Single Cell Assignment

2024-05-07

Background

The data used is from the following study: https://doi.org/10.1016/j.molcel.2021.10.013. Data was downloaded from GEO using the following accession number: GSE173682.
Specifically, we are focusing on an endometrial cancer tumor that metastasized to the ovary in a human patient. The dataset contains nearly 6000 genes.

Set the working directory, clear environment, and load necessary packages

```
#set working directory
setwd("/Users/cbl284/Desktop/NYU/Classes/Spring2024/SingleCell")

#remove environment
rm(list = ls(all.names = TRUE))

#load packages
library(data.table)
library(dplyr)
library(ggplot2)
library(infercnv)
library(glmGamPoi)
library(Matrix)
library(patchwork)
library(Seurat)
```

Read in the relevant data, and create seurat object

```
#Read genes, cells, metadata, and expression
genes <- fread("./Genes.txt", header = FALSE)
meta <- fread("./Meta-data.csv", header = TRUE)
cells <- fread("./Cells.csv", header = TRUE)
exp <- readMM("./Exp_data_UMIcounts.mtx")

#add genes as rownames of expression matrix
rownames(exp) <- genes$V1
colnames(exp) <- cells$cell_name

#create seurat object
obj <- CreateSeuratObject(counts = exp)
```

```
## Warning: Data is of class dgTMatrix. Coercing to dgCMatrix.
```

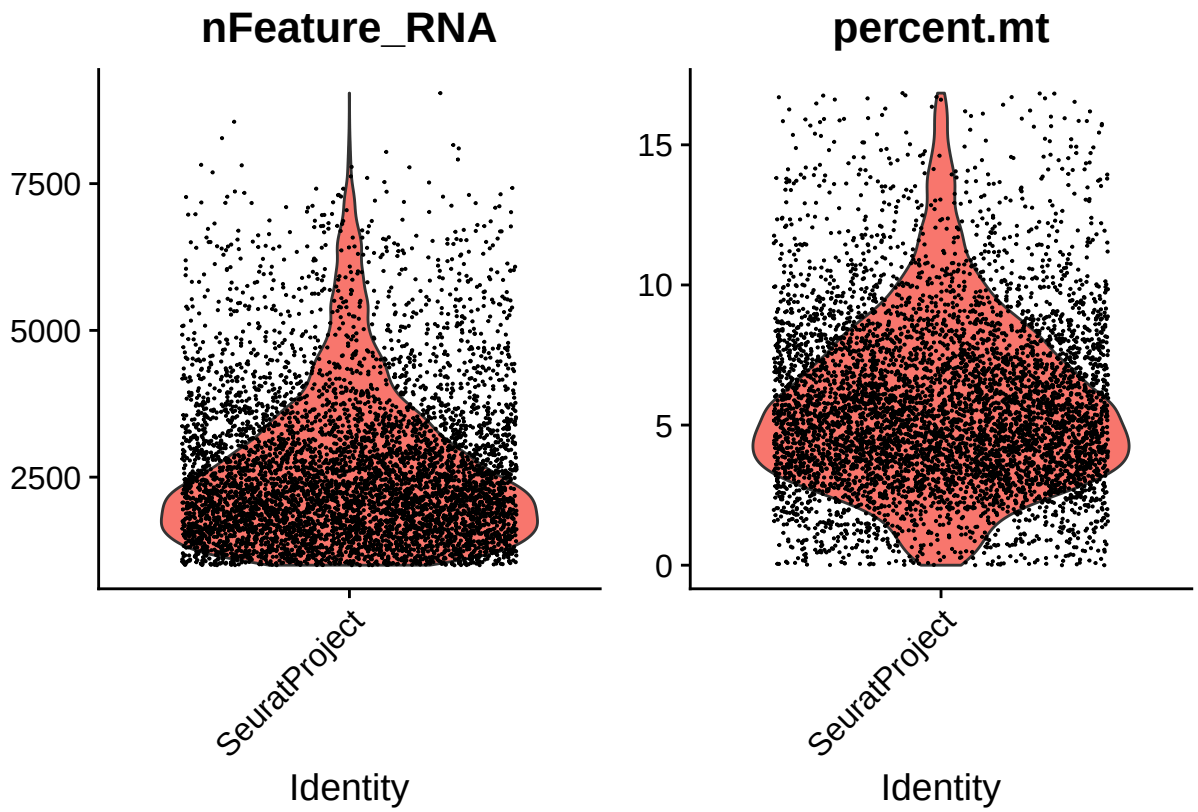
Add metadata (which includes previously annotated cell type labels), save the seurat object

This includes annotated cell type information from the paper. We attempt to annotate the cell types using a reference in a below chunk.

```
#add metadata for cell names and other interesting metadata
rownames(cells) <- cells$cell_name
obj <- AddMetaData(object = obj, metadata = cells)

#add cancer status metadata
cells$CancerStatus <- ifelse(cells$cell_type == "Malignant", "Cancer", "NotCancer")
obj@meta.data$CancerStatus <- cells$CancerStatus

#save object
saveRDS(obj, file="./obj.rds")
```

