

SA02 - Cluster analysis using final cut-offs, initial doublet exclusion

Stefan Siebert
November 1, 2018

Summary

We recluster the cells using >300 <7k genes, >500UMI < 50k UMIs as cut-offs and explore the dataset. The resulting data object is the starting point for lineage subclusterings.

Preliminaries

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

# Function to find the full ID for gene of interest Gene ids have are of the
# format txxxaeplswiss_prot_id_best_hit
hFind <- function(x) {
  return(ds.ds@data@Dimnames[[1]][grep(x, ds.ds@data@Dimnames[[1]], ignore.case = T)])
}

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

Load object from permissive analysis

```
# Load object from permissive analysis
ds.ds <- readRDS("objects/ds.ds.g200_8k_U400_70k_PC1_19.rds")
ds.ds <- MakeSparse(ds.ds)
```

Apply selected cut-offs for genes and UMIs

```
# Apply final gene/UMI cut-offs
ds.ds <- FilterCells(object = ds.ds, subset.names = c("nGene", "nUMI"), low.thresholds = c(300,
  500), high.thresholds = c(7000, 50000))
```

Pre-clustering workflow

We scale the data and identify genes that vary more than expected for their expression level (Fig. 1).

```
# Identify highly variable genes
ds.ds <- FindVariableGenes(object = ds.ds, mean.function = ExpMean, dispersion.function = LogVMR,
  x.low.cutoff = 0.05, x.high.cutoff = 4, y.cutoff = 0.5)

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

Perform linear dimensional reduction

In preparation for graph-based clustering we perform PCA on the scaled data using genes in object@var.genes. 2244 genes are identified as variable given the selected cut-offs for dispersion and expression level.

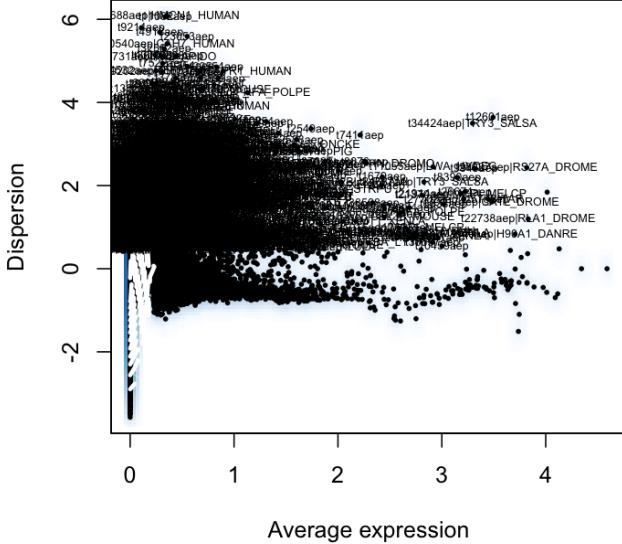


Figure 1: Variable genes using a dispersion cut-off of 0.5 and an expression cut-off of 0.05.

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

We score each gene in the dataset (including genes not included in the PCA) based on their correlation with the calculated components.

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

Community detection clustering

We cluster the cells.

```
# Find cluster
ds.ds <- FindClusters(object = ds.ds, reduction.type = "pca", dims.use = 1:19, force.recalc = TRUE,
  resolution = 1.5, print.output = 0)
```

t-SNE embedding

We apply t-SNE dimensionality reduction for visualization purposes. 43 clusters are recovered using a resolution parameter of 1.5 that were annotated using published expression patterns (Fig. 2).

```
# Run t-SNE
ds.ds <- RunTSNE(object = ds.ds, dims.use = c(1:19), do.fast = T)

# Save object saveRDS(ds.ds, 'objects/ds.ds.g300_7k_U500_50k_PC1_19.rds')
# Since t-SNE is not deterministic we here load the object of our original
# analysis
ds.ds <- readRDS("objects/ds.ds.g300_7k_U500_50k_PC1_19.rds")

# Label clusters Store cluster numbering
ds.ds <- StashIdent(object = ds.ds, save.name = "cluster_numbering")
ds.ds <- SetAllIdent(ds.ds, "res.1.5")

current.cluster.ids <- as.character(0:42)

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

```

cluster.names <- c("enEp_SC1", "enEp_SC2", "ecEp_SC1", "ecEp_SC2", "i_SC", "i_nb1",
  "i_nb2", "enEp_SC3", "ecEp_nem1(pd)", "i_smgc", "i_nc_prog", "enEp_foot", "i_gmgc",
  "i_nb3", "ecEp_bat", "ecEp_head", "i_nc1", "i_gc_nc_prog", "i_nc2", "i_zmg1",
  "i_fmg11", "ecEp_bd", "ecEp_nem2(mp)", "i_fmg12", "enEp_head", "i_nc3", "i_nc4",
  "enEp_tent", "i_nb3", "i_mgl1", "i_nb4", "i_nc5", "i_nc6", "enEp_nem1(pd)", "i_nc7",
  "i_mg12", "i_zmg2", "db1", "db2", "unident", "i_nc8", "enEp_tent(pd)", "db3")

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

TSNEPlot(object = ds.ds, do.return = T, do.label = T, no.legend = TRUE, pt.size = 0.5) +
  ggtitle("g>300 <7k genes, UMI>500<50k, 26,843 cells")

```

Doublet identification

Exploratory analyses of gene expression revealed doublet transcriptional signatures for cells in clusters db1, db2 and db3 (Fig. 2). Cells in cluster db1 show expression for both endodermal epithelial cell and zymogen gland cell marker genes (Fig. 3). Since zymogen gland cells reside in between endodermal epithelial cells these cells may correspond to dissociation doublets. Gland cells are somatic differentiation products of the interstitial lineage, a lineage distinct from the endodermal epithelial cell lineage (1). Cells in cluster db2 are positive for endodermal and ectodermal epithelial marker genes (Fig. 4). These cells may correspond to Drop-seq doublets (two cells were encapsulated with a single bead). We exclude these clusters from downstream analyses.

Cells in cluster db3 were positive for histone expression (*H2BL1* (t11585), *H10A* (t38683)) as well as for endodermal epithelial cell markers. These histones were furthermore found to be expressed in cells of multiple clusters (Fig. 5 A, B). RNA in situ hybridization demonstrated that these two histones are exclusively expressed in cells of the male germ-line (Fig. 5 C-F). 87.7% of cells that express either histone *H2BL1* and/or *H10A* outside the male germline clusters originated from suspensions that intentionally had polyps with testes in them (library 03-MA: male-spike in, 06-MA: all male library). We hypothesize that small spermatids and sperm progenitors were missed in the cell suspension leading to inaccurate cell concentrations in the Drop-seq experiments and to the generation of Drop-seq doublets. We here exclude all cells positive for these histones outside the male germline clusters. We perform NMF analysis (analysis whole transcriptome - wt_K96) using the remaining cells (25,052) to identify metagenes, sets of genes that are co-expressed in specific populations of cells (supplementary file SA07_NMF, supplementary methods).

```

# Cells that co-express gland cell and endodermal epithelial cell genes p <-
# FeaturePlot(ds.ds,
# c(hFind('t31900aep'), hFind('t34741aep'), hFind('t30697aep'), hFind('t4961aep'), hFind('t18356aep'), hFind('t
# cols.use=c('grey', 'blue'), do.return=TRUE)

# plot plot_grid(p[[1]], p[[2]], p[[3]], p[[4]], p[[5]], p[[6]], p[[7]], p[[8]],
# labels='AUTO', label_size = 20, align = 'h', ncol=4)

d1 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t31900aep"), hFind("t18356aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d2 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t34741aep"), hFind("t8678aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d3 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t30697aep"), hFind("t14102aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d4 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t4961aep"), hFind("t20198aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)

# Plot with tsne

plot_grid(d1[[1]], d2[[1]], d3[[1]], d4[[1]], ncol = 4)

# p <- FeaturePlot(ds.ds,
# c(hFind('t31900aep'), hFind('t34741aep'), hFind('t30697aep'), hFind('t4961aep'), hFind('t13977aep'),
# hFind('t24244aep'), hFind('t12614aep'), hFind('t23162aep')), cols.use=c('grey',
# 'blue'), do.return=TRUE)

```

$g > 300$ <7k genes, UMI >500<50k, 26,843 cells

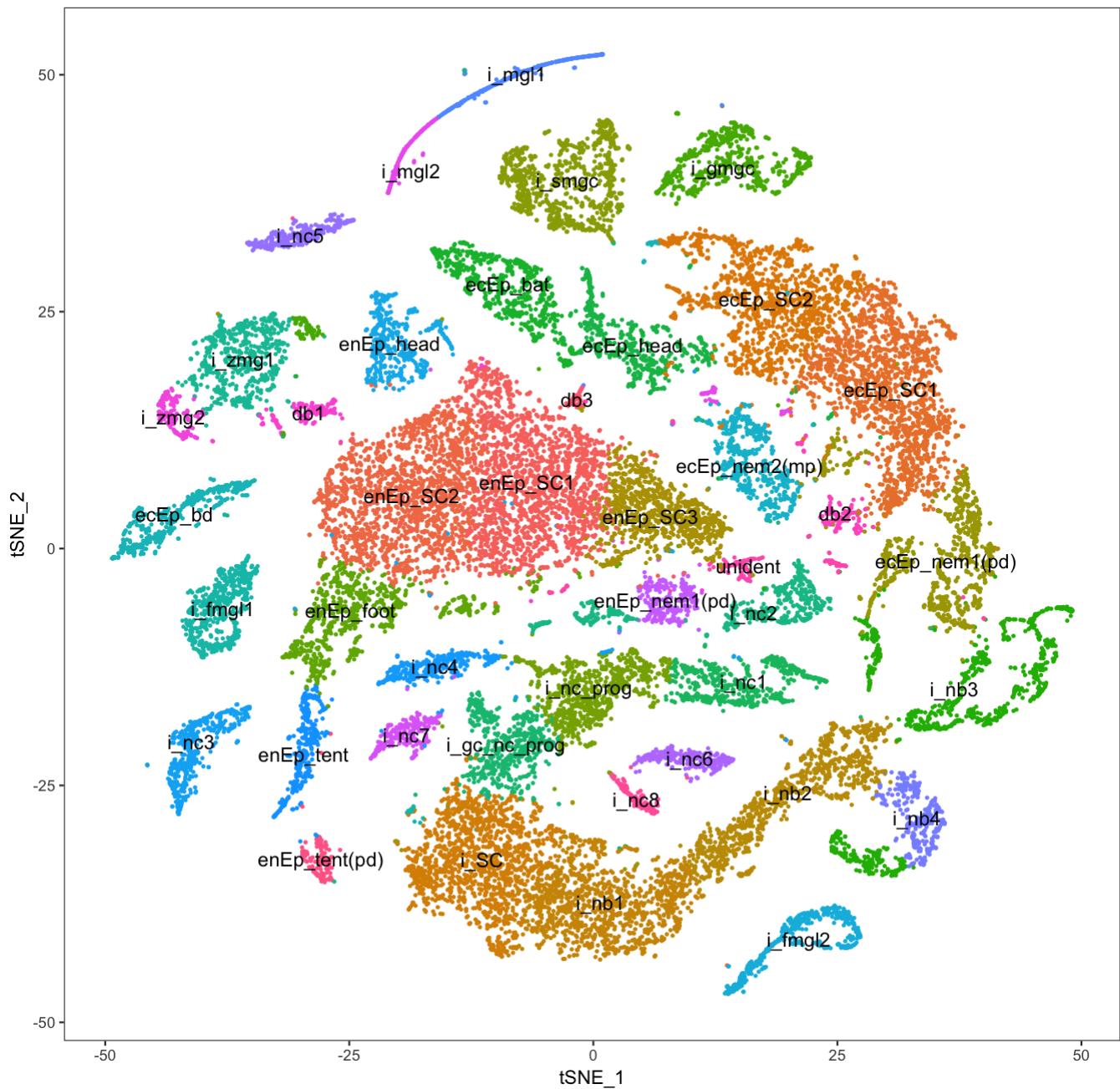


Figure 2: Annotated t-SNE - final gene and UMI cut-offs. bat: battery cell, db: doublet, ecEP: ectodermal epithelial cell, enEP: endodermal epithelial cell, fmg1: female germ-line, gc: gland cell, gmgc: granular mucous gland cell, i: cell of the interstitial lineage, id: integration doublet, mgl: male germline, mp: multiplet, nb: nematoblast, nc: neuronal cell, nem: nematocyte, pd: suspected phagocytosis doublet, prog: progenitor, SC: stem cell, smgc: spumous mucous gland cell, tent: tentacle, zmg: zymogen gland cell.

