SA03 - Subclusterings of epithelial cells

Stefan Siebert

November 1, 2018, updated on May 25, 2019

Summary

We subcluster and curate epithelial cells from both the endodermal and the ectodermal lineages to obtain lineage specific t-SNE plots. Batch effects are addressed. The resulting data objects are the starting point for URD trajectory reconstructions.

Preliminaries

```
library(Seurat)
library(dplyr)
library(Matrix)
library(gtable)
library(grid)
library(gridExtra)
library(rlang)

# We assume a folder 'objects' in the markdown directory that contains our raw
# count object and all Seurat objects
```

Subsetting - ectodermal cells

We load the full data set and extract ectodermal epithelial clusters. We consider cells that express more than 500 genes and include the battery cell clusters but omit body column clusters that are also positive for nematoblast/nematocyte expression (Fig. 1). This subset was used to perform non-negative matrix factorization (NMF) to identify metagenes expressed in ectodermal epithelial cells (SA07_nmf.rmd, NMF analysis ec_K76).

```
# Read full data object
ds.s1 <- readRDS("objects/Hydra_Seurat_Whole_Transcriptome.rds")

# Suspected doublet cluster db (68 cells) was excluded from downstream analyses
ds.s1 <- SubsetData(object = ds.s1, ident.remove = c("db"), subset.raw = TRUE)

# Run this to restore original cluster numbering
ds.s1 <- SetAllIdent(object = ds.s1, id = "cluster_numbering")

# Ectodermal clusters excluding doublet clusters but including the tentacle
# cluster (battery cell)
ds.ec <- SubsetData(object = ds.s1, ident.use = c("2", "3", "10", "26", "37", "14"),
    subset.raw = TRUE)
length(ds.ec@meta.data$nGene)

p1 <- TSNEPlot(object = ds.s1, group.by = "res.1.5", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- TSNEPlot(object = ds.ec, group.by = "res.1.5", do.return = T, do.label = T,
    no.legend = TRUE)

plot_grid(p1, p2, ncol = 2, labels = "AUTO", label_size = 20, align = "h")
```

Batch effects

We observe library specific effects (batch) in epithelial cells which are more pronounced in case of the ectodermal subset. There are potential sources for such effects. Drop-seq runs were conducted on different days and the culture may have been in a different state despite maintaining standard culturing conditions. Four libraries included sexually reproducing polyps and the physiological state may be reflected in the transcriptional signatures of ectodermal epithelial cells. In some *Hydra* strains sexual reproduction leads to dramatic exhaustion and even death (*1*). Additional confounding effects may have been introduced with the choice of medium used in the dissociations. The first three sets of libraries (01-, 02-, 03-) were generated using cells that were dissociated in *Hydra* culture medium. The remaining libraries were generated using cells that were dissociated using isotonic *Hydra* dissociation medium (see Material & Methods).

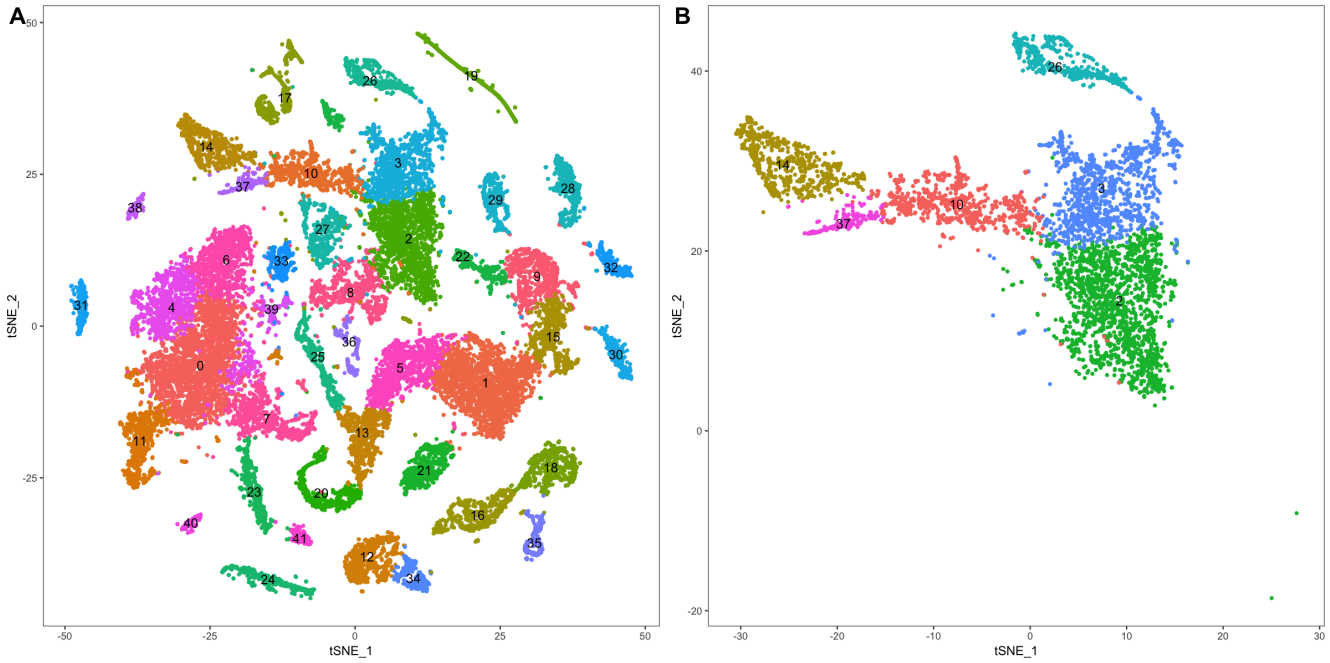

Figure 1: Subset of endodermal cells. A) t-SNE plot for all cells in the data set. B) Endodermal clusters.

To reduce the number of confounding effects we here consider cells from five selected libraries in the subclustering of ectodermal cells. Three libraries were collected on two consecutive days using ChemGenes beads and using *Hydra* culture medium in the dissociations (libraries 02-). Two additional libraries were collected on a single day using Biosearch beads and using *Hydra* dissociation medium to generate the cell suspensions (libraries 11-). No obvious batch effects are observed when clustering cells from experiments 02- and 11- separately. An additional NMF analysis was performed using this subset of cells (ec_K79).

Clustering of cells

We first cluster the cells without any batch regression to evaluate library set specific effects. Cells from 02- libraries separate from cells from 11- libraries (Fig. 2). We are able to integrate the two sets of cells following the approach described by Buttler et al. (*2*)(Fig. 3).

```
# Retrieve cell ids for selected libraries
p3 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-CO"]
p4 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-P1"]
p5 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-PB"]

p12 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "11-PO"]
p13 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "11-BU"]

# Combine cell ids from libraries 02- and 11-
sel <- c(p3, p4, p5, p12, p13)

# Create object for subset
ds.ec <- SubsetData(object = ds.ec, cells.use = sel, subset.raw = TRUE)

# Identify highly variable genes
ds.ec <- FindVariableGenes(object = ds.ec, mean.function = ExpMean, dispersion.function = LogVMR,
    x.low.cutoff = 0.1, x.high.cutoff = 4, y.cutoff = 0.7)

# Scale
ds.ec <- ScaleData(object = ds.ec)

# Run PCA on highly variable genes
ds.ec <- RunPCA(object = ds.ec, pc.genes = ds.ec@var.genes, pcs.compute = 40, do.print = TRUE,
    pcs.print = 1:5)
```