Quality check

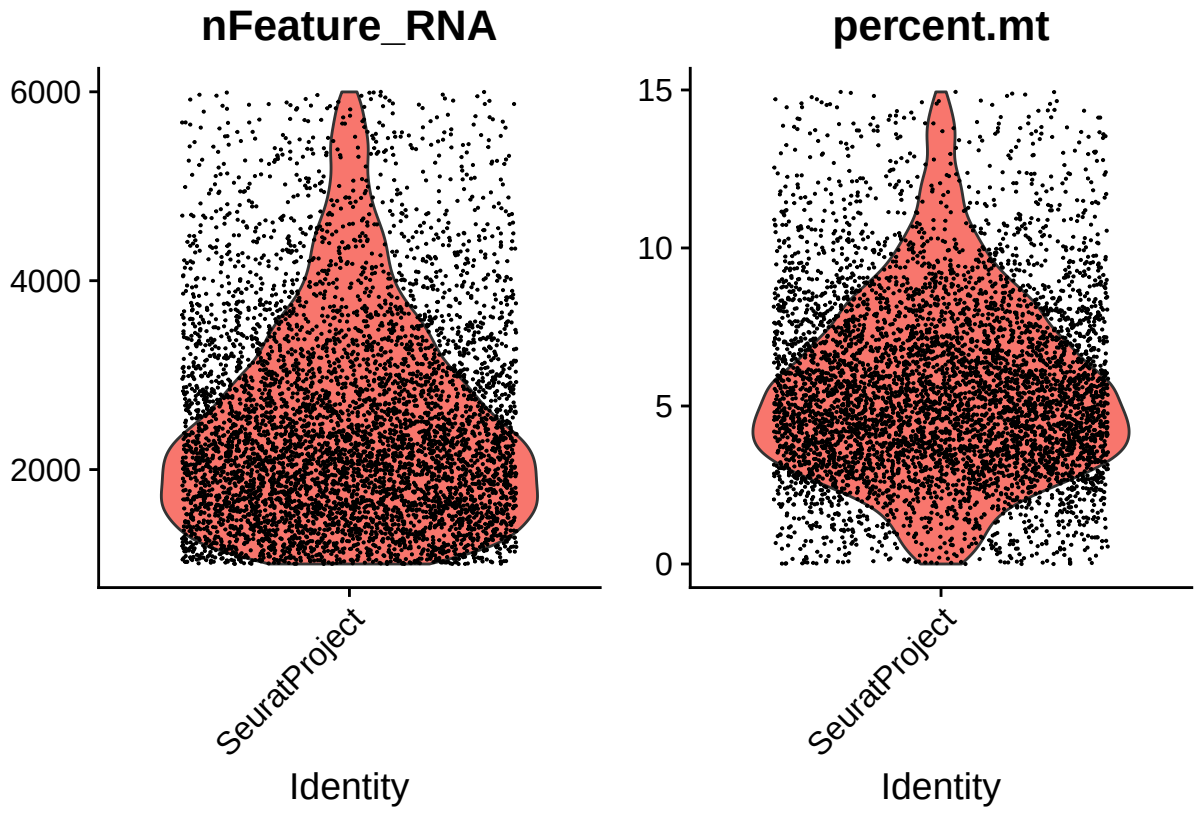
```
#set percent.mt, plot number of unique features and percent.mt
obj <- PercentageFeatureSet(obj, pattern = "^MT-", col.name = "percent.mt")
VlnPlot(obj, features = c("nFeature_RNA", "percent.mt"), ncol = 2)
```



Based on the distribution, we will remove remove cells that have less than 200 unique molecules, more than 6000 unique molucules, and more than 15% mitochondrial reads.

```
#filter seurat object, plot again
obj <- subset(obj, subset = nFeature_RNA > 200 & nFeature_RNA < 6000 & percent.mt < 15)
VlnPlot(obj, features = c("nFeature_RNA", "percent.mt"), ncol = 2)
```

```
## Warning: Default search for "data" layer in "RNA" assay yielded no results;
## utilizing "counts" layer instead.
```




```
#save seurat object
saveRDS(obj, file="./obj_QC.rds")

#filter data for clustering using scVI
write.csv(obj$cell_name, './filtered_cell_names.csv')
```

Process using SCTransform

```
#process with sctransform
obj <- SCTransform(obj, vars.to.regress = "percent.mt", return.only.var.genes = FALSE)
```

```
## Running SCTransform on assay: RNA
```

