SA04 - Subclustering of cells from the interstitial lineage

Stefan Siebert

November 1, 2018

Summary

We subcluster cells from the interstitial cell lineage. The resulting data object is the starting point for URD trajectory reconstruction.

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
hFind <- function(x) {
    return(ds.s1@data@Dimnames[[1]][grep(x, ds.s1@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
```

Subsetting - cells of the interstitial lineage

We load the full data set and extract interstitial clusters (Fig. 1). This subset was used to perfom NMF to identify gene expression modules expressed in cells of the interstitial lineage (NMF analysis ic_K75).

```
# Read data object for the whole dataset

# Read full 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)

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

# Clusters of the interstitial lineage
ds.ic <- SubsetData(object = ds.s1, ident.use = c("1", "5", "9", "12",
    "13", "15", "16", "17", "18", "19", "20", "21", "22", "24", "25", "28",
    "29", "30", "31", "32", "34", "35", "36", "38", "40"), subset.raw = TRUE)

p1 <- TSNEPlot(object = ds.s1, group.by = "res.1.5", do.return = T, do.label = T,
    no.legend = TRUE)
p2 <- TSNEPlot(object = ds.ic, 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 (Fig. 2).

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

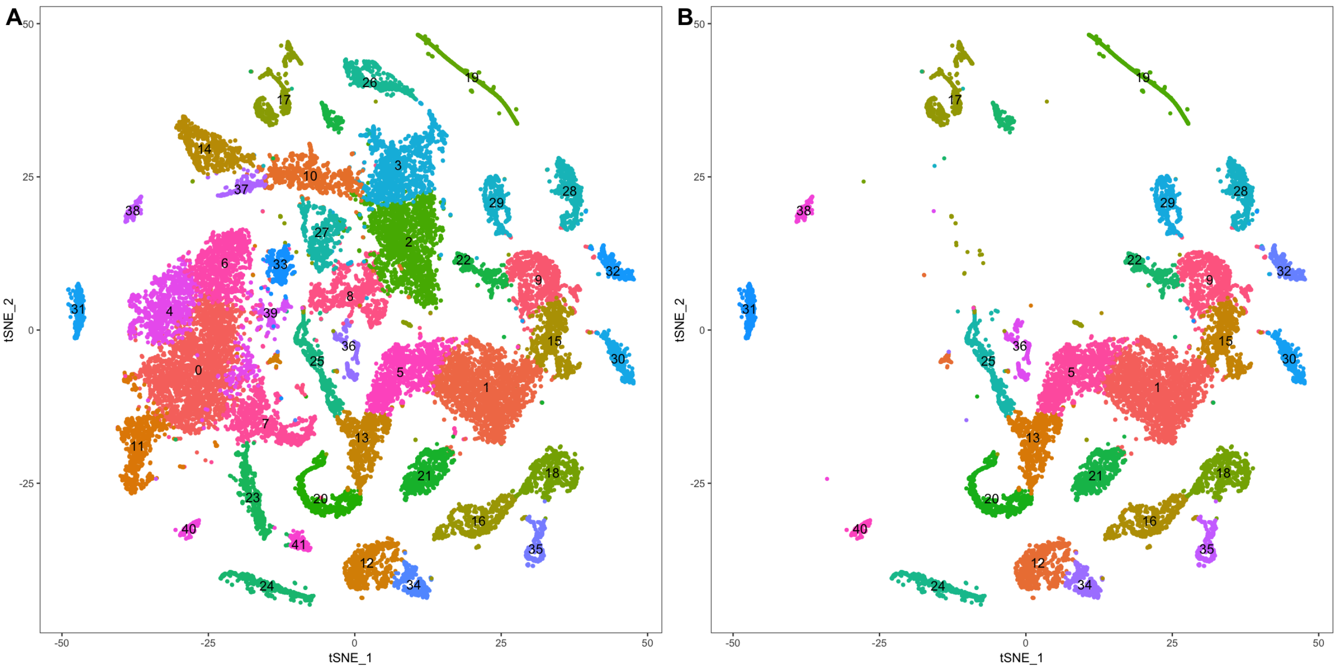

Figure 1: Subsetting cells of the interstitial lineage. A) Full t-SNE, B) Interstitial clusters.

```