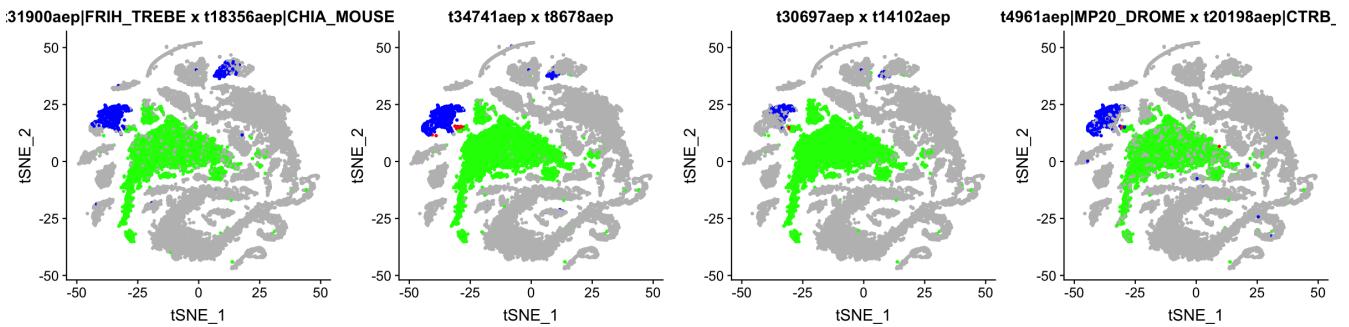


Figure 3: Doublet identification (endodermal epithelial cell - zymogen gland cell). Co-expression (red) of endodermal epithelial cell (green) and zymogen gland cells markers (blue). Cluster (db1) with cells co-expressing both sets of markers was excluded from downstream analyses.

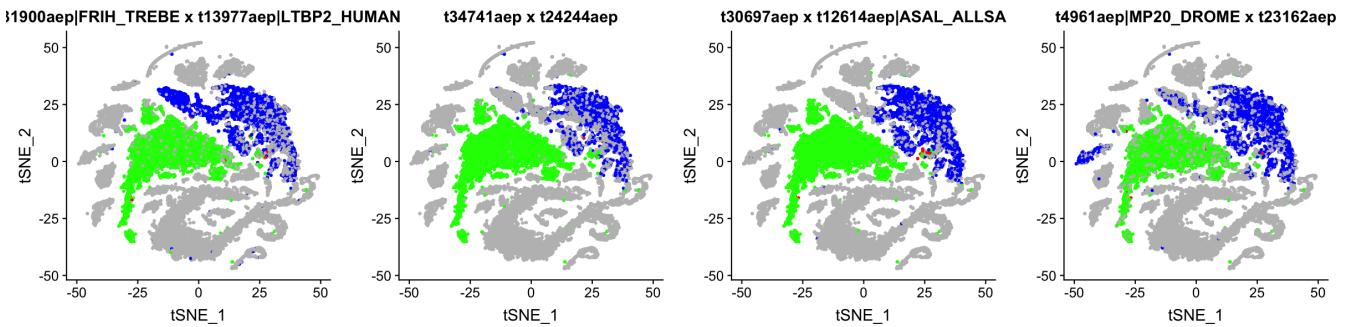


Figure 4: Doublet identification (endodermal epithelial cell - ectodermal epithelial cell). Co-expression (red) of endodermal epithelial cell (green) and ectodermal epithelial cell markers (blue). Cluster (db2) with cells co-expressing both sets of markers was excluded from downstream analyses.

```
# plot_grid(p[[1]], p[[2]], p[[3]], p[[4]], p[[5]], p[[6]], p[[7]], p[[8]],
# labels='AUTO', label_size = 20, align = 'h', ncol=4) dev.off()

d1 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t31900aep"), hFind("t13977aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d2 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t34741aep"), hFind("t24244aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d3 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t30697aep"), hFind("t12614aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)
d4 <- FeaturePlot(object = ds.ds, features.plot = c(hFind("t4961aep"), hFind("t23162aep")),
  cols.use = c("grey", "green", "blue", "red"), overlay = TRUE, no.legend = TRUE,
  do.return = TRUE)

# Plot with tsne
plot_grid(d1[[1]], d2[[1]], d3[[1]], d4[[1]], ncol = 4)

## Doublet exclusion, cells that express male germline specific histones outside
## the germline

# Explore co-expression of histones FeaturePlot(object = ds.ds, features.plot =
# c('t11585aep/H2BL1_PSAMI', 't38683aep/H10A_XENLA'), no.legend = TRUE, min.cutoff
# = 0.2, max.cutoff = 0.5) FeaturePlot(object = ds.ds, features.plot =
# c('t11585aep/H2BL1_PSAMI', 't38683aep/H10A_XENLA'), no.legend = TRUE,
# cols.use=c('grey', 'blue')) FeaturePlot(object = ds.ds, features.plot =
# c('t11585aep/H2BL1_PSAMI', 't38683aep/H10A_XENLA'), cols.use = c('grey', 'red',
# 'blue', 'green'), overlay = TRUE, no.legend = FALSE)
```