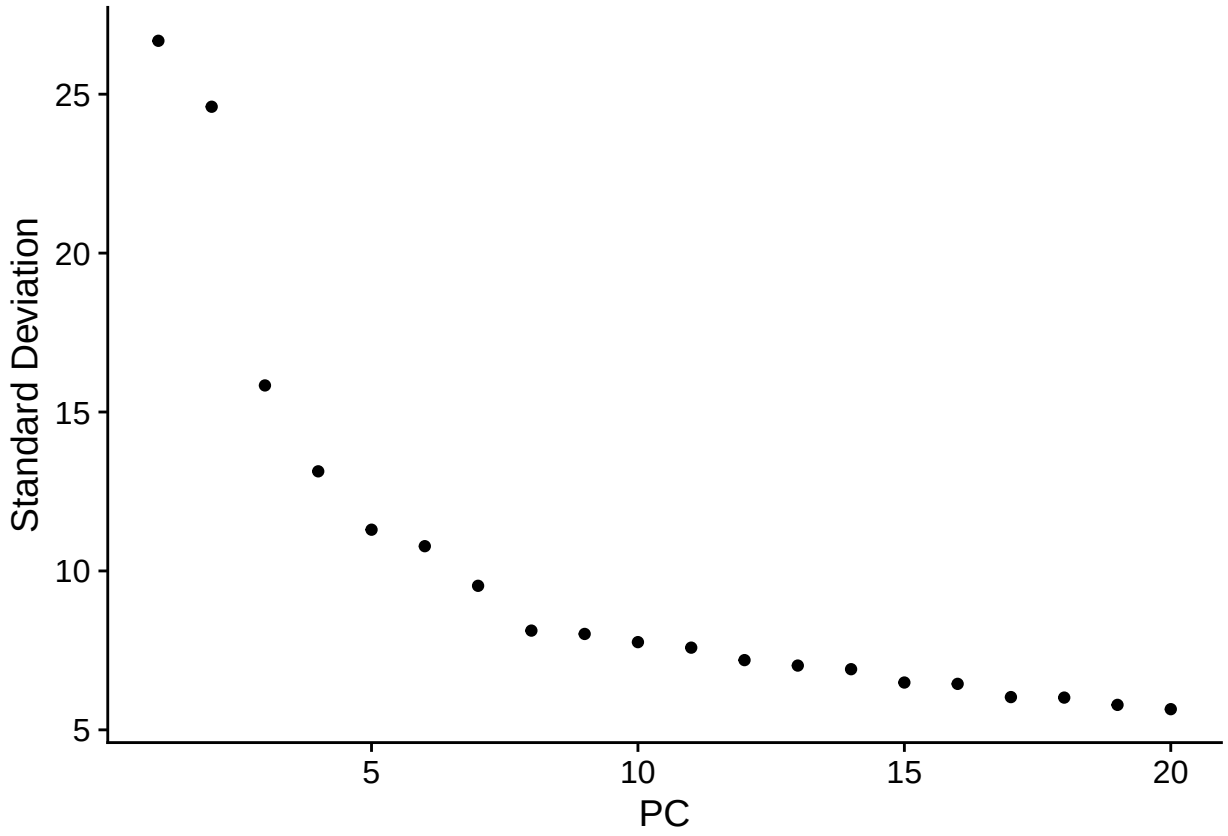
```
## Running SCTransform on layer: counts
```

```
## vst.flavor='v2' set. Using model with fixed slope and excluding poisson genes.

## Variance stabilizing transformation of count matrix of size 18141 by 5748

## Model formula is y ~ log_umi

## Get Negative Binomial regression parameters per gene

## Using 2000 genes, 5000 cells

## Found 64 outliers - those will be ignored in fitting/regularization step

## Second step: Get residuals using fitted parameters for 18141 genes

## Computing corrected count matrix for 18141 genes

## Calculating gene attributes

## Wall clock passed: Time difference of 44.21666 secs

## Determine variable features

## Regressing out percent.mt

## Centering data matrix

## Getting residuals for block 1(of 2) for counts dataset

## Getting residuals for block 2(of 2) for counts dataset

## Regressing out percent.mt

## Centering data matrix

## Finished calculating residuals for counts

## Set default assay to SCT
```

```
#save processed object
saveRDS(obj, file="./obj_sct.rds")
```

Run PCA, plot elbow plot

```
#run pca, plot elbow plot
obj <- RunPCA(obj, verbose = FALSE)
ElbowPlot(obj, ndims = 20, reduction = "pca")
```



Run UMAP, find neighbors, and find clusters

From the elbow plot, it appears that 12 PCs explains the majority of the variance in the data as the slope appears to plateau. Therefore, we will use 12 PCs for the downstream analysis.