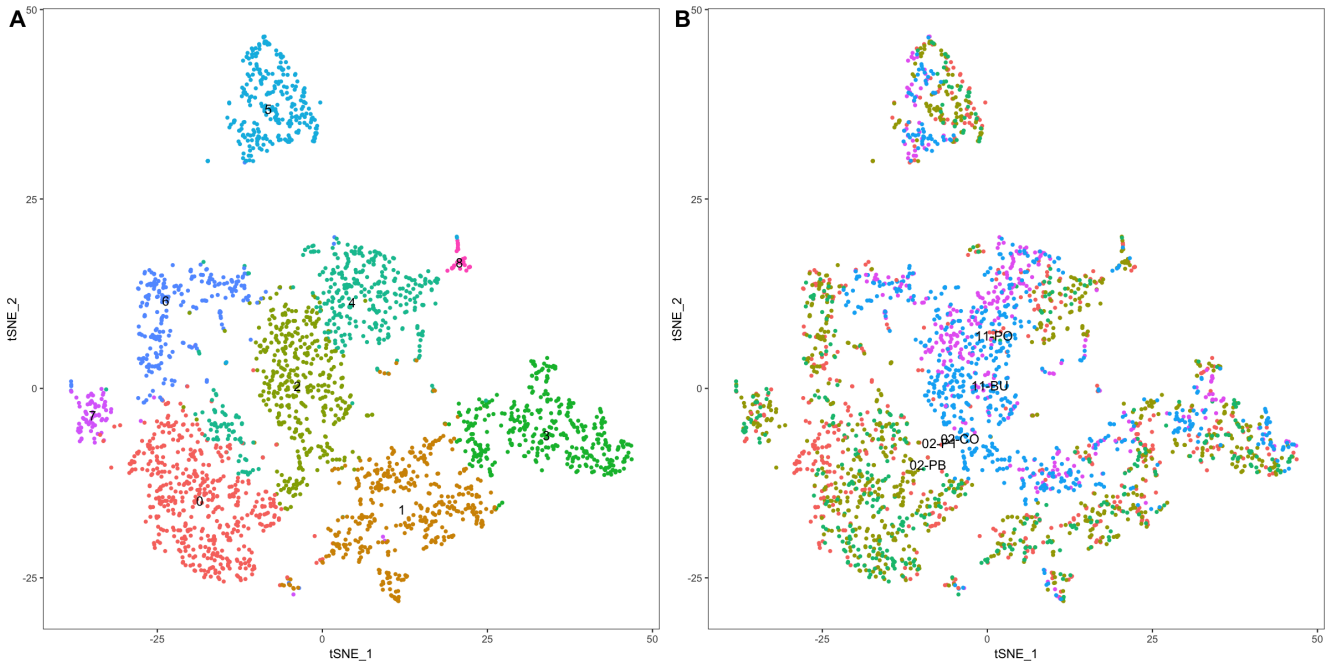

Figure 2: Clustering without batch regression reveals batch effects in epithelial cells of the ectodermal subset. A) t-SNE plot with clusters labeled. B) t-SNE plot with cells labeled by batch. Cells from libraries 02- separate from cells from libraries 11-.

```
# Project PCA
ds.ec <- ProjectPCA(object = ds.ec)

# determine statistically significant PCs ds.ec <- JackStraw(object = ds.ec,
# num.pc = 40, num.replicate = 100, do.print = FALSE) JackStrawPlot(object =
# ds.ec, PCs=1:40)

# Look at a plot of the standard deviations of the principle components and draw
# cutoff where there is a clear elbow in the graph.
PCElbowPlot(object = ds.ec, num.pc = 40)

# Find cluster
ds.ec <- FindClusters(object = ds.ec, reduction.type = "pca", dims.use = 1:14, force.recalc = TRUE,
    resolution = 0.6, print.output = 0)
# Run t-SNE
ds.ec <- RunTSNE(object = ds.ec, dims.use = c(1:14), do.fast = T)

# Save object saveRDS(ds.ec,'objects/ds.ec.pca_libr2_11.rds')
```

```
# Load object from original analysis: ds.ec <-
# readRDS('objects/ds.ec.pca_libr2_11.rds')

p1 <- TSNEPlot(object = ds.ec, group.by = "res.0.6", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- TSNEPlot(object = ds.ec, group.by = "orig.ident", do.return = T, do.label = T,
    no.legend = TRUE)

plot_grid(p1, p2, ncol = 2, labels = "AUTO", label_size = 20, align = "h")
```

```
# Integration of 02- and 11- cells following approach by Butler et al., 2018

# Combine cells from 02- libraries
l02 <- c(p3, p4, p5)

# Combine cells from 11- libraries
l11 <- c(p12, p13)

# Create subsets
```

```
l11 <- SubsetData(object = ds.ec, cells.use = l11, subset.raw = TRUE)
l02 <- SubsetData(object = ds.ec, cells.use = l02, subset.raw = TRUE)

# We consider cells expressing a minimum of 500 genes

# Filter, normalize, scale
l11@meta.data$stim <- "l11"
l11 <- FilterCells(l11, subset.names = "nGene", low.thresholds = 500, high.thresholds = Inf)
l11 <- NormalizeData(l11)
l11 <- ScaleData(l11, display.progress = F)

# Filter, normalize, scale
l02@meta.data$stim <- "l02"
l02 <- FilterCells(l02, subset.names = "nGene", low.thresholds = 500, high.thresholds = Inf)
l02 <- NormalizeData(l02)
l02 <- ScaleData(l02, display.progress = F)

# Gene selection for input to CCA
l11 <- FindVariableGenes(l11, do.plot = F)
l02 <- FindVariableGenes(l02, do.plot = F)
g.1 <- head(rownames(l11@hvg.info), 1000)
g.2 <- head(rownames(l02@hvg.info), 1000)
genes.use <- unique(c(g.1, g.2))
genes.use <- intersect(genes.use, rownames(l11@scale.data))
genes.use <- intersect(genes.use, rownames(l02@scale.data))

# Run CCA
ds.ec.ss <- RunCCA(l11, l02, genes.use = genes.use, num.cc = 30)

# Visualize results of CCA plot CC1 versus CC2 and look at a violin plot p1 <-
# DimPlot(object = ds.ec.ss, reduction.use = 'cca', group.by = 'stim', pt.size =
# 0.5, do.return = TRUE) p2 <- VlnPlot(object = ds.ec.ss, features.plot = 'CC1',
# group.by = 'stim', do.return = TRUE) plot_grid(p1, p2)

# Explore gene loadings PrintDim(object = ds.ec.ss, reduction.type = 'cca',
# dims.print = 1:2, genes.print = 100)

# Explores CCs
p3 <- MetageneBicorPlot(ds.ec.ss, grouping.var = "stim", dims.eval = 1:30, display.progress = FALSE)

# DimHeatmap(object = ds.ec.ss, reduction.type = 'cca', cells.use = 100, dim.use
# = 1, do.balanced = TRUE, margins=c(12,16))

# Align subspaces
ds.ec.ss <- AlignSubspace(ds.ec.ss, reduction.type = "cca", grouping.var = "stim",
    dims.align = 1:20)

# p1 <- VlnPlot(object = ds.ec.ss, features.plot = 'ACC1', group.by = 'stim',
# do.return = TRUE) p2 <- VlnPlot(object = ds.ec.ss, features.plot = 'ACC2',
# group.by = 'stim', do.return = TRUE) plot_grid(p1, p2)

ds.ec.ss <- RunTSNE(ds.ec.ss, reduction.use = "cca.aligned", dims.use = 1:20, do.fast = T)

ds.ec.ss <- FindClusters(ds.ec.ss, reduction.type = "cca.aligned", resolution = 1.2,
    dims.use = 1:20)

# Save object saveRDS(ds.ec.ss,'objects/ds.ec.cca_l2_l11_int.rds')
```

```
# Load object from original analysis ds.ec.ss <-
# readRDS('objects/ds.ec.cca_l2_l11_int.rds')

p1 <- TSNEPlot(ds.ec.ss, do.label = T, do.return = T, pt.size = 0.5, group.by = "res.1.2")
p2 <- TSNEPlot(ds.ec.ss, do.return = T, pt.size = 0.5, group.by = "stim")

plot_grid(p1, p2, ncol = 2, labels = "AUTO", label_size = 20, align = "h")
```

We use the metagene scores from the whole dataset NMF analysis (wt_K96) to identify signatures of contamination within the population of ectodermal cells. Cells from cluster 12 are strongly positive for metagenes expressed in endodermal epithelial cells (Fig. 4). We exclude cells from this cluster and recluster the cells (Fig. 5). The resulting set of cells was used for URD trajectory reconstruction.

