Aim 2 - Finding differentially expressed features (cluster biomarkers)

Yilin Qiu 08/04/2021

The following tutorials were used as references to generate this script

https://satijalab.org/seurat/articles/pbmc3k_tutorial.html

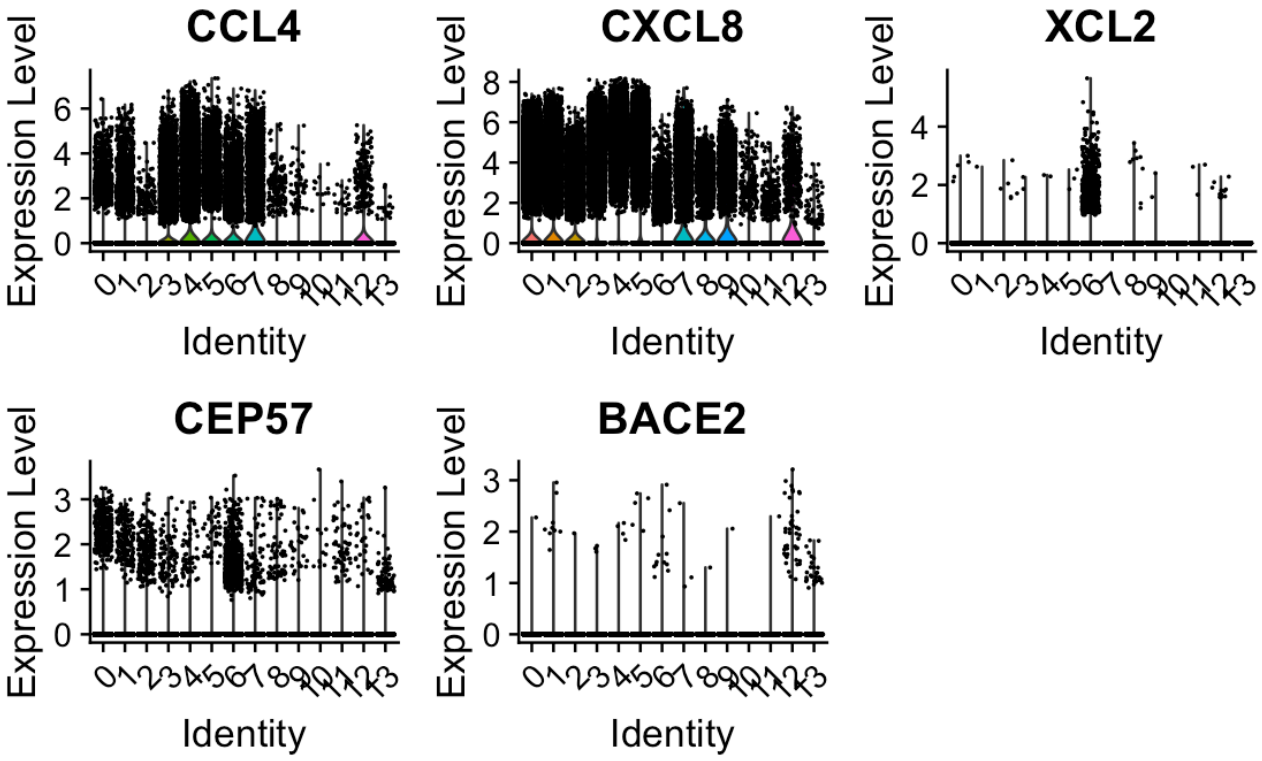
https://satijalab.org/seurat/articles/integration_introduction.html

