

4-Functional characterization of clustered populations

BAI Qiang*

2022-03-25 17:20:17 +0100

Contents

1 Description	2
2 Load data and packages	2
3 Distribution of cells in clusters	2
4 Expression of macrophage markers and Cluster 1/3/4 signatures	4
5 Expression of cluster 2 signature	5
6 Focus on Cluster 3 and recluster	8
6.1 subsetdata	8
6.2 Processing data (only cluster 3)	9
6.3 Re-cluster the Cluster 3	10
6.4 Statistic summary about the subpopulations in Cluster 3 (Monocyte-derived AM)	11
6.5 Functional markers in subpopulations of Cluster 3 (Monocyte-derived AM)	14
7 Compare to Mould et al. 2020	16
8 Find DE genes in subpopulations of the cluster 3 (Monocyte-derived AM)	18
9 Scoring of Monocyte-derived AM and AM signatures	20
9.1 Create signature from DE gene lists	20
9.2 Scoring for Monocyte-derived AM and AM signatures	20
9.3 Present signature score with existing embedding in Seurat object	20
10 Session information	27
References	29

*University Liege, mail qiang.bai@uliege.be

1 Description

Functional characterization of the cluster 1, 2, 3 and 4 by examining expression of deferentially expressing (DE) genes and functional markers with Seurat package[1].

To compare bulk and scRNA-seq data, AFlo and AFhi AM signatures were calculated by comparing AFlo AM samples with AFhi AM samples using DESeq2 package. A threshold of P adjusted < 0.05 and a biological FC > 2 was applied to obtain the signatures. For each cell, a score of signatures was calculated with VISION package [2] and the scores were presented using Seurat FeaturePlot function with the same embedding as in Figure 4A.

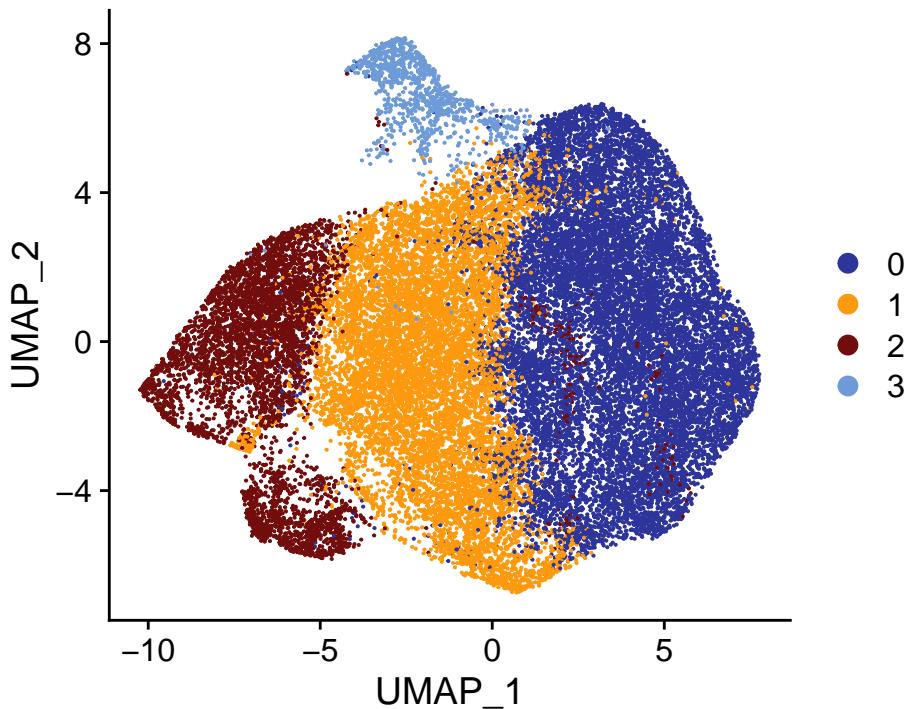
Notice: The cluster 1/2/3/4 in the manuscript refers to cluster 0/1/2/3 in the following codes.

2 Load data and packages

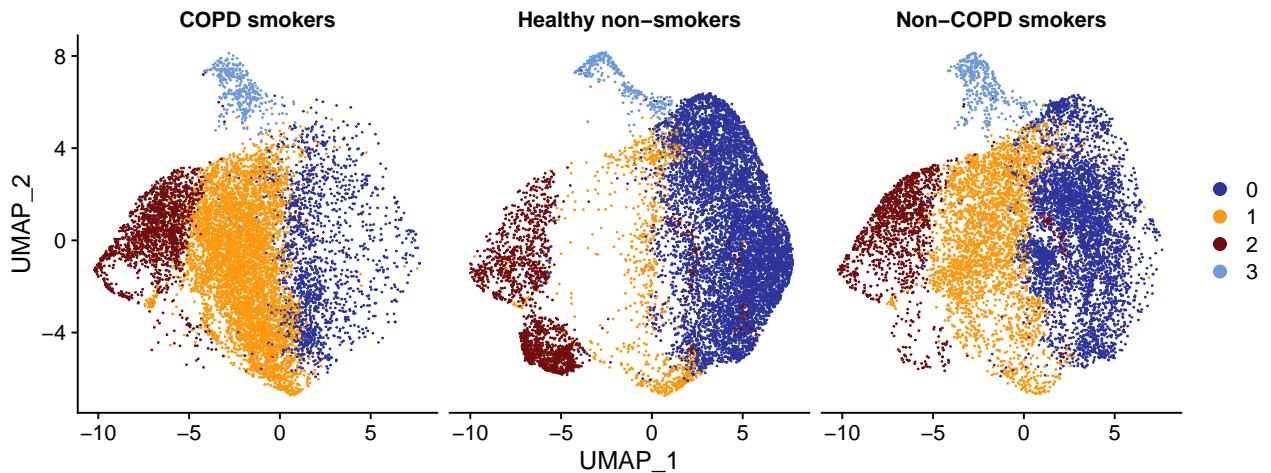
```
suppressMessages(library(Seurat))  
suppressMessages(library(dplyr))  
suppressMessages(library(ggplot2))  
results <- readRDS(file = "../3-Merge_and_cell_typing/so.merged_clusters.  
seuratObject.Rds")
```

3 Distribution of cells in clusters

```
pal_4c <- c("#2E359A", "#FC990E", "#720DOD", "#6E9BD8")  
p <- DimPlot(results, cols = pal_4c)  
p
```



UMAPplot split by group

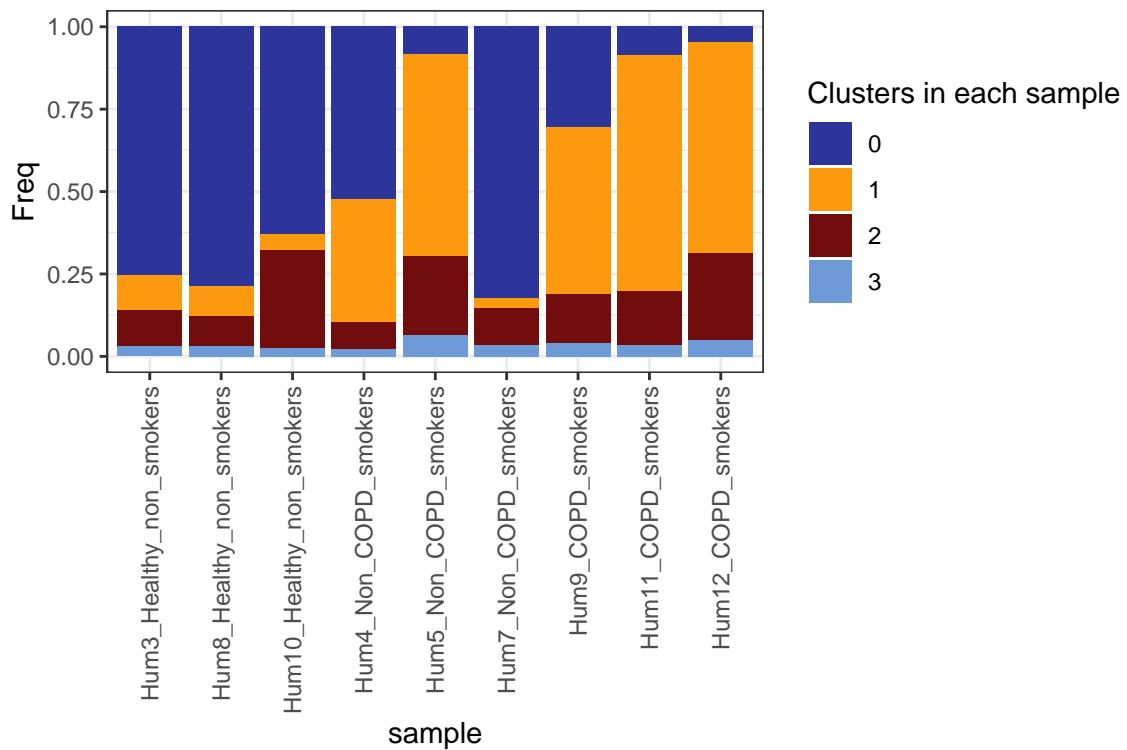


Barplot split by group

```

source("../R/barChart.R")
source("../R/SeuratFreqTable.R")
freq.celltype.list <- list(
  Hum3_Healthy_non_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum3"), slotName = "seurat_clusters"),
  Hum8_Healthy_non_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum8"), slotName = "seurat_clusters"),
  Hum10_Healthy_non_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum10"), slotName = "seurat_clusters"),
  Hum4_Non_COPD_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum4"), slotName = "seurat_clusters"),
  Hum5_Non_COPD_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum5"), slotName = "seurat_clusters"),
  Hum7_Non_COPD_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum7"), slotName = "seurat_clusters"),
  Hum9_COPD_smokers = Seurat2CellFreqTable(subset(results, subset = origin
    == "LBA_Hum9"), slotName = "seurat_clusters"),
  Hum11_COPD_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum11"), slotName = "seurat_clusters"),
  Hum12_COPD_smokers = Seurat2CellFreqTable(subset(results, subset =
    origin == "LBA_Hum12"), slotName = "seurat_clusters")
)
p <- barChart(freq.celltype.list) + labs(fill = "Clusters in each sample")
+ scale_fill_manual(values = pal_4c) + theme(axis.text.x = element_
text(angle = 90, vjust = 0.5, hjust=1))
p

```



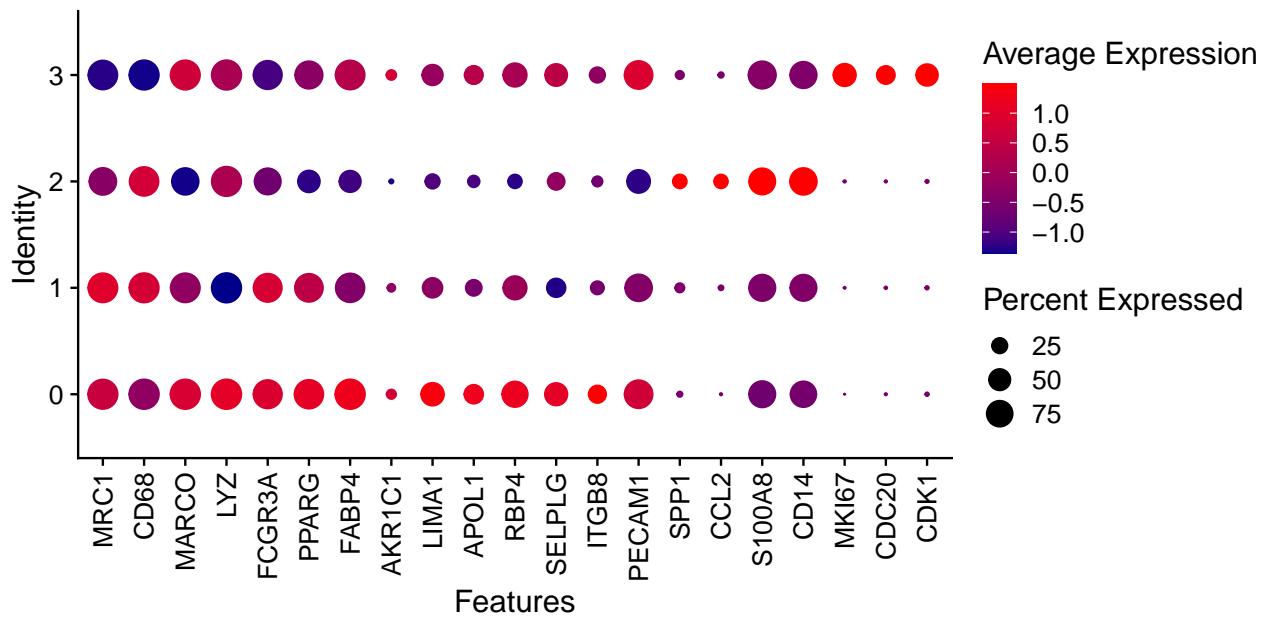
4 Expression of macrophage markers and Cluster 1/3/4 signatures

```

p <- DotPlot(results, features = c("MRC1", "CD68", "MARCO", "LYZ", "FCGR3A", # high expression of core macrophage genes
                                    "PPARG", # AM-associated transcription factor
                                    "FABP4", "AKR1C1", "LIMA1", "APOL1", "RBP4", # Cluster 1 upregulated
                                    "SELPLG", "ITGB8", "PECAM1", # Cluster 1 overexpressed genes coding for cell adhesion molecules
                                    "SPP1", "CCL2", "S100A8", "CD14", # Cluster 3 overexpressed transcripts encoding monocyte lineage-associated molecules
                                    "MKI67", "CDC20", "CDK1" # Cluster 4 upregulated cycling-related genes
), assay = "RNA",
scale.by = "size",
cols = c("dark_blue", "red")) +
theme(axis.text.x = element_text(angle = 90,
vjust = 0.5,
hjust=1))