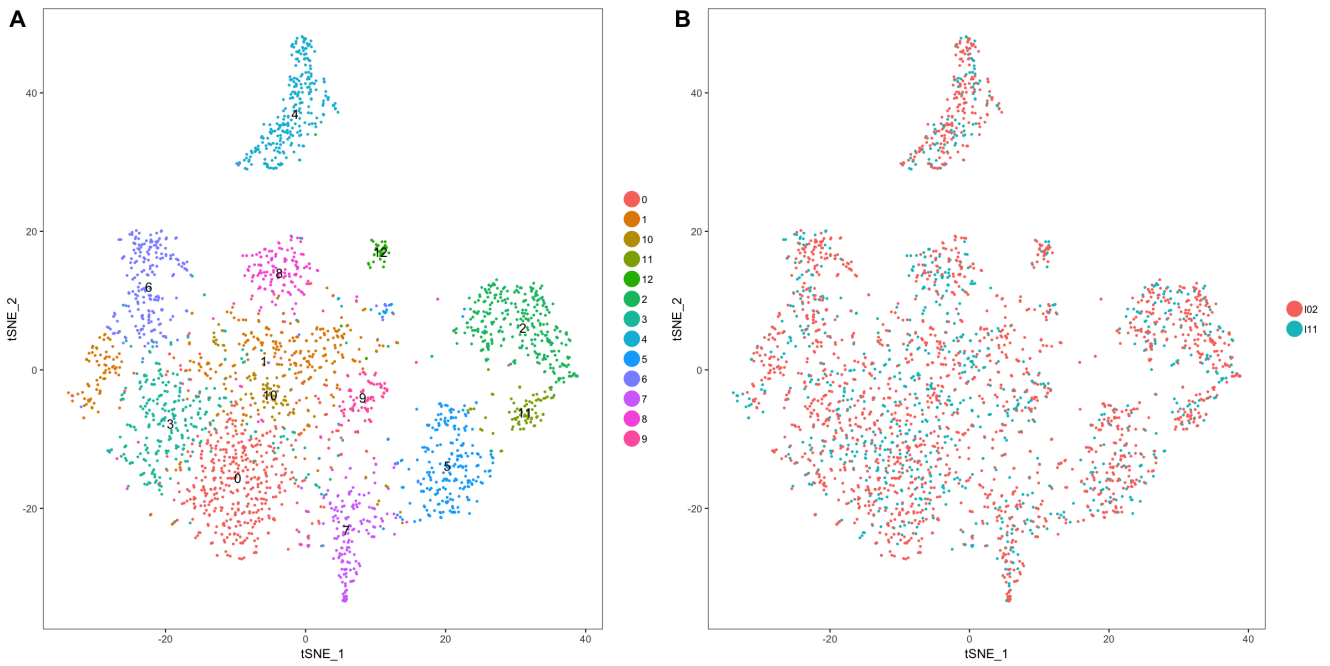

Figure 3: Integrated ectodermal epithelial cells from two sets of libraries. A) t-SNE plot with clusters labeled. B) t-SNE representation with cells labeled by batch. l02: libraries 02-, l11: libraries 11-.

```
# NMF results for whole dataset (wt_K96)

# Import good metagenes
cellScores <- read.csv("nmf/wt_K96/GoodMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(t(cellScores))

# Fix the geneIDs
rownames(cellScores) <- substring(rownames(cellScores), 2)
rownames(cellScores) <- gsub("[.]", "-", rownames(cellScores))

# Add cell scores as metagene columns to seurat object
cellScores <- cellScores[match(rownames(ds.ec.ss@meta.data), rownames(cellScores)),
    ]
ds.ec.ss@meta.data <- cbind(ds.ec.ss@meta.data, cellScores)
```

```
p1 <- FeaturePlot(ds.s1, "wt2", cols.use = c("grey", "blue"), do.return = TRUE, no.legend = FALSE)
p2 <- FeaturePlot(ds.ec.ss, "wt2", cols.use = c("grey", "blue"), do.return = TRUE,
    no.legend = FALSE)
plot_grid(p1[[1]], p2[[1]], ncol = 2, labels = "AUTO", label_size = 24, align = "h")
```

```
ds.ec.ss <- SetAllIdent(ds.ec.ss, "res.1.2")
# We remove cluster 12 and integrate as before
ds.ec.ss <- SubsetData(object = ds.ec.ss, ident.remove = 12, subset.raw = TRUE)
# save object for NMF K79 saveRDS(ds.ec.ss,'objects/ds.ec.cca1_10_12rm.rds')

# Retrieve cell ids for each batch
p3 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-CO"]
p4 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-P1"]
p5 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "02-PB"]
p12 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "11-PO"]
p13 <- rownames(ds.ec@meta.data)[ds.ec@meta.data[, "orig.ident"] == "11-BU"]

# cells from 02- libraries
l02 <- c(p3, p4, p5)

# cells from 11- libraries
l11 <- c(p12, p13)
```

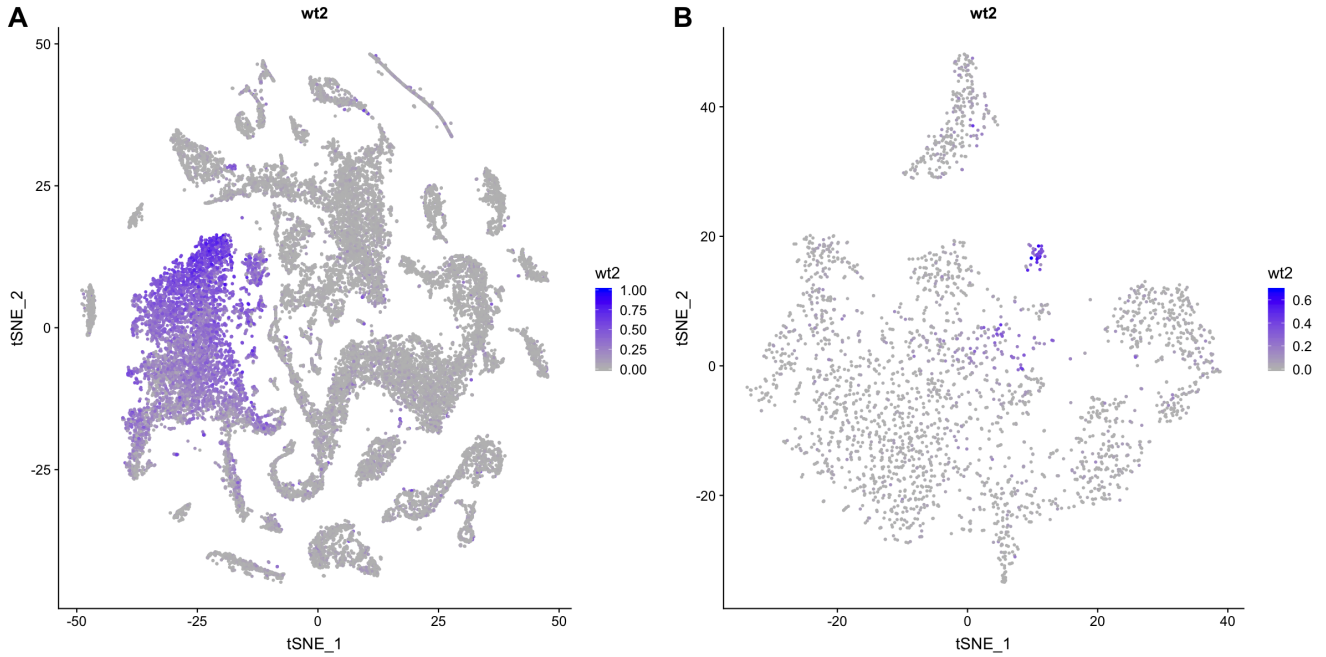

Figure 4: Expression of endodermal epithelial cell metagene wt2 in ectodermal epithelial cells. A) t-SNE plot for all cells in the transcriptome data set. Metagene wt2 is expressed in endodermal epithelial cells. B) t-SNE plot for ectodermal epithelial cells with cells from cluster 12 receiving high scores for endodermal metagene wt2.

```
l11 <- SubsetData(object = ds.ec.ss, cells.use = l11, subset.raw = TRUE)
l02 <- SubsetData(object = ds.ec.ss, cells.use = l02, subset.raw = TRUE)

l11@meta.data$stim <- "l11"
l11 <- FilterCells(l11, subset.names = "nGene", low.thresholds = 500, high.thresholds = Inf)
l11 <- NormalizeData(l11)
l11 <- ScaleData(l11, display.progress = F)

l02@meta.data$stim <- "l02"
l02 <- FilterCells(l02, subset.names = "nGene", low.thresholds = 500, high.thresholds = Inf)
l02 <- NormalizeData(l02)
l02 <- ScaleData(l02, display.progress = F)

# Gene selection for input to CCA
l11 <- FindVariableGenes(l11, do.plot = F)
l02 <- FindVariableGenes(l02, do.plot = F)
g.1 <- head(rownames(l11@hvg.info), 1000)
g.2 <- head(rownames(l02@hvg.info), 1000)
genes.use <- unique(c(g.1, g.2))
genes.use <- intersect(genes.use, rownames(l11@scale.data))
genes.use <- intersect(genes.use, rownames(l02@scale.data))

# Run CCA
ds.ec.ss <- RunCCA(l11, l02, genes.use = genes.use, num.cc = 30)

# Visualize results of CCA plot CC1 versus CC2 and look at a violin plot p1 <-
# DimPlot(object = ds.ec.ss, reduction.use = 'cca', group.by = 'stim', pt.size =
# 0.5, do.return = TRUE) p2 <- VlnPlot(object = ds.ec.ss, features.plot = 'CC1',
# group.by = 'stim', do.return = TRUE) plot_grid(p1, p2)

# PrintDim(object = ds.ec.ss, reduction.type = 'cca', dims.print = 1:2,
# genes.print = 100)

p3 <- MetageneBicorPlot(ds.ec.ss, grouping.var = "stim", dims.eval = 1:30, display.progress = FALSE)
```