# Scale
ds.ic <- ScaleData(object = ds.ic)
# PCA on highly variable genes
ds.ic <- RunPCA(object = ds.ic, pc.genes = ds.ic@var.genes, pcs.compute = 40,
    do.print = TRUE, pcs.print = 1:5, genes.print = 20)

# Project PCA to find genes that weren't scored as highly variable, but
# should belong to a given PC and include them.
ds.ic <- ProjectPCA(object = ds.ic)

## Perform permutation test to directly calculate p-values ds.ic <-
## JackStraw(object = ds.ic, num.pc = 40, num.replicate = 100, do.print
## = FALSE) JackStrawPlot(object = ds.ic, PCs=1:40)

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

# We run RunTSNE() for a range of principal components, seed and
# perplexities on compute cluster PCs 1:26 though 1:40 seed: 1, 300,
# 400 perplexity: 20, default (30), 40 The selected analysis considered
# PCs 1:31, perplexity 40, seed 300

ds.ic <- FindClusters(object = ds.ic, reduction.type = "pca", dims.use = 1:31,
    force.recalc = TRUE, resolution = 1.5, print.output = 0)
ds.ic <- RunTSNE(object = ds.ic, dims.use = c(1:31), do.fast = T, perplexity = 40)

# saveRDS(ds.ic,'objects/ds.ic.s300_pc31_p40.rds')
```

```
# Since t-SNE is not deterministic we here load the object of our
# original analysis
ds.ic <- readRDS("objects/ds.ic.s300_pc31_p40.rds")

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

Cluster annotation

We annotate the t-SNE using published marker genes (Fig. 3, 4).

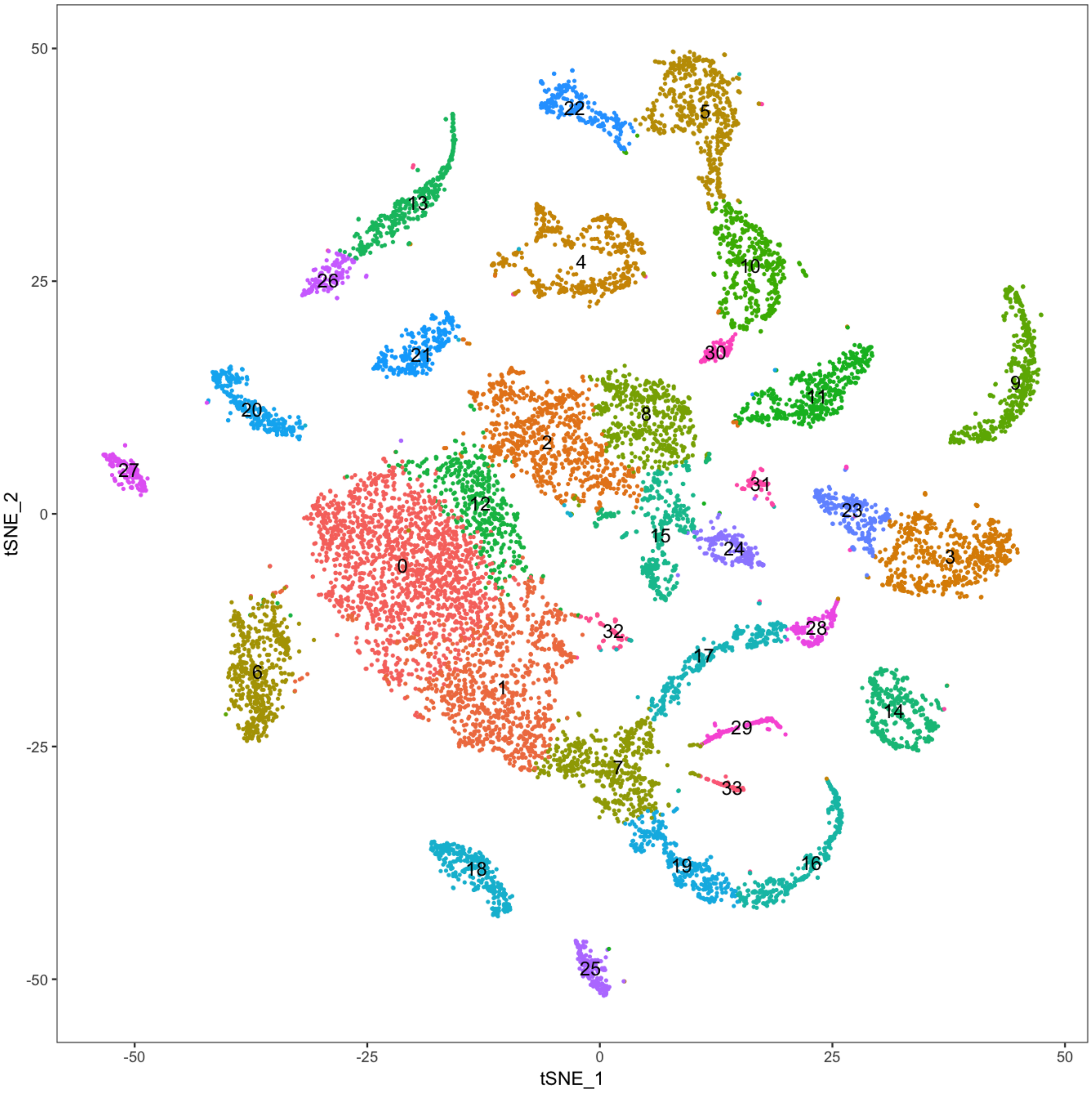

Figure 2: t-SNE plot for cells from the interstitial cell lineage.

