MTGOsc: MTGO analysis on single cells

Nelson Nazzicari, Simone Marini, Danila Vella

2019-03-08

Contents

Example data: bladder basal epithelial cells

MTGOsc comes with some example data from the Mouse Atlas in the form of a (simplified for storage reasons) Seurat object. Let's start with loading Seurat and MTGO:

```
library(Seurat)
library(MTGOsc)
```

Loading MTGOsc also loads three objects: *bladder*, *markers*, and *mouse.pathways*.

```
#this is a (simplified) Seurat object
print(bladder)
```

```
## An object of class seurat in project Bladder_rm.batch_dge.txt
##  269 genes across 1028 samples.
```

```
#this come from applying Seurat function FindAllMarkers() to the bladder dataset
head(markers)
```

```
##                p_val avg_logFC pct.1 pct.2     p_val_adj
## krt15  1.523860e-202  1.959457 0.994 0.269 2.070317e-198
## igfbp2 3.880320e-196  1.857702 1.000 0.322 5.271803e-192
## trf    2.219344e-186  1.747788 0.896 0.189 3.015200e-182
## krt5   1.329655e-180  1.730088 0.976 0.276 1.806470e-176
## gsdmc2 1.235398e-172  1.711701 0.908 0.208 1.678412e-168
## gsto1  2.506894e-172  1.629429 0.985 0.299 3.405866e-168
##                                cluster   gene
## krt15  Basal epithelial cell(Bladder)  krt15
## igfbp2 Basal epithelial cell(Bladder) igfbp2
## trf    Basal epithelial cell(Bladder)    trf
## krt5   Basal epithelial cell(Bladder)   krt5
## gsdmc2 Basal epithelial cell(Bladder) gsdmc2
## gsto1  Basal epithelial cell(Bladder)  gsto1
```

```
#this is a simple gene -> pathway dictionary
head(mouse.pathways)
```

```
##    gene                                  pathway
## 1 43160                               Metabolism
## 2 43160                    Biological oxidations
## 3 43160 Phase I - Functionalization of compounds
## 4 43161                               Metabolism
## 5 43161                    Biological oxidations
```

```
## 6 43161 Phase I - Functionalization of compounds
```

Bladder object contains three clusters:

```
table(bladder@ident)
```

```
##
## Basal epithelial cell(Bladder) Stromal cell_Dpt high(Bladder)
##                            327                             651
##         Umbrella cell(Bladder)
##                             50
```

Calling MTGOsc

For this tutorial we'll focus on Basal epitelial cells cluster, which is of intermediate size. We create two "selector" arrays, one for cells and one for genes.

```
cells.selected = bladder@ident == 'Basal epithelial cell(Bladder)'
genes.selected = subset(markers, cluster == 'Basal epithelial cell(Bladder)')$gene

my.bladder = bladder
my.bladder@data = my.bladder@data[genes.selected, cells.selected]
```

MTGOsc needs a folder where to save temporary files and final results.

```
root = tempdir() #change this to your preferred local path
dir.create(root, recursive = TRUE, showWarnings = FALSE)
```

We are now ready to start with MTGOsc:

```
#building a genes-pathways dictionary
dict = write.dictionary(genes=mouse.pathways$gene,
                        terms = mouse.pathways$pathway, outfolder = root)

#computign gene coexpression (default function is 'cor')
coexp = write.coexpressionMatrix(geneExpression = my.bladder, outfolder = root)

#thinning coexpression network via scale criterion
edges = write.edges(coexpression = coexp, outfolder = root,
                    keep.weights = FALSE, fun = scale_free_threshold)
```

```
## gamma: 2.05513085766315 (target:2)
## threshold: 0.3
## network edges: 33
```

```
#writing a parameter file, useful for MGTO
write.paramFile(outfolder = root)

#actual call to MTGO
call.MTGO(outfolder = root, verbose = TRUE)
```

```
## ======> MTGO executed with the following OUTPUT:
##  /tmp/RtmpGI1Rpm/
## Q: -0.09366391184573018
## QGO: 0.0
## Cluster Mean Density: 0.05000000000000001
## Q: 0.46235078053259737
## QGO: 0.85
```

```
## Cluster Mean Density: 0.20039682539682538
## Q: 0.45224977043158743
## QGO: 0.9
## Cluster Mean Density: 0.18115079365079367
## Q: 0.4412304866850309
## QGO: 0.9
## Cluster Mean Density: 0.1773989898989899
## Q: 0.4531680440771338
## QGO: 0.9
## Cluster Mean Density: 0.1875
## Q: 0.4522497704315873
## QGO: 0.9
## Cluster Mean Density: 0.17876984126984125
## Q: 0.4522497704315873
## QGO: 0.9
## Cluster Mean Density: 0.17876984126984125
##
## ======> MTGO executed with the following ERRORS:
```

```
#building and saving representation of resulting network
network.collapsed = export.network.modules(infolder = root, collapse.modules = TRUE)
network.full = export.network.modules(infolder = root, collapse.modules = FALSE)
```

At this point networks are saved on disk, in "Network" subfolder, in the form of two html files. In the first one (ClusterNetwork.html) each functional module is collapsed to a single node, the second one (FullNetwork.html) where each gene is represented and functional modules are color coded.

Gene module enrichment on Reactome

Here we look for Reactome pathway enrichment of the genes constituting the thinned network. This procedure is complementary to the exctraction of Reactome pathways by MTGO-SC.

```
# load libraries for gene enrichment on Reactome (those are on Bioconductor, not CRAN)
library(ReactomePA)
library(clusterProfiler)
library(org.Mm.eg.db)

#a support function to take care of gene upper/lower case convention
firstup = function(x) {
  x = tolower(x)
  substr(x, 1, 1) = toupper(substr(x, 1, 1))
  return(x)
}

#the list of all genes involved in the cluster
genes = unique(c(as.character(edges$gene1), as.character(edges$gene2)))

#correct casing of gene names
genes = firstup(genes)

#translating gene names to ENTREZID via org.Mm.eg.db database
genes = bitr(genes, fromType="SYMBOL", toType="ENTREZID", OrgDb="org.Mm.eg.db")
```

```
## Warning in bitr(genes, fromType = "SYMBOL", toType = "ENTREZID", OrgDb =
## "org.Mm.eg.db"): 9.09% of input gene IDs are fail to map...
```

```
#the actual enrichment
enriched = enrichPathway(gene=genes$ENTREZID, pvalueCutoff=0.05,
                         readable=T, organism = "mouse", pAdjustMethod = "BH")
```