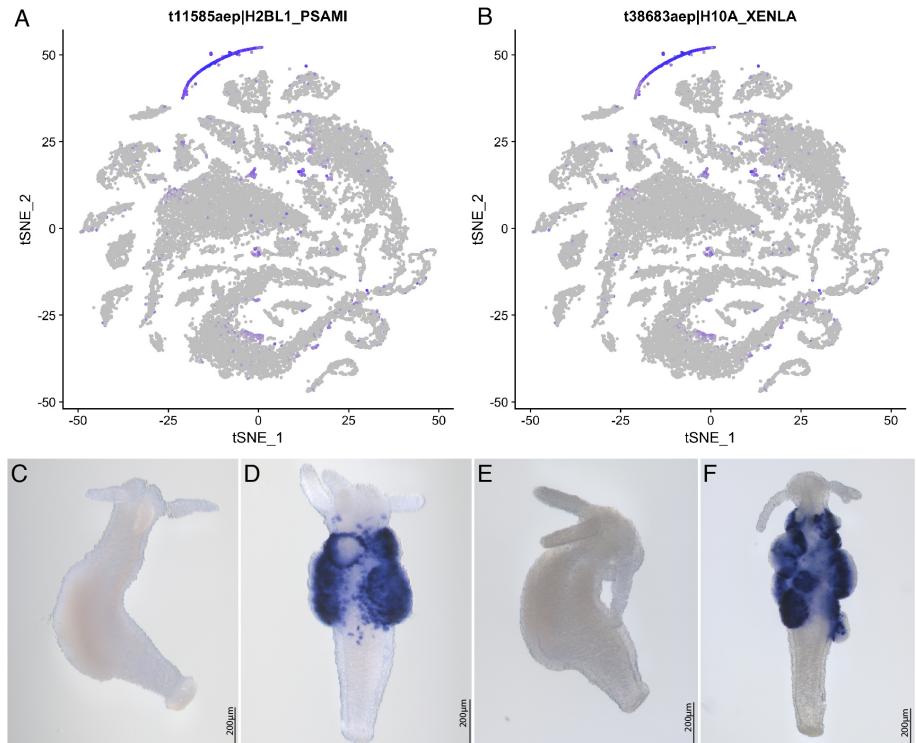


Figure 5: Expression of male germline specific histone proteins. A) t-SNE plot showing expression of histone *H2BL1* (t11585), B) t-SNE plot showing expression of histone *H10A* (t38683). C-D) RNA in situ hybridization using histone *H2BL1* probe. C) No expression was detected in female polyps with developing eggpatch. D) Expression in polyps with testes. E-F) RNA in situ hybridization using histone *H10A* probe. E) No expression was detected in female polyps with developing eggpatch. F) Expression in polyps with testes.

```

# jpeg('SA_figures/SA02_figures/histones.jpeg'), width=300, height=150, units =
# 'mm', res=300, bg ='white') FeaturePlot(object = ds.ds, features.plot =
# c('t11585aep/H2BL1_PSAMI', 't38683aep/H10A_XENLA'), no.legend = TRUE,
# cols.use=c('grey', 'blue')) dev.off()

## Exclude histone positive cells outside the germline clusters i_mgl1 and i_mgl2;
## exclude doublet clusters db1 and db2

# We isolate cells from male germline clusters
ds.male <- SubsetData(object = ds.ds, ident.use = c("i_mgl1", "i_mgl2"), subset.raw = TRUE)
# 602 cells

## Plot germline cluster cells

# TSNEPlot(object = ds.male, do.return=T, do.label = T, no.legend=TRUE, pt.size =
# 0.5)

## There are a few histone positive cells that are assigned to the germline
## clusters but group with epithelial cells We manually select these cells using
## argument do.identify.

# select.cells <- TSNEPlot(object = ds.male, do.identify = TRUE)
# length(select.cells) 66 cells

# cells to be excluded > select.cells [1] '11-BU_CGGGCAATGCCA'
# '06-MA_ACAGCGGCCCTC' '06-MA_CCTTGGTCCAAC' '01-D1_TTCAAATTCAAC'
# '01-P2_CCGCTTAAGGAN' '03-MA_GGGTACTCTTCC' '03-MA_CTGTGATGTGAN' [8]
# '03-MA_AATCATGGAGG' '03-MA_TCCTTCGCTGGA' '03-MA_CTCACCCCTCGT'
# '06-MA_CTCAGCCGCATC' '06-MA_CTCGCCGCATCN' '06-MA_GTTCCCTCGCGG'
# '06-MA_GCTTCTTGACA' [15] '06-MA_GCAGGAACCTCA' '06-MA_AGGATAACAGTG'
# '06-MA_CAGACTGAAGAC' '06-MA_CGGAAGACGCGA' '06-MA_GAACCCCTGTTG'
# '06-MA_AATAGGTACTGT' '06-MA_TTGGAAATTCAAC' [22] '06-MA_GGTAGGTGAGAG'
# '06-MA_ACCATGTTGTTN' '06-MA_ACCTATGTTGTT' '06-MA_CGTCGTTGAAAT'
# '06-MA_ATTTGTGTTAG' '06-MA_AGCCTCTATTGT' '06-MA_AATGGTACTGTN' [29]
# '06-MA_ATCGCGTGTATCN' '06-MA_CAGCTGAAGACN' '06-MA_CGGAGACCGCGAN'
# '06-MA_AGTACGACCTTG' '06-MA_GGTCGGTAACAT' '06-MA_TGGGTTATCGGGG'
# '06-MA_AGGTAACAGTGN' [36] '06-MA_AGTGACCTTGN' '06-MA_GGATTGAATCAG'
# '06-MA_GGTGGTGGAGAGN' '06-MA_CGCAACGGAATC' '06-MA_CGCACGGAAATCN'
# '06-MA_CCACCGGCCATC' '06-MA_ATCAGCGTGTATC' [43] '06-MA_TGTTGCGTGCAG'
# '06-MA_CTCGGGAGCAGG' '06-MA_TCTGTTTCAAG' '06-MA_CAAATCATGACTT'
# '06-MA_TTCCCCCAGGAT' '06-MA_AAATACGCTTCA' '06-MA_CTAGCTTGTGTTTC' [50]
# '06-MA_CACCAATCTCCA' '06-MA_GACTCAACACCG' '06-MA_GCCTAAATGGATG'
# '06-MA_CGTCGTTGAGTTA' '06-MA_TACTTAGTTTAT' '06-MA_ACTAACTCTGT'
# '06-MA_GCGCTATCGTAT' [57] '06-MA_GGGTGGATTATA' '06-MA_AAGCATCCCGTT'
# '06-MA_TGGGGGTGCCCG' '06-MA_TCAGAGGCCGA' '06-MA_CTGACCGAAACG'
# '06-KI_TAGCCTCGTTCT' '11-BU_GTGTAGGGGTCA' [64] '11-BU_GCATAGAACCTG'
# '06-MA_GTGGATGCAAAT' '06-MA_GTCGCCGATTCA' '06-MA_GTCGCCGATTCA'

# We here provide ids of cells that were manually selected and excluded
select.cells <- c("11-BU_CGGGCAATGCCA", "06-MA_ACAGCGGCCCTC", "06-MA_CCTTGGTCCAAC",
"01-D1_TTCAAATTCAAC", "01-P2_CCGCTTAAGGAN", "03-MA_GGGTACTCTTCC", "03-MA_CTGTGATGTGAN",
"03-MA_AATCATGGAGG", "03-MA_TCCTTCGCTGGA", "03-MA_CTCACCCCTCGT", "06-MA_CTCAGCCGCATC",
"06-MA_CTCGCCGCATCN", "06-MA_GTTCCCTCGCGG", "06-MA_GCTTCTTGACA", "06-MA_GCAGGAACCTCA",
"06-MA_AGGATAACAGTG", "06-MA_CAGACTGAAGAC", "06-MA_CGGAAGACGCGA", "06-MA_GAACCCCTGTTG",
"06-MA_AATAGGTACTGT", "06-MA_TTGGAAATTCAAC", "06-MA_GGTAGGTGAGAG", "06-MA_ACCATGTTGTTN",
"06-MA_ACCATGTTGTT", "06-MA_CGTCGTTGAAAT", "06-MA_ATTTGTGTTAG", "06-MA_AGGTACTTGT",
"06-MA_AATGGTACTGTN", "06-MA_ATCGCGTGTATCN", "06-MA_CAGCTGAAGACN", "06-MA_CGGAGACCGCGAN",
"06-MA_AGTACGACCTTG", "06-MA_GGTCGGTAACAT", "06-MA_TGGGTTATCGGGG", "06-MA_AGGTAACAGTGN",
"06-MA_AGTGACCTTGN", "06-MA_GGATTGAATCAG", "06-MA_GGTGGTGGAGAGN", "06-MA_CGCAACGGAATC",
"06-MA_CGCAACGGAATCN", "06-MA_GCACCGGCCATC", "06-MA_ATCAGCGTGTATC", "06-MA_TGTTGCGTGCAG",
"06-MA_CTCGGGAGCAGG", "06-MA_TCTGTTTCAAG", "06-MA_CAAATCATGACTT", "06-MA_TTCCCCCAGGAT",
"06-MA_AAATACGCTTCA", "06-MA_CTAGCTTGTGTTTC", "06-MA_CACCAATCTCCA", "06-MA_GACTCAACACCG",
"06-MA_GCCTAAATGGATG", "06-MA_CGTCGTTGAGTTA", "06-MA_TACTTAGTTTAT", "06-MA_ACTAACTCTGT",
"06-MA_GCGCTATCGTAT", "06-MA_GGGTGGATTATA", "06-MA_AAGCATCCCGTT", "06-MA_TGGGGGTGCCCG",
"06-MA_TCAGAGGCCGA", "06-MA_CTGACCGAAACG", "06-KI_TAGCCTCGTTCT", "11-BU_GTGTAGGGGTCA",
"11-BU_GCATAGAACCTG", "06-MA_GTGGATGCAAAT", "06-MA_GTCGCCGATTCA")

# Get all cell ids associated with germ-line clusters i_mgl1 and i_mgl2
cells <- ds.male@data@Dimnames[[2]]
# Identify cells to keep
cells.keep <- setdiff(cells, select.cells)

# Subset the dataset

```

```

ds.male <- SubsetData(object = ds.male, cells.use = cells.keep, subset.raw = TRUE)

# Subset of cells without male germline, cluster db1 (zymogen gland
# cell/endodermal epithelial cell doublet cluster), cluster db2 (endodermal
# epithelial cell / ectodermal epithelial cell doublet cluster)
ds.else <- SubsetData(object = ds.ds, ident.remove = c("i_mgl1", "i_mgl2", "db1",
  "db2"), subset.raw = TRUE)
# 25917 cells

# Keep cells not expressing H2BL1 in cells outside germline clusters
gate1 <- WhichCells(ds.else, subset.name = "t11585aep|H2BL1_PSAMI", accept.high = 0.5)
# length(gate1) [1] 24694

# Keep cells not expressing H10A in cells outside germline clusters
gate2 <- WhichCells(ds.else, subset.name = "t38683aep|H10A_XENLA", accept.high = 0.5)
# Length(gate2) [1] 24833

# identify cells not expressing either H2BL1 or H10A
cells.keep.else <- intersect(gate1, gate2)
length(cells.keep.else)
# 24517

# Create Seurat subset for cells outside male germline
ds.else <- SubsetData(ds.else, cells.use = cells.keep.else, subset.raw = TRUE)
# Merge germline subset with all other cells
ds.s1 <- MergeSeurat(ds.male, ds.else)

# Remove objects that are no longer needed
rm(ds.ds)
rm(ds.male)
rm(ds.else)

```

Clustering of cells

We cluster the cells as before. We tested PCs from 1:19 through 1:37 with three different seeds (1 (default), 100, 4024) and three perplexities, 20, 30 (default), 40. No principal components were excluded. The selected analysis used PCs 1:31 and a perplexity of 40 (seed=100) (Fig. 6).

```

# Identify highly variable genes
ds.s1 <- FindVariableGenes(object = ds.s1, mean.function = ExpMean, dispersion.function = LogVMR,
  x.low.cutoff = 0.05, x.high.cutoff = 4, y.cutoff = 0.5)

# Scale
ds.s1 <- ScaleData(object = ds.s1)

# PCA on highly variable genes
ds.s1 <- RunPCA(object = ds.s1, pc.genes = ds.s1@var.genes, pcs.compute = 40, do.print = TRUE,
  pcs.print = 1:5)

# Project PCA
ds.s1 <- ProjectPCA(object = ds.s1)

## Determine statistically significant PCs
ds.s1 <- JackStraw(object = ds.s1,
  ## num.pc = 40, num.replicate = 100, do.print = FALSE)
JackStrawPlot(object =
  ## ds.s1, PCs=1:40)

# Approximate amount of variance encoded by each PC
# 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.s1, num.pc = 40)

# Find cluster
ds.s1 <- FindClusters(object = ds.s1, reduction.type = "pca", dims.use = 1:19, force.recalc = TRUE,
  resolution = 1.5, print.output = 0)

# Run t-SNE
ds.s1 <- RunTSNE(object = ds.s1, dims.use = c(1:31), seed.use = 100, perplexity = 40,
  do.fast = T)

## Save object
saveRDS(ds.ds,paste0(data.out,'objects/ds.s1.s100_pc31_p40.rds'))

```

```

## saveRDS(ds.ds,'objects/Hydra_Seurat_Whole_Transcriptome.rds'))
# Since t-SNE is not deterministic we here load the object of our original
# analysis ds.s1 <- readRDS(paste0(data.path, 'objects/ds.s1.s100_pc31_p40.rds'))
ds.s1 <- readRDS("objects/Hydra_Seurat_Whole_Transcriptome.rds")

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

TSNEPlot(object = ds.s1, do.return = T, do.label = T, no.legend = TRUE, pt.size = 0.5)

```

Cluster annotation

We annotate the t-SNE representation using marker genes (Fig. 7, 8). We save the object: Hydra_Seurat_Whole_Transcriptome.rds.

```

# Annotate t-SNE Selected genes used for annotation
gene.names <- c(hFind("t31074aep"), hFind("t14194aep"), hFind("t25396aep"), hFind("t16043aep"),
  hFind("t4498aep"), hFind("t16456aep"), hFind("t11407aep"), hFind("t3974aep"),
  hFind("t8678aep"), hFind("t12596aep"), hFind("t7059aep"), hFind("t13480aep"),
  hFind("t23176aep"), hFind("t11117aep"), hFind("t11585aep"))

# Update gene names
new.names <- c("armininia1a", "HyWnt3", "CnNK-2", "PPOD1", "ks1", "HyAlx", "Cnnos1",
  "ELAV2 (t3974)", "HyDkk1/2/4 A", "HyTSR1", "MUC2 (t7059)", "nematogalectin B",
  "nematocilin A", "periculinia1a", "H2BL1 (t11585)")

# Annotate
update.names(gene.names, new.names)

# Plot with tsne
p1 <- TSNEPlot(object = ds.s1, group.by = "res.1.5", do.label = T, label.size = 5,
  pt.size = 0.5, cex.names = 6, no.legend = TRUE, do.return = TRUE)
p2 <- FeaturePlot(ds.s1, c("armininia1a", "HyWnt3", "CnNK-2", "PPOD1", "ks1", "HyAlx",
  "Cnnos1", "ELAV2 (t3974)", "HyDkk1/2/4 A", "HyTSR1", "MUC2 (t7059)", "nematogalectin B",
  "nematocilin A", "periculinia1a", "H2BL1 (t11585)"), cols.use = c("grey", "blue"),
  do.return = TRUE)

plotlist <- prepend(p2, list(p1))

plot_grid(plotlist = plotlist, labels = "AUTO", label_size = 20, align = "h", ncol = 4)

## Label clusters Store cluster numbering
ds.s1 <- StashIdent(object = ds.s1, save.name = "cluster_numbering")

current.cluster.ids <- as.character(0:42)

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

cluster.names <- c("enEp_SC1", "i_SC", "ecEp_SC2", "ecEp_SC3", "enEp_SC2", "i_nb1",
  "enEp_SC3", "enEp_head", "ecEp_nem1(pd)", "i_nc_prog", "ecEp_head", "enEp_foot",
  "i_smgc2", "i_nb2", "ecEp_bat2(mp)", "i_nc_gc_prog", "i_gmgc", "i_nem", "i_zmg1",
  "i_mgl1", "i_nb3", "i_fmg11", "i_nc1", "enEp_tent", "i_fmg12_nurse", "i_nb4",
  "ecEp_bd", "ecEp_nem2(id)", "i_nc2", "i_nc3", "i_nc4", "i_nc5", "i_nc6", "enEp_nem1(pd)",
  "i_smgc1", "i_zmg2", "i_nb4", "ecEp_bat1(mp)", "i_nc7", "enEp_nem2(pd)", "i_nc8",
  "enEp_tent(pd)", "db")

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

# Store state labels
ds.s1 <- StashIdent(object = ds.s1, save.name = "states")

TSNEPlot(object = ds.s1, do.return = T, do.label = T, no.legend = TRUE, pt.size = 0.5)

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