```
# Genes to be plotted
gene.names <- c(hFind("t11407aep"), hFind("t15393aep"), hFind("t3974aep"),
    hFind("t27424aep"), hFind("t12642aep"), hFind("t15237aep"), hFind("t22117aep"),
    hFind("t23176aep"), hFind("t11117aep"), hFind("t38683aep"), hFind("t7059aep"),
    hFind("t35863aep"), hFind("t14102aep"), hFind("t8678aep"), hFind("t18356aep"))

# Updated gene names for readability

new.names <- c("Cnnos1", "HvSoxC", "ELAV2 (t3974)", "Myb domain (t27424)",
    "FOXL1 (t12642)", "nowa-1", "HyDkk-3", "nematocilin A", "periculin1a",
    "H10A (t38683)", "MUC2 (t7059)", "HyTSR1", "HyDkk1/2/4 C", "HyDkk1/2/4 A",
    "CHIA (t18356)")

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

# Plot with tsne
p1 <- TSNEPlot(object = ds.ic, do.label = T, label.size = 5, pt.size = 0.5,
    cex.names = 6, no.legend = TRUE, do.return = TRUE)
p2 <- FeaturePlot(ds.ic, c("Cnnos1", "HvSoxC", "ELAV2 (t3974)", "Myb domain (t27424)",
    "FOXL1 (t12642)", "nowa-1", "HyDkk-3", "nematocilin A", "periculin1a",
    "H10A (t38683)", "MUC2 (t7059)", "HyTSR1", "HyDkk1/2/4 C", "HyDkk1/2/4 A",
    "CHIA (t18356)"), cols.use = c("grey", "blue"), do.return = TRUE)

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

plot_grid(plotlist = plotlist, labels = "AUTO", label_size = 30, align = "h",
    ncol = 4)
```

```
# Since t-SNE is not deterministic we here load the object of our
# original analysis ds.ic <- readRDS(paste0(data.path,
# 'objects/ds.ic.s300_pc31_p40.rds')

