lung_reference

2025-07-18

R Markdown

This script prepares a publicly-available single cell RNA sequencing data set, annotated by the cell type, for processing as a reference for clustering using Seurat.

Citation:

Zepp, J. A., [...], Morrisey, E. E. (2021). Genomic, epigenomic, and biophysical cues controlling the emergence of the lung alveolus. Science (American Association for the Advancement of Science), 371(6534). https://doi.org/10.1126/science.abc3172

Click here to download data

```
library(anndata)
library(Seurat)
library(MuDataSeurat)
library(BPCells)
library(reticulate)
library(Seurat)
library(biomaRt)
library(tidyverse)
library(ggpubr)
```

```
## [1] Inf
```

Read data

```
path <- "./data/mouse_atlas/cellXgene.h5ad"
# h5ad file was formatted in Python; convert to a format available to R
numpy <- import('numpy', convert = FALSE)
anndata <- import('anndata', convert = FALSE)
scipy <- import('scipy', convert = FALSE)

adata <- py_to_r(read_h5ad(path))
mat <- as.sparse(adata$X)
# Transpose
tmat <- t(mat)
```

At this point, row names are Ensembl IDs.

To match the intended Seurat object, gene names are needed.

This can be matched using BioMart

```
mart <- useDataset("mmusculus_gene_ensembl", useMart("ensembl"))
gIDs <- row.names(tmat)
gNames <- getBM(filters = "ensembl_gene_id",
                attributes = c(
                  "ensembl_gene_id",
                  "external_gene_name"),
```

```
                values = gIDs,
                mart = mart)

# Some IDs returned results, but no gene names
# In order to use as row names, empty rows will not be compatible
# Fill those in with the ensembl IDs
for (i in 1:nrow(gNames)) {
  if (gNames$external_gene_name[i] == "") {
    gNames$external_gene_name[i] <- gNames$ensembl_gene_id[i]
  }
}
# Some IDs returned nothing and those also need to be added in
# Create a dummy data set with those IDs and add that in
dummy <- cbind(gIDs[which(!(rownames(tmat) %in% gNames$ensembl_gene_id))],
               gIDs[which(!(rownames(tmat) %in% gNames$ensembl_gene_id))])
colnames(dummy) <- colnames(gNames)
gNames <- rbind(gNames, dummy)
# Gene names df needs to match order of the matrix to correctly use as row names
gNames <- gNames[match(rownames(tmat), gNames$ensembl_gene_id),]
# Final check before replacing row names
if (identical(gNames$ensembl_gene_id, rownames(tmat)) == FALSE) {
  stop()
}
rownames(tmat) <- make.unique(gNames$external_gene_name)
```

Create the Seurat object

```
lung <- CreateSeuratObject(counts = tmat,
                           meta.data = adata$obs )
# Intended use is for comparison to adult
# Dropping other developmental stages from the reference will speed processing time
lung <- subset(lung, development_stage == "prime adult stage")
```

Look at the QC metrics

```
plots <- c()
feats <- c("nCount_RNA", "nFeature_RNA", "percent.mito")
# Plot each feature separately
for (i in 1:length(feats)) {
  temp <- VlnPlot(lung,
                  features = feats[i]) +
    theme(axis.text.x = element_blank(),
          axis.title.x = element_blank(),
          legend.position = "none")
  plots <- c(plots, list(temp))
}
```