```

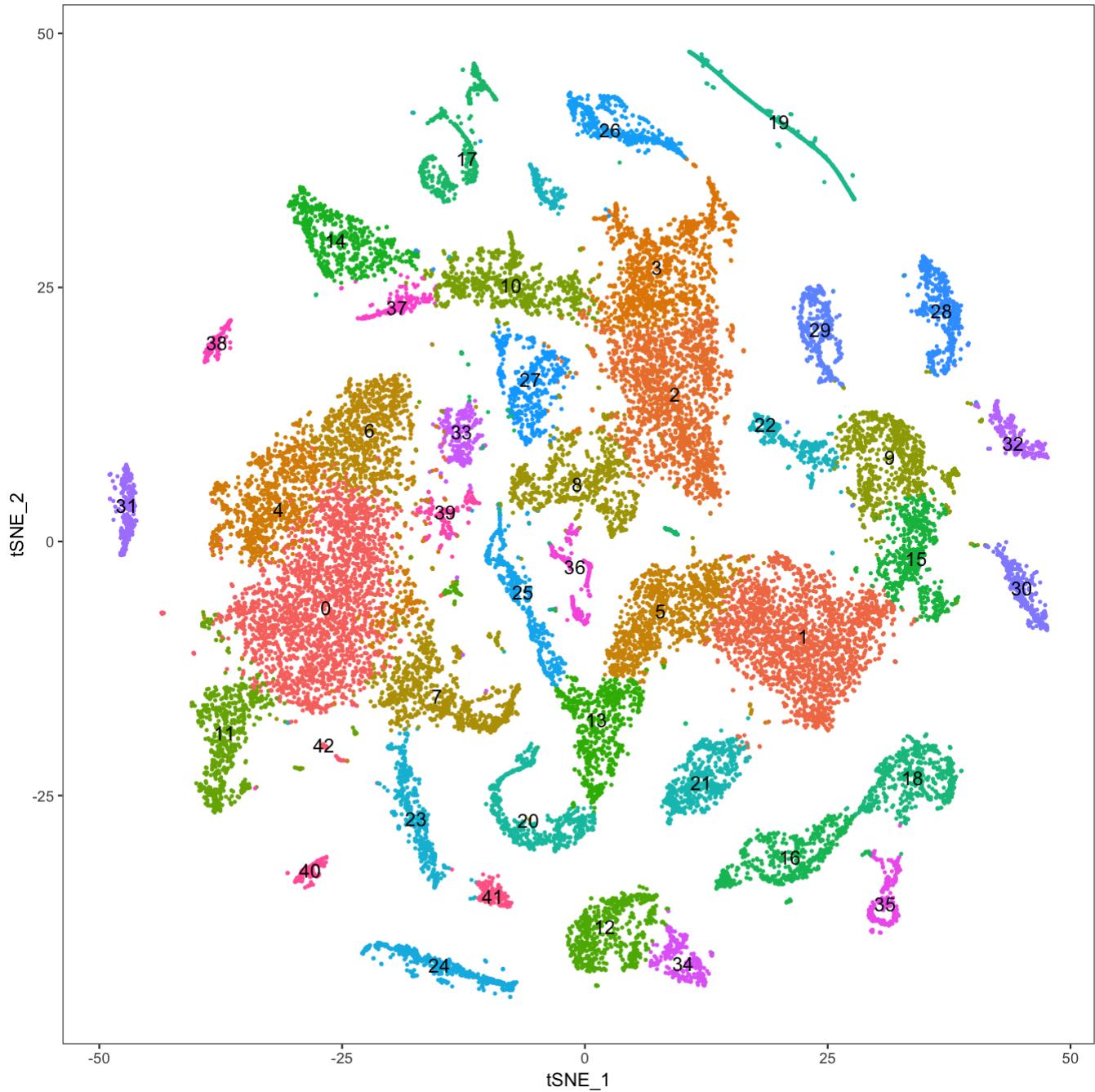


Figure 6: t-SNE plot after applying final cut-offs and initial doublet exclusion.

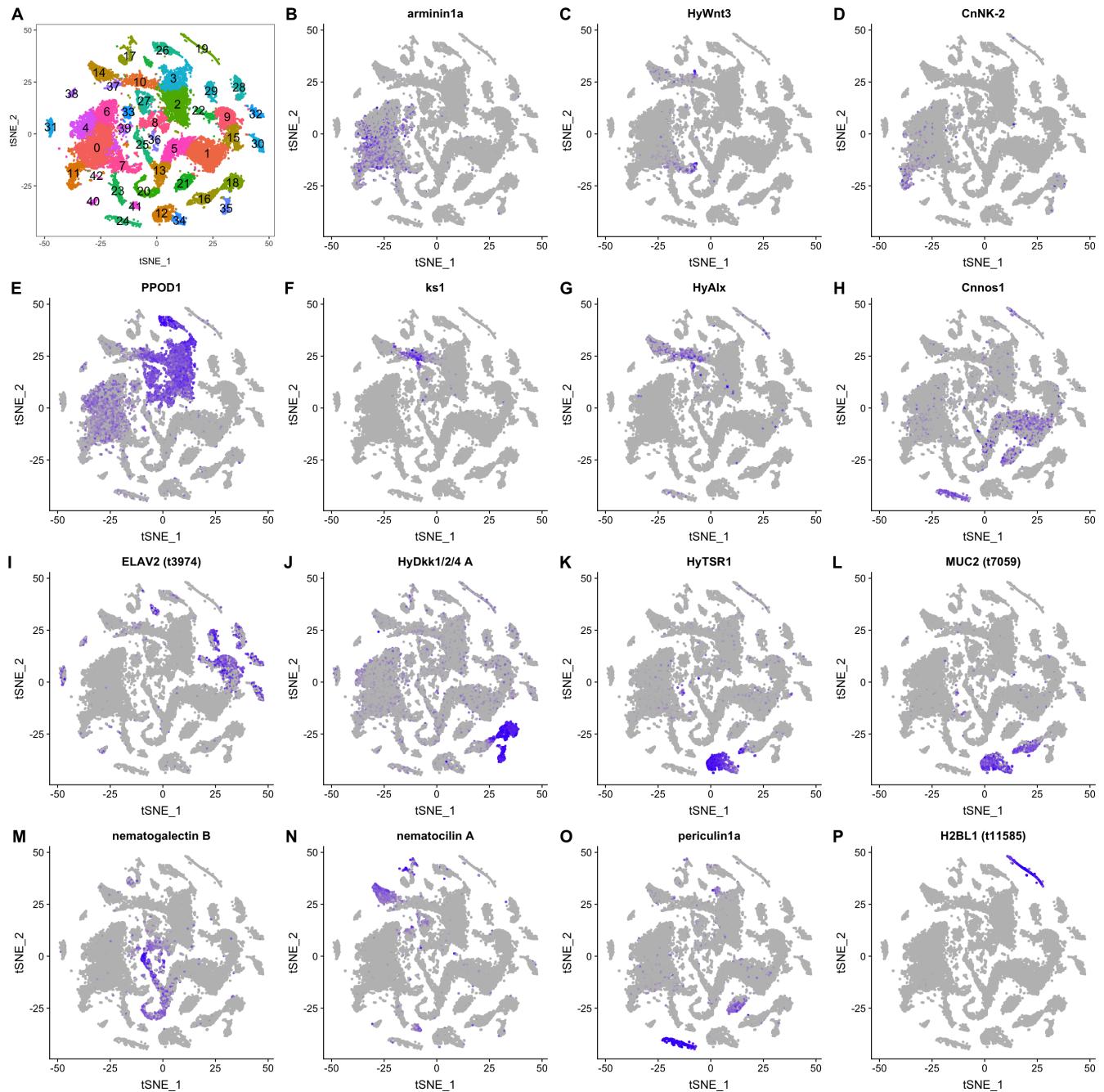


Figure 7: Selected markers used for cluster annotation. A) t-SNE plot. B) endoderm - *arminin1a* (2). C) endoderm/ectoderm hypostome - *Wnt3* (3). D) endoderm foot/peduncle - *CnNK-2* (4). E) ectoderm/endoderm - *PPOD1* (5). F) ectoderm head - *ks1* (6). G) ectoderm tentacle - *HyAlx* (7). H) multipotent i-cells/progenitors/female and male germline - *Cnnos1* (8), I) neuron progenitor/neurons - *ELAV2* (t3974, this study), J) zymogen gland cell - *HyDkk1/2/4-A* (9, 10). K) mucous gland cells - *HyTSR1* (11). L) mucous gland cell - *MUC2* (t7059). M) nematogenesis/biological doublets - *nematoglectin B* (12). N) differentiated nematocytes/battery cell - *nematocillin A* (13). O) female germ line - *periculin1a* (14). P) male germline - histone *H2BL1* (t11585, this study).

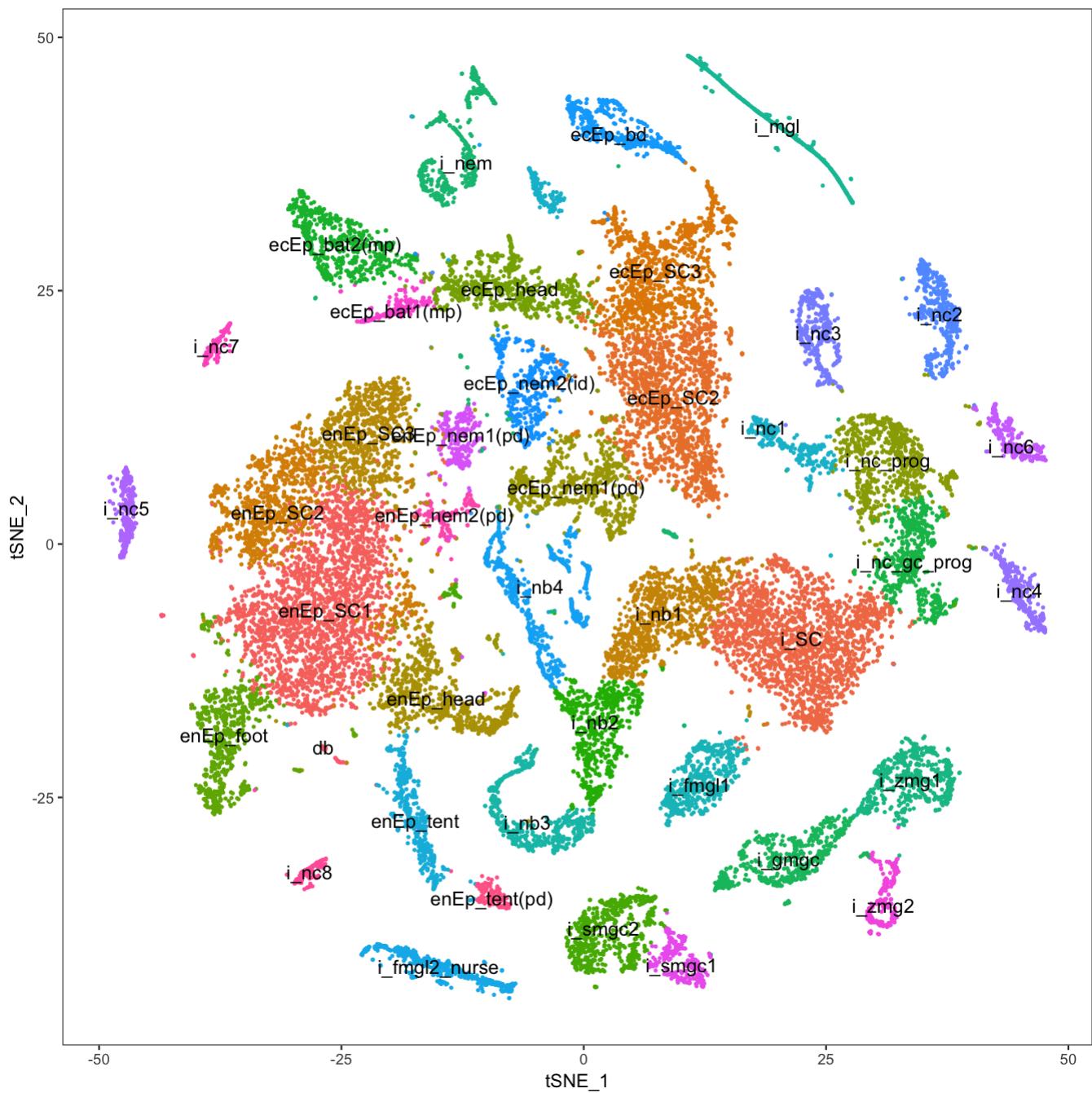


Figure 8: Annotated t-SNE plot (final gene/UMI cut-offs, post initial doublet filtering). bat: battery cell, db: doublet, ecEP: ectodermal epithelial cell, enEP: endodermal epithelial cell, fmg1: female germ-line, gc: gland cell, gmgc: granular mucous gland cell, i: cell of the interstitial lineage, id: integration doublet, mgl: male germline, mp: multiplet, nb: nematoblast, nc: neuronal cell, nem: nematocyte, pd: suspected phagocytosis doublet, prog: progenitor, SC: stem cell, smgc: spumous mucous gland cell, tent: tentacle, zmg: zymogen gland cell.

