Trajectory inference

Ricardo Martins-Ferreira

2023-07-31

Trajectory analysis of microglia activation from homeostatic to DAM

```
libs <- c("Seurat", "tidyverse", "monocle3", "SeuratWrappers",
          "pheatmap","circlize", "data.table")
suppressMessages(
  suppressWarnings(sapply(libs, require, character.only = TRUE))
)
```

```
##         Seurat      tidyverse       monocle3 SeuratWrappers       pheatmap
##           TRUE           TRUE           TRUE           TRUE           TRUE
##       circlize     data.table
##           TRUE           TRUE
```

Upload and process HuMicA Seurat object

The Seurat object consists of the 64,438 nuclei already clustered and annotated for clinicopathological features of the subjects. This object is divided in 15 clusters. The active assay is the normalized and scaled "RNA" assay.

```
DefaultAssay(Seurat)<- "RNA"
Seurat <- NormalizeData(Seurat)
Seurat <- ScaleData(Seurat); gc()
```

The HuMicA object is then subsetted to include only the main homeostatic cluster (Homeos1, cluster 0), the two pre-activation homeostatic clusters (Homeos2, cluster 10; Homeos4, cluster 11) and DAM populations (Intermediate.DAM, cluster 4; Final.DAM, cluster 1).

```
subset_seurat <- subset(Seurat, idents=c("0","1","4","10","11"))
```

Convert Seurat to cell_data_set object

```
cds <- as.cell_data_set(subset_seurat)
```

Monocle3 workflow

```
# to get cell metadata
colData(cds)

# to see gene metadata
fData(cds)
rownames(fData(cds))[1:10]
```

```
# Add the gene_short_name column to metadata
fData(cds)$gene_short_name <- rownames(fData(cds))

# ...2. Cluster cells (using clustering info from seurat's UMAP)
# let's use the clustering information have

# assign paritions (only 1 particion because all cells are microglia)
reacreate.partition <- c(rep(1,length(cds@colData@rownames)))
names(reacreate.partition) <- cds@colData@rownames
reacreate.partition <- as.factor(reacreate.partition)

cds@clusters$UMAP$partitions <- reacreate.partition

# Assign the cluster info

list_cluster <- subset_seurat@active.ident
cds@clusters$UMAP$clusters <- list_cluster

# Assign UMAP coordinate - cell embeddings

cds@int_colData@listData$reducedDims$UMAP<-subset_seurat@reductions$umap@cell.embeddings

# ...3. Learn trajectory graph
cds <- learn_graph(cds, use_partition = FALSE)
# the

plot_cells(cds,
           # color_cells_by = 'redefined_cluster',
           label_groups_by_cluster = FALSE,
           label_branch_points = FALSE,
           label_roots = FALSE,
           label_leaves = FALSE,
           group_label_size = 5)

# ...4. Order the cells in pseudotime
## the cells from Homeos1 were used as starting point

cds <- order_cells(cds, reduction_method = 'UMAP')

plot_cells(cds,
           color_cells_by = 'pseudotime',show_trajectory_graph = F,
           label_groups_by_cluster = FALSE,
           label_branch_points = FALSE,
           label_roots = FALSE,
           label_leaves = FALSE)

# cells ordered by monocle3 pseudotime (boxplot)
pseudotime(cds)
cds$monocle3_pseudotime <- pseudotime(cds)
data.pseudo <- as.data.frame(colData(cds))

ggplot(data.pseudo,
       aes(monocle3_pseudotime, reorder(seurat_clusters, monocle3_pseudotime, median),
```

```
           fill = seurat_clusters)) +
  geom_boxplot()+ theme_pubr()

# ...5. Finding genes that change as a function of pseudotime

## Calculate size factors using built-in function in monocle3
cds <- estimate_size_factors(cds)

## Add gene names into CDS
cds@rowRanges@elementMetadata@listData[["gene_short_name"]]<-rownames(subset_seurat[["RNA"]])

## Calculation of the differential expression across the trajectory
deg <- graph_test(cds, neighbor_graph = 'principal_graph', cores = 8)


# isolate significant degs
deg %>%
  arrange(q_value) %>%
  filter(status == 'OK') -> deg_ok

deg_ok$rank <- 1:length(deg_ok$status)


# heatmap with top 150 degs
pt.matrix <-as.matrix(cds@assays@data$counts[match(rownames(deg_ok[1:150,]),
                                                   rowData(cds)[,1]),
                                             order(pseudotime(cds))])

pt.matrix.1 <- t(apply(pt.matrix,1,function(x){smooth.spline(x,df=3)$y}))
pt.matrix.1 <- t(apply(pt.matrix.1,1,function(x){(x-mean(x))/sd(x)}))

cells <- colnames(pt.matrix)
colnames(pt.matrix.1) <- cells

subset_seurat@meta.data %>%
  dplyr::select(integrated_snn_res.0.25) %>%
  as.matrix() -> annotation

annotation[match(colnames(pt.matrix.1), rownames(annotation)),] %>%
  as.matrix() -> annotation



anno_colors <- list(V1 = c("0"="#E6AB02","1" = "#FC8D62", "4" = "#E5D8BD",
                           "10" = "#344D67", "11" = "#B3CDE3"))



pheatmap(as.matrix(pt.matrix.1),cluster_rows=TRUE,
         show_rownames=T, show_colnames = FALSE,
         cluster_cols=FALSE, scale = 'row',
         clustering_method = 'average', fontsize = 5,
         clustering_distance_rows = 'euclidean',
```

```
         annotation_col = as.data.frame(annotation),
         annotation_colors = c(anno_colors),
         #annotation_row = anno_rows,
         use_raster = F)

#######################
# Plot individual genes

genes <- c("P2RY12","APOE","SPP1")
cds_subset <- cds[rowData(cds)$gene_short_name %in% genes]

monocle3::plot_genes_in_pseudotime(cds_subset,panel_order = c("P2RY12","APOE","SPP1"),
                                   color_cells_by="seurat_clusters",
                                   min_expr=0.5)+
  scale_color_manual(breaks = c("0", "1", "4","10","11"),
                     values=c("#E6AB02", "#FC8D62",  "#E5D8BD", "#344D67", "#B3CDE3"))
```