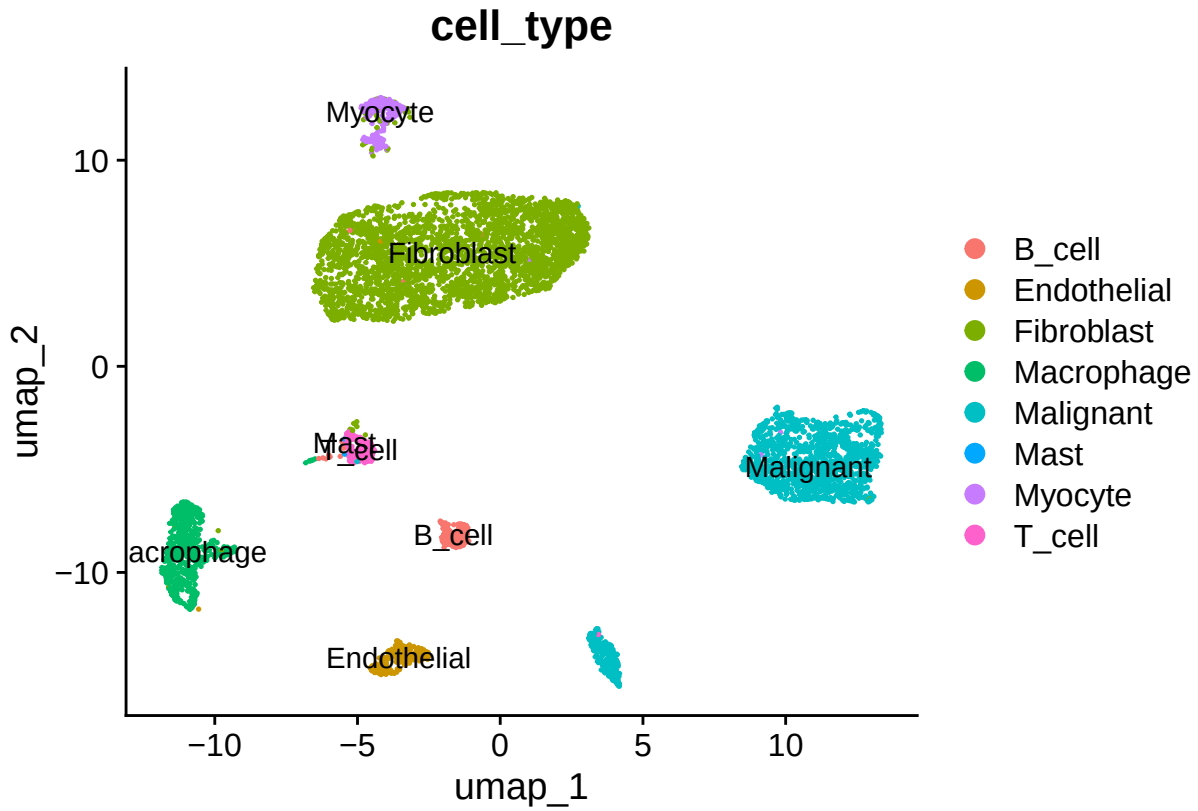
```
#run umap, fing neighbors, and find clusters using 12 PC's (from elbow plot)
obj <- RunUMAP(obj, dims = 1:12, verbose = FALSE)
```

```
## Warning: The default method for RunUMAP has changed from calling Python UMAP via reticulate to the R
## To use Python UMAP via reticulate, set umap.method to 'umap-learn' and metric to 'correlation'
## This message will be shown once per session
```

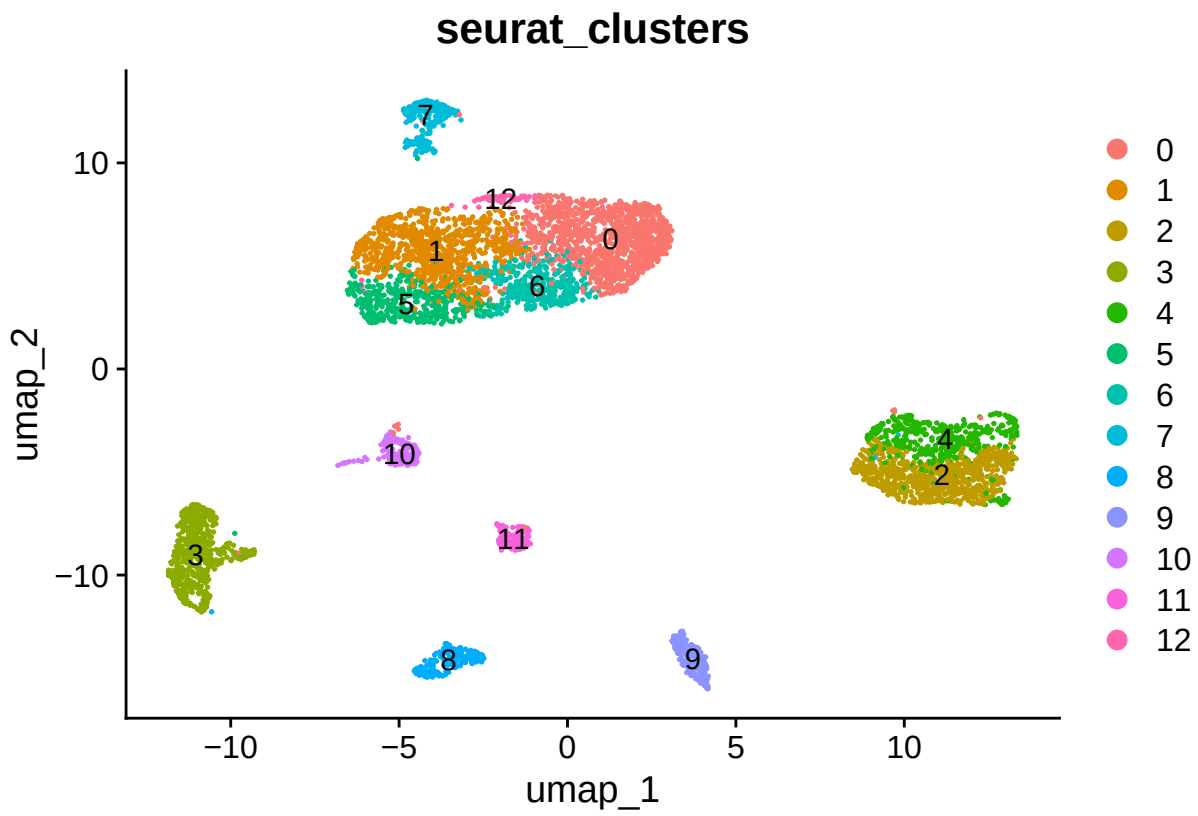
```
obj <- FindNeighbors(obj, dims = 1:12, verbose = FALSE)
obj <- FindClusters(obj, verbose = FALSE)

#save processed object
saveRDS(obj, file="./obj_sct.rds")
```