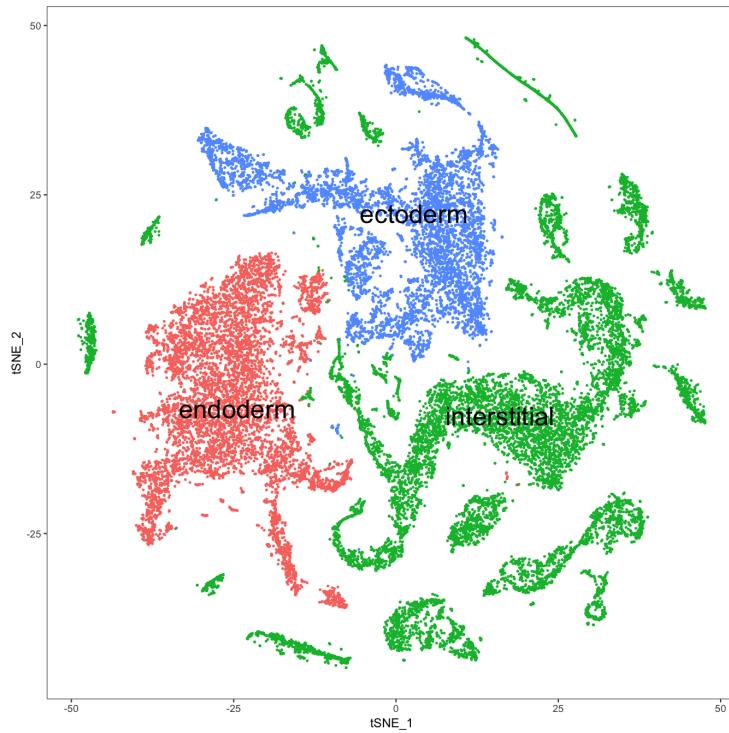


Figure 9: Annotated t-SNE plot colored by cell lineage.

Cluster annotation by cell lineage

We annotate the t-SNE representation by cell lineage (Fig. 9).

```
# Restore original cluster numbering
ds.s1 <- SetAllIdent(object = ds.s1, id = "cluster_numbering")

cluster.names <- c("endoderm", "interstitial", "ectoderm", "ectoderm", "endoderm",
  "interstitial", "endoderm", "endoderm", "ectoderm", "interstitial", "ectoderm",
  "endoderm", "interstitial", "interstitial", "ectoderm", "interstitial", "interstitial",
  "interstitial", "interstitial", "interstitial", "interstitial", "interstitial",
  "interstitial", "endoderm", "interstitial", "interstitial", "ectoderm", "ectoderm",
  "interstitial", "interstitial", "interstitial", "interstitial", "interstitial",
  "endoderm", "interstitial", "interstitial", "interstitial", "interstitial",
  "endoderm", "interstitial", "endoderm", "endoderm")

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

# Store lineage labels
ds.s1 <- StashIdent(object = ds.s1, save.name = "lineages")

TSNEPlot(object = ds.s1, do.return = T, do.label = T, no.legend = TRUE, label.size = 8,
  pt.size = 0.5)
```

Plot selected metagenes

NMF identifies sets of genes that are co-expressed (metagene, gene module) and outputs scores that indicate how strongly a metagene is expressed in a particular cell. We visualize these scores on the t-SNE representation (Fig. 10). We provide a supplementary file with the top 30 genes that characterize a metagene in the repository (nmf/wt_K96/GoodMeta_Top30.csv). NMF analysis wt_K96 was performed using cells in Seurat object Hydra_Seurat_Whole_Transcriptome.rds.

```
# Load state labels
ds.s1 <- SetAllIdent(object = ds.s1, id = "states")
```

```

# Load metagene scores for each cell
cellScores <- read.csv("nmf/wt_K96/GoodMeta_CellScores.csv", row.names = 1, check.names = F)
cellScores <- as.data.frame(cellScores)
# Make metagenes columns
cellScores <- t(cellScores)
# Fix cell ids
rownames(cellScores) <- sub("X", "", rownames(cellScores))
rownames(cellScores) <- sub("\\.", "-", rownames(cellScores))

# Add scores
cellScores.ds.s1 <- cellScores[match(rownames(ds.s1@meta.data), rownames(cellScores)),
]
ds.s1@meta.data <- cbind(ds.s1@meta.data, cellScores.ds.s1)

## Save object with metagene information
## saveRDS(ds.s1,paste0(data.path,'objects/ds.s1.s100_pc31_p40.rds'))

p1 <- TSNEPlot(object = ds.s1, do.label = T, label.size = 3, pt.size = 0.5, cex.names = 6,
no.legend = TRUE, do.return = TRUE)
p2 <- FeaturePlot(ds.s1, c("wt18", "wt11", "wt62", "wt2", "wt30", "wt82", "wt89",
"wt76", "wt23", "wt13", "wt8", "wt45", "wt33", "wt75", "wt41", "wt31", "wt71",
"wt70", "wt7", "wt25", "wt15", "wt22", "wt32", "wt66"), cols.use = c("grey",
"blue"), do.return = TRUE)

# generate plotlist
plotlist <- prepend(p2, list(p1))

# Plot metagenes
plot_grid(plotlist = plotlist, labels = "AUTO", label_size = 20, align = "h", ncol = 5)

```

Multiplet exploration

We use metagenes to further explore cells that express both epithelial and nematocyte genes and categories of these associations (Fig. 11). Whereas mature nematocytes can be mounted in epithelial cells this is not the case for nematoblasts which reside in interstitial spaces. Metagene co-expression analysis reveals doublet populations that include nematoblasts or mature nematocytes (Fig. 11).

```

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

# Plot highlighting nematoblast/epithelial cell, mature nematocyte/epithelial
# cell doublets

p <- TSNEPlot(object = ds.s1, do.label = T, label.size = 4, pt.size = 0.5, cex.names = 6,
no.legend = TRUE, do.return = TRUE)
d1 <- FeaturePlot(object = ds.s1, features.plot = c("wt11", "wt45"), cols.use = c("grey",
"green", "blue", "red"), max.cutoff = 0.5, overlay = TRUE, no.legend = TRUE,
do.return = TRUE)
d2 <- FeaturePlot(object = ds.s1, features.plot = c("wt11", "wt33"), cols.use = c("grey",
"green", "blue", "red"), max.cutoff = 0.5, overlay = TRUE, no.legend = TRUE,
do.return = TRUE)
d3 <- FeaturePlot(object = ds.s1, features.plot = c("wt2", "wt45"), cols.use = c("grey",
"green", "blue", "red"), max.cutoff = 0.5, overlay = TRUE, no.legend = TRUE,
do.return = TRUE)
d4 <- FeaturePlot(object = ds.s1, features.plot = c("wt2", "wt33"), cols.use = c("grey",
"green", "blue", "red"), max.cutoff = 0.5, overlay = TRUE, no.legend = TRUE,
do.return = TRUE)

# Plot with tsne
p0 <- FeaturePlot(ds.s1, c("wt11", "wt2"), cols.use = c("grey", "green"), do.return = TRUE,
max.cutoff = 0.5)
p1 <- FeaturePlot(ds.s1, c("wt45", "wt33"), cols.use = c("grey", "blue"), do.return = TRUE,
max.cutoff = 0.5)
p2 <- FeaturePlot(ds.s1, c("nematogalectin B", "nematocilin A"), cols.use = c("grey",
"blue"), do.return = TRUE)

plot_grid(p, p2[[1]], p2[[2]], p0[[1]], p1[[1]], d1[[1]], p0[[1]], p1[[2]], d2[[1]],
p0[[2]], p1[[1]], d3[[1]], p0[[2]], p1[[2]], d4[[1]], labels = "AUTO", label_size = 30,

```

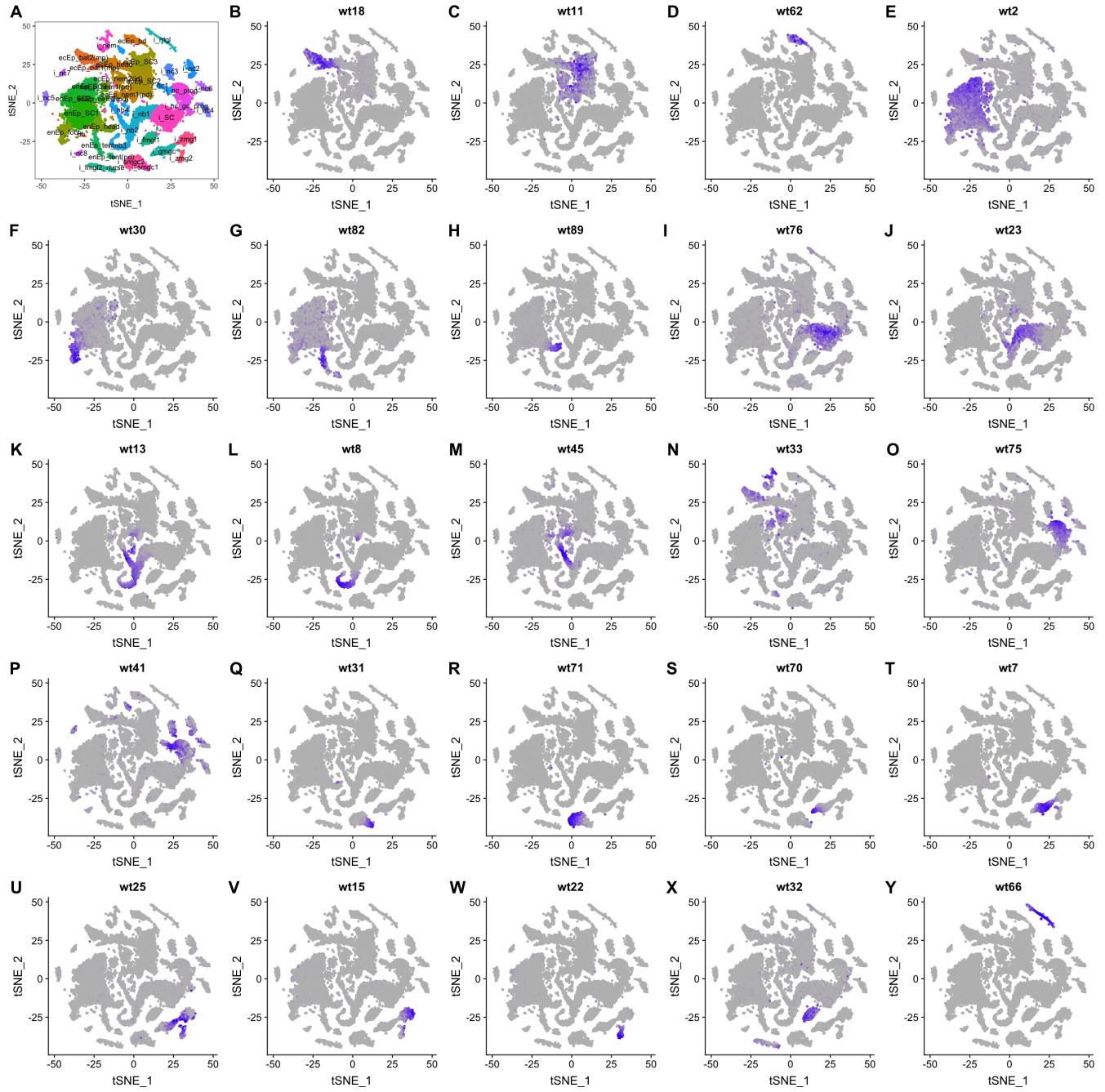


Figure 10: Selected metagenes identified in NMF analysis for the whole dataset (wt_K96). A metagene describes a set of genes co-expressed in the highlighted cell population. A) Annotated t-SNE. B) Tentacle ectodermal epithelial cells. This metagene includes transcripts that are expressed in the epithelial cell of a battery cell complex since no expression is found in neuronal or nematocyte cell populations. C) Ectodermal epithelial cells, body column. D) Ectodermal epithelial cells, basal disk. E) Endodermal epithelial cells, body column. F) Endodermal epithelial cells, foot. G) Endodermal epithelial cells, tentacle. H) Endodermal epithelial cells, hypostome. I) Interstitial stem cells and early progenitors. J) Early stage nematoblast, singletons and phagocytosed. K) Mid stage nematoblast, singletons and phagocytosed. L,M) Late nematoblast, singletons and phagocytosed. N) Mature nematocyte, singletons and integrated. O) Nerve cell progenitors. P) Differentiated neurons. Q) Spumous mucous gland cells, hypostome. R) Spumous mucous gland cells, mid/lower head. S) Granular mucous gland cells, hypostome. T) Granular mucous gland cells, mid/lower head. U) Granular mucous gland cells/zymogen gland cells. V,W) Zymogen gland cells. X) Female germline cells. Y) Male germline cells.

```
    align = "h", ncol = 3)
```