# Annotate cluster

# Stash
ds.ic <- StashIdent(object = ds.ic, save.name = "cluster_numbering")

# Choose resolution, in case multiple resolutions were run
ds.ic <- SetAllIdent(ds.ic, "res.1.5")

# Current ids
current.cluster.ids <- as.character(0:33)

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

# Load annotations
cluster.names <- c("SC/nb", "nb1", "nc_gc_prog", "smgc2", "nem", "zmg1",
    "fmgl1", "nb2", "nc_prog", "fmgl2_nurse", "gmgc_head", "nc1", "SC/prog",
    "mgl2", "nc2", "nc_prog/nc9", "nb5", "nb4", "nc3", "nb3", "nc4", "nc5",
    "zmg2", "smgc1", "nc8", "nc6", "mgl1", "nc7", "nb6", "nb7", "gmgc_hyp",
    "db1", "db2", "nb8")

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

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

Identification of doublet clusters

We load gene modules that were identified in NMF analyses for the whole dataset (wt_K96). We plot scores for selected epithelial metagenes on the t-SNE representation for the whole dataset and the t-SNE representation for the cells of the interstitial lineage (Fig. 5). High scores for epithelial gene modules suggest an epithelial component in the transcriptomes of cells in interstitial clusters db1 and db2. These cells were excluded from downstream analyses.