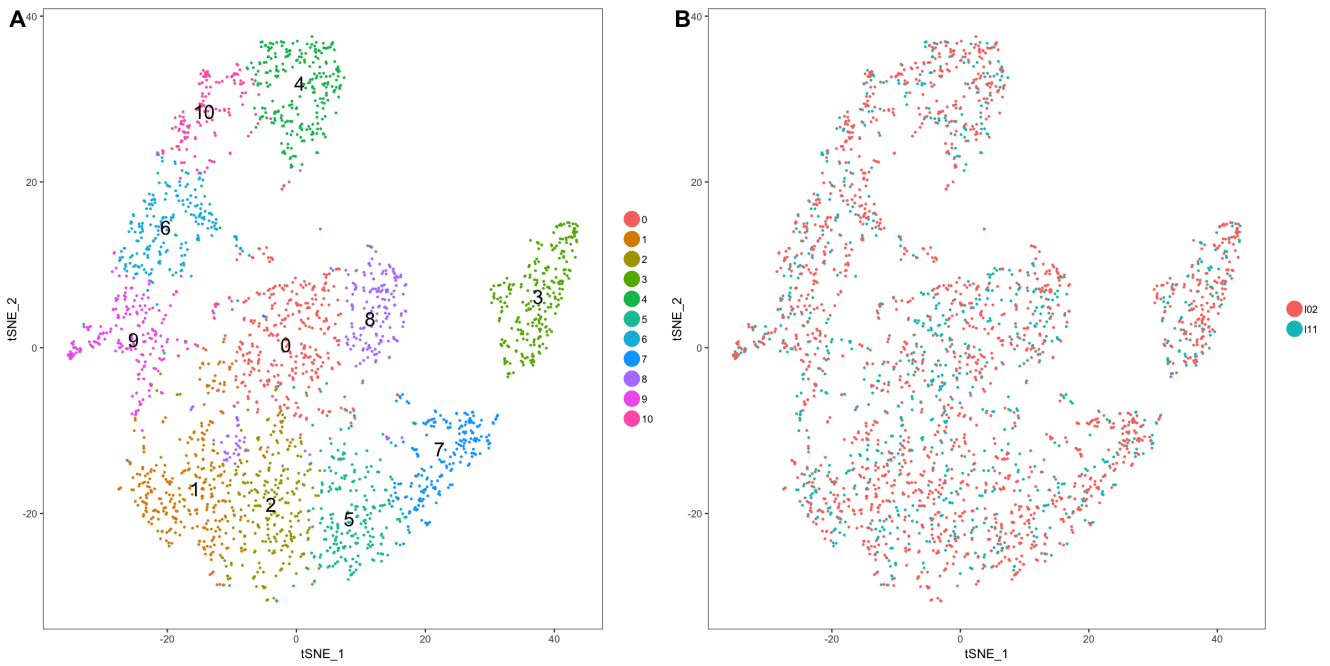

Figure 5: Integrated cells from two sets of libraries after exclusion of suspected endoderm/ectoderm doublets. A) t-SNE plot with clusters. B) t-SNE plot with cells labeled by batch. l02: libraries 02-, l11: libraries.

```
# Explore CC heatmaps DimHeatmap(object = ds.ec.ss, reduction.type = 'cca',
# cells.use = 100, dim.use = 1, do.balanced = TRUE, margins=c(12,16))

# Align subspaces
ds.ec.ss <- AlignSubspace(ds.ec.ss, reduction.type = "cca", grouping.var = "stim",
    dims.align = 1:10)

# p1 <- VlnPlot(object = ds.ec.ss, features.plot = 'ACC1', group.by = 'stim',
# do.return = TRUE) p2 <- VlnPlot(object = ds.ec.ss, features.plot = 'ACC2',
# group.by = 'stim', do.return = TRUE) plot_grid(p1, p2)

ds.ec.ss <- RunTSNE(ds.ec.ss, reduction.use = "cca.aligned", dims.use = 1:10, do.fast = T)

ds.ec.ss <- FindClusters(ds.ec.ss, reduction.type = "cca.aligned", resolution = 1.5,
    dims.use = 1:10)
```

```
# since t-SNE is not deterministic we here load the object of our original
# analysis ds.ec.ss <-
# readRDS(paste0(data.path,'objects/ds.ec.cca_l2_l11_int_12rm.rds'))
ds.ec.ss <- readRDS("objects/Hydra_Seurat_Ecto.rds")

# restore cluster numbering
ds.ec.ss <- SetAllIdent(object = ds.ec.ss, id = "cluster_numbering")
# drop metagenes that had already been loaded
ds.ec.ss@meta.data <- ds.ec.ss@meta.data[, -grep("^ec", colnames(ds.ec.ss@meta.data))]

p1 <- TSNEPlot(ds.ec.ss, do.label = T, do.return = T, label.size = 6, pt.size = 0.5)
p2 <- TSNEPlot(ds.ec.ss, do.return = T, pt.size = 0.5, label.size = 8, group.by = "stim")
plot_grid(p1, p2, ncol = 2, labels = "AUTO", label_size = 20, align = "h")
```

Cluster annotation and plot ectodermal metagene expression

We annotate the clusters (Fig. 6) and plot ectodermal metagene expression (ec_K76)(Fig. 7).

```
# Load metagene scores for each cell NMF ec_K76 was calculated for all ectodermal
# cells NMF ec_K79 was calculated for cells from libraries l02- and l11-
```

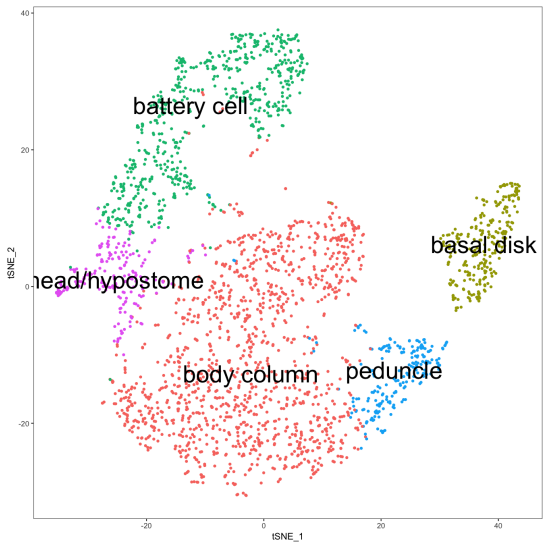

Figure 6: t-SNE representation for subclustered ectodermal cells. Labels indicate cell origin based on marker gene expression.

```
# Import good metagenes
cellScores <- read.csv("nmf/ec_K76/GoodMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(t(cellScores))

# Fix the geneIDs
rownames(cellScores) <- substring(rownames(cellScores), 2)
rownames(cellScores) <- gsub("[.]", "-", rownames(cellScores))
colnames(cellScores) <- sub("^", "ec", colnames(cellScores))

# Add cell scores as metagene columns to seurat object
cellScores <- cellScores[match(rownames(ds.ec.ss@meta.data), rownames(cellScores)),
    ]
ds.ec.ss@meta.data <- cbind(ds.ec.ss@meta.data, cellScores)

# Metagenes along the body column Plotting good metagenes
ds.ec.ss@meta.data[is.na(ds.ec.ss@meta.data)] <- 0

p <- TSNEPlot(object = ds.ec.ss, do.return = T, do.label = T, no.legend = TRUE, return = TRUE)
p1 <- FeaturePlot(ds.ec.ss, c("ec4", "ec48", "ec75", "ec38", "ec57", "ec12", "ec56",
    "ec47", "ec35", "ec20", "ec36", "ec34", "ec13", "ec14"), do.return = TRUE, cols.use = c("grey",
    "blue"))

plotlist <- prepend(p1, list(p))

plot_grid(plotlist = plotlist, ncol = 3, labels = "AUTO", label_size = 20, align = "h")

# saveRDS(ds.ec.ss,'objects/Hydra_Seurat_Ecto.rds')
```

Subsetting - endodermal cells

We load the full data set, extract endodermal clusters but omit body column clusters that are also positive for nematoblast/nematocyte expression (Fig. 8). This subset was used to perform non-negative matrix factorization (NMF) to identify gene expression modules expressed in endodermal epithelial cells (SA07_nmf.rmd, NMF analysis en_K40). We adjust the lower cut-offs for genes and UMIs to 500 and 2k, respectively, to consider only cells of high quality.

```
# Read data object
ds.s1 <- readRDS("objects/Hydra_Seurat_Whole_Transcriptome.rds")

# Suspected doublet cluster db (68 cells) was excluded from from downstream
# analyses
ds.s1 <- SubsetData(object = ds.s1, ident.remove = c("db"), subset.raw = TRUE)
```