Plot UMAP, label by cell type and seurat cluster

```
#Plot by cell type
DimPlot(obj, label = TRUE, label.size = 4, group.by="cell_type")
```



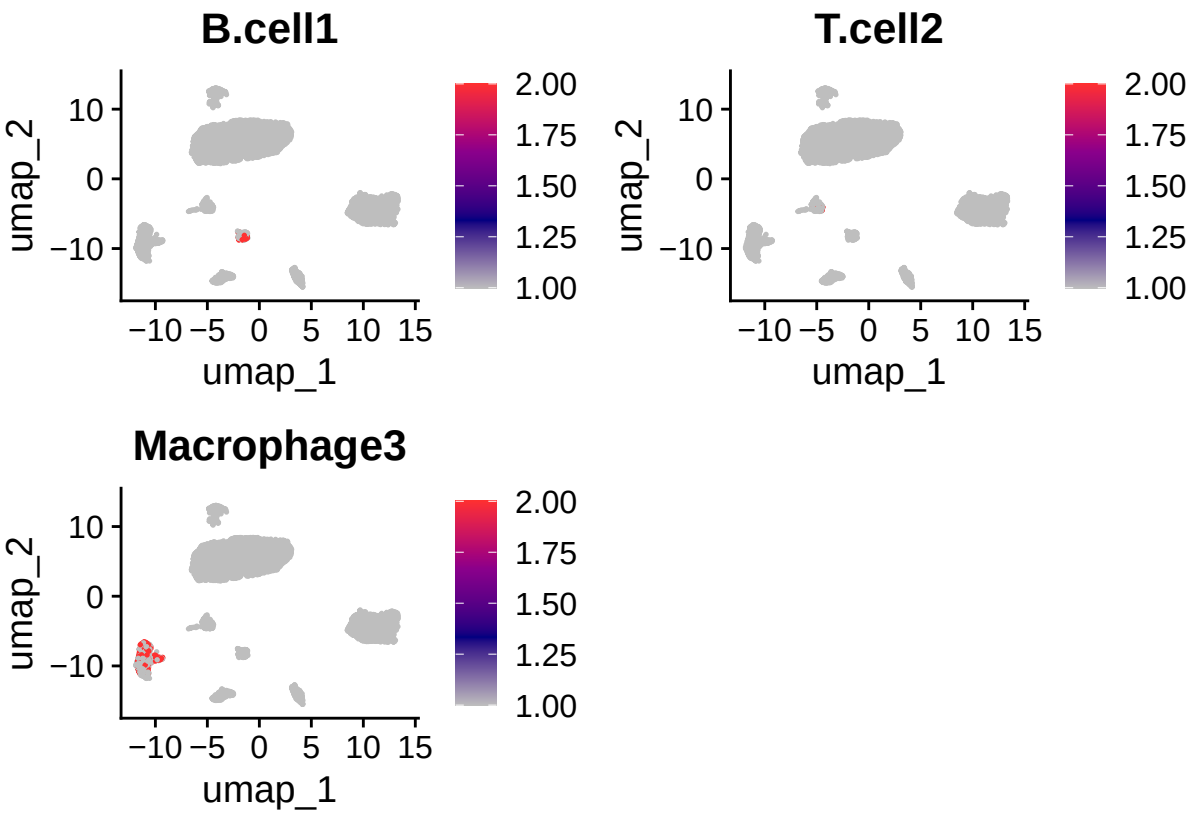

```
#Plot by cluster
DimPlot(obj, label = TRUE, label.size = 4, group.by="seurat_clusters")
```