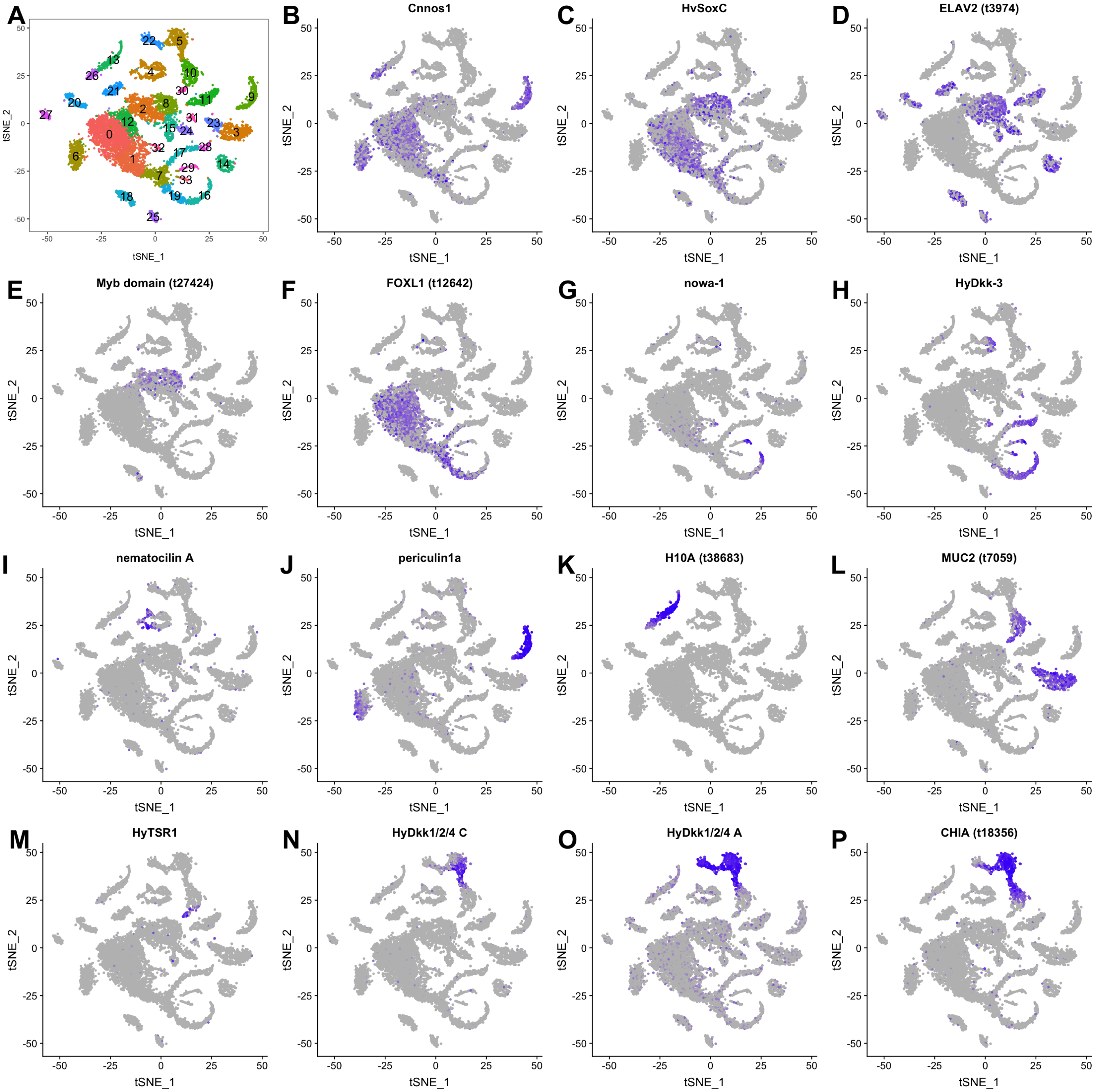

Figure 3: Selected markers used for cluster annotation. A) t-SNE plot. B) Interstitial stem cells, progenitor, germline - Cnnos1 (*1*). C) Differentiating progenitors - HvSoxC (this study)(*2*). D) Neuronal cells - ELAV2 (t3974) (this study). E) Neurogenesis, gland cell differentiation - Myb-like (t27424) (this study). F) Interstitial stem cells, nematoblasts - FOXL1 (t12642) (this study). G) Nematoblasts - Nowa-1 (*3*). H) Nematoblasts - HyDkk-3 (*4*). I) Differentiated nematocyte - nematocillin A (*5*). J) Female germline - periculin1a (*6*). K) Male germline - histone H10A (t3863) (this study). L) Granular and spumous mucous gland cells - MUC2 (t7059). M) Granular mucous gland cells - HyTSR1 (*7*). N) Zymogen gland cell - Hydkk1/2/4 C (*8*). O) Zymogen gland cell - Hydkk1/2/4 A (*8*). P) Zymogen gland cell - CHIA (t18356) (this study).