```

1
2
3
4
5
6
7
8
9
10
11
12
13

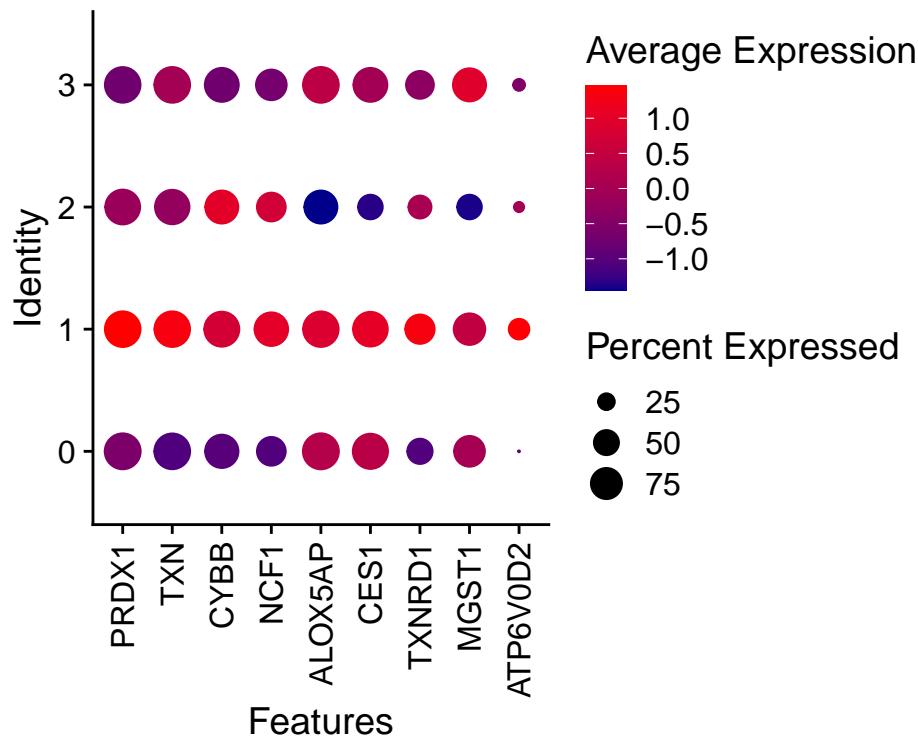


5 Expression of cluster 2 signature

```

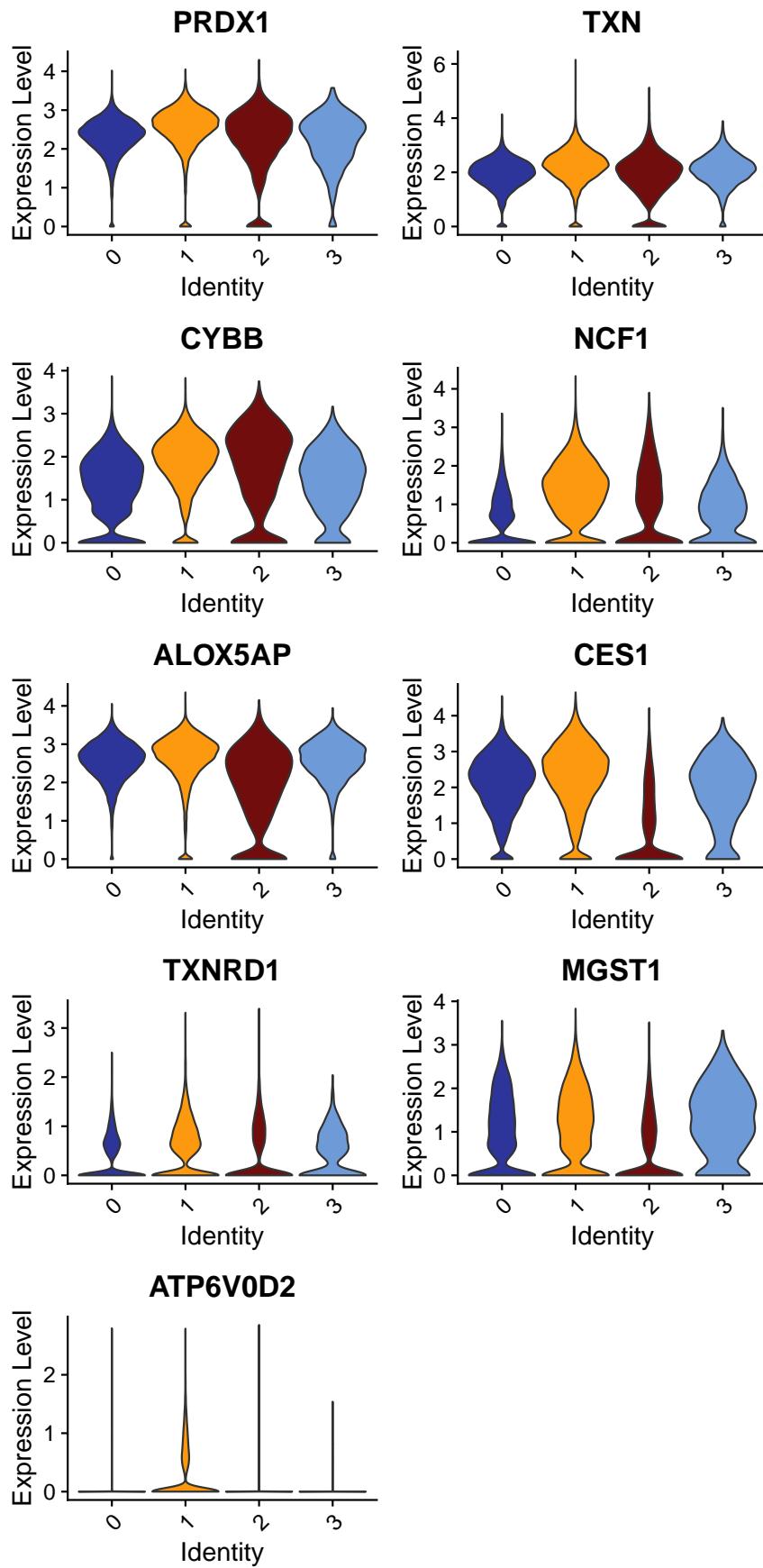
1 p <- DotPlot(results, features = c("PRDX1", "TXN", "CYBB", "NCF1", " "
2   ALOX5AP", "CES1", "TXNRD1", "MGST1", "ATP6V0D2"),
3     assay = "RNA",
4     scale.by = "size",
5     cols = c("darkblue", "red")) +
6     theme(axis.text.x = element_text(angle = 90,
7       vjust = 0.5,
8       hjust=1))

```



Signature of Cluster 2 in vlnplot:

```
p <- VlnPlot(results, features = c("PRDX1", "TXN", "CYBB", "NCF1", "ALOX5AP", "CES1", "TXNRD1", "MGST1", "ATP6V0D2"),
              pt.size = 0,
              cols = pal_4c,
              ncol = 2)
p
```



6 Focus on Cluster 3 and recluster

NOTE: The cluster 3 in the manuscript refers to cluster 2 in the following codes.

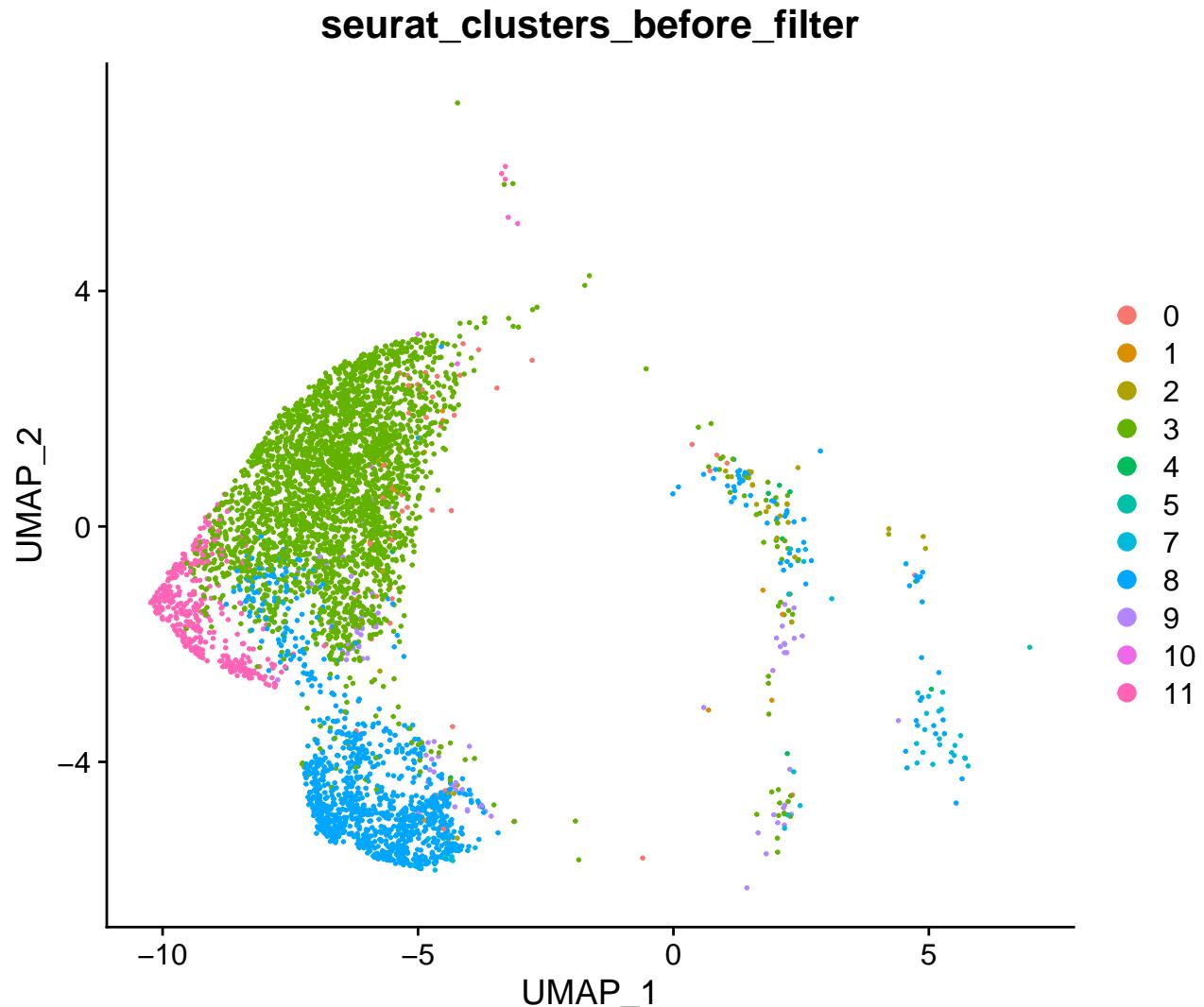
6.1 subsetdata

```
results.c2 <- subset(results, idents = 2) 1
results.c2 2

## An object of class Seurat
## 20050 features across 4897 samples within 1 assay 1
## Active assay: RNA (20050 features, 2000 variable features) 2
## 3 dimensional reductions calculated: pca, umap, tsne 3
## 4
```