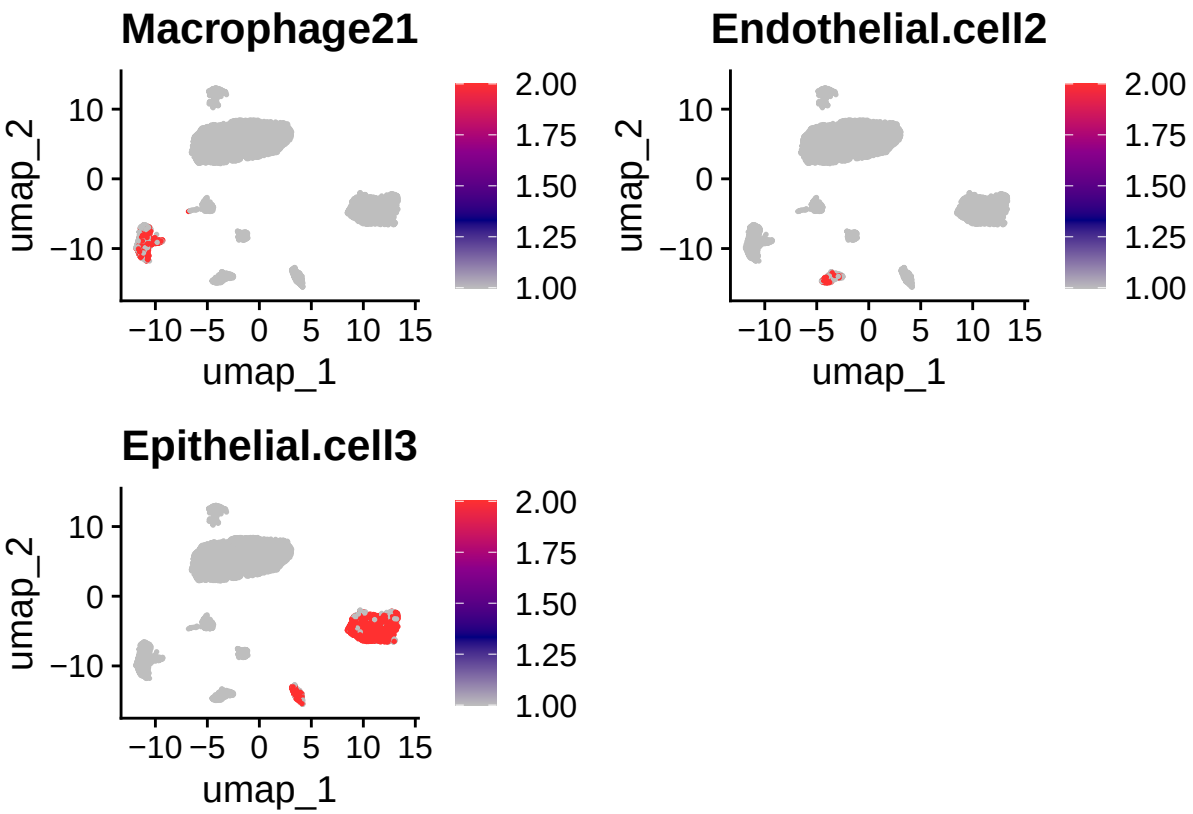
Annotating cells with cell type

There is no database for endometrial normal or cancer tissue that we can find. Therefore, we are using markers from the following study that also looked at endometrial cancer: https://doi.org/10.1038/s41467-022-33982-7.
We are using the AddModuleScore feature to score each cluster using all marker gene sets, and then assigning the highest scoring clusters as that cell type.