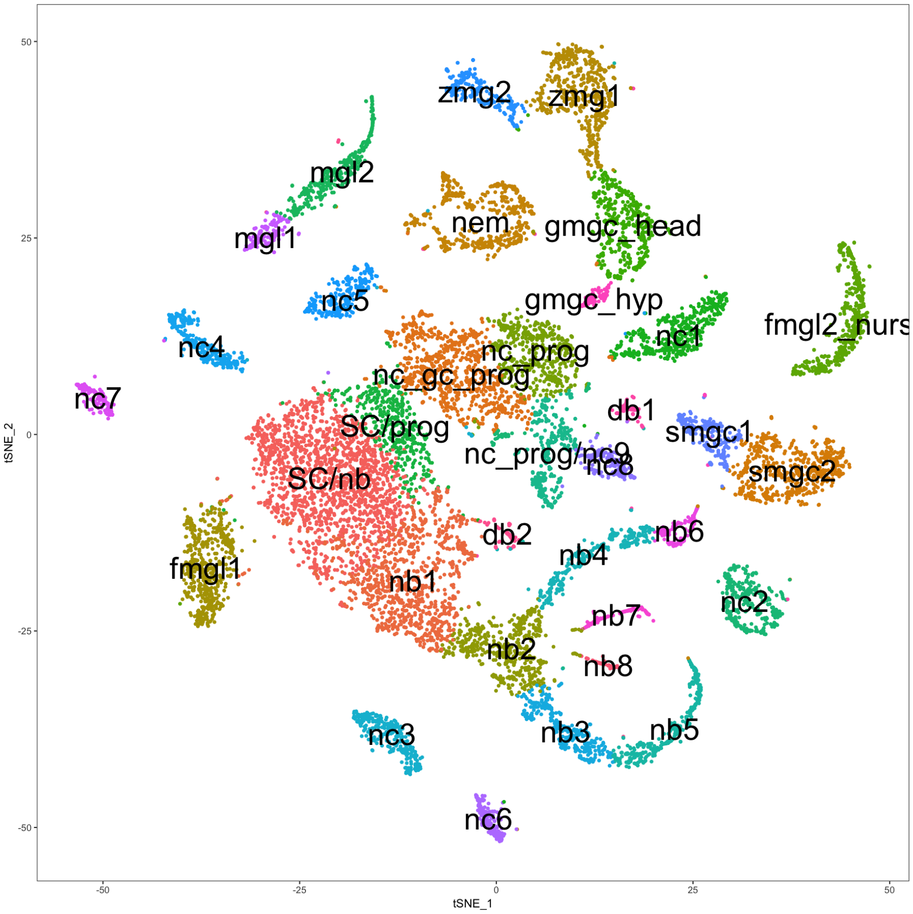

Figure 4: Annotated t-SNE representation for cells of the interstitial cell lineage. db: doublet, fmgl: female germline, gc: gland cell, gmgc: granular mucous cell, mgl: male germline, nc: neuronal cell, nb: nematoblast, nem: nematocyte, smgc: spumous mucous cell, prog: progenitor, SC: stem cell, zmg: zymogen gland cell.

```
# Ic metagene scores
cellScores <- read.csv("nmf/ic_K75/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))

# Match and load
cellScores <- cellScores[match(rownames(ds.ic@meta.data), rownames(cellScores)),
    ]
ds.ic@meta.data <- cbind(ds.ic@meta.data, cellScores)
head(ds.ic@meta.data)

# Whole metagene scores
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))

# Match and load
cellScores <- cellScores[match(rownames(ds.ic@meta.data), rownames(cellScores)),
    ]
ds.ic@meta.data <- cbind(ds.ic@meta.data, cellScores)
head(ds.ic@meta.data)

# saveRDS('objects/Hydra_Seurat_IC.rds')
ds.ic <- readRDS("objects/Hydra_Seurat_IC.rds")
```

```
# Plot epithelial metagene scores on full t-SNE and intertitial t-SNE
p <- FeaturePlot(ds.s1, c("wt11", "wt36", "wt40", "wt42", "wt12", "wt28",
    "wt30", "wt38"), do.return = TRUE, cols.use = c("grey", "blue"))
p1 <- FeaturePlot(ds.ic, c("wt11", "wt36", "wt40", "wt42", "wt12", "wt28",
    "wt30", "wt38"), do.return = TRUE, cols.use = c("grey", "blue"))

plot_grid(p[[1]], p[[2]], p[[3]], p[[4]], p1[[1]], p1[[2]], p1[[3]], p1[[4]],
    p[[5]], p[[6]], p[[7]], p[[8]], p1[[5]], p1[[6]], p1[[7]], p1[[8]],
    ncol = 4, labels = "AUTO", label_size = 20, align = "h")
```