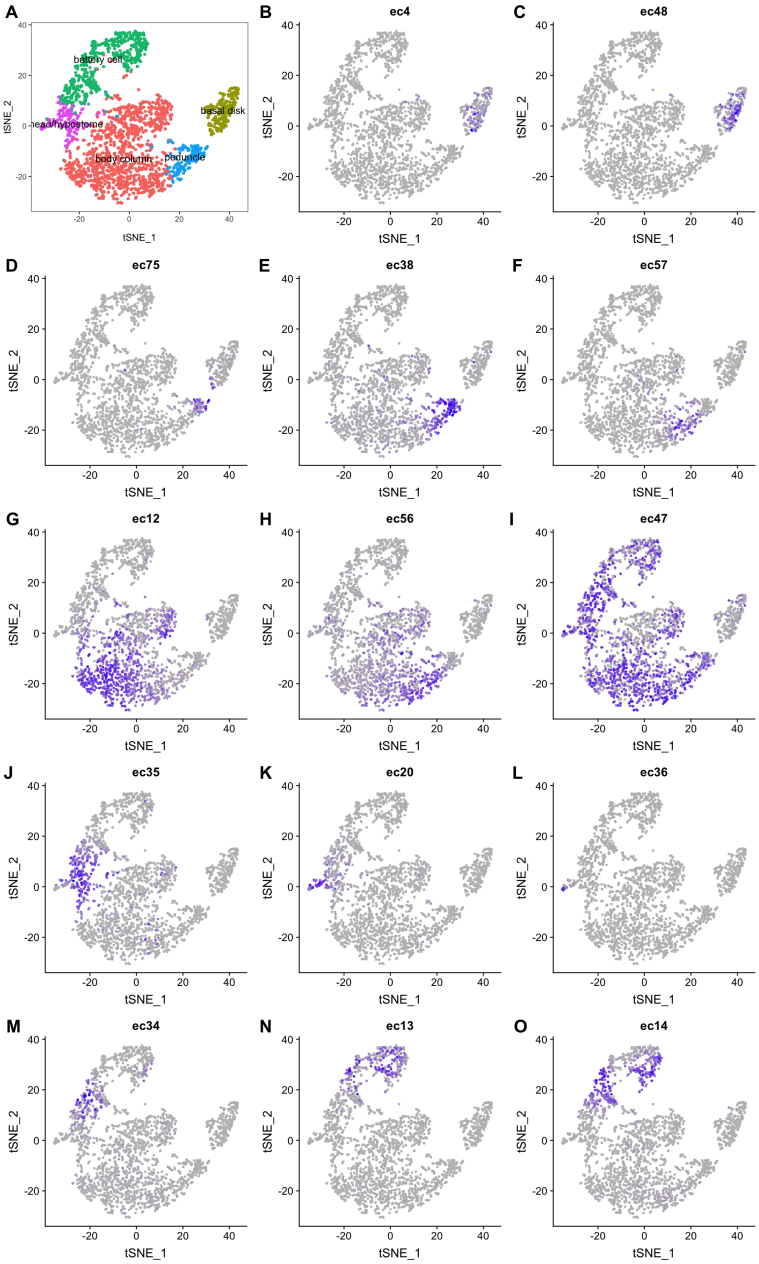

Figure 7: Metagenes expressed in ectodermal epithelial cells along the body column. A) t-SNE plot for ectodermal subset. B-O) t-SNE plots with metagene cell scores visualized. These metagenes were identified in a NMF analysis considering ectodermal cells from all libraries (ec K76).

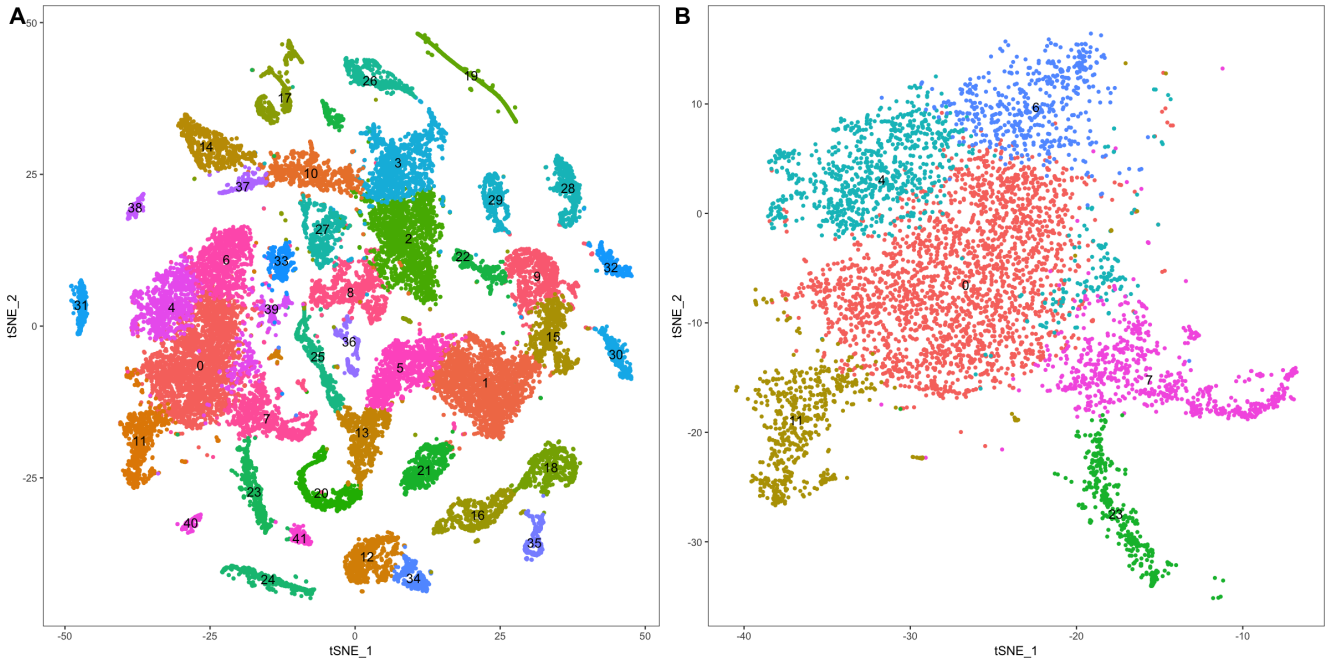

Figure 8: Subset of endodermal cells. A) t-SNE plot for all cells in the data set. B) Endodermal clusters.

```
# Run this to restore original cluster numbering
ds.s1 <- SetAllIdent(object = ds.s1, id = "cluster_numbering")

# Endodermal cluster without nematocyte doublet cluster
ds.en <- SubsetData(object = ds.s1, ident.use = c("0", "4", "6", "7", "11", "23"),
    subset.raw = TRUE)

# Save object for NMF analysis saveRDS(ds.en,'objects/ds.en.rds'))

# There are a few cells that are assigned to the endodermal clusters but group
# with interstitial cells These cells were identified them using the do.identify
# argument in function TSNEPlot() and were subsequently excluded. select.cells <-
# TSNEPlot(object = ds.en, do.identify = TRUE)

# Load cell ids to be excluded
select.cells <- c("11-PO_TTAAGTAGGGCG", "01-D1_CTGGCGCAGATC", "02-CO_GATGCAGTCATG",
    "02-P1_TACCCTTCTTAN", "02-P1_CCCCCCAGTGCC", "03-KI_ATGTGAGTTGCA", "03-KI_CAAGTATTCCCC",
    "03-KI_ATTCGAGACGCG", "06-FM_TTTCGGCGGTTC")

# All endodermal cells
cells <- ds.en@data@Dimnames[[2]]
# Identify cells to keep
cells.keep <- setdiff(cells, select.cells)
# Subset
ds.en <- SubsetData(object = ds.en, cells.use = cells.keep, subset.raw = TRUE)

# Filter cells
ds.en <- FilterCells(object = ds.en, subset.names = c("nGene", "nUMI"), low.thresholds = c(500,
    2000), high.thresholds = c(7000, 50000))

p1 <- TSNEPlot(object = ds.s1, group.by = "res.1.5", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- TSNEPlot(object = ds.en, group.by = "res.1.5", do.return = T, do.label = T,
    no.legend = TRUE)

plot_grid(p1, p2, ncol = 2, labels = "AUTO", label_size = 20, align = "h")
```

