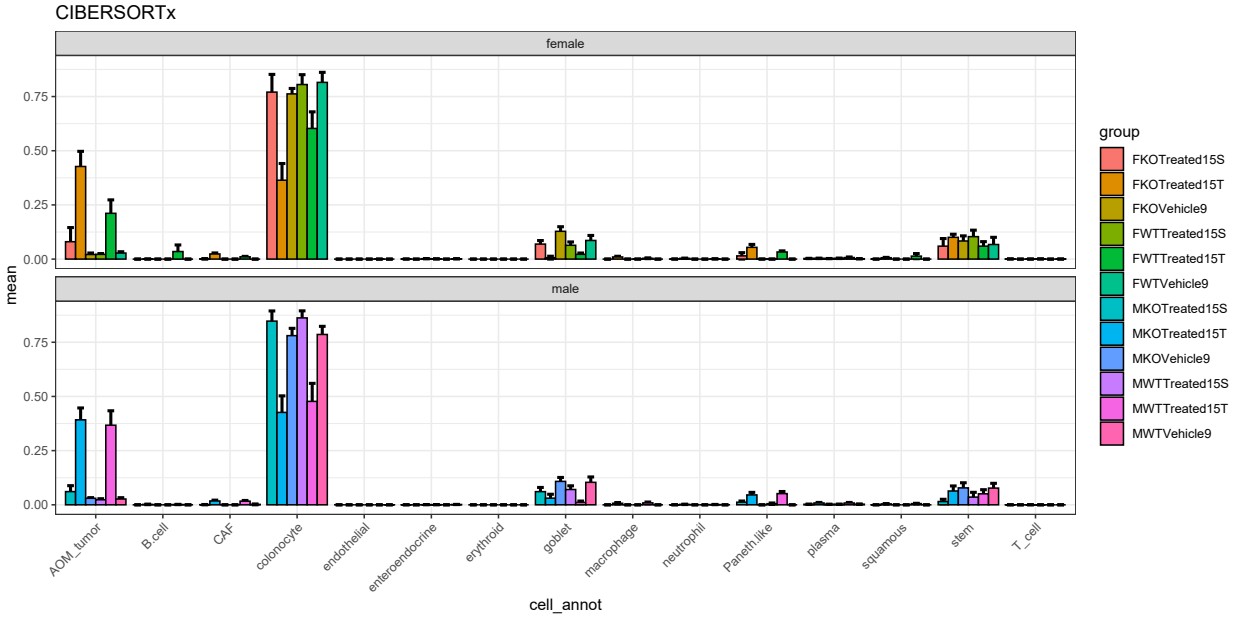
Bulk RNA-seq count matrix

```
df <- RNAseq$counts %>%
  as.data.frame() %>%
  rownames_to_column(var = 'geneID') %>%
  dplyr::mutate(geneID = dplyr::recode(geneID,
                                       !!!setNames(RNAseq$gene_meta$external_gene_name,
                                                   RNAseq$gene_meta$ensembl_gene_id))) %>%
  dplyr::filter(!duplicated(geneID))
```

```
# write.table(df,
#             file="data/CIBERSORTx/CIBERSORTx_bulkRNAseq_counts.txt",
#             row.names=FALSE,
#             col.names=TRUE,
#             sep="\t",
#             quote = FALSE)
```

CIBERSORTx deconvolution results

```
df <- read.table(file = 'results/CIBERSORTx_deconvolution_cell_type_fractions.txt',
                 header = TRUE,
                 sep = '\t',
                 quote = '') %>%
  column_to_rownames(var = 'Mixture') %>%
  round(4)

df <- cbind(RNAseq$sample_meta, df)

celltypes <- sc_dat@meta.data %>%
  dplyr::filter(orig.ident %in% c('AOM_DSS','wildtype')) %>%
  dplyr::filter(cell_annot != 'APC_tumor') %>%
  dplyr::mutate(cell_annot = dplyr::recode(cell_annot, 'CAF1' = 'CAF', 'CAF2' = 'CAF',
                                           'Paneth-like' = 'Paneth.like', 'B cell' = 'B.cell')) %>%
  dplyr::mutate(cell_annot = as.character(cell_annot)) %>%
  dplyr::pull(cell_annot) %>%
  unique()

df2 <- lapply(setNames(celltypes, nm = celltypes), function(x){
  df %>%
    group_by(group) %>%
    dplyr::select(group, x) %>%
    summarize_each(funs(mean, sd, se=sd(.)/sqrt(n())), x) %>%
    mutate(cell_annot = x)
}) %>%
  bind_rows() %>%
  left_join(df %>%
              dplyr::select(group, treatment, genotype, sex, tissue) %>%
              dplyr::filter(!duplicated(group)),
            by = 'group')

ggplot(df2, aes(x=cell_annot, y=mean, fill=group, ymin=mean-se, ymax=mean+se)) +
  geom_errorbar(width=.6, lwd=1, position = position_dodge(width=0.9)) +
  geom_bar(stat='identity', position = position_dodge(), col='black') +
  facet_wrap(~sex, ncol=1) +
  ggtitle('CIBERSORTx') +
  theme_bw() +
  theme(axis.text.x.bottom = element_text(angle = 45, hjust = 1, vjust = 1))
```



SessionInfo

```
sessionInfo()
```

```
## R version 4.2.1 (2022-06-23 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] patchwork_1.1.2             DESeq2_1.38.1
##  [3] SummarizedExperiment_1.28.0 Biobase_2.58.0
##  [5] MatrixGenerics_1.10.0       matrixStats_0.63.0
##  [7] GenomicRanges_1.50.1        GenomeInfoDb_1.34.9
##  [9] IRanges_2.32.0              S4Vectors_0.36.0
## [11] BiocGenerics_0.44.0         biomaRt_2.54.1
## [13] SeuratDisk_0.0.0.9020       SeuratObject_4.1.3
## [15] Seurat_4.3.0.1              magrittr_2.0.3
## [17] lubridate_1.9.2             forcats_1.0.0
## [19] stringr_1.5.0               dplyr_1.1.0
## [21] purrr_1.0.1                 readr_2.1.4
## [23] tidyr_1.3.0                 tibble_3.2.1
## [25] ggplot2_3.4.2               tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##   [1] BiocFileCache_2.6.1    plyr_1.8.8             igraph_1.3.5
##   [4] lazyeval_0.2.2         sp_2.0-0               splines_4.2.1
##   [7] BiocParallel_1.32.3    listenv_0.9.0          scattermore_1.2
##  [10] digest_0.6.30          htmltools_0.5.5        fansi_1.0.3
##  [13] memoise_2.0.1          tensor_1.5             cluster_2.1.4
##  [16] ROCR_1.0-11            tzdb_0.4.0             annotate_1.76.0
##  [19] globals_0.16.2         Biostrings_2.66.0      timechange_0.2.0
##  [22] spatstat.sparse_3.0-2  prettyunits_1.1.1      colorspace_2.0-3
##  [25] rappdirs_0.3.3         blob_1.2.4             ggrepel_0.9.2
##  [28] xfun_0.39              RCurl_1.98-1.9         crayon_1.5.2
##  [31] jsonlite_1.7.2         progressr_0.13.0       spatstat.data_3.0-1
##  [34] survival_3.5-5         zoo_1.8-12             glue_1.6.2
##  [37] polyclip_1.10-4        gtable_0.3.3           zlibbioc_1.44.0
##  [40] XVector_0.38.0         leiden_0.4.3           DelayedArray_0.24.0
##  [43] future.apply_1.11.0    abind_1.4-5            scales_1.2.1
##  [46] DBI_1.1.3              spatstat.random_3.1-5  miniUI_0.1.1.1
##  [49] Rcpp_1.0.9             progress_1.2.2         viridisLite_0.4.2
##  [52] xtable_1.8-4           reticulate_1.18        bit_4.0.5
##  [55] htmlwidgets_1.6.2      httr_1.4.6             RColorBrewer_1.1-3
##  [58] ellipsis_0.3.2         ica_1.0-3              farver_2.1.1
##  [61] pkgconfig_2.0.3        XML_3.99-0.12          dbplyr_2.3.3
##  [64] uwot_0.1.16            deldir_1.0-9           locfit_1.5-9.6
##  [67] utf8_1.2.2             labeling_0.4.2         tidyselect_1.2.0
##  [70] rlang_1.1.1            reshape2_1.4.4         later_1.3.1
##  [73] AnnotationDbi_1.60.2   munsell_0.5.0          tools_4.2.1
##  [76] cachem_1.0.6           cli_3.4.1              generics_0.1.3
##  [79] RSQLite_2.2.19         ggridges_0.5.4         evaluate_0.21
##  [82] fastmap_1.1.0          yaml_2.3.7             goftest_1.2-3
##  [85] knitr_1.43             bit64_4.0.5            fitdistrplus_1.1-11
##  [88] RANN_2.6.1             KEGGREST_1.38.0        pbapply_1.7-2
##  [91] future_1.33.0          nlme_3.1-162           mime_0.12
##  [94] xml2_1.3.5             hdf5r_1.3.8            compiler_4.2.1
##  [97] rstudioapi_0.15.0      filelock_1.0.2         curl_5.0.1
## [100] plotly_4.10.2          png_0.1-8              spatstat.utils_3.0-3
## [103] geneplotter_1.76.0     stringi_1.7.8          highr_0.10
## [106] lattice_0.20-41        Matrix_1.5-3           vctrs_0.6.2
## [109] pillar_1.9.0           lifecycle_1.0.3        spatstat.geom_3.2-2
## [112] lmtest_0.9-40          RcppAnnoy_0.0.21       bitops_1.0-7
## [115] data.table_1.14.6      cowplot_1.1.1          irlba_2.3.5.1
## [118] httpuv_1.6.11          R6_2.5.1               promises_1.2.0.1
## [121] KernSmooth_2.23-22     gridExtra_2.3          parallelly_1.36.0
## [124] codetools_0.2-19       MASS_7.3-60            withr_2.5.0
## [127] sctransform_0.3.5      GenomeInfoDbData_1.2.9 parallel_4.2.1
```

```
## [130] hms_1.1.3              grid_4.2.1             rmarkdown_2.23
## [133] Rtsne_0.16             spatstat.explore_3.2-1 shiny_1.7.4.1
```