As the DC-like population (cluster 11 in the pre-filter clustering) is mainly enriched in this cluster, let's first identify these cells.

```
DimPlot(results.c2, group.by = "seurat_clusters_before_filter") 1
```



6.2 Processing data (only cluster 3)

NOTE: The cluster 3 in the manuscript refers to cluster 2 in the following codes.

```

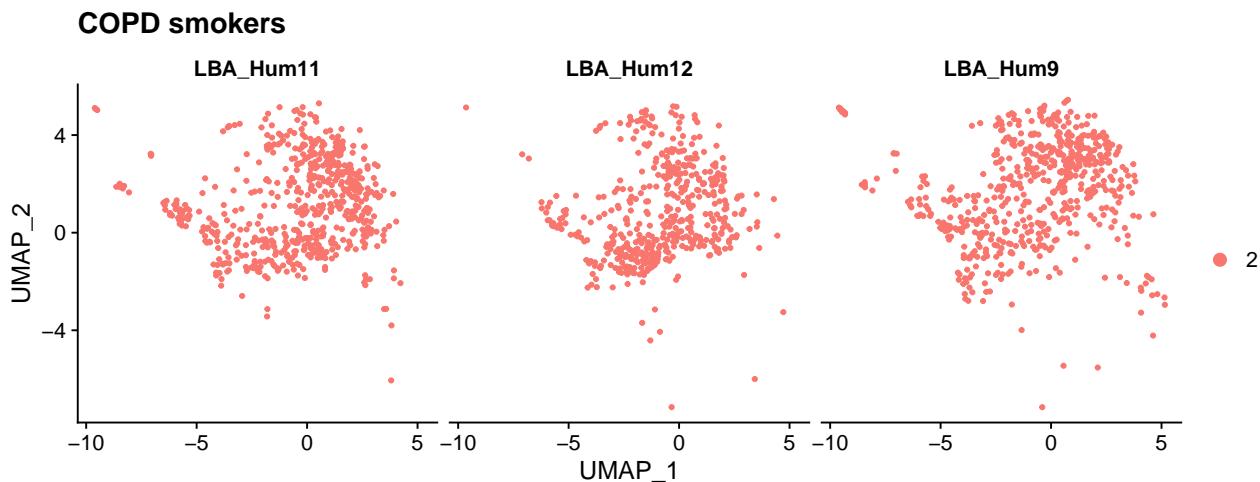
results.c2 <- NormalizeData(results.c2, verbose = FALSE)          1
results.c2 <- FindVariableFeatures(results.c2, selection.method = "vst", 2
  nfeatures = 2000, verbose = FALSE)
results.c2 <- ScaleData(results.c2, features = rownames(results.c2), 3
  verbose = FALSE)
results.c2 <- RunPCA(results.c2, features = VariableFeatures(results.c2), 4
  verbose = FALSE)
results.c2 <- RunUMAP(results.c2, dims = 1:10, verbose = FALSE)      5

```

```

DimPlot(subset(results.c2, subset = group == "COPD_U_smokers"), split.by = " 1
  origin") + ggtitle("COPD_U_smokers")

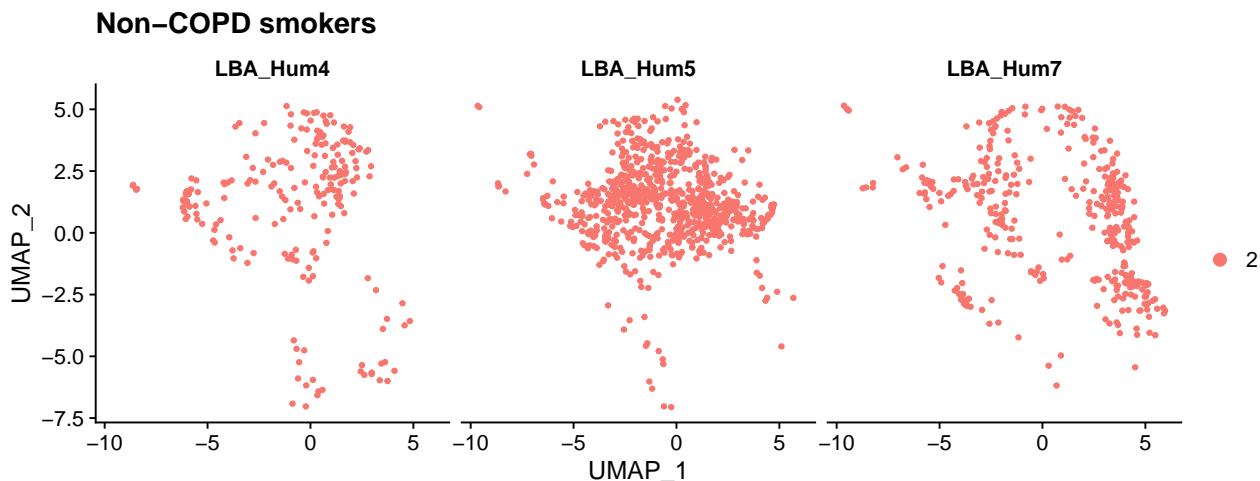
```



```

DimPlot(subset(results.c2, subset = group == "Non-COPD_U_smokers"), split.by = " 1
  origin") + ggtitle("Non-COPD_U_smokers")

```

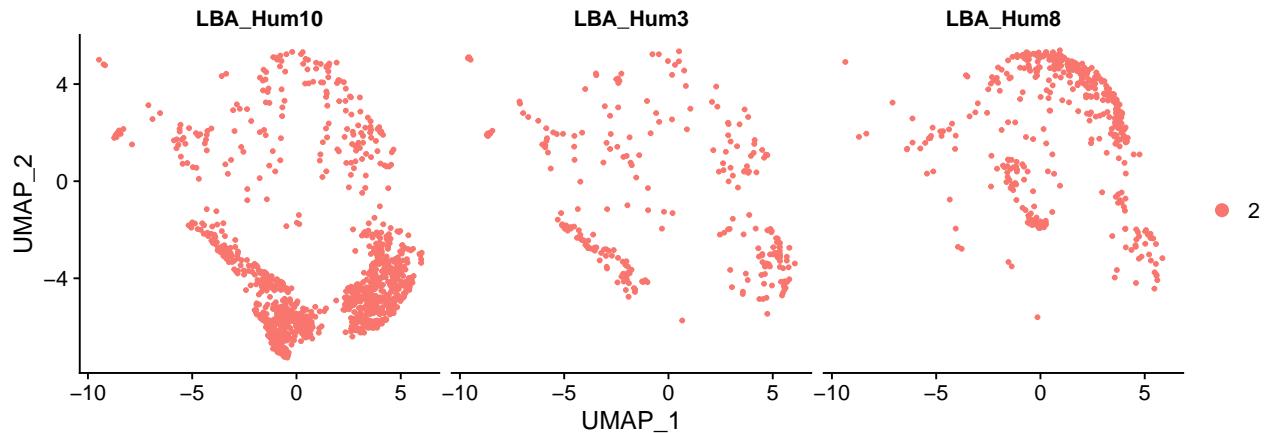


```

DimPlot(subset(results.c2, subset = group == "Healthy_U_non-smokers"), split 1
  .by = "origin") + ggtitle("Healthy_Unon-smokers")

```

Healthy non-smokers



Except the sample Hum10, other samples have relatively equal distribution of all subsets. the Hum10 represent two exceptional lower subsets.

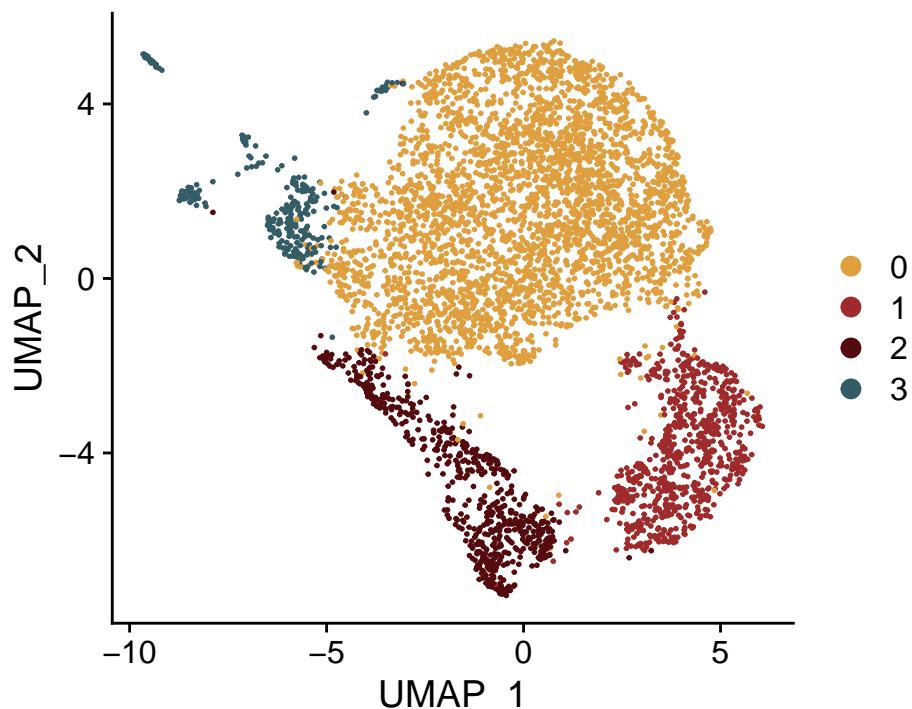
6.3 Re-cluster the Cluster 3

NOTE: The cluster 3 in the manuscript refers to cluster 2 in the following codes.

```

pal_4c.c2 <- c("#e09f3e", "#9e2a2b", "#540b0e", "#335c67")           1
results.c2 <- FindNeighbors(results.c2, reduction = "pca", dims = 1:10,      2
    verbose = FALSE)
results.c2 <- FindClusters(results.c2, resolution = 0.12, verbose = FALSE) 3
p <- DimPlot(results.c2, cols = pal_4c.c2)                                4
p

```



```

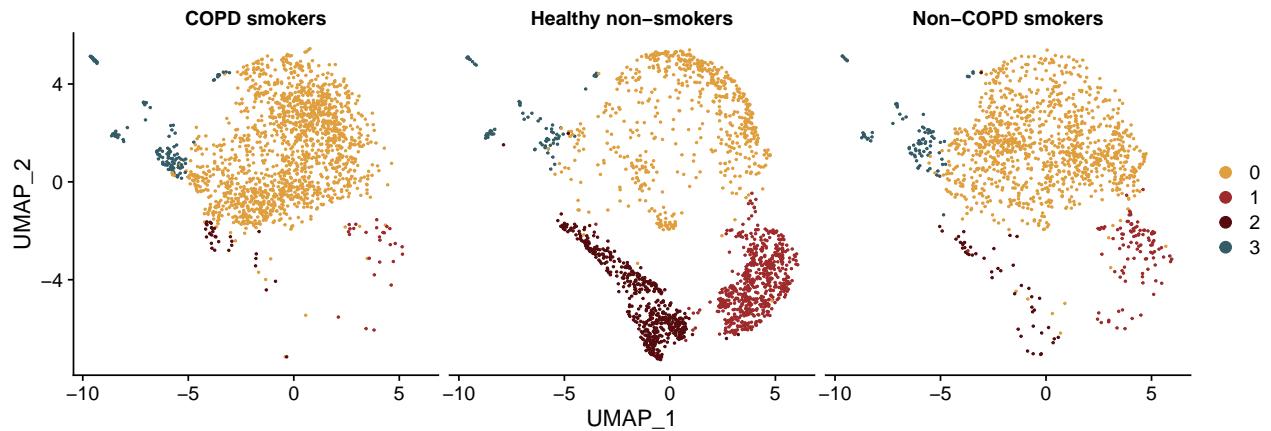
pal_4c.c2 <- c("#e09f3e", "#9e2a2b", "#540b0e", "#335c67")           1
results.c2 <- FindNeighbors(results.c2, reduction = "pca", dims = 1:10,      2
    verbose = FALSE)

```

```

results.c2 <- FindClusters(results.c2, resolution = 0.12, verbose = FALSE) 3
p <- DimPlot(results.c2, cols = pal_4c.c2, split.by = "group") 4
p 5