Clustering of cells

We cluster the cells after regressing out the library (batch) as source of variation by using the vars.to.regress argument in the Seurat function ScaleData(). Coloring cells by batch reveals clusters (8, 10) that are composed of cells originating exclusively from a specific set of libraries (01- through 03-) (Fig. 9 A-C). We evaluate NMF metagenes to get insights into transcriptional signatures of cells within these clusters. We load the scores for metagenes identified when performing NMF on all cells (wt_K96) and when using the subset of endodermal cells (en_K40). For metagene en19 expression we find a strong relation with batch origin (Fig. 9 D). We exclude cells positive for metagene en19 expression. NMF analysis of endodermal cells also reveals neuronal gene expression within cells of the endodermal subset (Fig. 10). High scoring genes of metagene en36 include LWamide and a sequence with similarity to RFamide. Neuronal expression is not random suggesting integrated neurons, phagocytic activity or dissociation doublets as possible sources. We want to retain these cells with partial neuronal signatures for interrogation since they may provide spatial information, but do not want neuronal genes to play a role when clustering the cells. We therefore remove the neuronal genes of metagene en36 from the list of variable genes considered when calculating principal components prior to subclustering (object used for lineage plot: Hydra_Seurat_Endo_lineage_plot.rds) (Fig. 11). We, however, exclude these cells prior to URD trajectory reconstruction (object used for URD trajectory reconstruction: Hydra_Seurat_Endo.rds).

```
# Identify highly variable genes
ds.en <- FindVariableGenes(object = ds.en, mean.function = ExpMean, dispersion.function = LogVMR,
    x.low.cutoff = 0.05, x.high.cutoff = 4, y.cutoff = 0.55)
# this identifies 2280 genes as variable

# Scale
ds.en <- ScaleData(object = ds.en, vars.to.regress = "orig.ident")
# Do PCA on highly variable genes
ds.en <- RunPCA(object = ds.en, pc.genes = ds.en@var.genes, pcs.compute = 40, do.print = TRUE,
    pcs.print = 1:5)
# Project PCA
ds.en <- ProjectPCA(object = ds.en)

# determine statistically significant PCs ds.en <- JackStraw(object = ds.en,
# num.pc = 40, num.replicate = 100, do.print = FALSE) JackStrawPlot(object =
# ds.en, PCs=1:40)

# Approximate amount of variance encoded by each PC
PCElbowPlot(object = ds.en, num.pc = 40)

ds.en <- FindClusters(object = ds.en, reduction.type = "pca", dims.use = 1:14, force.recalc = TRUE,
    resolution = 1.2, print.output = 0)
ds.en <- RunTSNE(object = ds.en, dims.use = c(1:14), do.fast = T)

TSNEPlot(object = ds.en, group.by = "res.1.2", do.return = T, do.label = T, no.legend = TRUE)
TSNEPlot(object = ds.en, group.by = "orig.ident", do.return = T, do.label = T)

# saveRDS(ds.en, 'objects/ds.en.orig.ident.pc14.rds')
```

```
# Load object from original analysis
ds.en <- readRDS("objects/ds.en.orig.ident.pc14.rds")

# Load NMF cell scores

# NMF results for endoderm subset (en_K40)

# Import good metagenes
cellScores <- read.csv("nmf/en_K40/GoodMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(t(cellScores))

# Fix the geneIDs
rownames(cellScores) <- substring(rownames(cellScores), 2)
rownames(cellScores) <- gsub("[.]", "-", rownames(cellScores))

# Add cell scores as metagene columns to seurat object
cellScores <- cellScores[match(rownames(ds.en@meta.data), rownames(cellScores)),
    ]
ds.en@meta.data <- cbind(ds.en@meta.data, cellScores)

# Import bad metagenes
```

```
cellScores <- read.csv("nmf/en_K40/BadMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(t(cellScores))

# Fix the geneIDs
rownames(cellScores) <- substring(rownames(cellScores), 2)
rownames(cellScores) <- gsub("[.]", "-", rownames(cellScores))

# Add cell scores as metagene columns to seurat object
cellScores <- cellScores[match(rownames(ds.en@meta.data), rownames(cellScores)),
    ]
ds.en@meta.data <- cbind(ds.en@meta.data, cellScores)

# NMF results for whole dataset (wt_K96)

# Import good metagenes
cellScores <- read.csv("nmf/wt_K96/GoodMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(t(cellScores))

# Fix the geneIDs
rownames(cellScores) <- substring(rownames(cellScores), 2)
rownames(cellScores) <- gsub("[.]", "-", rownames(cellScores))

# Add cell scores as metagene columns to seurat object
cellScores <- cellScores[match(rownames(ds.en@meta.data), rownames(cellScores)),
    ]
ds.en@meta.data <- cbind(ds.en@meta.data, cellScores)
```

```
# Metagene en19 plot
p1 <- TSNEPlot(object = ds.en, group.by = "res.1.2", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- TSNEPlot(object = ds.en, group.by = "orig.ident", do.return = T, no.legend = FALSE)
p3 <- TSNEPlot(object = ds.en, group.by = "orig.ident", colors.use = c("grey", "grey",
    "grey", "grey", "grey", "grey", "grey", "blue", "blue", "blue", "blue",
    "blue", "blue", "blue"), do.return = T)
p4 <- FeaturePlot(ds.en, "en19", cols.use = c("grey", "blue"), do.return = T)

plot_grid(p1, p2, p3, p4[[1]], ncol = 2, labels = "AUTO", label_size = 30, align = "h")
```

```
# Metagene en36 plot
p1 <- TSNEPlot(object = ds.en, group.by = "res.1.2", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- FeaturePlot(ds.en, "en36", cols.use = c("grey", "blue"), do.return = T)

plot_grid(p1, p2[[1]], ncol = 2, labels = "AUTO", label_size = 30, align = "h")
```

```
# Object for trajectory analysis

# We remove cells positive for metagenes en19 and en36
ds.en.s1.tr <- SubsetData(object = ds.en, subset.name = "en19", accept.high = 0.2,
    subset.raw = TRUE)
ds.en.s1.tr <- SubsetData(object = ds.en.s1.tr, subset.name = "en36", accept.high = 0.2,
    subset.raw = TRUE)

# Cluster the cells as before ds.en.s1.tr <- ScaleData(object = ds.en.s1.tr,
# vars.to.regress = 'orig.ident') ds.en.s1.tr <- RunPCA(object = ds.en.s1.tr,
# pc.genes = ds.en.s1@var.genes, pcs.compute = 40, do.print = TRUE, pcs.print =
# 1:5, genes.print = 20) ds.en.s1.tr <- ProjectPCA(object = ds.en.s1.tr)
# PCElbowPlot(object = ds.en.s1.tr, num.pc = 40) ds.en.s1.tr <-
# FindClusters(object = ds.en.s1.tr, reduction.type = 'pca', dims.use = 1:10,
# force.recalc = TRUE, resolution = 1.2, #print.output = 0)

# Save object for URD analysis

# saveRDS(ds.en.s1.tr,'objects/Hydra_Seurat_Endo.rds'))
```

```
# Object for lineage plot

# We want to keep the neuronal cell doublets in the data set but do not want them
# to play a role in the clustering.

# Remove cells positive for metagene en19 - batch specific cluster
ds.en.s1 <- SubsetData(object = ds.en, subset.name = "en19", accept.high = 0.2, subset.raw = TRUE)
# excludes 637 cells
```

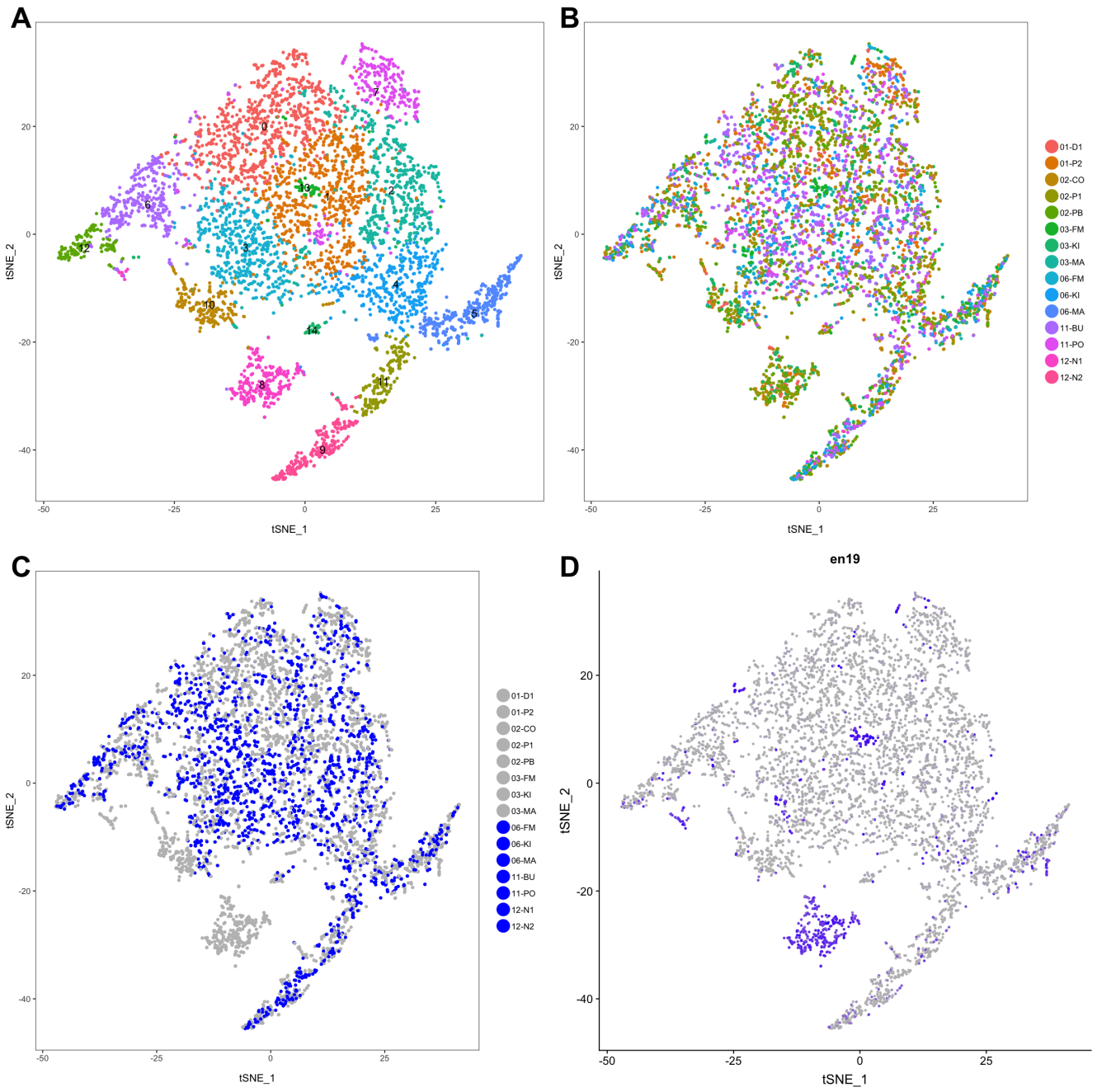

Figure 9: Endodermal cluster exploration. A) t-SNE plot for the endodermal epithelial cell subset. B) t-SNE plot with cells colored by batch. C) t-SNE representation highlighting batches generated using Hydra culture medium (grey) or isotonic Hydra dissociation medium (blue). D) Metagene en19 is expressed in a subset of cells from libraries 01- through 03-.