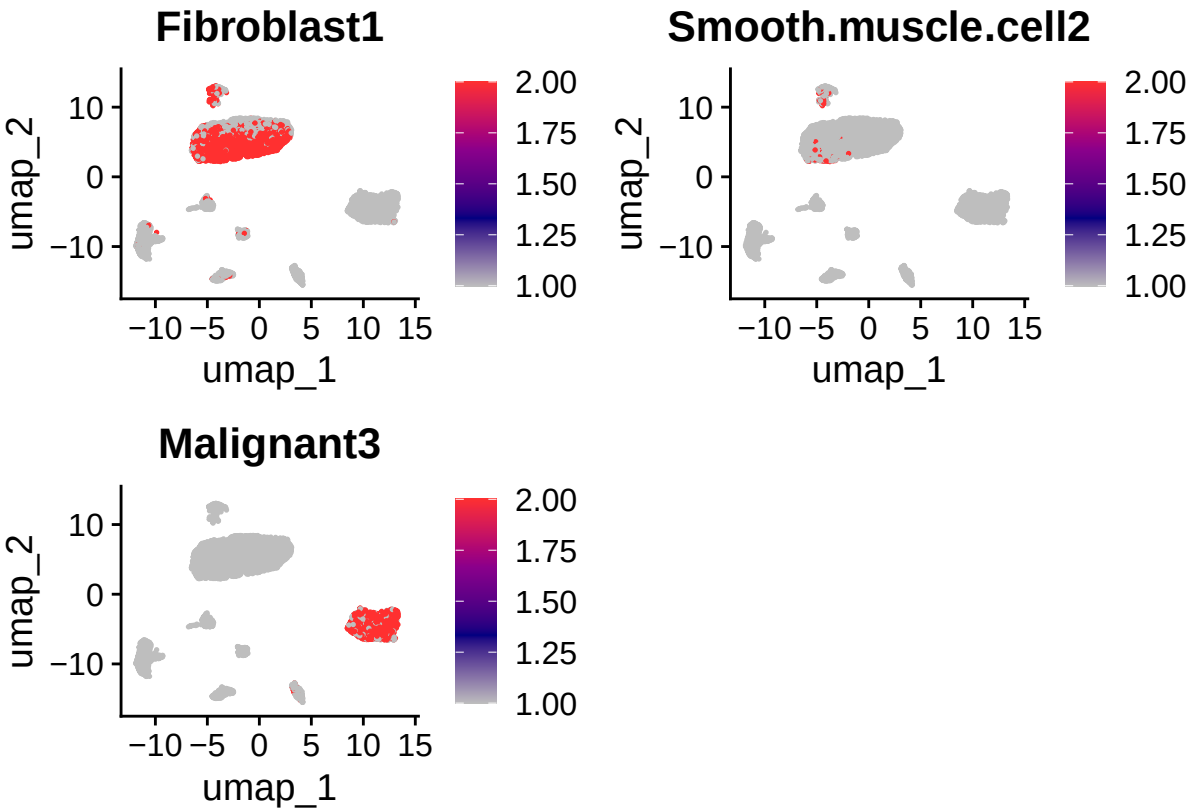
```
#Make a list containing marker genes. Doing this in sets of three to improve visualization.
#Score each cell with the AddModuleScore function for each gene set, and visualize on the UMAP plot.
ec.type <- list()
ec.type[["B.cell"]] <-c("CD79A", "CD79B", "IGKC", "CD14")
ec.type[["T.cell"]] <-c("CD3D", "CD3E", "CD3G", "CD4")
ec.type[["Macrophage"]] <-c("CD14", "FCGR3A", 'FCGR1A', "CD68", "TFRC", "CCR5")
obj <- AddModuleScore(obj, ec.type, name = names(ec.type), assay="SCT")
FeaturePlot(obj, features=paste0(names(ec.type),1:3), cols=c("grey","navy","magenta4","firebrick1"))
```




```
ec.type <- list()
ec.type[["Macrophage2"]] <-c("LYZ","IL1B","CD14","HLA-DQA1","MS4A6A","CD68")
ec.type[["Endothelial.cell"]] <- c("CLDN5", "PECAM1", "VWF", "RNASE1","PECAM1","PCDH17","VWF","ADGRL4")
ec.type[["Epithelial.cell"]]<-c("CDH1","EPCAM","KLF5","KRT7","KRT8","KRT19","PAX8","CD9", "FOXJ1","DYNL
obj <- AddModuleScore(obj, ec.type, name = names(ec.type), assay="SCT")
FeaturePlot(obj, features=paste0(names(ec.type),1:3), cols=c("grey","navy","magenta4","firebrick1"))
```




```
ec.type <- list()
ec.type[["Fibroblast"]] <-c("COL1A1", "COL1A2", "BGN", "DCN", "LUM","DCN","COL6A3","VIM")
ec.type[["Smooth.muscle.cell"]]<-c("ACTA2","MCAM","BGN","NOTCH3","GUCY1A2","RSG5")
ec.type[["Malignant"]] <- c("MUC16","WFDC2")
obj <- AddModuleScore(obj, ec.type, name = names(ec.type), assay="SCT")
FeaturePlot(obj, features=paste0(names(ec.type),1:3), cols=c("grey","navy","magenta4","firebrick1"))
```



The annotated cell types largely correspond to the ones found using the AddModuleScore function. Therefore, we will stick with the annotations provided with the data set as it should not alter our downstream analysis.
Interestingly, we saw three fibroblast clusters in the seurat clustering and the scVI clustering (attached). Both the annotated labels and our own annotations confirm that they are all fibroblasts. We suspect that these are likely different subtypes of fibroblasts that may serve unique niches within the tumor microenvironment.
We repeatedly see two clusters of malignant cells. These may be a different clone, potentially defined by CNVs. Therefore, we will run InferCNV to see if there are any obvious clones.

InferCNV - Prepare input files

InferCNV requires a count matrix of the reference cells (B cells and T cells) and observation cells (malignant cells), and metadata that labels what cells are what cell types.

```
#Subset seurat object for b cells, t cells, and malignant cells
obj <- SetIdent(obj, value = obj@meta.data$cell_type)
obj_cells <- subset(x = obj, idents = c("Malignant", "T_cell", "B_cell"))

#Prepare count matrix
counts_mat = as.matrix(obj_cells@assays$RNA$counts[,colnames(obj_cells)])

#Prepare metadata
sc_meta <- as.data.frame(matrix(nrow=ncol(obj_cells),ncol=1))
rownames(sc_meta) <- colnames(obj_cells)
sc_meta$V1 <- obj_cells$cell_type #cell types of interest
```

```
rownames(sc_meta) <- colnames(obj_cells)
sc_meta$V1 <- sc_meta$V1 #cell types

#Write count matrix and metadata
write.table(round(counts_mat, digits=3), file="./infercnv/sc.10x.counts.matrix", quote=F, sep="\t")
write.table(sc_meta, file="./infercnv/sc.10x.meta.matrix", quote=T, sep="\t", col.names=F)
```