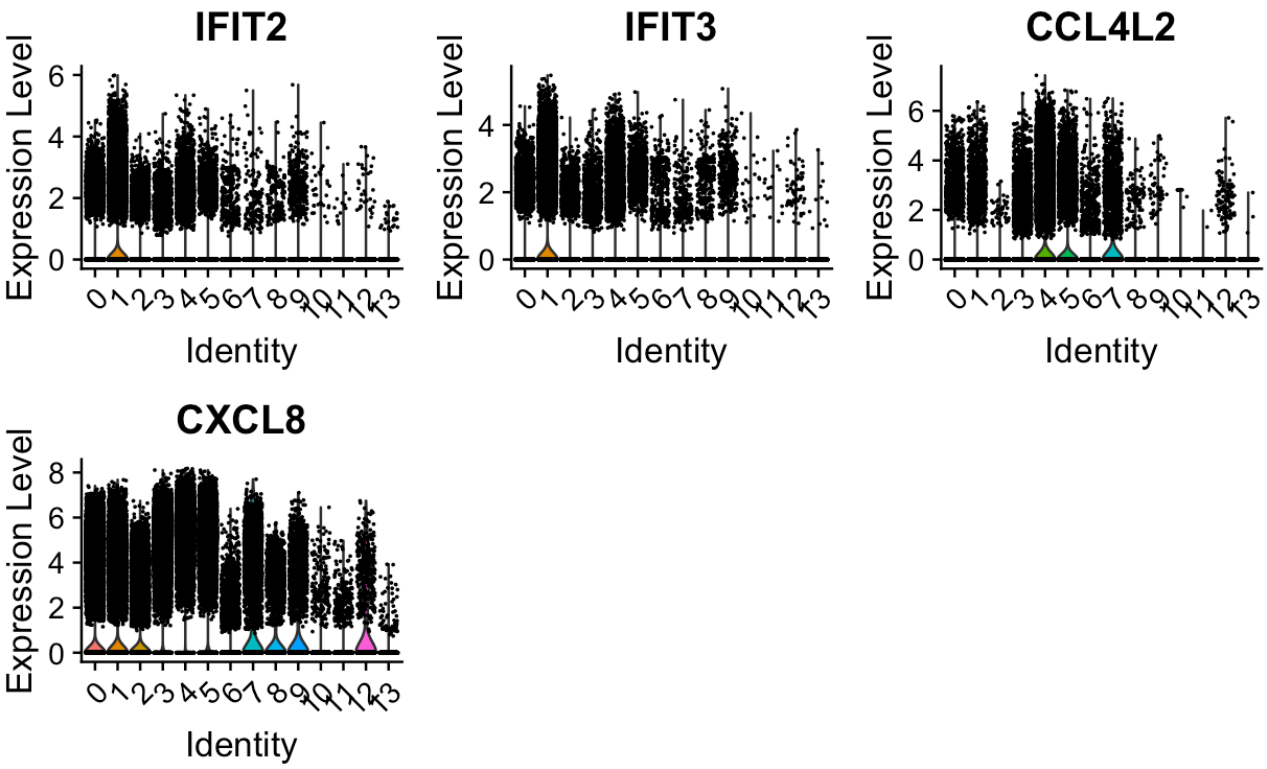
The dataset using in this script is provided in the script called "Preprocessing-and-Filtering-scRNA-Data.md"

```
#load library
library(dplyr)
library(Seurat)
library(patchwork)
library(data.table)
dt <- readRDS("Aim_3/All_Samples_Merged_Filtered.txt.gz")

# find markers for every cluster compared to all remaining cells, report only the
positive o, ""nes
dt.markers <- FindAllMarkers(dt, only.pos = TRUE, min.pct = 0.25, logfc.threshold
= 0.25)
dt.markers_2 <- dt.markers %>% group_by(cluster) %>% top_n(n = 2, wt = avg_log2FC)

# VlnPlot() (shows expression probability distributions across clusters) we are
plotting the top 8 differentiated gene that we are interested in across different
clusters. (These gene hits are provided by script by DE_trevor.md)

v1 <- VlnPlot(dt, features = c("IFIT2", "IFIT3", "CCL4L2", "CXCL8"))
v1

v2 <- VlnPlot(dt, features = c("CCL4", "CXCL8", "XCL2", "CEP57", "BACE2"))
v2
```