```

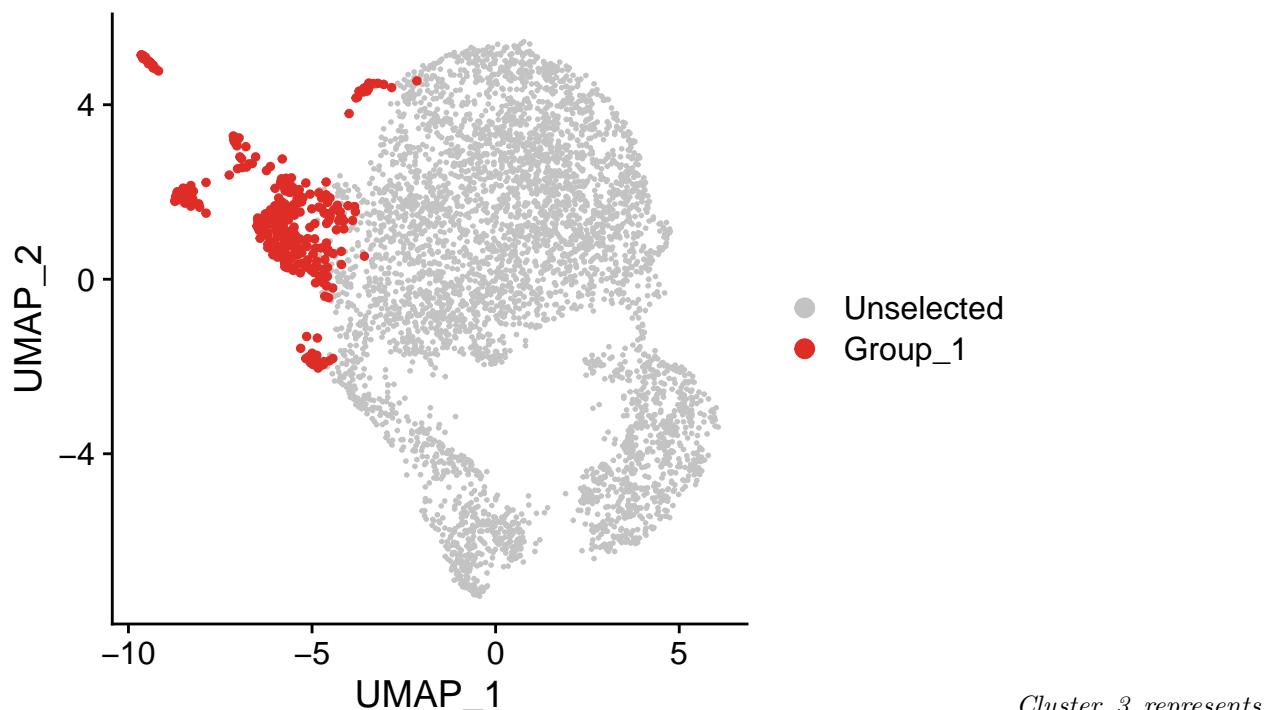


Where is the DC-like population located?

```

DimPlot(results.c2, cells.highlight = WhichCells(results.c2, expression =
  seurat_clusters_before_filter == "11")) 1

```



the old cluster 11 DC(-like).

Cluster 3 represents

```

saveRDS(results.c2, file = "./cluster2_clustered.seuratObject.Rds")

```

1

6.4 Statistic summary about the subpopulations in Cluster 3 (Monocyte-derived AM)

Cell number of each cluster for each samples

```

names.col <- unique(results.c2$origin)
names.row <- as.character(0:3)
df <- sapply(names.col, function(x) table(subset(results.c2, subset =
  origin == x)$seurat_clusters)[names.row])
df <- as.data.frame(df)
df

```

```

## # A tibble: 4 x 9
##   LBA_Hum3 LBA_Hum4 LBA_Hum5 LBA_Hum7 LBA_Hum8 LBA_Hum9 LBA_Hum10
##   <int>     <int>     <int>     <int>     <int>     <int>     <int>    <
## 1      85      151      715      255      351      527      166
##           539
## 2      77       17        7       77       41       13      497
##           11
## 3      70       12       13       22        5       13      464
##           8
## 4      35       28       34       29       16       50       28
##           49
## # ... with 1 more variable: LBA_Hum12 <int>

```

Distribution in %:

```

df <- apply(df, 2, function(x) round(x/sum(x)*100, 2) )
df <- as.data.frame(df)
df

```

```

## # A tibble: 4 x 9
##   LBA_Hum3 LBA_Hum4 LBA_Hum5 LBA_Hum7 LBA_Hum8 LBA_Hum9 LBA_Hum10
##   <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>    <
## 1      31.8      72.6      93.0      66.6      85.0      87.4      14.4
##           88.8
## 2      28.8      8.17      0.91      20.1      9.93      2.16      43.0
##           1.81
## 3      26.2      5.77      1.69      5.74      1.21      2.16      40.2
##           1.32
## 4      13.1      13.5      4.42      7.57      3.87      8.29      2.42
##           8.07
## # ... with 1 more variable: LBA_Hum12 <dbl>

```

```

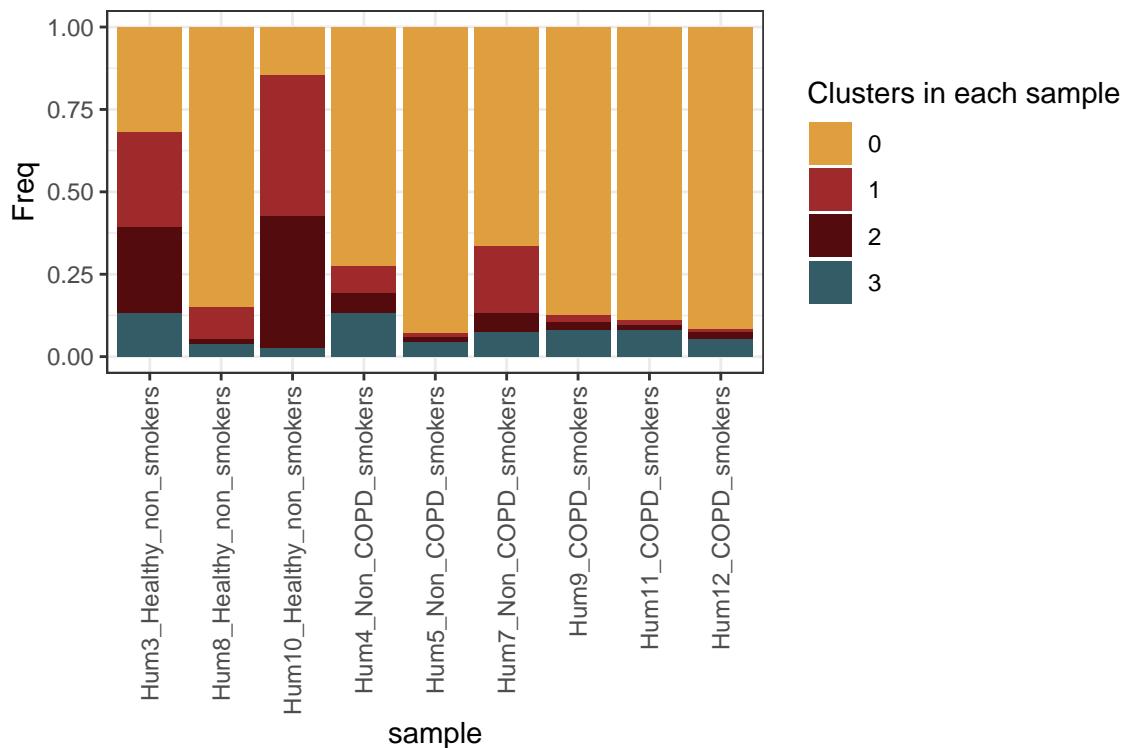
freq.celltype.list <- list(
  Hum3_Healthy_non_smokers = Seurat2CellFreqTable(subset(results.c2,
    subset = origin == "LBA_Hum3"), slotName = "seurat_clusters"),
  Hum8_Healthy_non_smokers = Seurat2CellFreqTable(subset(results.c2,
    subset = origin == "LBA_Hum8"), slotName = "seurat_clusters"),
  Hum10_Healthy_non_smokers = Seurat2CellFreqTable(subset(results.c2,
    subset = origin == "LBA_Hum10"), slotName = "seurat_clusters"),
  Hum4_Non_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
    origin == "LBA_Hum4"), slotName = "seurat_clusters"),

```

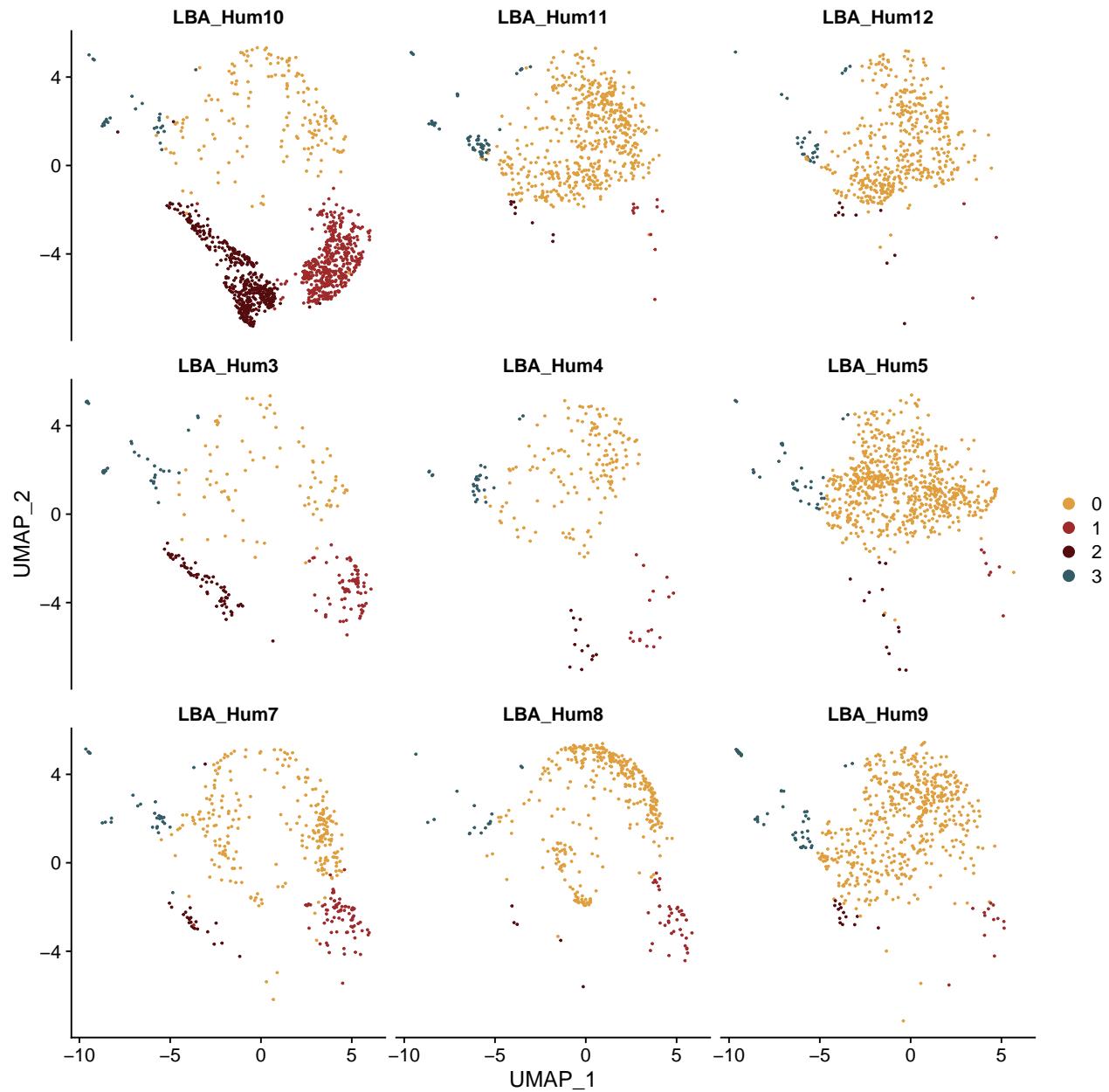
```

    Hum5_Non_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
      origin == "LBA_Hum5"), slotName = "seurat_clusters"),
    Hum7_Non_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
      origin == "LBA_Hum7"), slotName = "seurat_clusters"),
    Hum9_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
      origin == "LBA_Hum9"), slotName = "seurat_clusters"),
    Hum11_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
      origin == "LBA_Hum11"), slotName = "seurat_clusters"),
    Hum12_COPD_smokers = Seurat2CellFreqTable(subset(results.c2, subset =
      origin == "LBA_Hum12"), slotName = "seurat_clusters")
)
p <- barChart(freq.celltype.list) + labs(fill = "Clusters in each sample") +
  scale_fill_manual(values = pal_4c.c2) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
p