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

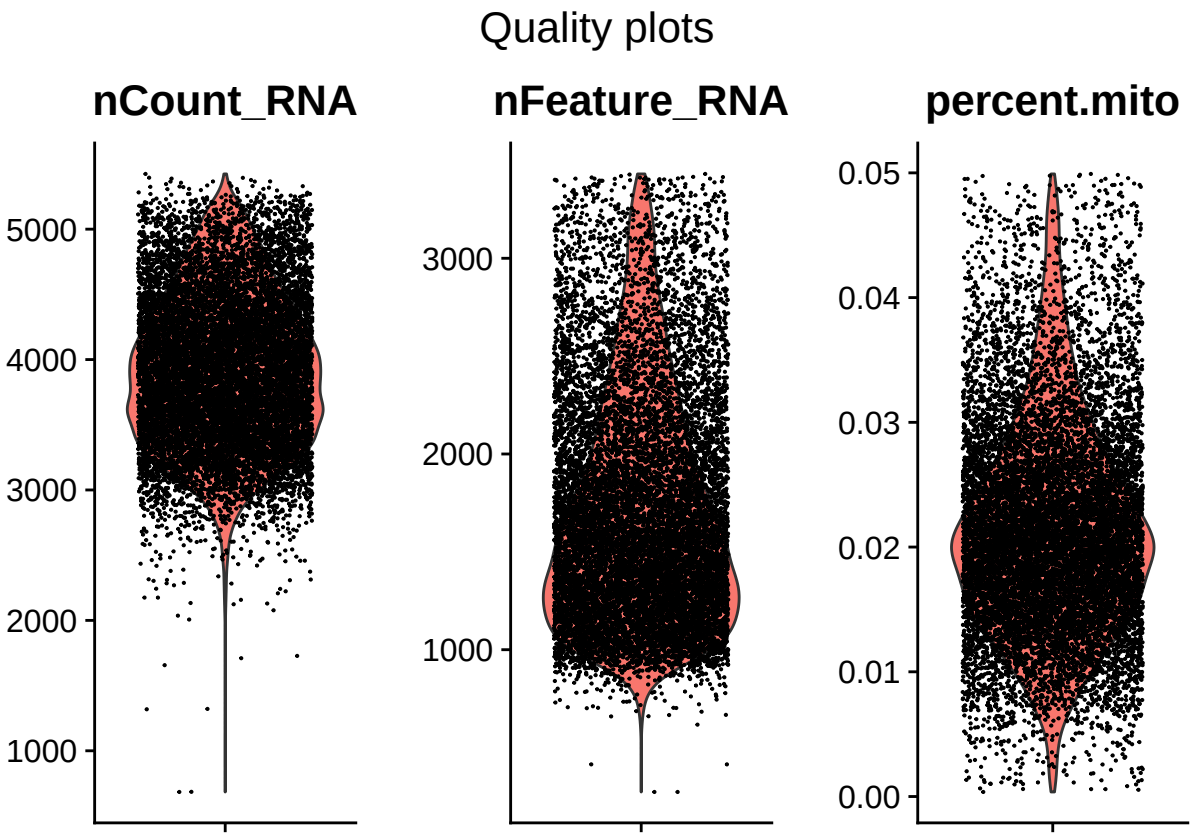
```
## Warning: The `slot` argument of `FetchData()` is deprecated as of SeuratObject 5.0.0.
## i Please use the `layer` argument instead.
## i The deprecated feature was likely used in the Seurat package.
##   Please report the issue at <https://github.com/satijalab/seurat/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
```

```
## generated.
```

```
## Warning: `PackageCheck()` was deprecated in SeuratObject 5.0.0.
## i Please use `rlang::check_installed()` instead.
## i The deprecated feature was likely used in the Seurat package.
##   Please report the issue at <https://github.com/satijalab/seurat/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

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

```
# Generate plot
ggarrange(plotlist = plots, nrow = 1) %>%
  annotate_figure(top = text_grob("Quality plots",
                                  size = 16))
```



The data quality looks good and was very likely pre-filtered before upload.

No additional filtering necessary.

Normalize and scale

```
lung <- NormalizeData(lung)
```

```
## Normalizing layer: counts
```

```
all.genes <- rownames(lung)
lung <- ScaleData(lung, features = all.genes)
```

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

Cluster

```
lung <- FindVariableFeatures(lung)
```

```
## Finding variable features for layer counts
```

```
lung <- RunPCA(lung,
                  features = VariableFeatures(object = lung))
```