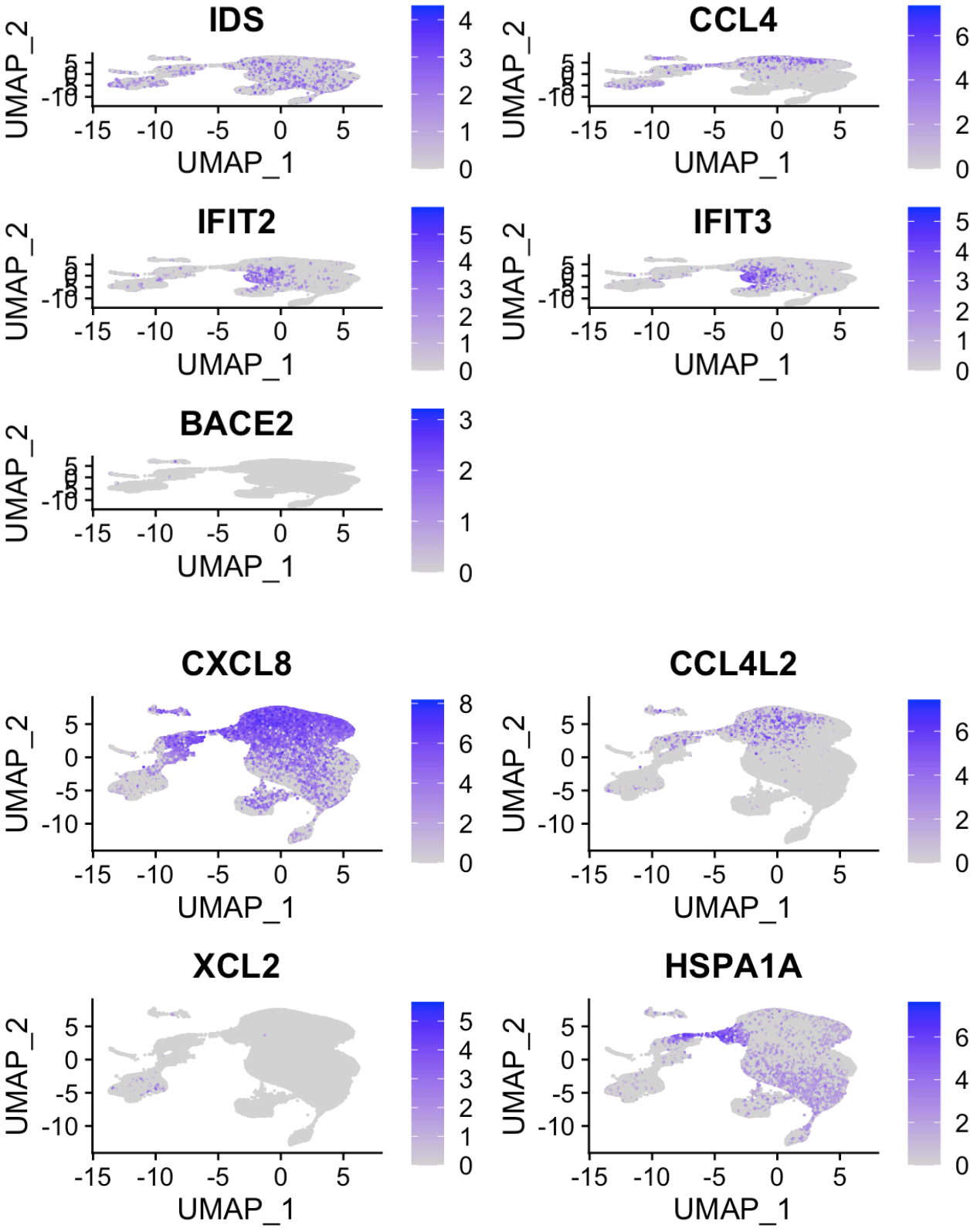
```
#load library
```

```
library(dplyr)
library(Seurat)
library(patchwork)
library(data.table)
dt <- readRDS("Aim_3/All_Samples_Merged_Filtered.txt.gz")

# find markers for every cluster compared to all remaining cells, report only the positive o, ""nes
dt.markers <- FindAllMarkers(dt, only.pos = TRUE, min.pct = 0.25, logfc.threshold = 0.25)
dt.markers_2 <- dt.markers %>% group_by(cluster) %>% top_n(n = 2, wt = avg_log2FC)

# VlnPlot() (shows expression probability distributions across clusters) we are plotting the top 8 differentiated gene that we are interested in across different clusters. (These gene hits are provided by script by DE_trevor.md)

v1 <- VlnPlot(dt, features = c("IFIT2", "IFIT3", "CCL4L2", "CXCL8"))
v1

v2 <- VlnPlot(dt, features = c("CCL4", "CXCL8", "XCL2", "CEP57", "BACE2"))
v2
```

