RSCORE_Tutorials

This is an example of RSCORE. Data comes from NCBI GEO with accession GSE81861. After you download the raw data, we have to do pretreatment. For your data, you can do it by yourself and finally provide a Seurat class object, or you can provide a clean matrix data and use our mat2seurat function.

```
# change the directory to yours
# download.file('https://www.ncbi.nlm.nih.gov/geo/download/?
acc=GSE81861&format=file&file=GSE81861%5FCell%5FLine%5FFPKM%2Ecsv%2Egz', destfile =
'RCA_FPKM.csv.gz')
# gunzip('RCA_FPKM.csv.gz', 'RCA_FPKM.csv')
RCA_count <- read.csv('RCA_FPKM.csv', header=T, row.names = 1)
RCA_count <- log(RCA_count+1, 2)
row_names <- strsplit(row.names(RCA_count), '_')
gene_names <- c()
for (i in 1:length(row_names)){
  temp = unlist(row_names[i])
  gene_names[i] = temp[2]
}

row.names(RCA_count) <- make.names(gene_names, unique=TRUE)

RCA_seurat <- CreateSeuratObject(counts = RCA_count, min.cells = 10, min.features = 1000,
                                 names.field = 3, names.delim = '_',
                                 assay = 'RNA', project = 'RCA')
```

We also suggest doing normalization and feature selection. Although we have given some default parameters, it depends on your data specifically.

```
RCA_seurat <- ScaleData(object = RCA_seurat)
RCA_seurat <- FindVariableFeatures(object = RCA_seurat, selection.method = 'vst', nfeatures =
8000)
```

PPI data is necessary. You can provide the adjacent matrix of PPI network by yourself,

```
# change the directory to yours
hs_network <- as.matrix(readRDS(system.file('extdata','hs_network_matrix_Biogrid-
3.5.173.Rda',package = 'RSCORE')))
```

and then the parameter ‘PPI’ is just the matrix.

```
RCA_seurat <- R.SCORE(Data = RCA_seurat, PPI = hs_network)
#> module num:  1435
```

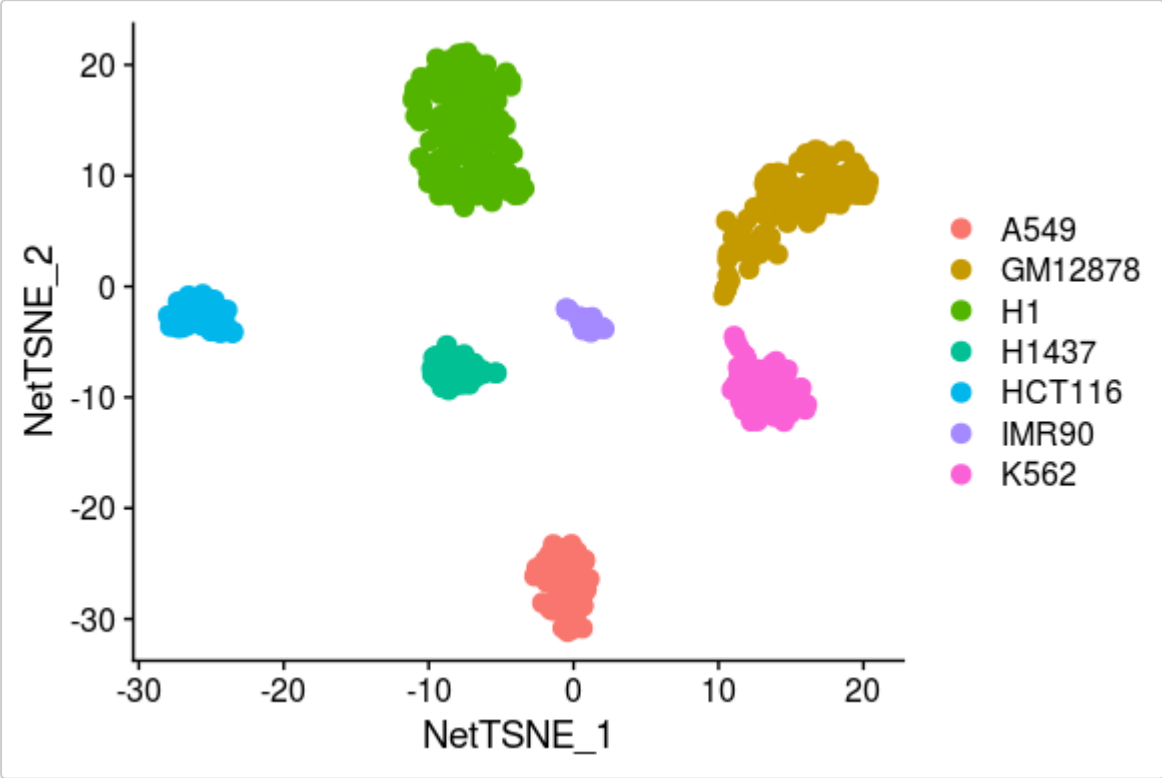
Or you can get it by our functions. Then you have to set the parameter ‘PPI’ as ‘String’ or ‘Biogrid’. This means we will download PPI data from STRING or BioGRID (It will cost some time, depends on your Internet speed). Both of these two choices should give the species (default is 9606, Homosapiens).

```
# RCA_seurat <- R.SCORE(Data = RCA_seurat, PPI = 'String', species = 9606)
# or
# RCA_seurat <- R.SCORE(Data = RCA_seurat, PPI = 'Biogrid', species = 9606)
```