```
## PC_ 1
## Positive:  Ehd4, Hpgd, Clec14a, Tspan7, Pecam1, Ramp2, Tspan13, Cldn5, BC028528, Egfl7
##     Calcrl, Clec1a, Pltp, Ctla2a, Aqp1, Ly6e, Eng, Cav2, H2-D1, Srgn
##     Tmem100, Ace, Esam, Rasip1, Ly6a, Acvrl1, Cd9, H2-K1, Cyp4b1, Arhgap31
## Negative:  Rarres2, Fxyd1, Plxdc2, Pcolce, Prelp, Il11ra3, Serping1, Ogn, Cd302, Col1a1
##     Ppp1r14a, Pdgfra, Col6a1, Npnt, Col3a1, Bgn, Tcf21, Hsd11b1, Cxcl14, Cd81
##     Cdo1, Ndrg2, Il11ra1, Aldh1a1, Tmt1a, Celf2, Itga8, Gpc3, Pcolce2, Ms4a4d
## PC_ 2
## Positive:  Nkx2-1, Atp1b1, Cldn18, Chchd10, Krt18, Krt8, Bex4, Lamp3, Cystm1, Cldn3
##     Mal2, Slc39a8, Lgi3, Aqp5, Ctsh, Hc, Sfta2, Muc1, Sftpd, Slc34a2
##     Dram1, Spint2, Alcam, Chia1, Sec14l3, Sftpc, Pi4k2b, Cxcl15, S100g, Car8
## Negative:  Ifitm3, Slc43a3, Plpp3, Thbd, Gng11, Septin4, Prex2, Tcf4, Plxdc2, Pdgfra
##     Tcf21, Tmsb10, Tuba1a, Col13a1, Ifngr1, Fmo2, Ms4a4d, Clic4, Eln, Colec12
##     Crispld2, Col1a1, Gstm1, Tmem100, Itga8, Atp1a2, Jun, Acvrl1, Cxcl14, Fxyd1
## PC_ 3
## Positive:  Pdgfra, Fmo2, Ogn, Gstm1, Cfh, Colec12, Ms4a4d, Ces1d, Gpc3, Col13a1
##     Mgst1, Nebl, Fn1, Slc43a3, Limch1, Efemp1, Trf, Pcolce2, Cp, 1810010H24Rik
##     Rbp1, Gsta3, Loxl1, Crispld2, Slc7a10, Plxdc2, Atp1a2, Clec3b, Scube2, Il11ra3
## Negative:  Higd1b, Ndufa4l2, Agtr1a, Vsnl1, Vtn, Lipg, Kcnk3, Fam162b, Rgs4, Postn
##     Cox4i2, Cstdc2, Gucy1b1, Art3, Tmem178, Ebf1, Gap43, Emid1, Il34, Plxdc1
##     Serpina1b, Nkain4, Des, Ltbp2, Pde5a, Pcdh18, Lmcd1, Tnfrsf21, Trarg1, Foxs1
## PC_ 4
## Positive:  Tgfbi, Cygb, Col14a1, Ccdc80, Olfml2b, Serpinf1, Aspn, Igfbp4, Dcn, Rbp4
##     Scara5, Pdgfrl, Mustn1, Tshz2, Gdf10, Has1, Ifitm3, Gpc6, Timp1, Fxyd6
##     Ccl11, Mt1, Pi16, Smoc2, Fstl1, Aebp1, Myc, Map1b, Dpt, Sfrp1
## Negative:  Hopx, Emp2, Chst1, Septin4, Car4, Ednrb, Nebl, Tbx2, Kitl, Pmp22
##     Ptp4a3, Slc7a10, Tspan8, Prx, Rprml, Tsc22d3, Npnt, Fam174b, Col13a1, Enho
##     Gm14964, Scube2, Tbx3os1, Serpine1, Gyg1, Meox1, Colq, Cyp4b1, Rgs12, Scnn1g
## PC_ 5
## Positive:  Ednrb, Chst1, Clu, Hopx, Car4, Spock2, Fam174b, Pmp22, Cd34, Ptp4a3
##     Igfbp7, Gpm6a, Sema3e, Scnn1g, Col14a1, Tspan8, Timp2, Htra1, Emp2, Col4a3
##     Rtkn2, Gpnmb, Bdnf, Enho, Dcn, Olfml2b, Ccdc80, Igfbp2, Prx, Tacstd2
## Negative:  Sftpc, Lamp3, Hc, Sfta2, H2-Aa, Cldn3, Muc1, Slc34a2, Ifitm3, Apoc1
##     Dram1, Cxcl15, H2-Ab1, Chia1, Sftpd, Car8, Ctsc, Plvap, Cd74, Lgi3
##     S100g, Snhg11, Tmsb10, Bex2, Prex2, Bex4, Ptprb, Cd93, H2-Eb1, Malat1
```