Cells and UMIs across cell states

Considering all cells we find a median of 1936 genes and a median of 5672 UMIs per cell. We calculate metrics for each cell state/cluster (Fig. 12).

```
# Calculate gene/UMI metrics for states

# load state labels
ds.s1 <- SetAllIdent(object = ds.s1, id = "states")

# calculate median gene and UMI numbers for all cluster
mylist <- list()

for (i in levels(ds.s1@ident)) {
  vec <- numeric(3)
  s <- SubsetData(object = ds.s1, ident.use = i)
  vec[1] <- round(median(s@meta.data$nGene), digits = 0)
  vec[2] <- round(median(s@meta.data$nUMI), digits = 0)
  vec[3] <- length(s@meta.data$nGene)
  mylist[[i]] <- vec
}
df <- do.call("rbind", mylist) #combine all vectors into a matrix
df <- as.data.frame(df)
colnames(df) <- c("medianGene", "medianUMI", "cells")

df <- df[order(df$medianGene), ]
df <- df[, c(3, 1, 2)]

# Create table

# Function to scale each column to the range [0, 1]
norm <- function(x) {
  apply(x, 2, function(y) {
    (y - min(y))/(max(y) - min(y))
  })
}

bluecol <- colorRamp(c("red", "yellow", "green"))(norm(df))
bluecol <- rgb(bluecol[, 1], bluecol[, 2], bluecol[, 3], max = 255)

tt <- ttheme_default(core = list(bg_params = list(fill = bluecol)))

g <- tableGrob(df, theme = tt)
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 2, b = nrow(g),
  l = 1, r = ncol(g))
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 1, l = 1,
  r = ncol(g))
grid.draw(g)
```

Library stats

```
# Calculate number of cells, median gene and UMI numbers for all libraries
mylist <- list() #create an empty list

for (i in levels(ds.s1@meta.data$orig.ident)) {
  vec <- numeric(3) #preallocate a numeric vector
  id <- rownames(ds.s1@meta.data)[ds.s1@meta.data[, "orig.ident"] == i]
  s <- SubsetData(object = ds.s1, cells.use = id)
  vec[1] <- length(s@meta.data$nGene)
  vec[2] <- round(median(s@meta.data$nGene), digits = 0)
  vec[3] <- round(median(s@meta.data$nUMI), digits = 0)
  mylist[[i]] <- vec #put all vectors in the list
}
df <- do.call("rbind", mylist) #combine all vectors into a matrix
df <- as.data.frame(df)
```

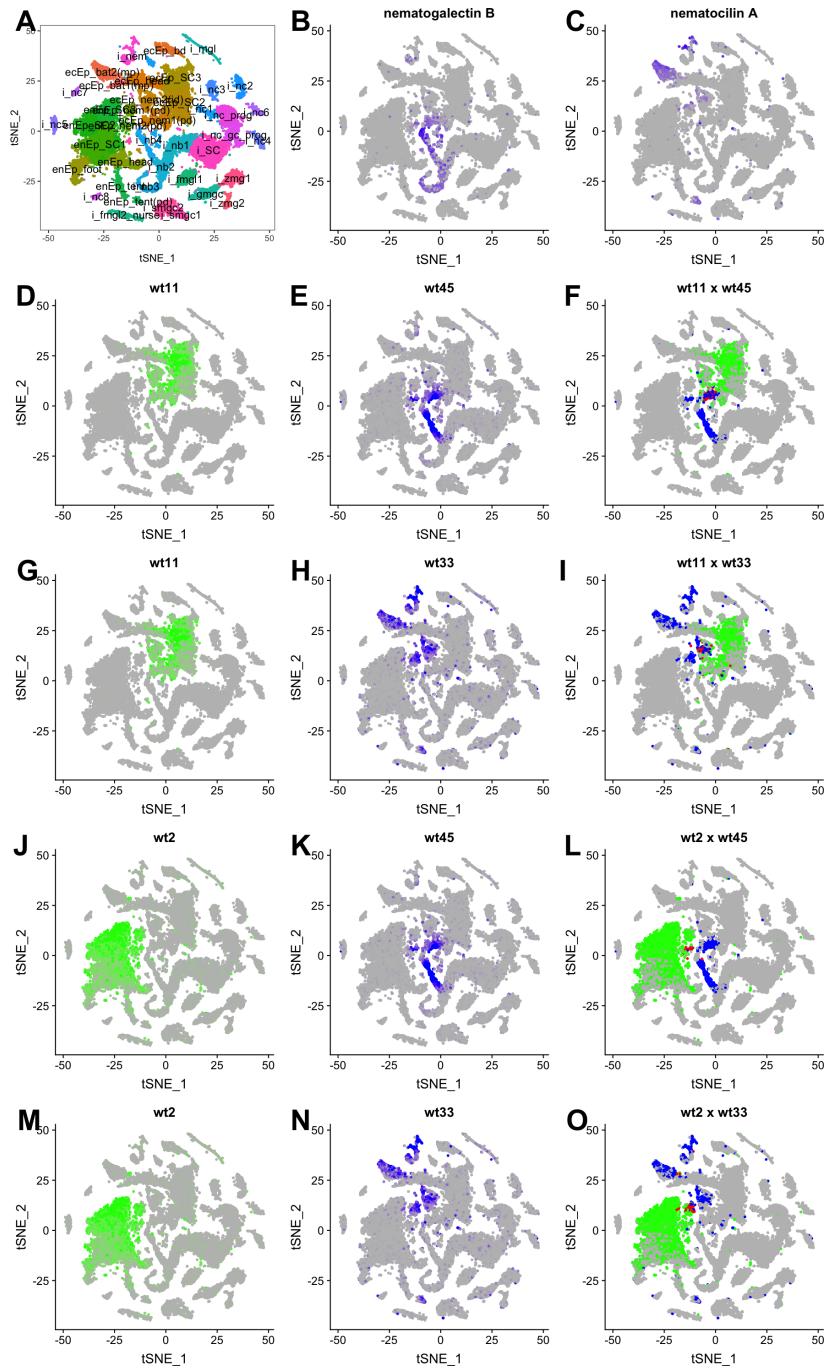


Figure 11: Biological multiplets and suspected phagocytosis doublets. Co-expression analyses of selected markers/metagenes identify multiplet categories. A) Annotated t-SNE plot. B) Nematoblast gene nematogalectin B [Hwang:2010hx] highlights metagene t45 as nematoblast metagene. C) Nematocilin A [Hwang:2008gd] is expressed in mature nematocytes and highlights t33 as nematocyte metagene. Note expression in battery cell clusters. D,G) Metagene wt11 is expressed in ectodermal epithelial cells of the body column. E,K) Metagene wt45 is expressed in developing nematocytes (nematoblasts). F) Co-expression of metagenes wt11 and wt45 (red cells) - nematoblast/epithelial cell phagocytosis doublet. H,N) Metagene wt33 is expressed in mature nematocytes. I) Co-expression of metagenes wt11 and wt33 (red cells) - mature nematocyte/epithelial cell integration doublet. J,M) Metagene wt2 is expressed in endodermal epithelial cells of the body column. L) Co-expression of metagenes wt2 and wt45 (red cells) - nematoblast/epithelial cell phagocytosis doublet. O) Co-expression of metagenes wt2 and wt33 (red cells) - mature nematocyte/epithelial cell integration doublet.

	cells	medianGene	medianUMI
<i>i_nc2</i>	442	492	958
<i>i_nc6</i>	267	501	920
<i>i_nc3</i>	366	512	1042
<i>i_nc1</i>	478	545	874
<i>i_nc7</i>	160	552	1316
<i>i_nc4</i>	311	556	1012
<i>i_nc5</i>	287	574	1128
<i>i_nc8</i>	143	628	1247
<i>i_nb4</i>	701	695	1636
<i>i_nem</i>	549	925	1767
<i>i_smgc1</i>	251	964	2575
<i>i_nb3</i>	529	1112	3607
<i>enEp_SC3</i>	841	1127	2479
<i>i_gmhc</i>	560	1132	3492
<i>i_nc_prog</i>	717	1203	2537
<i>i_zmg2</i>	246	1214	7788
<i>i_nb2</i>	608	1229	2770
<i>i_smgc2</i>	618	1356	4087
<i>i_zmg1</i>	537	1369	8670
<i>i_nc_gc_prog</i>	567	1780	4608
<i>enEp_tent</i>	458	1795	4921
<i>i_mgl</i>	535	1800	4072
<i>i_nb1</i>	902	1938	5404
<i>ecEp_bd</i>	452	1942	5957
<i>enEp_tent(pd)</i>	134	2122	6194
<i>i_SC</i>	1879	2153	6738
<i>ecEp_SC2</i>	1563	2303	7031
<i>ecEp_bat1(mp)</i>	200	2330	6357
<i>enEp_head</i>	825	2374	6783
<i>enEp_SC2</i>	961	2445	7775
<i>enEp_foot</i>	659	2571	7971
<i>i_ftngl1</i>	521	2653	7014
<i>enEp_nem1(pd)</i>	257	2968	9732
<i>ecEp_head</i>	695	3001	10351
<i>enEp_nem2(pd)</i>	143	3065	10704
<i>ecEp_bat2(mp)</i>	608	3153	9921
<i>ecEp_nem2(id)</i>	449	3492	12719
<i>enEp_SC1</i>	2203	3589	14226
<i>ecEp_SC3</i>	1145	3639	14978
<i>ecEp_nem1(pd)</i>	759	3660	14648
<i>i_ftngl2_nurse</i>	458	4640	16686

Figure 12: Median genes/UMIs per cell per state (all libraries).

	cells	medianGene	medianUMI
01-D1	1023	3400	15235
01-P2	1256	3090	10534
02-CO	2329	1941	5174
02-P1	3343	2413	7205
02-PB	1562	2596	8006
03-FM	886	2644	10309
03-KI	1958	2460	9914
03-MA	702	3056	12474
06-FM	1122	3722	13932
06-KI	1937	2075	7318
06-MA	384	1621	3466
11-BU	3207	1342	3281
11-PO	2058	955	2350
12-N1	1264	570	1085
12-N2	1953	557	1079