Reclustering of cells prior to URD trajectory reconstruction

We recalculate variable genes, PCs and recluster after excluding clusters db1 and db2. PCs and clustering from this analysis are used in URD trajectory reconstruction. We add them to our interstitial cell Seurat object (Hydra_Seurat_IC.rds) as ic.original.pca and clustering2.

```
# remove doublet cluster
ds.ic.s1 <- SubsetData(object = ds.ic, ident.remove = c("db1", "db2"),
    subset.raw = TRUE)

ds.ic.s1 <- FindVariableGenes(object = ds.ic.s1, mean.function = ExpMean,
    dispersion.function = LogVMR, x.low.cutoff = 0.05, x.high.cutoff = 4,
    y.cutoff = 0.5)
ds.ic.s1 <- ScaleData(object = ds.ic.s1)
ds.ic.s1 <- RunPCA(object = ds.ic.s1, pc.genes = ds.ic.s1@var.genes, pcs.compute = 40,
    do.print = TRUE, pcs.print = 1:5, genes.print = 20)
ds.ic.s1 <- ProjectPCA(object = ds.ic.s1)
ds.ic.s1 <- FindClusters(object = ds.ic.s1, reduction.type = "pca", dims.use = 1:31,
    force.recalc = TRUE, resolution = 1.5, print.output = 0)
ds.ic.s1 <- RunTSNE(object = ds.ic.s1, dims.use = c(1:31), do.fast = T,
    perplexity = 40)

# URD object
```