The result is saved in ‘Net’ assay of RCA_seurat (it has been set as default assay). You can plot the tsne

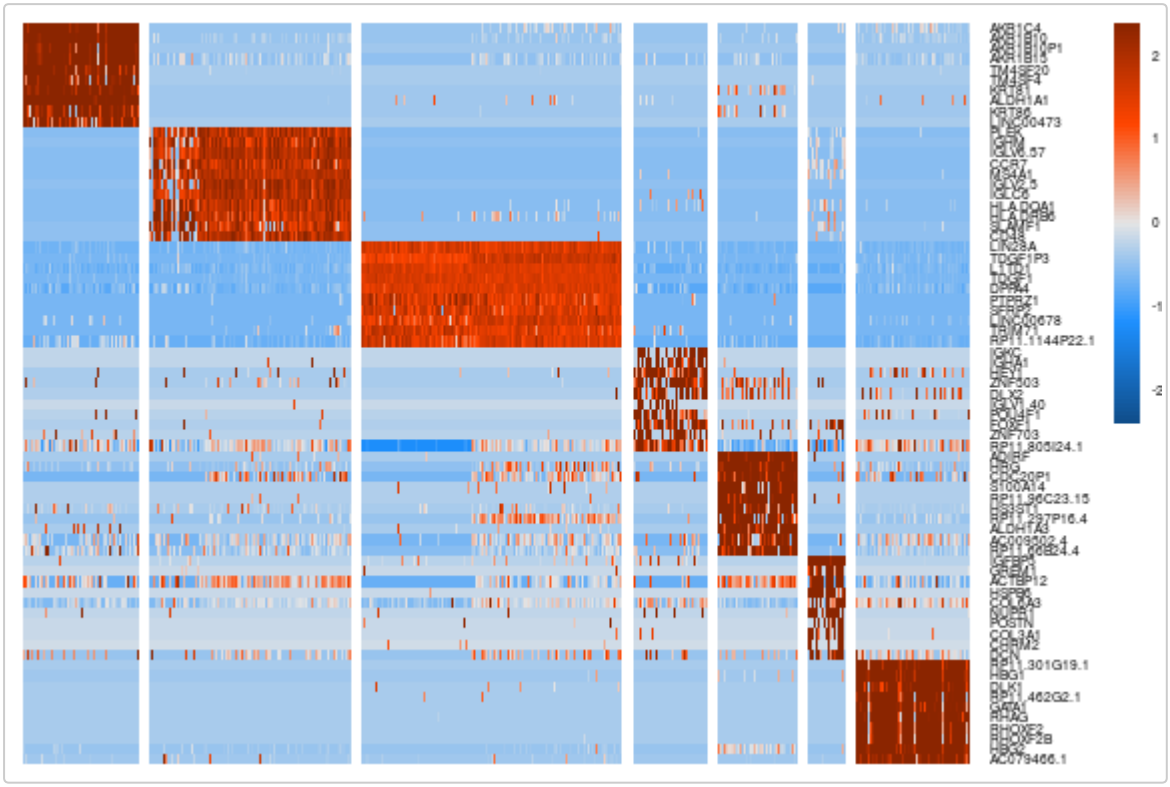
```
VariableFeatures(RCA_seurat) <- rownames(RCA_seurat)
RCA_seurat <- RunPCA(RCA_seurat, features = rownames(RCA_seurat), npcs = 30, reduction.name =
```

```
"NetPCA",
                      reduction.key = "NetPCA_", verbose = F)
RCA_seurat <- RunTSNE(RCA_seurat, reduction = "NetPCA", dims = 1:10,
                        reduction.name = "NetTSNE",  reduction.key = "NetTSNE_")
DimPlot(RCA_seurat, reduction = 'NetTSNE', pt.size = 3, group.by = 'orig.ident')
```