```
lung <- FindNeighbors(lung, dims = 1:16)
```

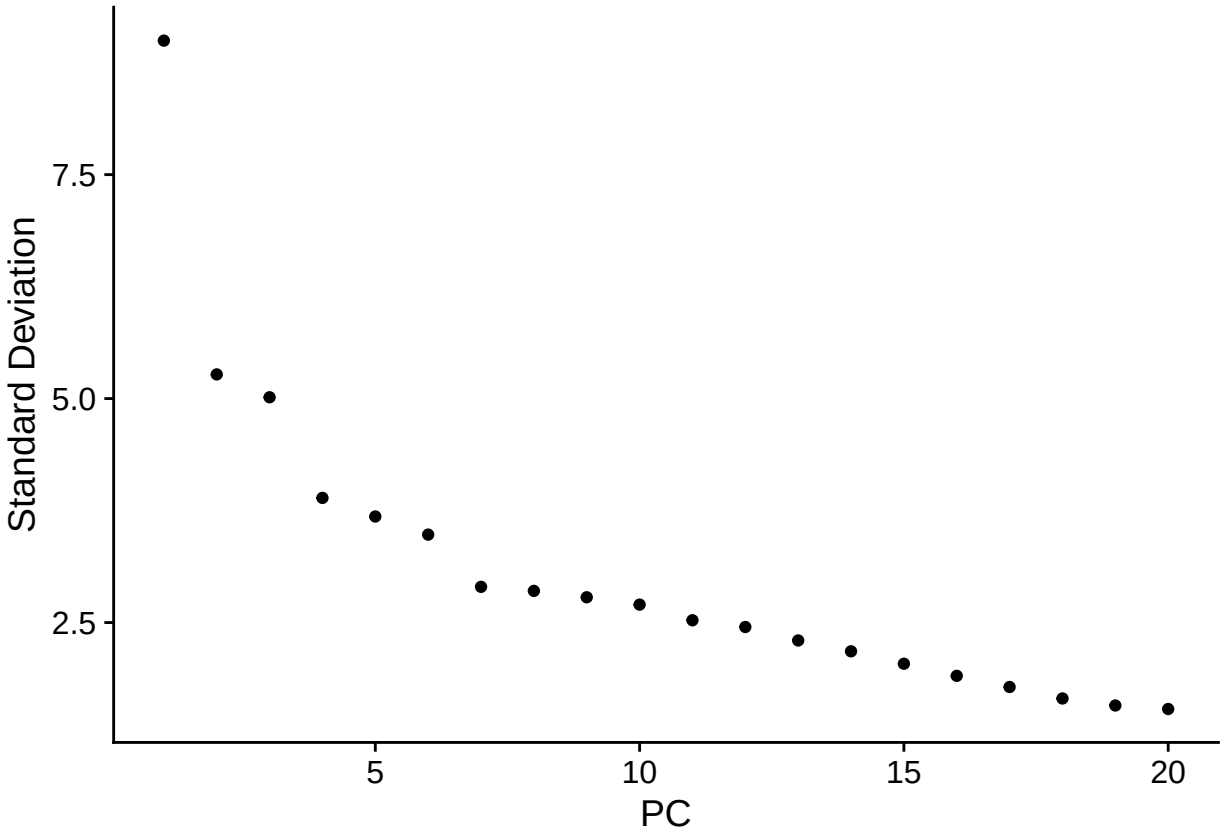
```
## Computing nearest neighbor graph
```