```



```
DimPlot(results.c2, split.by = "origin", ncol = 3, cols = pal_4c.c2) 1
```



6.5 Functional markers in subpopulations of Cluster 3 (Monocyte-derived AM)

```

1 p <- DotPlot(results.c2, features = c("FCER1A", "CD1A", "CD1C", "CD1E", "
2   CLEC10A",
3     "CCL18", "CCL4", "CCL3", "CXCL9", " "
4       CXCL10", "IL10",
5     "CALM1", "CALM2", "CCND2", "CCND3",
6     "FCGR2A", "FCGR2B",
7     "ADGRE5", "AGA", "ITGAM", "CLEC12A",
8     "ANXA11", "ARPC2", "ARPC5", "FCN1", " "
9       ICAM3"
10    ),
11   assay = "RNA",
12   scale.by = "size",
13 )

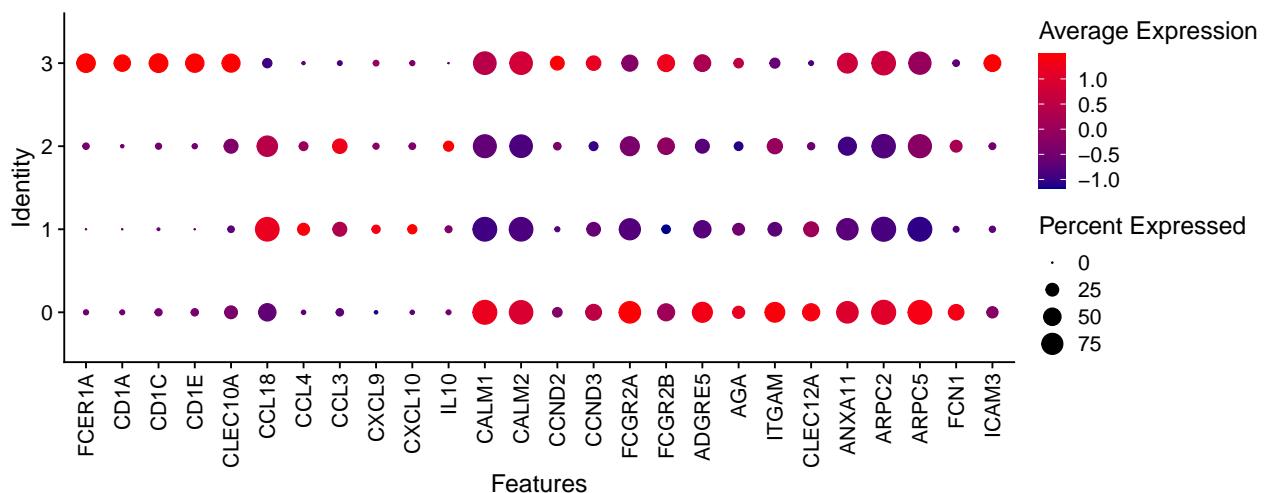
```

```

          cols = c("dark_blue", "red")) +
theme(axis.text.x = element_text(angle = 90,
vjust = 0.5,
hjust=1))
p

```

10
11
12
13
14



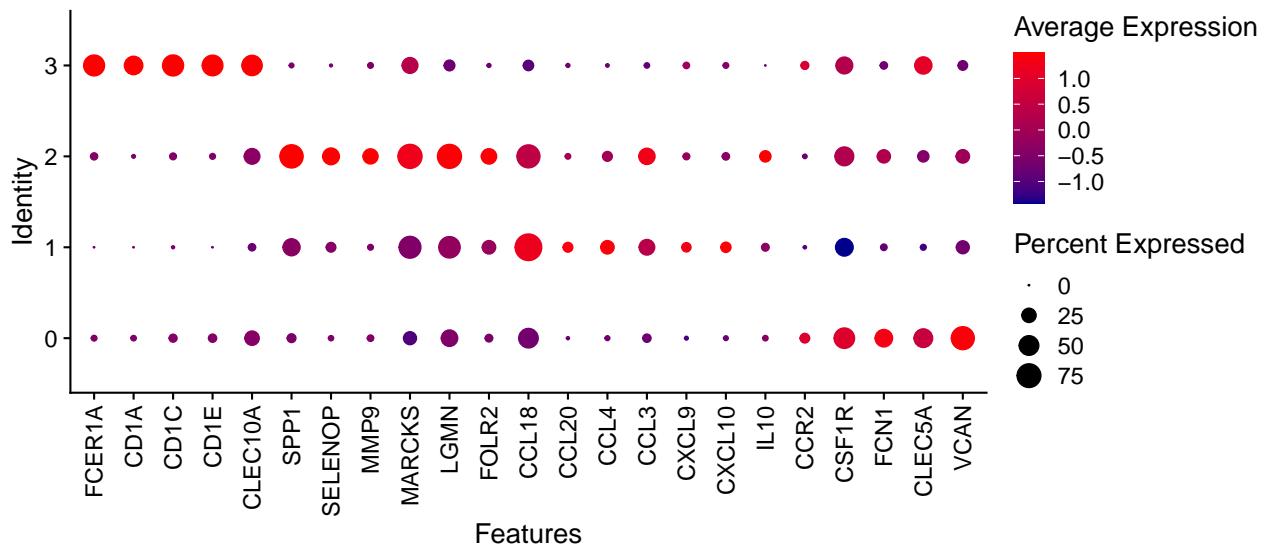
With another marker list

```

p <- DotPlot(results.c2, features = c("FCER1A", "CD1A", "CD1C", "CD1E", "CLEC10A",
                                         "SPP1", "SELENOP", "MMP9", "MARCKS",
                                         "LGMN", "FOLR2",
                                         "CCL18", "CCL20", "CCL4", "CCL3", "CXCL9", "CXCL10", "IL10",
                                         "CCR2", "CSF1R", "FCN1",
                                         "CLEC5A", "VCAN"),
              assay = "RNA",
              scale.by = "size",
              cols = c("dark_blue", "red")) +
theme(axis.text.x = element_text(angle = 90,
vjust = 0.5,
hjust=1))
p

```

1
2
3
4
5
6
7
8
9
10
11
12
13



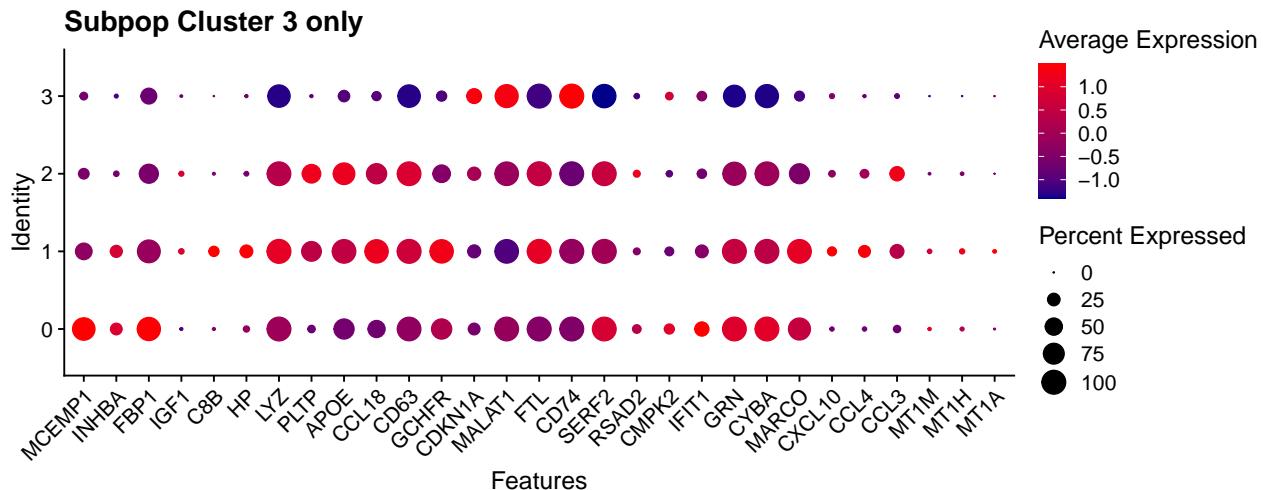
7 Compare to Mould et al. 2020

Let's focus on the Fig 2C of the report Mould et al. 2021[3]: Mould, K. J. et al. Airspace macrophages and monocytes exist in transcriptionally distinct subsets in healthy adults. Am. J. Respir. Crit. Care Med. (2021) doi:10.1164/RCCM.202005-1989OC.

```

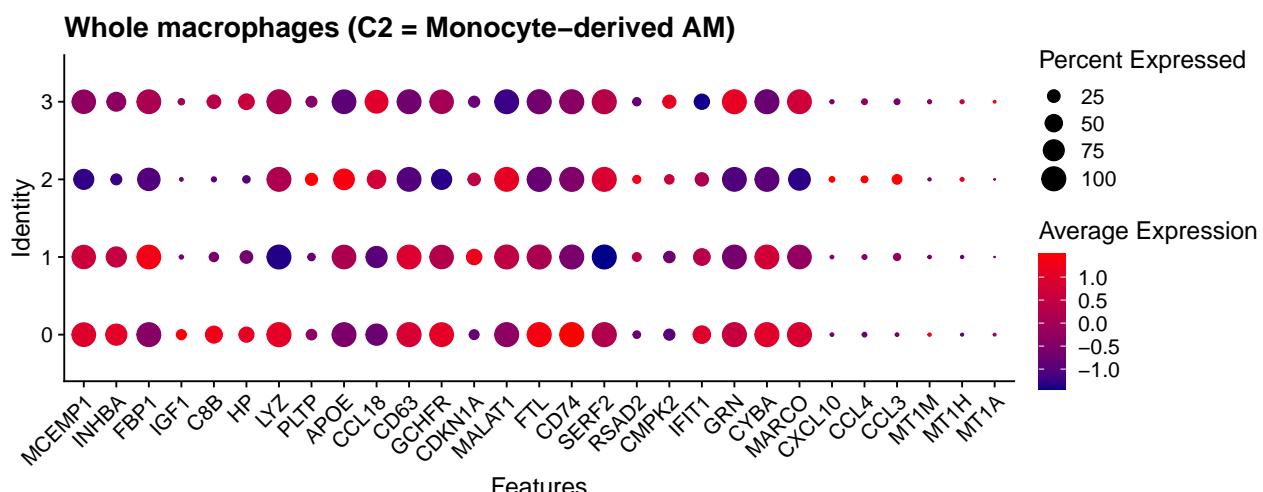
1 DotPlot(results.c2, features = c("MCEMP1", "INHBA", "FBP1",
2   "IGF1", "C8B", "HP",
3   "LYZ", "PLTP", "APOE",
4   "CCL18", "CD63", "GCHFR",
5   "CDKN1A", "MALAT1", "FTL",
6   "CD74", "SERF2", "RSAD2", "CMPK2",
7   "IFIT1", "GRN", "CYBA", "MARCO", "CXCL10"
8   ,
9   "CCL4", "CCL3",
10  "MT1M", "MT1H", "MT1A"),
11  scale.by = "size",
12  cols = c("darkblue", "red")) + theme(axis.text.x =
element_text(angle = 45, hjust=1))+
ggttitle("Subpop_Cluster_3_only")

```



How about in whole macrophages (not only cluster2)