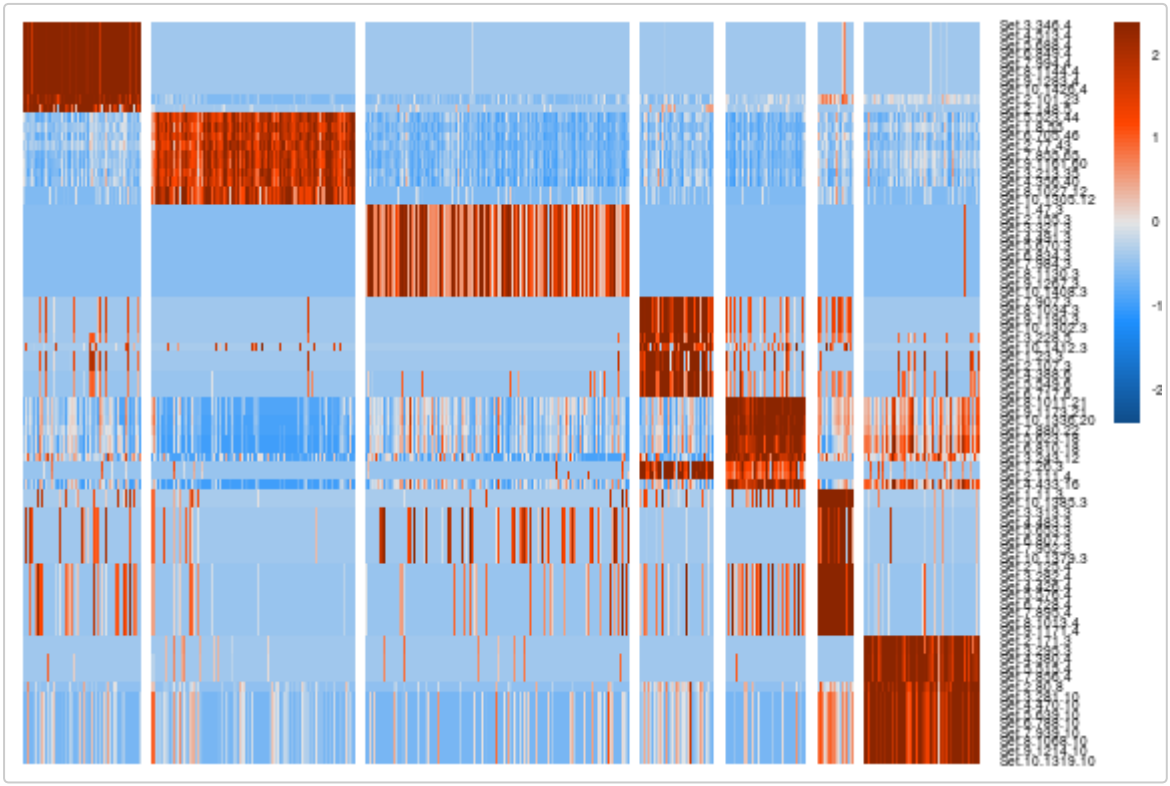



```
library(dplyr)
#>
#> Attaching package: 'dplyr'
#> The following objects are masked from 'package:igraph':
#>
#>     as_data_frame, groups, union
#> The following objects are masked from 'package:stats':
#>
#>     filter, lag
#> The following objects are masked from 'package:base':
#>
#>     intersect, setdiff, setequal, union
library(genesorteR)
#> Loading required package: Matrix
SCORE_DEGs_list <- Find_Markers(object = RCA_seurat, assay = 'RNA', FoldChange = 1.5)
#> Warning in sortGenes(expr_mtx, Idents(object), binarizeMethod =
#> binarizeMethod, : A Friendly Warning: Some genes were removed because
#> they were zeros in all cells after binarization. You probably don't need
#> to do anything but you might want to look into this. Maybe you forgot to
#> pre-filter the genes? You can also use a different binarization method.
#> Excluded genes are available in the output under '$removed'.

SCORE_DAMs_list <- Find_Markers(object = RCA_seurat, assay = 'Net', FoldChange = 1.5)

#Select the top n markers of each cluster
top10_DEGs <- SCORE_DEGs_list$Markers %>% group_by(Cluster) %>% top_n(n = 10, wt = Gene.Score)
top10_DAMs <- SCORE_DAMs_list$Markers %>% group_by(Cluster) %>% top_n(n = 10, wt = Gene.Score)


#genesorteR plotMarkerHeat function
plotMarkerHeat(exp = SCORE_DEGs_list$GeneSort$inputMat,
               classes = SCORE_DEGs_list$GeneSort$inputClass,
```

```
                markers = top10_DEGs$Marker,
                clusterGenes = FALSE,
                averageCells = 1)
```



```
plotMarkerHeat(exp = SCORE_DAMs_list$GeneSort$inputMat,
                classes = SCORE_DAMs_list$GeneSort$inputClass,
                markers = top10_DAMs$Marker,
                clusterGenes = FALSE,
                averageCells = 1)
```



You can also show steiner tree of given cluster

```
PlotSteinertree(RCA_seurat, ident = 'A549')
#> Warning in sortGenes(expr_mtx, Idents(object), binarizeMethod =
```

```
#> binarizeMethod, : A Friendly Warning: Some genes were removed because
#> they were zeros in all cells after binarization. You probably don't need
#> to do anything but you might want to look into this. Maybe you forgot to
#> pre-filter the genes? You can also use a different binarization method.
#> Excluded genes are available in the output under '$removed'.
#> calculate tree
```