Figure 13: Cells, median genes/UMIs per cell per libraries. Cells with >300 <7k genes and >500UMI <50k UMIs.

```

colnames(df) <- c("cells", "medianGene", "medianUMI")

tt <- ttheme_default(core = list(fg_params = list(hjust = 1, x = 0.9)), rowhead = list(fg_params = list(hj
  x = 0.95)))

g <- tableGrob(df, theme = tt)
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 2, b = nrow(g),
  l = 1, r = ncol(g))
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 1, l = 1,
  r = ncol(g))
grid.draw(g)

```

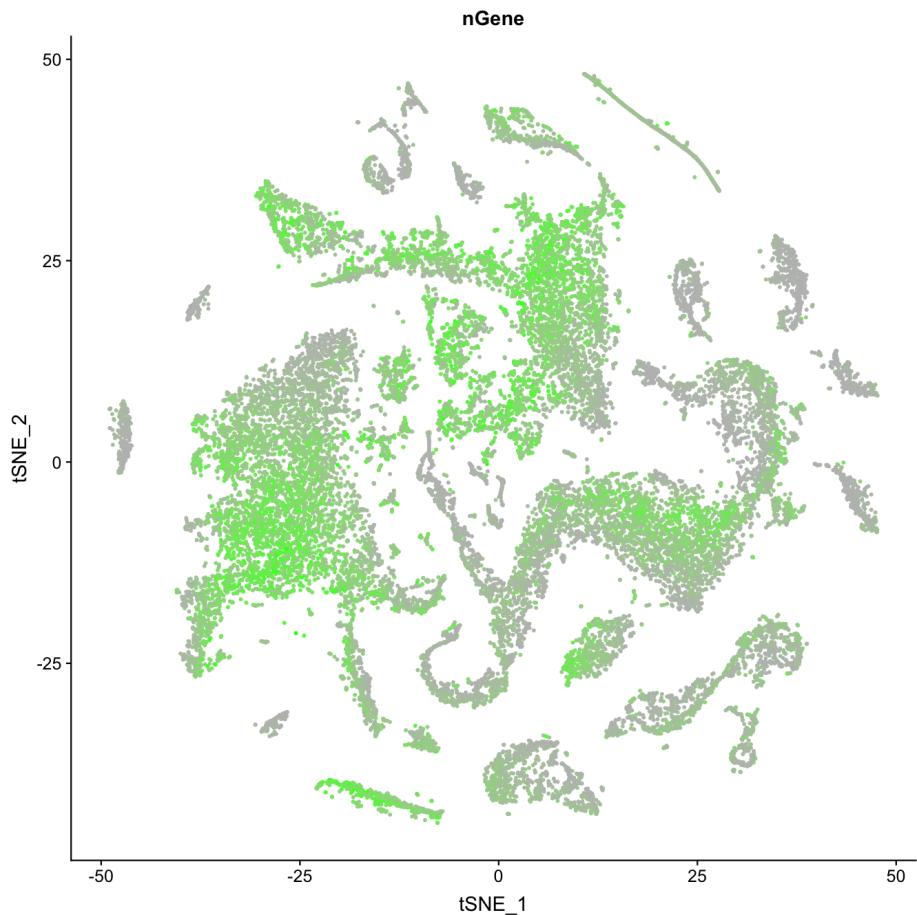


Figure 14: t-SNE plot. Number of genes.

Quality Metrics

Finally, we take a look at number of genes, UMIs and the contribution from each library (batch) (Figs. 14, 15, 16).

```
# number of genes
FeaturePlot(object = ds.s1, features.plot = "nGene", cols.use = c("grey", "green"))

# number of umis
FeaturePlot(object = ds.s1, features.plot = "nUMI", cols.use = c("grey", "red"))

# by batch
TSNEPlot(object = ds.s1, group.by = "orig.ident")
```

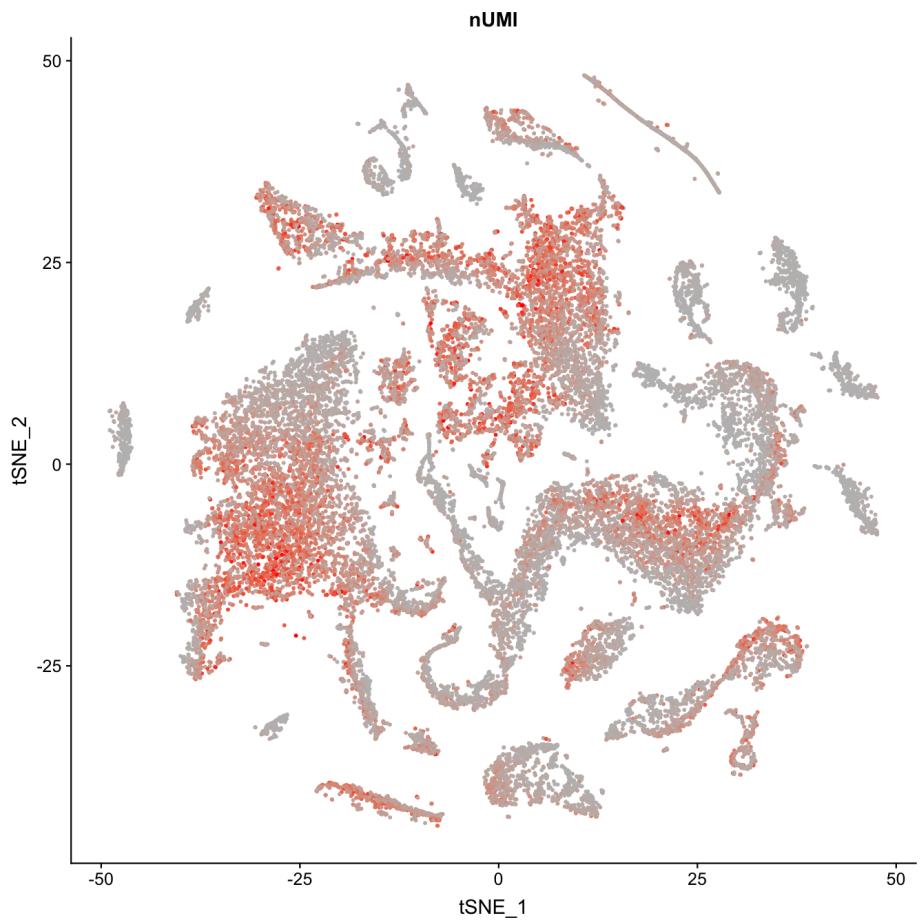


Figure 15: t-SNE plot. Number of UMIs.

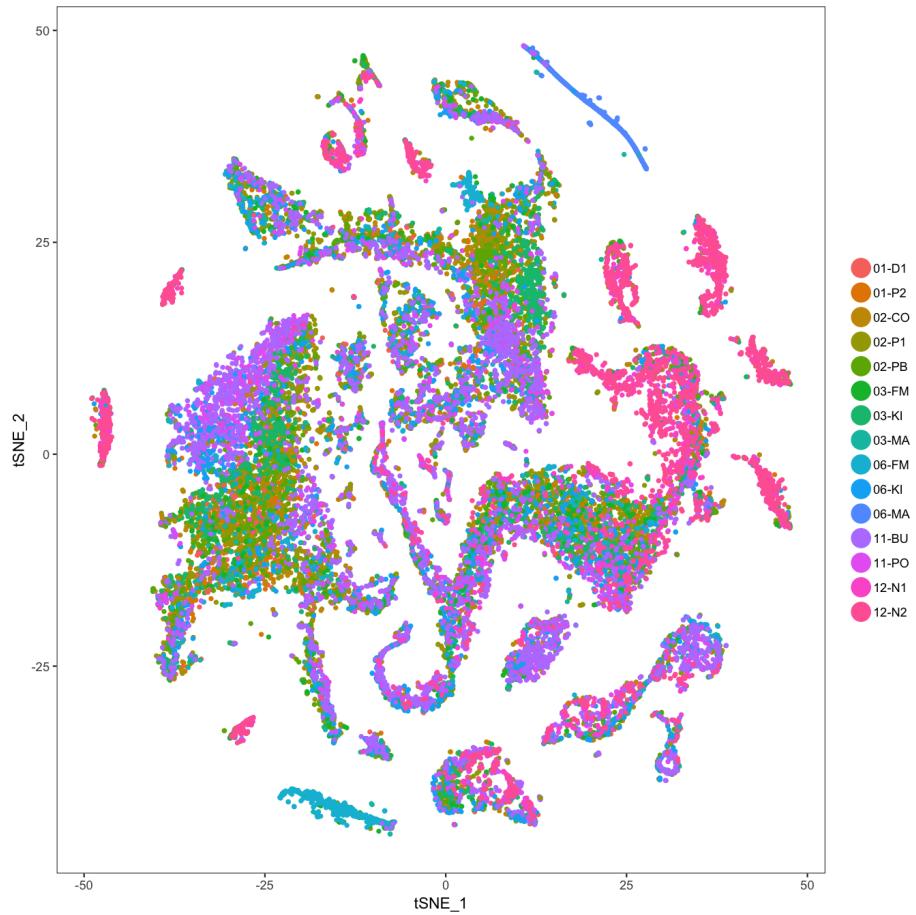


Figure 16: t-SNE plot. Cells by library.

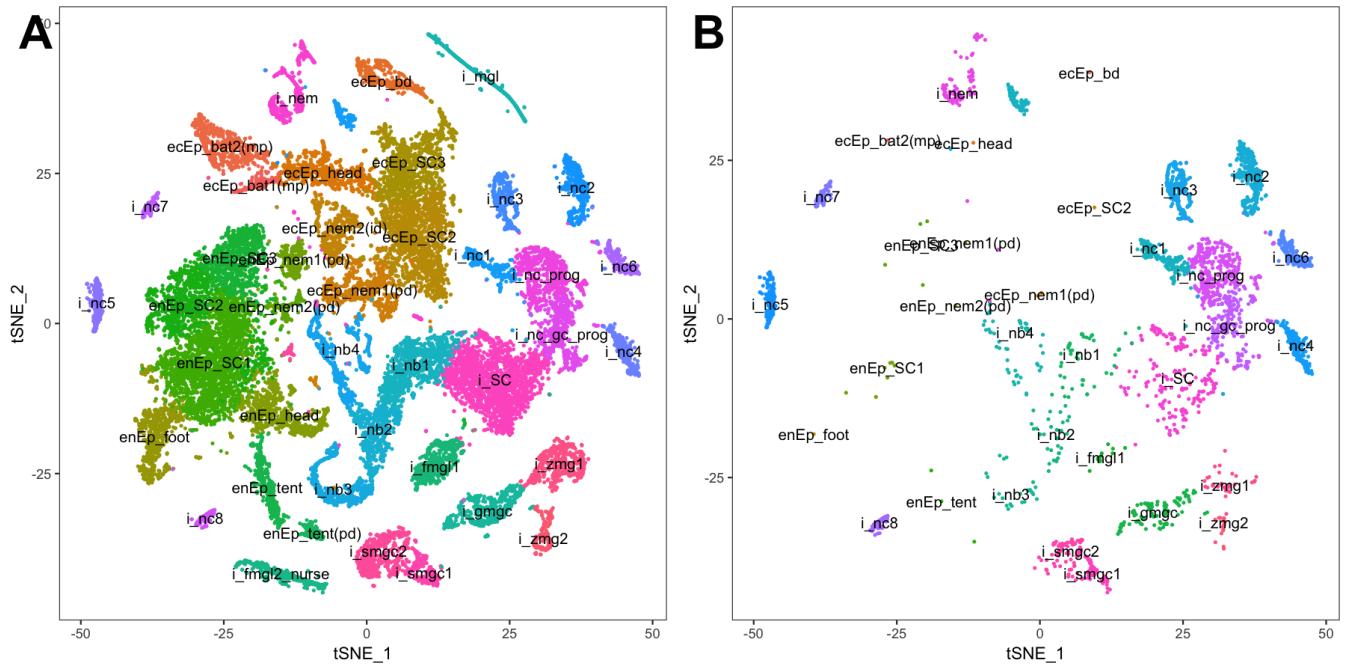


Figure 17: Cells from neuronal libraries 12-. A) Annotated t-SNE plot - whole data set. B) Annotated t-SNE plot - subset of cells from neuronal libraries 12-.