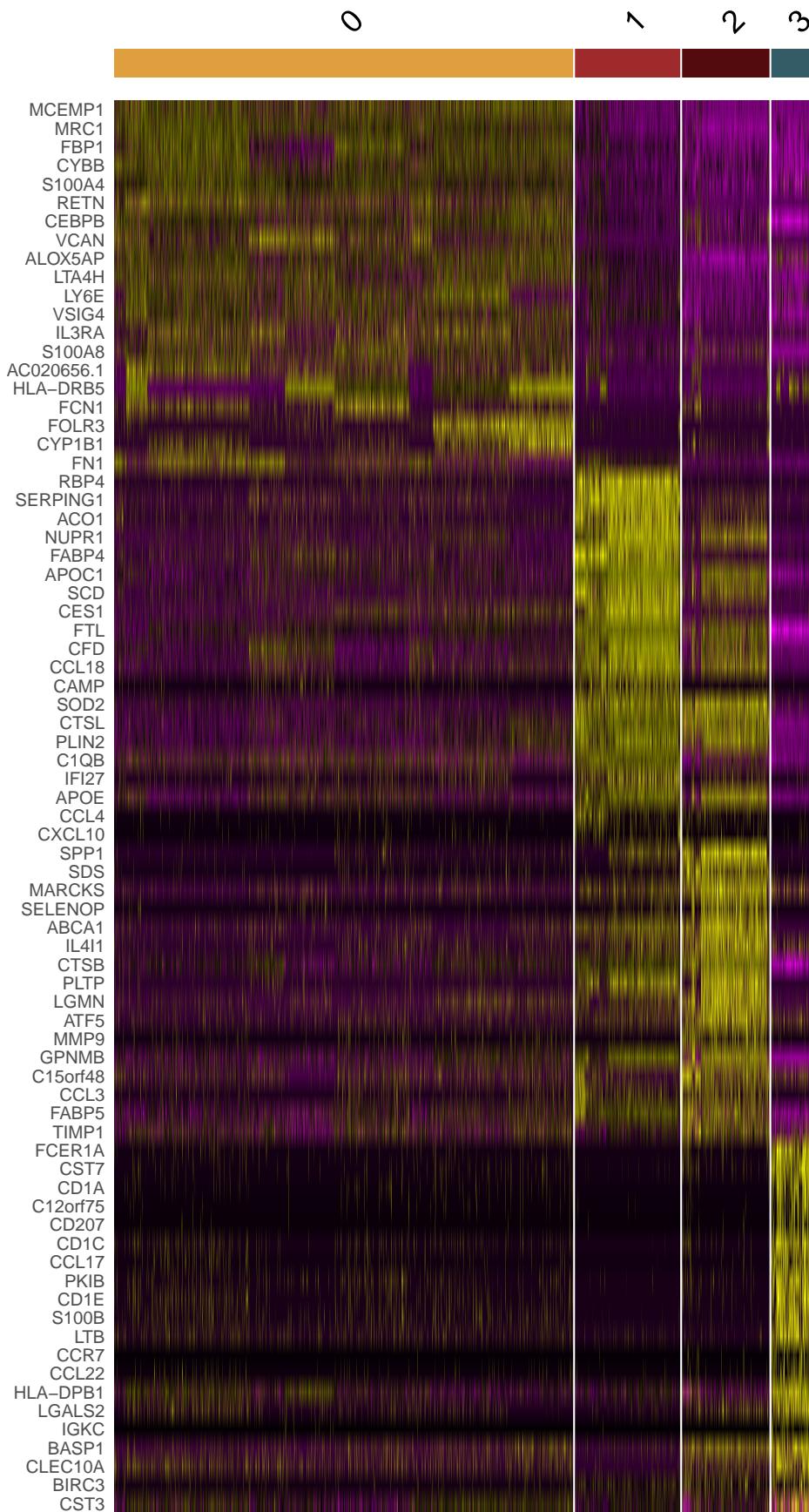
```
DotPlot(results, features = c("MCEMP1", "INHBA", "FBP1",
                               "IGF1", "C8B", "HP",
                               "LYZ", "PLTP", "APOE",
                               "CCL18", "CD63", "GCHFR",
                               "CDKN1A", "MALAT1", "FTL",
                               "CD74", "SERF2", "RSAD2", "CMPK2",
                               "IFIT1", "GRN", "CYBA", "MARCO", "CXCL10",
                               ,
                               "CCL4", "CCL3",
                               "MT1M", "MT1H", "MT1A"),
        scale.by = "size",
        cols = c("dark_blue", "red")) + theme(axis.text.x =
          element_text(angle = 45, hjust=1)) +
        ggtitle("Whole_macrophages_(C2_Monocyte-derived_AM)")
```



Compare to Mould et al. Fig 2C, 1) the Monocyte-derived AM (C2 of whole macrophages) expresses higher cytokine, chemokines (CXCL10, CCL4, CCL3), thus may be the pro-inflammatory macrophage (m5) mentioned in their report; 2) Cluster 0 and cluster 1 are quite similar except IGF1 C8B HP, these 2 pops should represent continuous development of AM.

8 Find DE genes in subpopulations of the cluster 3 (Monocyte-derived AM)

```
all_cluster.markers <- FindAllMarkers(results.c2) 1  
  
require(dplyr)  
top20 <- all_cluster.markers %>% group_by(cluster) %>% top_n(n = 20, wt = 2  
    avg_log2FC)  
DoHeatmap(results.c2, features = top20$gene, group.colors = pal_4c.c2 3  
    ) + NoLegend() 4
```



9 Scoring of Monocyte-derived AM and AM signatures

```
library(Seurat)
library(VISION)
library(ggplot2)
```

9.1 Create signature from DE gene lists

The Monocyte-derived AM and AM signatures are the top 100 DE genes obtain from DESeq2 analysis with bulkRNAseq data (see bulkRNAseq_analysis for details).

```
sig <- read.table("./Mreg_MA_sig.csv", sep = "\t", header = T, as.is = T) 1
sig <- sig[-1, ] # remove description 2
sig <- as.data.frame(lapply(sig, unique), stringsAsFactors=FALSE) # remove 3
      doublon
sig <- as.data.frame(sapply(sig, as.character), stringsAsFactors=FALSE) 4
      # change to character
sig <- as.data.frame(sapply(sig, function(x) x <- x[! x == ""] ), 5
      stringsAsFactors=FALSE) # remove empty

sig.Monocyte_derived_AM <- c(rep(1, length(sig$Mreg_sig))) 6
names(sig.Monocyte_derived_AM) <- c(sig$Mreg_sig) 7
  8
  9

sig.AM <- c(rep(1, length(sig$MA_sig))) 10
names(sig.AM) <- c(sig$MA_sig) 11
  12

sig.Monocyte_derived_AM <- createGeneSignature(name = "Monocyte_derived_AM" 13
      ", sigData = sig.Monocyte_derived_AM")
sig.AM <- createGeneSignature(name = "AM", sigData = sig.AM) 14
sig.macro <- c(sig.Monocyte_derived_AM, sig.AM) 15
```

9.2 Scoring for Monocyte-derived AM and AM signatures

Let's calculate scores for the Monocyte-derived AM and AM signatures in either total macrophages and cells in cluster 3 (refers to C2 in the codes).

```
vis <- Vision(results,
               signatures = sig.macro) 1
  2
  3

vis.c2 <- Vision(results.c2,
                  signatures = sig.macro) 4
  5
```

Calculate score:

```
vis <- calcSignatureScores(vis)
vis.c2 <- calcSignatureScores(vis.c2)
```

9.3 Present signature score with existing embedding in Seurat object

Check cell names are the same in both analyses.

```
identical(colnames(results), rownames(vis@SigScores))
```

```

## [1] TRUE
1
identical(colnames(results.c2), rownames(vis.c2@SigScores))
1
## [1] TRUE
1
results$sig.Monocyte_derived_AM <- vis@SigScores[, "Monocyte_derived_AM"]
2
results$sig.AM <- vis@SigScores[, "AM"]
3
results.c2$sig.Monocyte_derived_AM <- vis.c2@SigScores[, "Monocyte_derived_AM"]
4
results.c2$sig.AM <- vis.c2@SigScores[, "AM"]
5

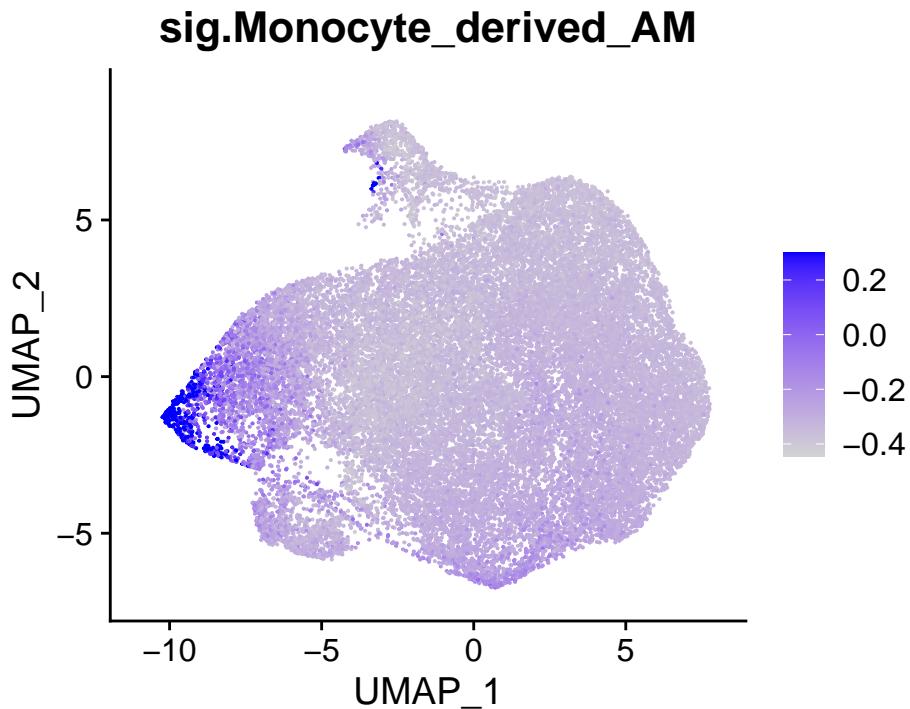
```

Plot signature scores with existing embedding in Seurat object:

```

FeaturePlot(results, features = "sig.Monocyte_derived_AM"
            , min.cutoff = -0.45, max.cutoff = 0.3)
1
2

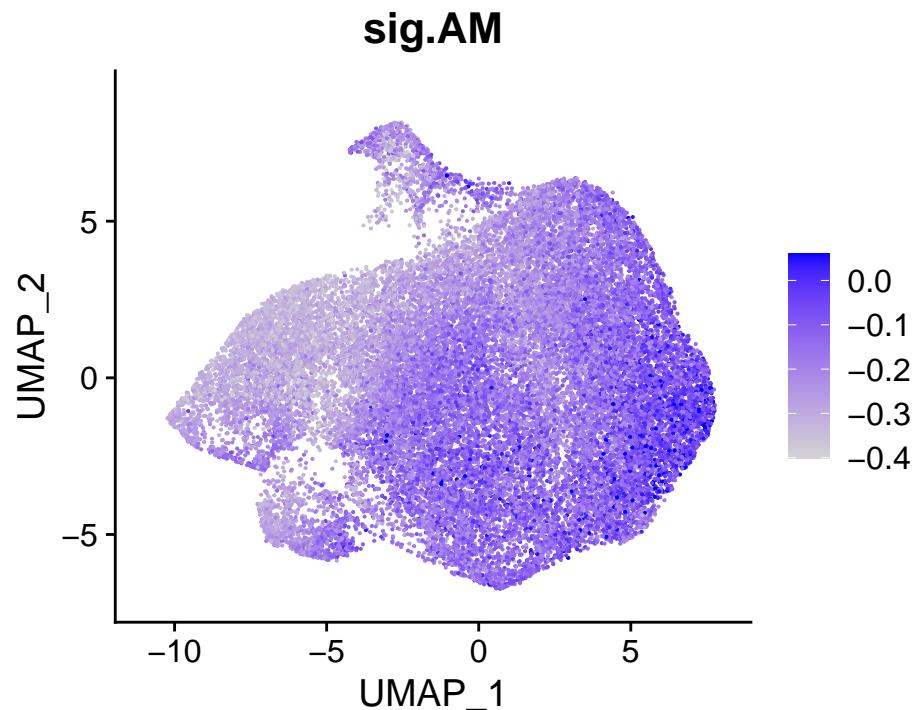
```



```

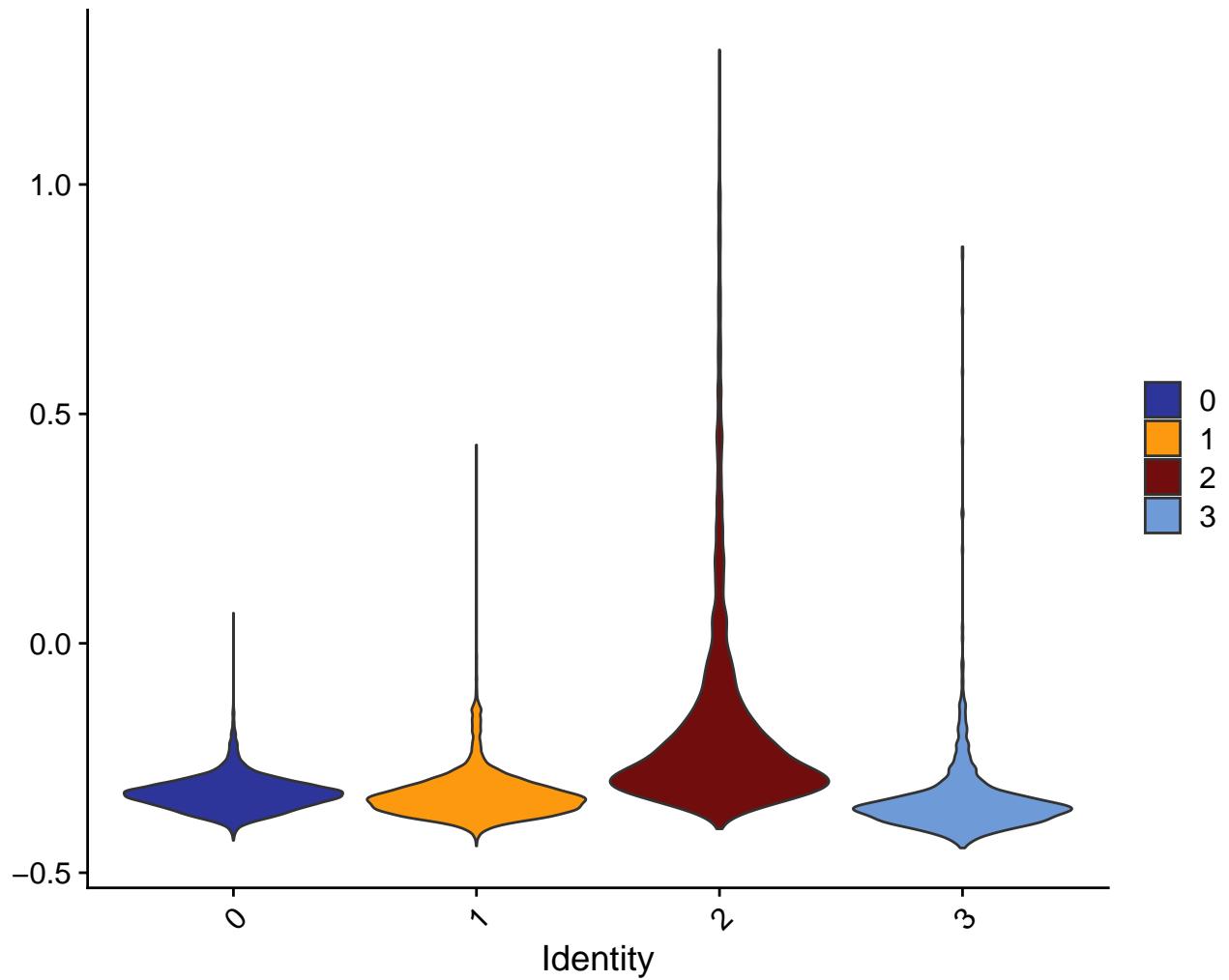
FeaturePlot(results, features = "sig.AM"
            , min.cutoff = -0.4, max.cutoff = 0.06)
1
2