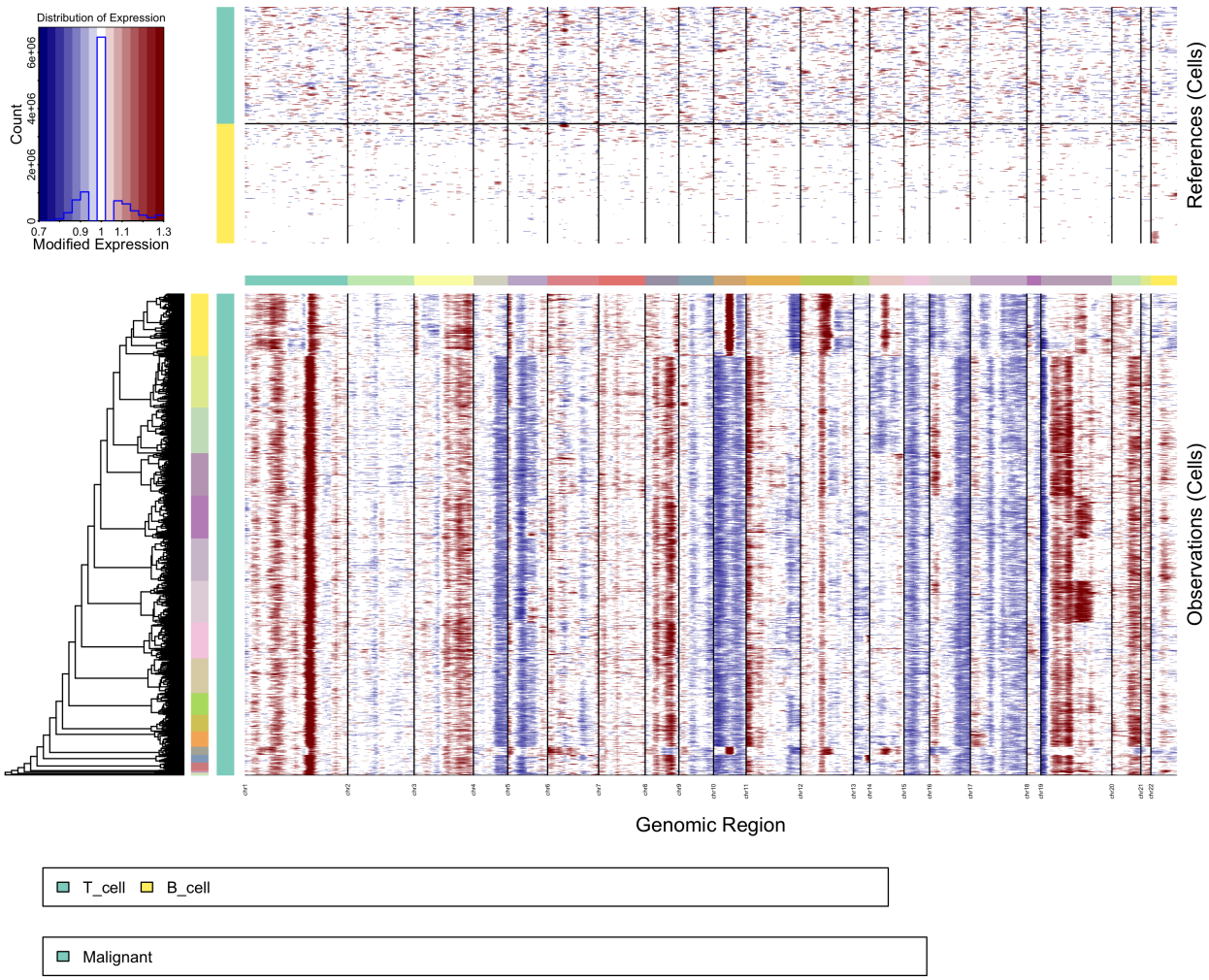
Infercnv - create infercnv object

```
#create infercnv object, run infercnv analysis
infercnv_obj = CreateInfercnvObject(raw_counts_matrix="./InferCNV/sc.10x.counts.matrix",
                                    annotations_file="./InferCNV/sc.10x.meta.matrix",
                                    delim="\t",
                                    gene_order_file="./InferCNV/gencode_v19_gene_pos.txt",
                                    ref_group_names=c("T_cell", "B_cell"))
```

InferCNV - run cnv analysis

```
infercnv_obj = infercnv::run(
  infercnv_obj,
  cutoff=0.1,
  out_dir="./InferCNV/",
  cluster_by_groups=TRUE,
  plot_steps=FALSE,
  denoise=TRUE,
  HMM=TRUE,
  no_prelim_plot=TRUE,
  png_res=300
)
```

There appears to be a clone in the cancer cells (top clone) that has a different CNV profile than the other samples. It does not have loss of Chr4 or Chr5, as all other cells do. This could potentially explain why there are two clusters for malignant cells.


inferCNV
Distribution of Expression
Count
0
2e+06
4e+06
6e+06
0.7
0.9
1
1.1
1.3
Modified Expression
References (Cells)
Observations (Cells)
Genomic Region
T_cell
B_cell
Malignant