Characterization of transgenic line nGreen

The transgenic line nGreen that was used in FACS shows mosaic GFP expression predominantly in the neuronal cell lineage caused by a random integration event after plasmid microinjection into the zygote. The expression is driven by the actin promoter, a promoter that is ubiquitously expressed in wild type *Hydra*. We extract cells from the neuronal libraries (12-)(Fig. 17) and use the cell state annotations to elucidate the composition of the GFP expressing population of cells (Fig. 18). 75.7% of the 3,218 cells that are retained after filtering are neuronal progenitors or differentiated neurons.

```
## Cell composition of neuronal libraries 12-
```

```
p1 <- TSNEPlot(object = ds.s1, do.label = T, label.size = 3.5, pt.size = 0.5, cex.names = 6,
no.legend = TRUE, do.return = TRUE)

# Get ids for cells from FACS libraries
n1 <- rownames(ds.s1@meta.data)[ds.s1@meta.data[, "orig.ident"] == "12-N1"]
n2 <- rownames(ds.s1@meta.data)[ds.s1@meta.data[, "orig.ident"] == "12-N2"]

facs <- c(n1, n2)

ds.fa <- SubsetData(object = ds.s1, cells.use = facs, subset.raw = TRUE)
p2 <- TSNEPlot(object = ds.fa, do.label = T, label.size = 3.5, pt.size = 0.5, cex.names = 6,
no.legend = TRUE, do.return = TRUE)

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

```

# calculate median gene and UMI numbers for all cluster
mylist <- list() #create an empty list

for (i in levels(ds.fa@ident)) {
  vec <- numeric(3) #preallocate a numeric vector
  # \vec[1] <- as.numeric(i)
  s <- SubsetData(object = ds.fa, ident.use = i)
  vec[1] <- round(median(s@meta.data$nGene), digits = 0)
  vec[2] <- round(median(s@meta.data$nUMI), digits = 0)
  vec[3] <- length(s@meta.data$nGene)
  mylist[[i]] <- vec #put all vectors in the list
}
df <- do.call("rbind", mylist) #combine all vectors into a matrix
df <- as.data.frame(df)
colnames(df) <- c("medianGene", "medianUMI", "cells")

df <- df[order(-df$cells), ]
df <- df[, c(3, 1, 2)]

tt <- ttheme_default(core = list(fg_params = list(hjust = 1, x = 0.9)), rowhead = list(fg_params = list(hj
  x = 0.95)))

g <- tableGrob(df, theme = tt)
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 2, b = nrow(g),
  l = 1, r = ncol(g))
g <- gtable_add_grob(g, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2)), t = 1, l = 1,
  r = ncol(g))
grid.draw(g)

```

	cells	medianGene	medianUMI
<i>i_nc_prog</i>	384	793	1498
<i>i_nc2</i>	381	457	862
<i>i_nc1</i>	367	490	787
<i>i_nc3</i>	266	437	869
<i>i_nc4</i>	238	506	871
<i>i_nc6</i>	215	456	839
<i>i_SC</i>	206	1154	2886
<i>i_nc5</i>	196	488	918
<i>i_nc_gc_prog</i>	171	718	1432
<i>i_nc7</i>	118	510	1198
<i>i_gmgc</i>	102	846	2218
<i>i_nc8</i>	99	541	1079
<i>i_nem</i>	99	665	1121
<i>i_smgc2</i>	87	905	2217
<i>i_nb4</i>	51	416	975
<i>i_smgc1</i>	47	642	1389
<i>i_nb1</i>	45	905	2214
<i>i_zmg1</i>	35	1465	7185
<i>i_nb2</i>	32	836	1768
<i>i_nb3</i>	22	828	2604
<i>i_zmg2</i>	21	1389	6269
<i>i_fmg1</i>	12	2489	7018
<i>enEp_SC1</i>	7	3463	11710
<i>enEp_SC3</i>	4	937	2188
<i>enEp_tent</i>	3	3553	15369
<i>ecEp_nem1(pd)</i>	2	4032	24592
<i>ecEp_bat2(mp)</i>	1	4964	22784
<i>ecEp_bd</i>	1	431	728
<i>ecEp_head</i>	1	2279	6486
<i>ecEp_SC2</i>	1	1373	3466
<i>enEp_foot</i>	1	740	1424
<i>enEp_nem1(pd)</i>	1	1940	5289
<i>enEp_nem2(pd)</i>	1	2391	6963

Figure 18: Composition of transgenic cell population from strain nGreen used in FACS.

Software versions

This document was computed on Thu Nov 01 14:22:07 2018 with the following R package versions.

```
R version 3.4.3 (2017-11-30)
Platform: x86_64-apple-darwin15.6.0 (64-bit)
Running under: OS X El Capitan 10.11.6

Matrix products: default
BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/3.4/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] bindrcpp_0.2.2 rlang_0.3.0.1  gridExtra_2.3  gtable_0.2.0
[5] dplyr_0.7.7   Seurat_2.2.1   Matrix_1.2-12  cowplot_0.9.2
[9] ggplot2_3.1.0 knitr_1.20

loaded via a namespace (and not attached):
 [1] diffusionMap_1.1-0    Rtsne_0.13          VGAM_1.0-5
 [4] colorspace_1.3-2       ggridges_0.4.1      class_7.3-14
 [7] modeltools_0.2-21     mclust_5.4          rprojroot_1.3-2
[10] htmlTable_1.11.2      base64enc_0.1-3    proxy_0.4-21
[13] rstudioapi_0.7         DRR_0.0.3           flexmix_2.3-14
[16] lubridate_1.7.3        prodlim_1.6.1      mvtnorm_1.0-7
[19] ranger_0.9.0           codetools_0.2-15   splines_3.4.3
[22] R.methodsS3_1.7.1     mnormt_1.5-5       robustbase_0.92-8
[25] tclust_1.3-1           RcppRoll_0.2.2     Formula_1.2-2
[28] caret_6.0-78          ica_1.0-1          broom_0.4.3
[31] ddalpha_1.3.1.1        cluster_2.0.6     kernlab_0.9-25
[34] R.oo_1.21.0            sfsmisc_1.1-2     compiler_3.4.3
[37] backports_1.1.2       assertthat_0.2.0  lazyeval_0.2.1
[40] formatR_1.5            lars_1.2            acepack_1.4.1
[43] htmltools_0.3.6       tools_3.4.3        igraph_1.2.2
[46] glue_1.3.0             reshape2_1.4.3    Rcpp_0.12.19
[49] trimcluster_0.1-2     gdata_2.18.0      ape_5.1
[52] nlme_3.1-131.1        iterators_1.0.9   fpc_2.1-11
[55] psych_1.7.8           timeDate_3043.102 xfun_0.1
[58] gower_0.1.2           stringr_1.3.0      irlba_2.3.2
[61] gtools_3.5.0           DEoptimR_1.0-8     MASS_7.3-49
[64] scales_1.0.0           ipred_0.9-6       parallel_3.4.3
[67] RColorBrewer_1.1-2    yaml_2.1.18       pbapply_1.3-4
[70] segmented_0.5-3.0     rpart_4.1-13      latticeExtra_0.6-28
[73] stringi_1.1.6          highr_0.6          foreach_1.4.4
[76] checkmate_1.8.5        calTools_1.17.1.1 lava_1.6
[79] dtw_1.18-1              SDMTools_1.1-221 pkgconfig_2.0.2
[82] prabclus_2.2-6         bitops_1.0-6       evaluate_0.10.1
[85] lattice_0.20-35        ROCR_1.0-7        purrr_0.2.5
[88] bindr_0.1.1             labeling_0.3      recipes_0.1.2
[91] htmlwidgets_1.0          CVST_0.2-1        tidyselect_0.2.5
[94] plyr_1.8.4              magrittr_1.5      bookdown_0.7
[97] R6_2.3.0                gplots_3.0.1      Hmisc_4.1-1
[100] dimRed_0.1.0            sn_1.5-1          pillar_1.2.1
[103] foreign_0.8-69          withr_2.1.2       mixtools_1.1.0
[106] survival_2.41-3        scatterplot3d_0.3-40 nnet_7.3-12
[109] tsne_0.1-3              tibble_1.4.2       crayon_1.3.4
[112] KernSmooth_2.23-15     rmarkdown_1.9      data.table_1.11.8
[115] FNN_1.1                 ModelMetrics_1.1.0 metap_0.8
[118] digest_0.6.18           diptest_0.75-7    numDeriv_2016.8-1
[121] tidyverse_0.8.0          R.utils_2.6.0      stats4_3.4.3
[124] munsell_0.5.0
```

References

1. H. R. Bode, S. Heimfeld, M. A. Chow, L. W. Huang, Gland cells arise by differentiation from interstitial cells in *Hydra attenuata*. *Developmental Biology*. **122**, 577–585 (1987).
2. R. Augustin *et al.*, Activity of the novel peptide arminin against multiresistant human pathogens shows the considerable potential of phylogenetically ancient organisms as drug sources. *Antimicrobial agents and chemotherapy*. **53**, 5245–5250 (2009).
3. B. Hobmayer *et al.*, WNT signalling molecules act in axis formation in the diploblastic metazoan *Hydra*. *Nature*. **407**, 186–189 (2000).
4. A. Grens, L. Gee, D. A. Fisher, H. R. Bode, CnNK-2, an NK-2 Homeobox Gene, Has a Role in Patterning the Basal End of the Axis in *Hydra*. **180**, 473–488 (1996).
5. S. Thomsen, T. C. G. Bosch, Foot differentiation and genomic plasticity in *Hydra*: lessons from the PPOD gene family. *Development genes and evolution*. **216**, 57–68 (2006).
6. I. Endl, J. U. Lohmann, T. C. Bosch, Head-specific gene expression in *Hydra*: complexity of DNA-protein interactions at the promoter of ks1 is inversely correlated to the head activation potential. *Proceedings of the National Academy of Sciences of the United States of America*. **96**, 1445–1450 (1999).
7. K. M. Smith, L. Gee, H. R. Bode, HyAlx, an aristaless-related gene, is involved in tentacle formation in *hydra*. *Development (Cambridge, England)*. **127**, 4743–4752 (2000).
8. K. Mochizuki, H. Sano, S. Kobayashi, C. Nishimiya-Fujisawa, T. Fujisawa, Expression and evolutionary conservation of nanos-related genes in *Hydra*. *Wilhelm Roux' Archiv für Entwicklungsmechanik der Organismen*. **210**, 591–602 (2000).
9. R. Augustin *et al.*, Dickkopf related genes are components of the positional value gradient in *Hydra*. **296**, 62–70 (2006).
10. C. Guder *et al.*, An ancient Wnt-Dickkopf antagonism in *Hydra*. *Development (Cambridge, England)*. **133**, 901–911 (2006).
11. S. Siebert, F. Anton-Erxleben, T. C. G. Bosch, Cell type complexity in the basal metazoan *Hydra* is maintained by both stem cell based mechanisms and transdifferentiation. *Developmental biology*. **313**, 13–24 (2008).
12. J. S. Hwang *et al.*, Nematogaelectin, a nematocyst protein with GlyXY and galectin domains, demonstrates nematocyte-specific alternative splicing in *Hydra*. *Proceedings of the National Academy of Sciences of the United States of America*. **107**, 18539–18544 (2010).
13. J. S. Hwang *et al.*, Cilium evolution: identification of a novel protein, nematocilin, in the mechanosensory cilium of *Hydra* nematocytes. *Molecular biology and evolution*. **25**, 2009–2017 (2008).
14. S. Fraune *et al.*, In an early branching metazoan, bacterial colonization of the embryo is controlled by maternal antimicrobial peptides. *Proceedings of the National Academy of Sciences of the United States of America*. **107**, 18067–18072 (2010).