```
## Computing SNN
```

```
lung <- FindClusters(lung, resolution = 0.8)
```

```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 12213
## Number of edges: 454300
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.8875
## Number of communities: 24
## Elapsed time: 1 seconds
```

```
ElbowPlot(lung)
```



The "elbow," point of diminishing returns, happens between the component ~15 and 20. (An ideal elbow plot would have a harder, more defined elbow.) I will use 16.

Dimensional reduction

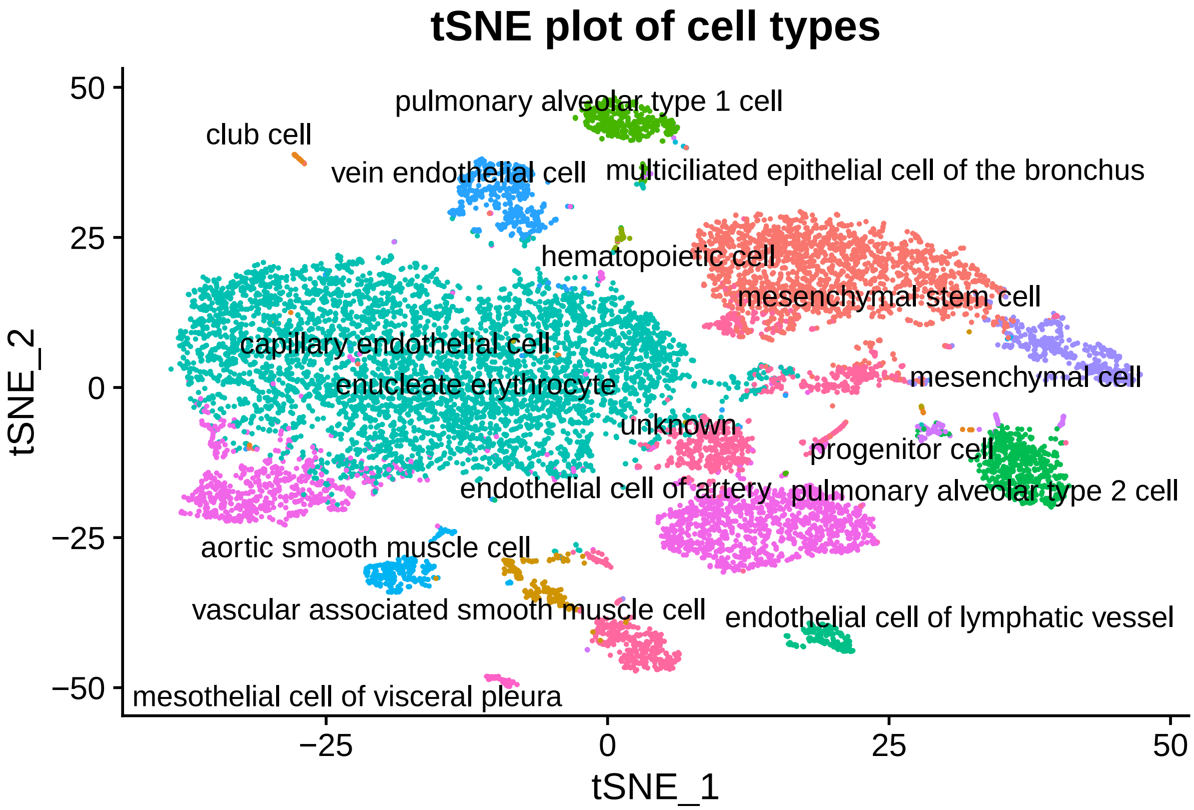
```
lung <- SCTransform(lung)
```

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