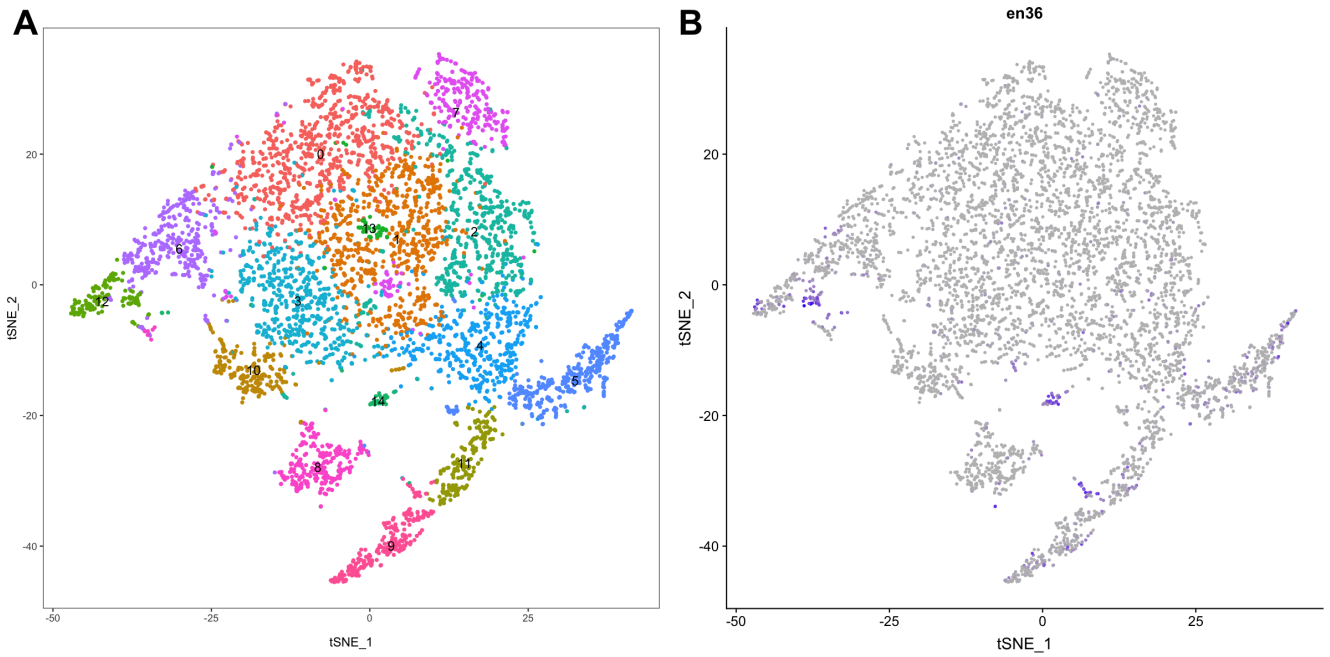

Figure 10: Neuron specific metagene expressed in endodermal epithelial cells. A) t-SNE plot for the endodermal epithelial cell subset. B) t-SNE plot with expression for neuronal metagene en36.

```
# Remove cells from cluster 10 - batch specific cluster
ds.en.s1 <- SubsetData(object = ds.en.s1, ident.remove = 10, subset.raw = TRUE)
# excludes 207 cells

# Remove top30 genes of metagene 36 from list of variable genes
mg36 <- read.csv("nmf/en_K40/BadMeta_Top30.csv", header = TRUE)
mg36 <- mg36$V1
mg36 <- levels(mg36)

# Load variable genes
var.gen <- ds.en.s1@var.genes

# Generate updated list of variable genes
new.var <- setdiff(var.gen, mg36)

# Update variable genes
ds.en.s1@var.genes <- new.var

# Cluster the cells as before
ds.en.s1 <- ScaleData(object = ds.en.s1, vars.to.regress = "orig.ident")
ds.en.s1 <- RunPCA(object = ds.en.s1, pc.genes = ds.en.s1@var.genes, pcs.compute = 40,
    do.print = TRUE, pcs.print = 1:5, genes.print = 20)
ds.en.s1 <- ProjectPCA(object = ds.en.s1)
PCElbowPlot(object = ds.en.s1, num.pc = 40)
ds.en.s1 <- FindClusters(object = ds.en.s1, reduction.type = "pca", dims.use = 1:10,
    force.recalc = TRUE, resolution = 1.2, print.output = 0)
ds.en.s1 <- RunTSNE(object = ds.en.s1, dims.use = c(1:10), do.fast = T)

# saveRDS(ds.en.s1,'objects/Hydra_Seurat_Endo_lineage_plot.rds')
```

```
# Load object from original analysis
ds.en.s1 <- readRDS("objects/Hydra_Seurat_Endo_lineage_plot.rds")

p1 <- TSNEPlot(object = ds.en.s1, group.by = "res.1.2", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- FeaturePlot(ds.en.s1, "en36", cols.use = c("grey", "blue"), do.return = T)
p3 <- FeaturePlot(ds.en.s1, "t11055aep|LWA_HYDEC", cols.use = c("grey", "blue"),
    do.return = T)
```

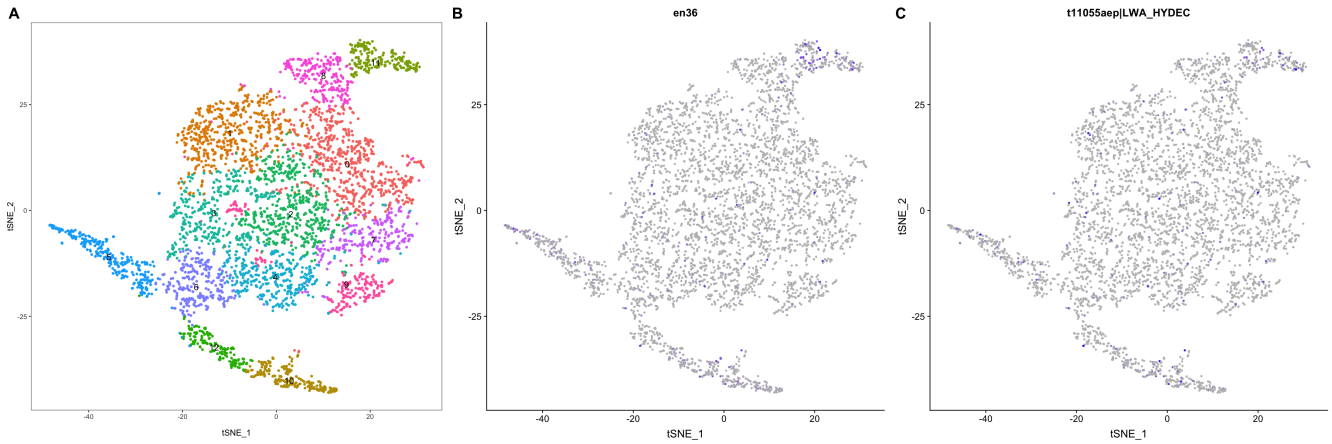

Figure 11: Neuron specific expression in endodermal epithelial cells after adjusting the set of variable genes considered in PCA. A) t-SNE plot for the endodermal epithelial cell subset. B) Top 30 genes for metagene en36 were removed from the list of variable genes considered when calculating principal components. Cells positive for metagene en36 no longer cluster with each other. C) Expression of neuron specific gene LWamide in cells of the endodermal subsets indicating multiplets.

```
plot_grid(p1, p2[[1]], p3[[1]], ncol = 3, labels = "AUTO", label_size = 20, align = "h")
```

Metagene expression and cluster annotation

We visualize endodermal metagene expression along the body axis (Fig. 12). Metagene gene loadings and marker gene expression allow us to annotate the t-SNE representation, e.g. hypostome marker *HyWnt3* is among the high scoring genes of metagene en13 (Fig. 13).

```
# Annotate t-SNE

ds.en.s1 <- SetAllIdent(ds.en.s1, "res.1.2")

# Stash
ds.en.s1 <- StashIdent(object = ds.en.s1, save.name = "cluster_numbering")
# Choose resolution ds.en.s1 <- SetAllIdent(ds.en.s1, 'res.1.2')

# Save cluster ids
current.cluster.ids <- as.character(0:12)

# Restore original cluster numbering before trying new names
ds.en.s1 <- SetAllIdent(object = ds.en.s1, id = "cluster_numbering")

cluster.names <- c("body column", "body column", "body column", "body column", "body column",
    "head/hypostome", "body column", "body column", "foot", "body column", "tentacle",
    "foot", "tentacle")

# Update names in Seurat object
ds.en.s1@ident <- plyr::mapvalues(x = ds.en.s1@ident, from = current.cluster.ids,
    to = cluster.names)
```

```
# Metagenes along the body column Plotting good metagenes except en18, en19
p <- TSNEPlot(object = ds.en.s1, do.return = T, do.label = T, no.legend = TRUE, label.size = 10,
    return = TRUE)
p1 <- FeaturePlot(ds.en.s1, c("en21", "en8", "en31", "en32", "en10", "en11", "en7",
    "en26", "en23", "en13", "en25", "en9", "en6", "en39"), do.return = TRUE, cols.use = c("grey",
    "blue"))

plotlist <- prepend(p1, list(p))

plot_grid(plotlist = plotlist, ncol = 3, labels = "AUTO", label_size = 20, align = "h")
```

```
TSNEPlot(object = ds.en.s1, do.return = T, do.label = T, label.size = 10, no.legend = TRUE)
```