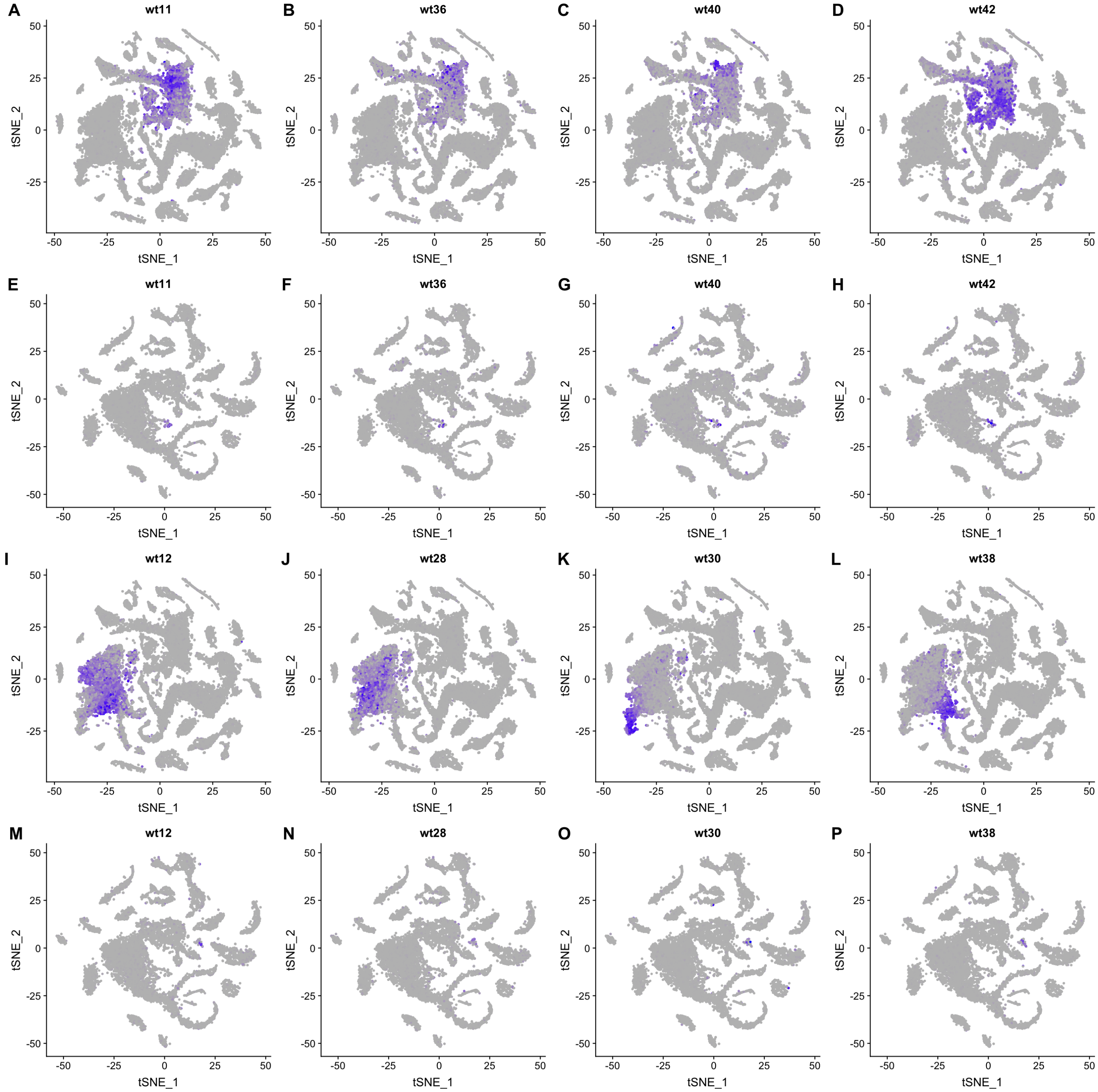

Figure 5: Expression of epithelial metagenes in cells of the interstitial lineage indicate hybrid transcriptomes in cells of clusters db1 and db2. A-D) Expression of four ectodermal gene modules visualized on the whole dataset t-SNE plot. E-H) Expression of ectodermal gene modules plotted on the t-SNE representation for the interstitial cell subset. I-L) Expression of four endodermal gene modules on the whole dataset t-SNE plot. M-P) Expression of endodermal gene modules plotted on the t-SNE representation for the interstitial cell subset.

```r
# We add the updated PCA and the updated clustering to our interstitial
# cell object

# Extract the clustering
ds.ic.s1.meta <- ds.ic.s1@meta.data

# Drop columns that are not needed
ds.ic.s1.meta$nGene <- NULL
ds.ic.s1.meta$nUMI <- NULL
ds.ic.s1.meta$orig.ident <- NULL
ds.ic.s1.meta$percent.mito <- NULL
ds.ic.s1.meta$cluster_numbering <- NULL

ds.ic.s1.meta <- as.data.frame(ds.ic.s1.meta)
# Rename clustering
names(ds.ic.s1.meta) <- c("clustering2")

# add gene id column
ds.ic@meta.data$id <- 1:nrow(ds.ic@meta.data)

# Add the clustering to metadata
ds.ic@meta.data <- merge(ds.ic@meta.data, ds.ic.s1.meta, by = "row.names",
    all.x = TRUE)
# Add row names
rownames(ds.ic@meta.data) <- ds.ic@meta.data$Row.names
ds.ic@meta.data[order(ds.ic@meta.data$id), ]

# Drop gene id column, row.name columns
ds.ic@meta.data$Row.names <- NULL
ds.ic@meta.data$id <- NULL

# Add updated PCA to Hydra_Seurat_IC.rds
ic.original.pca <- ds.ic.s1@dr$pca

ds.ic@dr$ic.original.pca <- ic.original.pca

# saveRDS(ds.ic,'/objects/Hydra_Seurat_IC.rds'))
```

Software versions

This document was computed on Thu Nov 01 15:40:14 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      caTools_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] tidyr_0.8.0          R.utils_2.6.0        stats4_3.4.3        
[124] munsell_0.5.0       
```

References

1. 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).

2. Hemmrich, Georg *et al.*, Molecular signatures of the three stem cell lineages in hydra and the emergence of stem cell function at the base of multicellularity. *Molecular biology and evolution.* **29**, 3267–3280 (2012).

3. Engel, Ulrike *et al.*, Nowa, a novel protein with minicollagen Cys-rich domains, is involved in nematocyst formation in Hydra. *Journal of cell science.* **115**, 3923–3934 (2002).

4. Fedders, Henning, Augustin, René, Bosch, Thomas C G, A Dickkopf- 3-related gene is expressed in differentiating nematocytes in the basal metazoan Hydra. *Wilhelm Roux' Archiv für Entwicklungsmechanik der Organismen.* **214**, 72–80 (2004).

5. 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).

6. 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).

7. 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).

8. R. Augustin *et al.*, Dickkopf related genes are components of the positional value gradient in Hydra. **296**, 62–70 (2006).