```

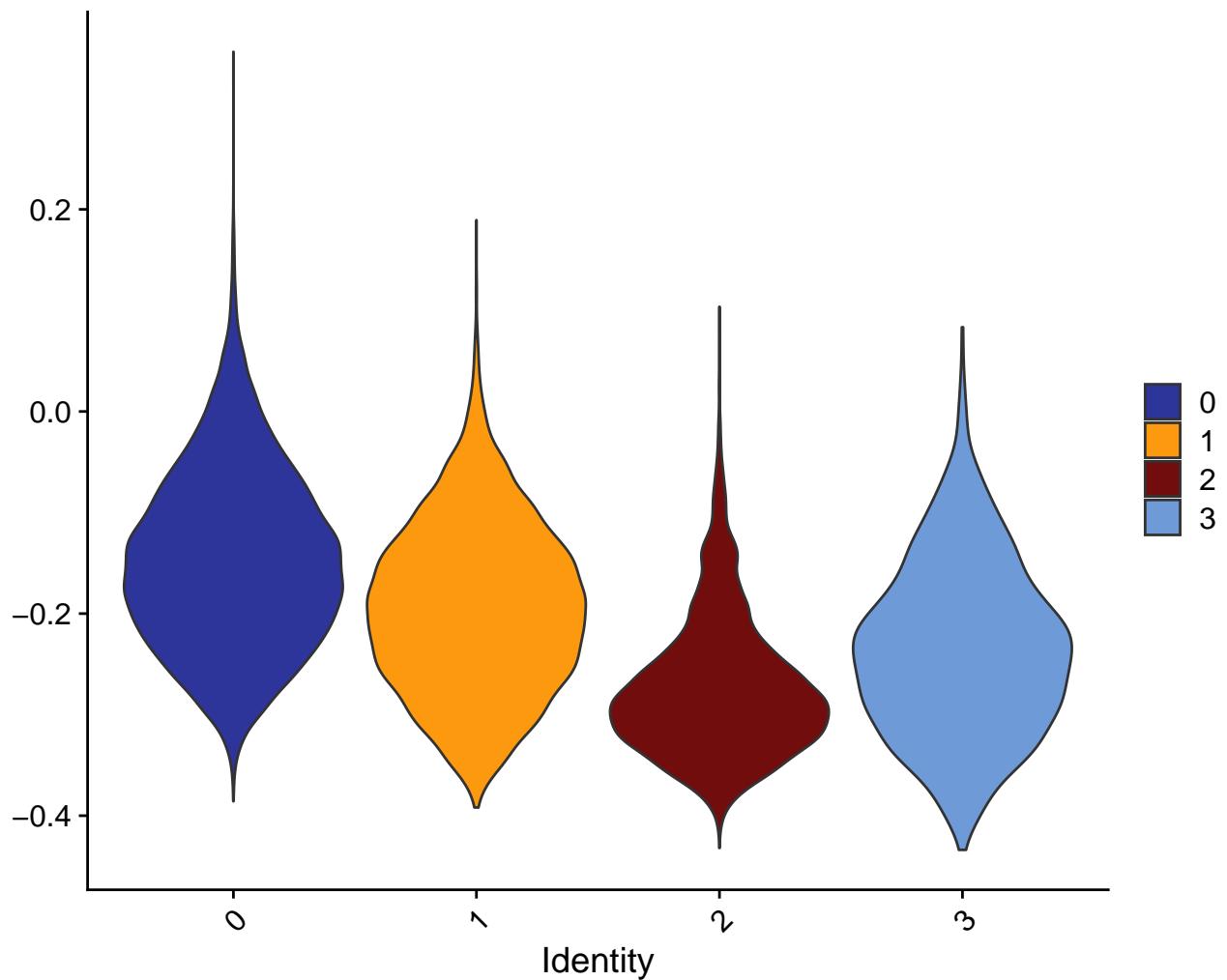


```
VlnPlot(results, features = "sig.Monocyte_derived_AM", pt.size = 0, cols = pal_4c)
```

sig.Monocyte_derived_AM



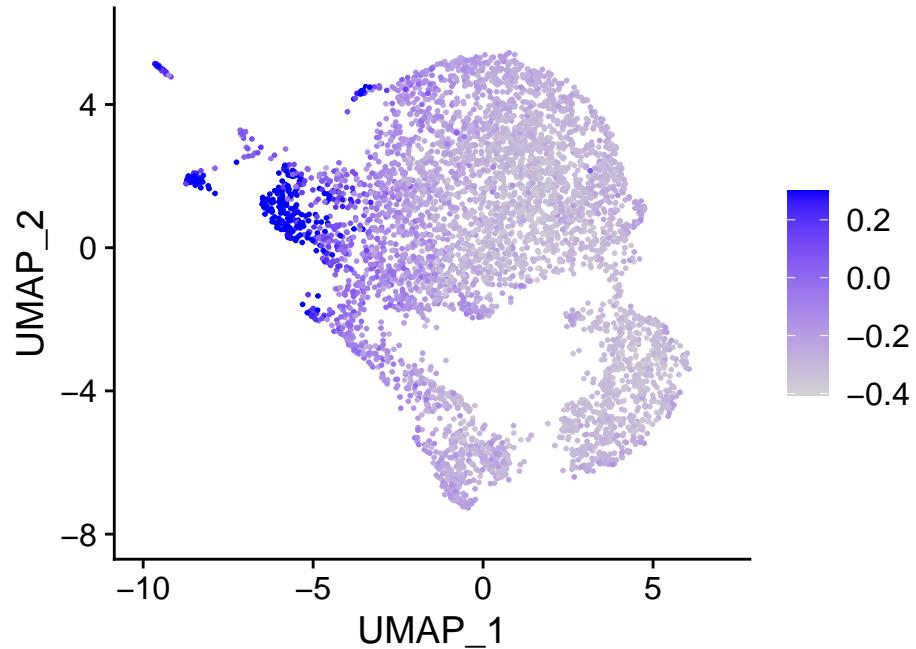
sig.AM



```
FeaturePlot(results.c2, features = "sig.Monocyte_derived_AM"  
           , min.cutoff = -0.45, max.cutoff = 0.3)
```

1
2

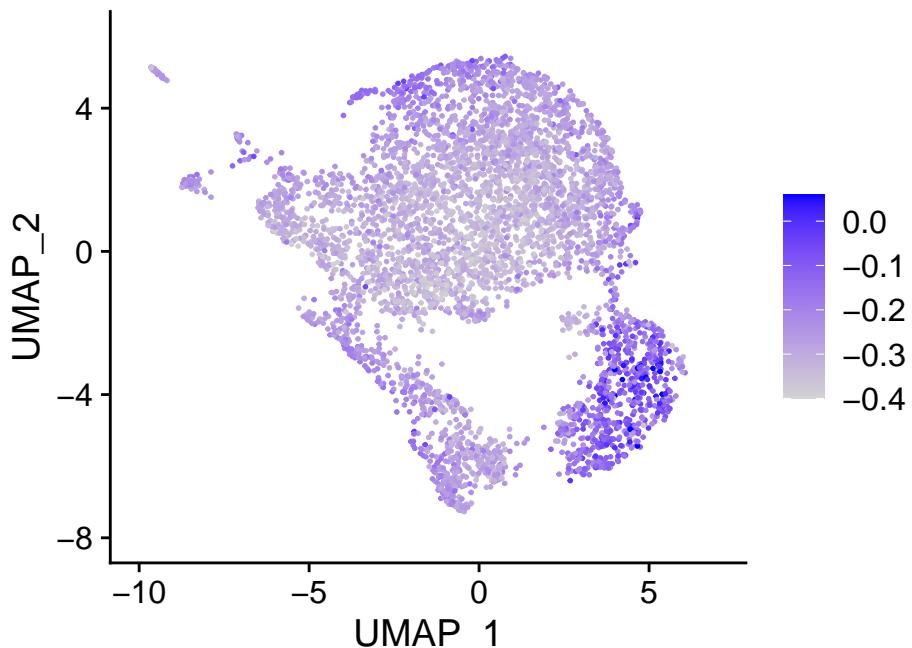
sig.Monocyte_derived_AM



```
FeaturePlot(results.c2, features = "sig.AM"  
           , min.cutoff = -0.4, max.cutoff = 0.06)
```