```
## Warning: The `slot` argument of `GetAssayData()` is deprecated as of SeuratObject 5.0.0.
## i Please use the `layer` argument instead.
## i The deprecated feature was likely used in the Seurat package.
##   Please report the issue at <https://github.com/satijalab/seurat/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## vst.flavor='v2' set. Using model with fixed slope and excluding poisson genes.
## Calculating cell attributes from input UMI matrix: log_umi
## Variance stabilizing transformation of count matrix of size 16167 by 12213
## Model formula is y ~ log_umi
## Get Negative Binomial regression parameters per gene
## Using 2000 genes, 5000 cells
## Found 3 outliers - those will be ignored in fitting/regularization step
## Second step: Get residuals using fitted parameters for 16167 genes
## Computing corrected count matrix for 16167 genes
## Calculating gene attributes
## Wall clock passed: Time difference of 1.395332 mins
## Determine variable features
## Centering data matrix
## Place corrected count matrix in counts slot
## Warning: The `slot` argument of `SetAssayData()` is deprecated as of SeuratObject 5.0.0.
## i Please use the `layer` argument instead.
## i The deprecated feature was likely used in the Seurat package.
##   Please report the issue at <https://github.com/satijalab/seurat/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
## Set default assay to SCT
lung <- RunUMAP(lung, dims = 1:16)
## 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
## 17:27:56 UMAP embedding parameters a = 0.9922 b = 1.112
## 17:27:56 Read 12213 rows and found 16 numeric columns
## 17:27:56 Using Annoy for neighbor search, n_neighbors = 30
## 17:27:56 Building Annoy index with metric = cosine, n_trees = 50
## 0%   10   20   30   40   50   60   70   80   90   100%
## [----|----|----|----|----|----|----|----|----|----|
## **************************************************|
## 17:27:57 Writing NN index file to temp file /var/folders/84/7fnlc30d2dl92_2kyb8f3q9h0000gn/T//RtmpFd
## 17:27:57 Searching Annoy index using 1 thread, search_k = 3000
## 17:28:00 Annoy recall = 100%
## 17:28:03 Commencing smooth kNN distance calibration using 1 thread with target n_neighbors = 30
## 17:28:04 Initializing from normalized Laplacian + noise (using RSpectra)
## 17:28:05 Commencing optimization for 200 epochs, with 527204 positive edges
## 17:28:05 Using rng type: pcg
## 17:28:08 Optimization finished
```

```r
lung <- RunTSNE(lung, dims = 1:16)
DimPlot(lung,
        reduction = "tsne",
        repel = TRUE,
        label = TRUE,
        group.by = "cell_type",
        ) +
  NoLegend() +
  ggtitle("tSNE plot of cell types")
```



Export data

```r
saveRDS(lung, "./data/mouse_atlas/lung_reference.rds")
```

```r
Sys.info()
```

```
##                                                                                                    sysn
##                                                                                                   "Darw
##                                                                                                    rele
##                                                                                                   "24.5
##                                                                                                    vers
## "Darwin Kernel Version 24.5.0: Tue Apr 22 19:54:26 PDT 2025; root:xnu-11417.121.6~2/RELEASE_ARM64_T81
##                                                                                                   noden
##                                                                                  "Emilys-MacBook-Air.loc
##                                                                                                    mach
##                                                                                                    "arm
##                                                                                                      lo
##                                                                                                     "ro
##                                                                                                       u
```

```
##                                                                                              "emilykibbl
##                                                                                              effective_u
##                                                                                              "emilykibbl
```