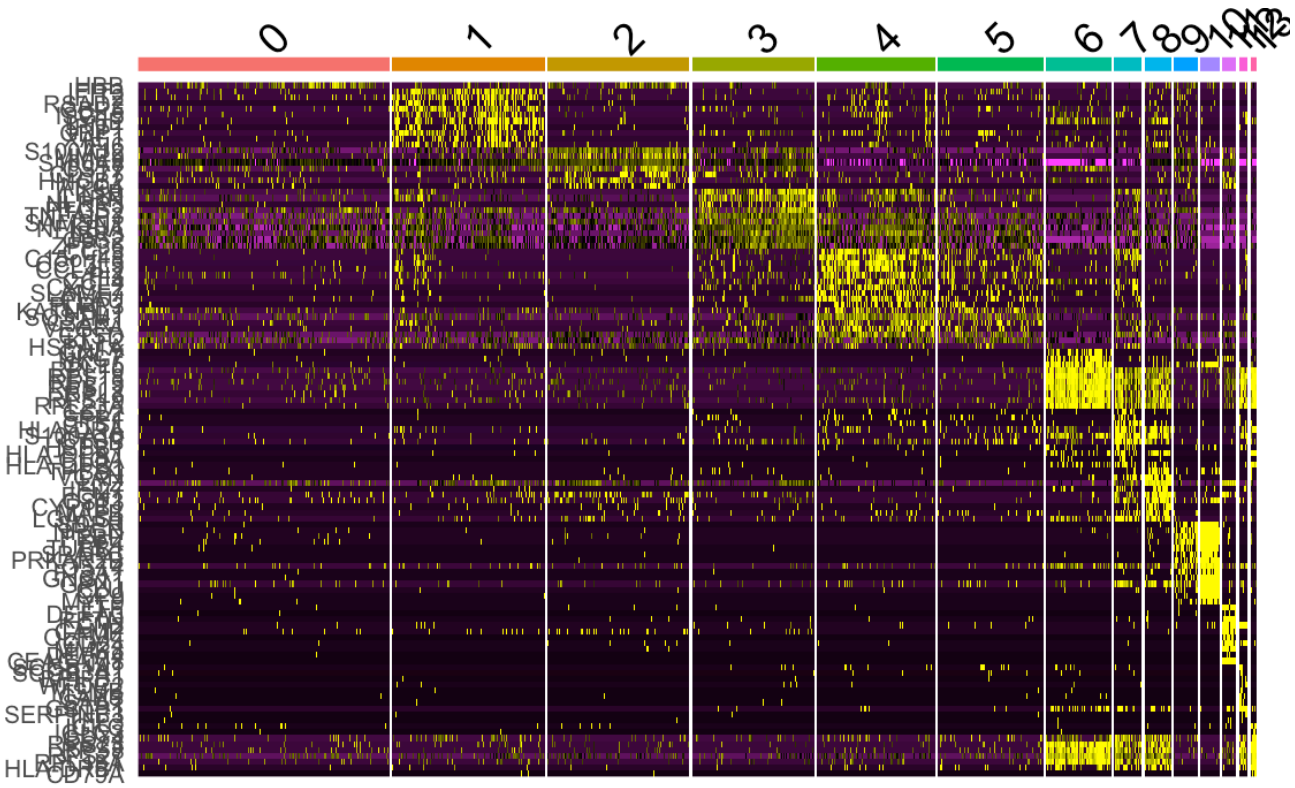




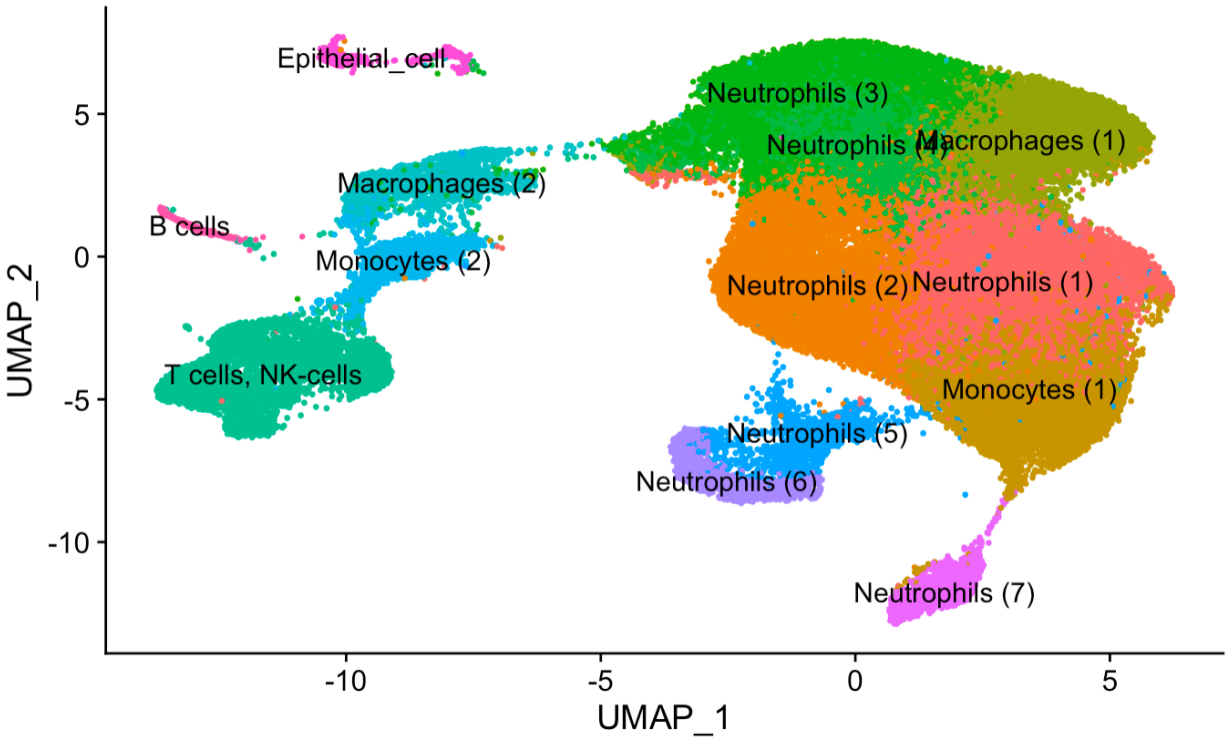

```
# FeaturePlot() helps to visualize gene expression on a dimensional reduction plot
f1 <- FeaturePlot(dt, features = c("CXCL8", "CCL4L2", "XCL2", "HSPA1A", "ACE2"))
f1


f2 <- FeaturePlot(dt, features = c("IDS", "CCL4", "IFIT2", "IFIT3", "BACE2"))
f2
```


IDS
CCL4
IFIT2
IFIT3
BACE2
CXCL8
CCL4L2
XCL2
HSPA1A
UMAP_1
UMAP_2

```
#DoHeatmap() generates an expression heatmap for given cells and features.
top10 <- dt.markers %>% group_by(cluster) %>% top_n(n = 10, wt = avg_log2FC)
DoHeatmap(dt, features = top10$gene) + NoLegend()
```




```
# Unfortunately in the case of this dataset, we cannot use canonical markers to
easily match the unbiased clustering to known cell types: so I did some research
to try to assign cell types for each cluster
new.cluster.ids <- c("Neutrophils (1)", "Neutrophils (2)", "Monocytes (1)",
"Macrophages (1)", "Neutrophils (3)", "Neutrophils (4)", "T cells, NK-cells",
"Macrophages (2)", "Monocytes (2)", "Neutrophils (5)", "Neutrophils (6)",
"Neutrophils (7)","Epithelial_cell", "B cells")


# draw the umap figure showing the cell type for each cell type
names(new.cluster.ids) <- levels(dt)
dt_2 <- RenameIdents(dt, new.cluster.ids)
DimPlot(dt_2, reduction = "umap", label = TRUE, pt.size = 0.5) + NoLegend()
```


Epithelial_cell
Neutrophils (3)
Neutrophils (4)
Macrophages (1)
Macrophages (2)
B cells
Monocytes (2)
Neutrophils (2)
Neutrophils (1)
T cells, NK-cells
Monocytes (1)
Neutrophils (5)
Neutrophils (6)
Neutrophils (7)
UMAP_2
UMAP_1
5
0
-5
-10
-10
-5
0
5