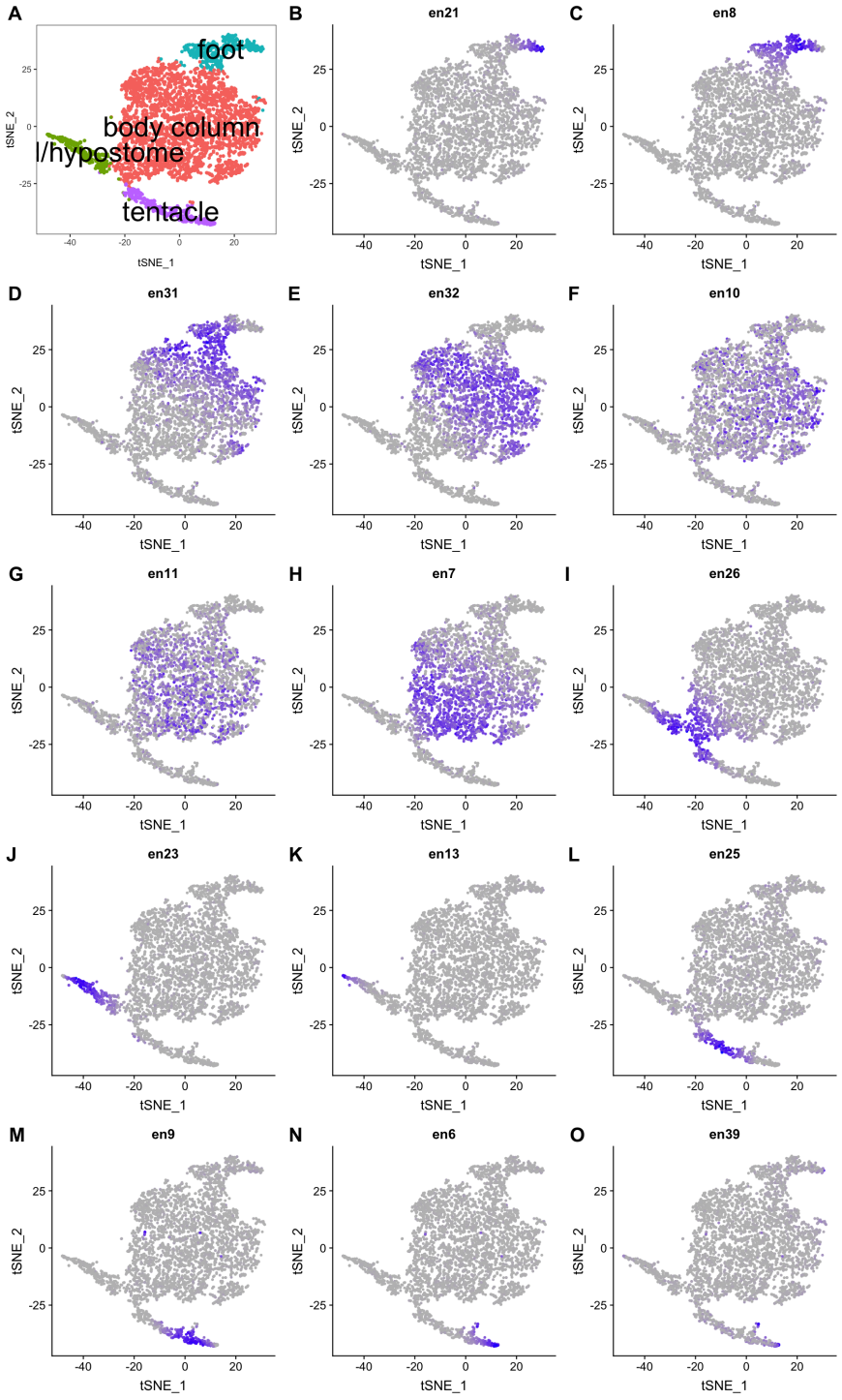

Figure 12: Metagenes expressed in endodermal epithelial cells along the body column. A) t-SNE plot for the endodermal epithelial cell subset. B-O) t-SNE plots with metagene cell scores visualized.

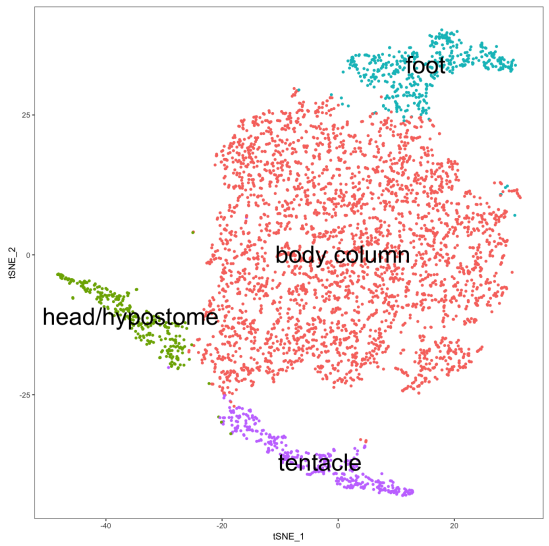

Figure 13: t-SNE representation for subclustered endodermal cells. Labels indicate cell origin based on marker gene expression.

Software versions

This document was computed on Mon Jul 08 19:47:58 2019 with the following R package versions.

```
R version 3.5.3 (2019-03-11)
Platform: x86_64-apple-darwin15.6.0 (64-bit)
Running under: macOS Mojave 10.14

Matrix products: default
BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] grid      stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
[1] rlang_0.3.4   gridExtra_2.3 gtable_0.3.0  dplyr_0.8.1   Seurat_2.3.4
[6] Matrix_1.2-17 cowplot_0.9.4 ggplot2_3.1.1 knitr_1.23

loaded via a namespace (and not attached):
  [1] Rtsne_0.15          colorspace_1.4-1    class_7.3-15
  [4] modeltools_0.2-22   ggridges_0.5.1      mclust_5.4.3
  [7] htmlTable_1.13.1    base64enc_0.1-3     rstudioapi_0.10
 [10] proxy_0.4-23        npsurv_0.4-0        flexmix_2.3-15
 [13] bit64_0.9-7         mvtnorm_1.0-10      codetools_0.2-16
 [16] splines_3.5.3       R.methodsS3_1.7.1   lsei_1.2-0
 [19] robustbase_0.93-5   jsonlite_1.6        Formula_1.2-3
 [22] ica_1.0-2           cluster_2.0.9       kernlab_0.9-27
 [25] png_0.1-7           R.oo_1.22.0         compiler_3.5.3
 [28] httr_1.4.0          backports_1.1.4     assertthat_0.2.1
 [31] lazyeval_0.2.2      formatR_1.6         lars_1.2
 [34] acepack_1.4.1       htmltools_0.3.6     tools_3.5.3
 [37] igraph_1.2.4.1      glue_1.3.1          RANN_2.6.1
 [40] reshape2_1.4.3      Rcpp_1.0.1          gdata_2.18.0
 [43] ape_5.3             nlme_3.1-140        iterators_1.0.10
 [46] fpc_2.2-1           gbRd_0.4-11         lmtest_0.9-37
 [49] xfun_0.7            stringr_1.4.0       irlba_2.3.3
 [52] gtools_3.8.1        DEoptimR_1.0-8      MASS_7.3-51.4
 [55] zoo_1.8-5           scales_1.0.0        doSNOW_1.0.16
```

```
 [58] parallel_3.5.3      RColorBrewer_1.1-2  yaml_2.2.0
 [61] reticulate_1.12     pbapply_1.4-0       rpart_4.1-15
 [64] segmented_0.5-4.0   latticeExtra_0.6-28 stringi_1.4.3
 [67] foreach_1.4.4       checkmate_1.9.3     caTools_1.17.1.2
 [70] bibtex_0.4.2        Rdpack_0.11-0       SDMTools_1.1-221.1
 [73] pkgconfig_2.0.2     dtw_1.20-1          prabclus_2.2-7
 [76] bitops_1.0-6        evaluate_0.13       lattice_0.20-38
 [79] ROCR_1.0-7          purrr_0.3.2         labeling_0.3
 [82] htmlwidgets_1.3     bit_1.1-14          tidyselect_0.2.5
 [85] plyr_1.8.4          magrittr_1.5        bookdown_0.10
 [88] R6_2.4.0            snow_0.4-3          gplots_3.0.1.1
 [91] Hmisc_4.2-0         pillar_1.4.1        foreign_0.8-71
 [94] withr_2.1.2         fitdistrplus_1.0-14 mixtools_1.1.0
 [97] survival_2.44-1.1   nnet_7.3-12         tsne_0.1-3
[100] tibble_2.1.1        crayon_1.3.4        hdf5r_1.2.0
[103] KernSmooth_2.23-15  rmarkdown_1.13      data.table_1.12.2
[106] metap_1.1           digest_0.6.19       diptest_0.75-7
[109] tidyr_0.8.3         R.utils_2.8.0       stats4_3.5.3
[112] munsell_0.5.0
```

References

1. P. Tardent, Gametogenesis in the Genus Hydra. *American Zoologist.* **14**, 447–456 (1974).

2. A. Butler, P. Hoffman, P. Smibert, E. Papalexi, R. Satija, Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology.* **36**, 411–420 (2018).