1
2

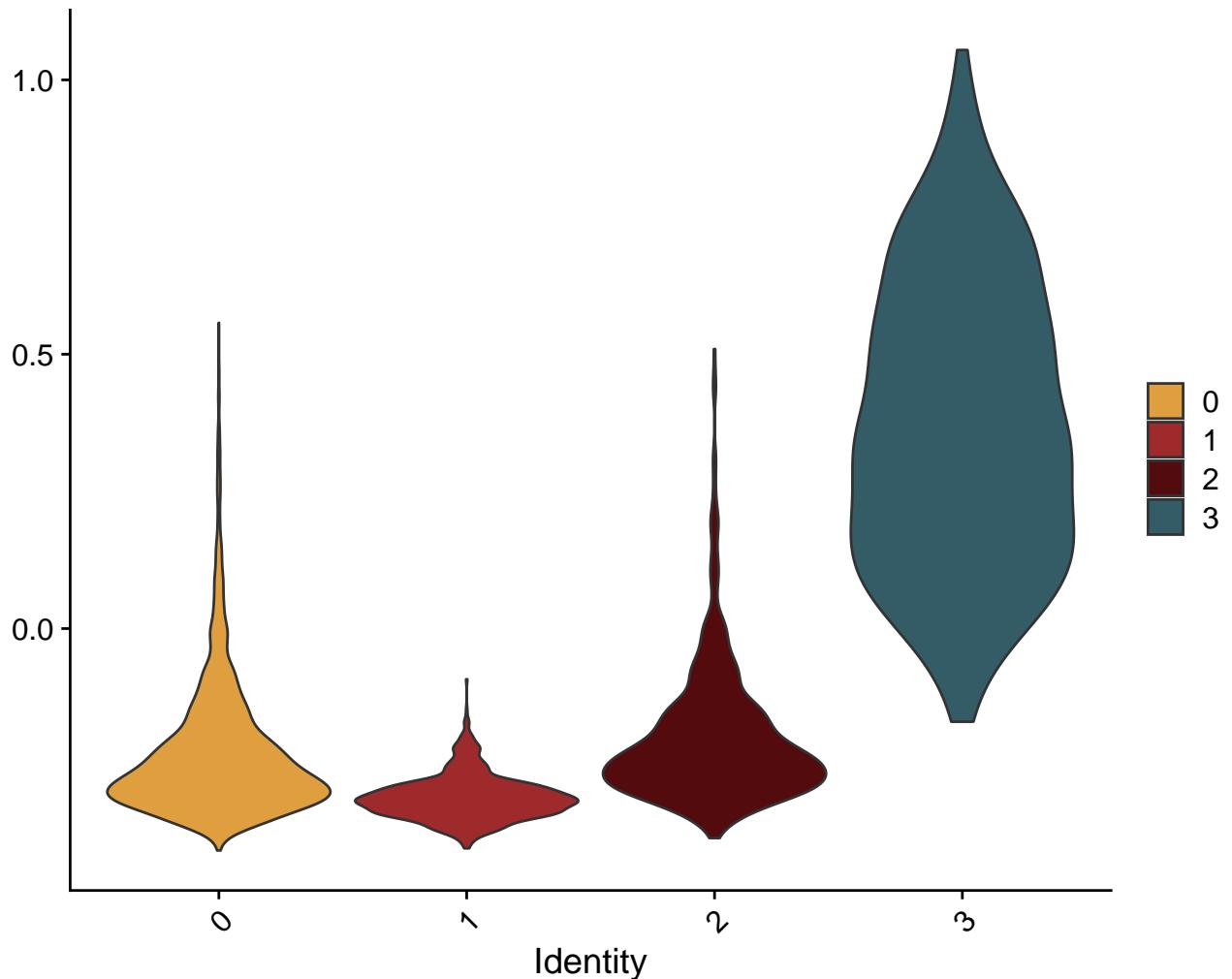
sig.AM



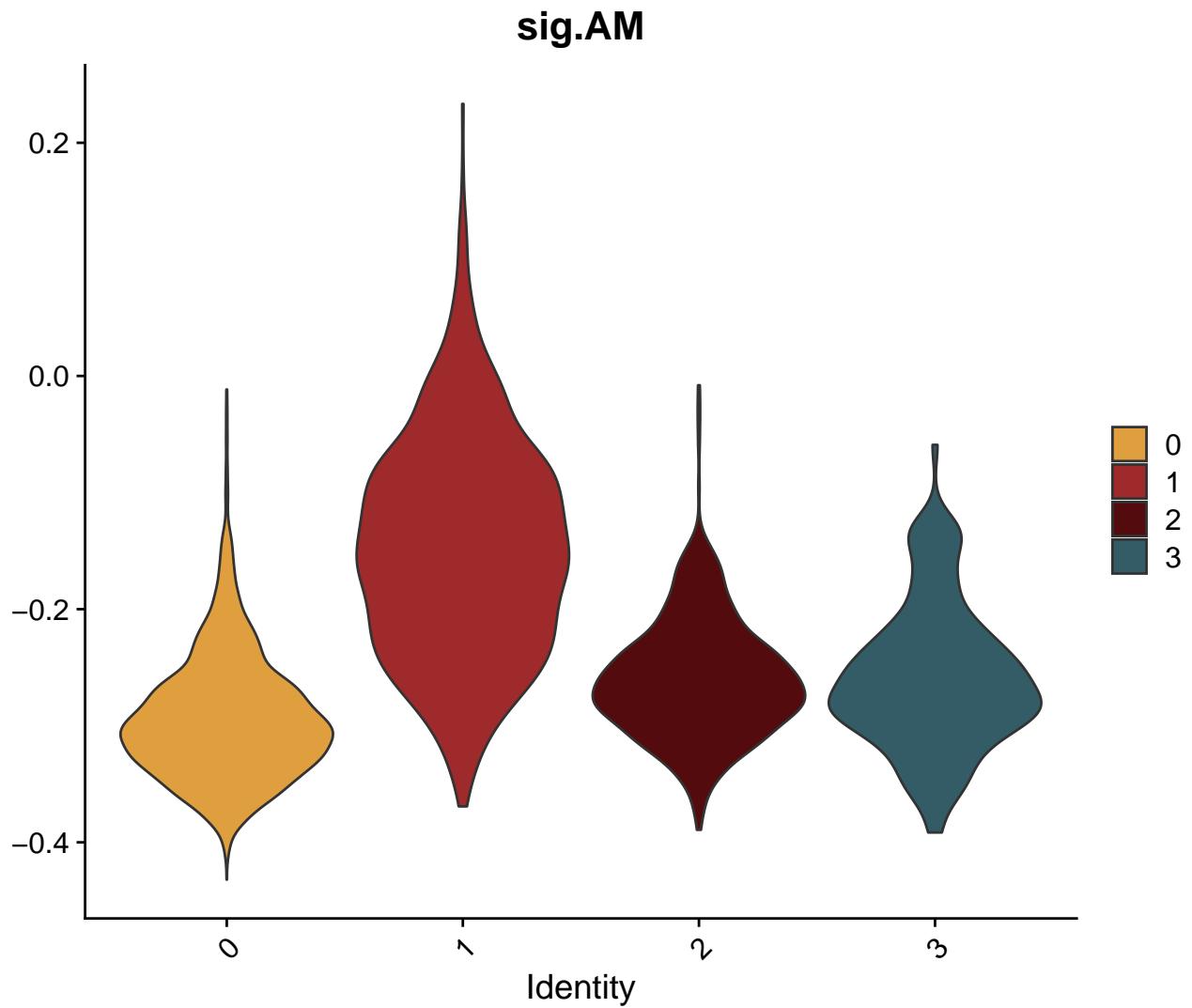
```
VlnPlot(results.c2, features = "sig.Monocyte_derived_AM", pt.size = 0,  
        cols = pal_4c.c2)
```

1
2

sig.Monocyte_derived_AM



```
VlnPlot(results.c2, features = "sig.AM", pt.size = 0, cols = pal_4c.c2) 1
```



10 Session information

```
sessionInfo()
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.3 LTS
##
## Matrix products: default
## BLAS:    /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK:  /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3
##
## locale:
## [1] LC_CTYPE=en_US.UTF-8          LC_NUMERIC=C
## [3] LC_TIME=en_GB.UTF-8          LC_COLLATE=en_US.UTF-8
## [5] LC_MONETARY=en_GB.UTF-8       LC_MESSAGES=en_US.UTF-8
## [7] LC_PAPER=en_GB.UTF-8          LC_NAME=C
## [9] LC_ADDRESS=C                  LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_GB.UTF-8   LC_IDENTIFICATION=C
```

```

##          16
## attached base packages:          17
## [1] stats      graphics   grDevices utils     datasets  methods   base    18
##          19
## other attached packages:         20
## [1] VISION_2.1.0        RColorBrewer_1.1-2 ggplot2_3.3.5       dplyr_1   21
##   .0.8
## [5] SeuratObject_4.0.4  Seurat_4.1.0
##          22
##          23
## loaded via a namespace (and not attached): 24
## [1] ggbeeswarm_0.6.0      Rtsne_0.15           colorspace_2.0-3  25
## [4] deldir_1.0-6          ellipsis_0.3.2      ggridges_0.5.3    26
## [7] mclust_5.4.9          rstudioapi_0.13     spatstat.data_2.1-2 27
## [10] leiden_0.3.9          listenv_0.8.0       farver_2.1.0      28
## [13] ggrepel_0.9.1         RSpectra_0.16-0     fansi_1.0.2       29
## [16] logging_0.10-108     codetools_0.2-18    splines_4.0.3     30
## [19] knitr_1.37            polyclip_1.10-0    jsonlite_1.7.3    31
## [22] ica_1.0-2             cluster_2.1.0      png_0.1-7        32
## [25] uwot_0.1.11           wordspace_0.2-6    shiny_1.7.1      33
## [28] sctransform_0.3.3     spatstat.sparse_2.1-0 plumber_1.1.0    34
## [31] compiler_4.0.3         httr_1.4.2          assertthat_0.2.1  35
## [34] Matrix_1.4-0          fastmap_1.1.0      lazyeval_0.2.2   36
## [37] limma_3.46.0          cli_3.2.0          later_1.3.0     37
## [40] htmltools_0.5.2       tools_4.0.3         rsvd_1.0.5       38
## [43] igraph_1.2.11         gtable_0.3.0       glue_1.6.1       39
## [46] RANN_2.6.1            reshape2_1.4.4     Rcpp_1.0.8       40
## [49] scattermore_0.8        vctrs_0.3.8        nlme_3.1-155    41
## [52] lmtest_0.9-39         spatstat.random_2.1-0 xfun_0.29     42
## [55] stringr_1.4.0         webutils_1.1       globals_0.14.0   43
## [58] mime_0.12              miniUI_0.1.1.1     lifecycle_1.0.1  44
## [61] irlba_2.3.5           goftest_1.2-3      future_1.24.0   45
## [64] MASS_7.3-53            zoo_1.8-9          scales_1.1.1    46
## [67] loe_1.1                spatstat.core_2.4-0 promises_1.2.0.1 47
## [70] spatstat.utils_2.3-0   parallel_4.0.3     swagger_3.33.1  48
## [73] yaml_2.3.5             reticulate_1.24    pbapply_1.5-0   49
## [76] gridExtra_2.3           ggrastr_1.0.1      rpart_4.1-15   50
## [79] fastICA_1.2-3          stringi_1.7.6     highr_0.9       51
## [82] permute_0.9-7          rlang_1.0.1        pkgconfig_2.0.3 52
## [85] matrixStats_0.61.0     evaluate_0.15     lattice_0.20-41 53
## [88] ROCR_1.0-11            purrr_0.3.4       tensor_1.5      54
## [91] patchwork_1.1.1        htmlwidgets_1.5.4  labeling_0.4.2   55
## [94] cowplot_1.1.1          tidyselect_1.1.1  parallelly_1.30.0 56
## [97] RcppAnnoy_0.0.19       plyr_1.8.6        magrittr_2.0.2   57
## [100] R6_2.5.1               generics_0.1.2     DBI_1.1.2      58
## [103] pillar_1.7.0           withr_2.4.3       mgcv_1.8-33    59
## [106] fitdistrplus_1.1-6     survival_3.2-7    abind_1.4-5    60
## [109] tibble_3.1.6            future.apply_1.8.1 crayon_1.5.0    61
## [112] KernSmooth_2.23-20    utf8_1.2.2        spatstat.geom_2.3-2 62
## [115] plotly_4.10.0           rmarkdown_2.11    grid_4.0.3      63
## [118] data.table_1.14.2      vegan_2.5-7      sparsesvd_0.2   64
## [121] digest_0.6.29          pbmcapply_1.5.0   xtable_1.8-4   65
## [124] tidyrr_1.2.0            httpuv_1.6.5     munsell_0.5.0   66
## [127] beeswarm_0.4.0          viridisLite_0.4.0  viper_0.4.5    67
## [130] iotools_0.3-2

```

References

1. Hao Y, Hao S, Andersen-Nissen E, Mauck III WM, Zheng S, Butler A, Lee MJ, Wilk AJ, Darby C, Zagar M, Hoffman P, Stoeckius M, Papalexi E, Mimitou EP, Jain J, Srivastava A, Stuart T, Fleming LB, Yeung B, Rogers AJ, McElrath JM, Blish CA, Gottardo R, Smibert P, Satija R. Integrated analysis of multimodal single-cell data. *Cell* [Internet] 2021; Available from: <https://doi.org/10.1016/j.cell.2021.04.048>.
2. DeTomaso D, Jones MG, Subramaniam M, Ashuach T, Ye CJ, Yosef N. Functional interpretation of single cell similarity maps. *Nature Communications* 2019;
3. Mould KJ, Moore CM, McManus SA, McCubrey AL, McClendon JD, Griesmer CL, Henson PM, Janssen WJ. Airspace macrophages and monocytes exist in transcriptionally distinct subsets in healthy adults. *American Journal of Respiratory and Critical Care Medicine* 2021;