Jaccard_Analysis

Ammar Sabir Cheema

14/05/2020

Loading libraries

```
library(pheatmap)
library(RColorBrewer)


aa <- list.files(path = "/home/cheemaa/jac_analysis/", pattern = "conta", all.files = TRUE,
            full.names = FALSE, recursive = FALSE,
            ignore.case = FALSE, include.dirs = FALSE)
 print(aa)
```

```
## [1] "cluster0.markers_wo_M1_wo_conta.txt" "cluster1.markers_wo_M1_wo_conta.txt"
## [3] "cluster2.markers_wo_M1_wo_conta.txt" "cluster3.markers_wo_M1_wo_conta.txt"
## [5] "cluster4.markers_wo_M1_wo_conta.txt" "cluster5.markers_wo_M1_wo_conta.txt"
## [7] "cluster6.markers_wo_M1_wo_conta.txt"
```

```
 dataset_GG <- lapply(aa, read.table)
 names(dataset_GG) <- list.files(path="/home/cheemaa/jac_analysis/", pattern = "conta", full.names=FALSE)

 names(dataset_GG) <- sub("cluster", "C", names(dataset_GG))
 names(dataset_GG) <- sub("markers_wo_M1_wo_conta.txt", "GG", names(dataset_GG))
 names(dataset_GG)
```

```
## [1] "C0.GG" "C1.GG" "C2.GG" "C3.GG" "C4.GG" "C5.GG" "C6.GG"
```

```
# I only keep the gene names.
 for (i in 1:length(dataset_GG)) {
   dataset_GG[[i]] <- rownames(dataset_GG[[i]])
}

 bb <- list.files(path = "/home/cheemaa/jac_analysis/", pattern = "only_Irf1_FF", all.files = TRUE,
            full.names = FALSE, recursive = FALSE,
            ignore.case = FALSE, include.dirs = FALSE)


 print(bb)
```

```
## [1] "cluster0.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [2] "cluster1.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [3] "cluster2.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [4] "cluster3.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [5] "cluster4.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [6] "cluster5.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
## [7] "cluster6.markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt"
```

```
 dataset_GF <- lapply(bb, read.table)
 names(dataset_GF) <- list.files(path="/home/cheemaa/jac_analysis/", pattern = "only_Irf1", full.names=FALSE)
 names(dataset_GF) <- sub("cluster", "C", names(dataset_GF))
 names(dataset_GF) <- sub("markers_only_Irf1_FF_only_Ikbkb_FF_res_2.txt", "GF", names(dataset_GF))
 names(dataset_GF)
```

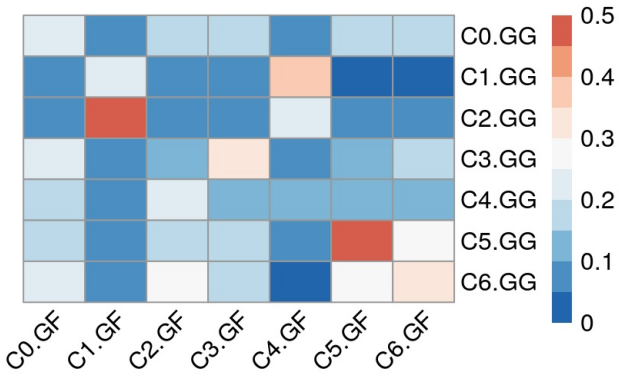
```
## [1] "C0.GF" "C1.GF" "C2.GF" "C3.GF" "C4.GF" "C5.GF" "C6.GF"
```

```
 # I only keep the gene names.
for (i in 1:length(dataset_GF)) {
  dataset_GF[[i]] <- rownames(dataset_GF[[i]])
}


# Compute the size of intersection between list of genes of pairs of different clusters of the tissues
# and compute the Jaccard index for each pair of clusters. Store both information in two different dataframes
jaccard_index_df = data.frame()
intersection_df = data.frame()
for(i in 1:length(dataset_GG)) {
  jaccard_index_set = vector()
  details_set = vector()
  for(j in 1:length(dataset_GF)) {
    jaccard_index = length( intersect( dataset_GG[[i]], dataset_GF[[j]])) / length( unique( c( dataset_GG[[i]], d
ataset_GF[[j]])))
    jaccard_index_set = append( jaccard_index_set, jaccard_index)
    details_set = append( details_set, length( intersect( dataset_GG[[i]], dataset_GF[[j]])))
  }
  jaccard_index_df = rbind( jaccard_index_df, jaccard_index_set)
  intersection_df = rbind( intersection_df, details_set)
}
names( jaccard_index_df) = names(dataset_GF)
row.names( jaccard_index_df) = names(dataset_GG)
names( intersection_df) = names(dataset_GF)
row.names( intersection_df) = names(dataset_GG)


# Display the data.frame of jaccard indexes as a heatmap
break_list = seq(0, 0.5, by = 0.05)
#pheatmap::pheatmap(
#  jaccard_index_df,
#  color = colorRampPalette(brewer.pal(n = 9, name = "Blues"))(length( break_list)),
#  cellwidth =40, cellheight =30,
#  fontsize = 12,
#  show_rownames = T, show_colnames = T,
#  angle_col = 45,
#  breaks = break_list,
#  display_numbers = matrix( paste0( as.matrix( intersection_df), " (", 100*signif( as.matrix( jaccard_index_df),
2), "%)"), ncol = length(dataset_GF)),
#  cluster_rows = FALSE, cluster_cols = FALSE,
#  treeheight_row = 0, treeheight_col=0
#)


pheatmap(
  jaccard_index_df,
  cellwidth =30, cellheight =20,
  color = colorRampPalette(rev(brewer.pal(n = 7, name =
  "RdBu")))(length( break_list)),
  fontsize = 12,
  show_rownames = T, show_colnames = T,
  angle_col = 45,
  breaks = break_list,
  cluster_rows = FALSE, cluster_cols = FALSE,
  treeheight_row = 0, treeheight_col=0,legend=T,cutree_rows = 4
)
```


C0.GG
C1.GG
C2.GG
C3.GG
C4.GG
C5.GG
C6.GG
C0.GF
C1.GF
C2.GF
C3.GF
C4.GF
C5.GF
C6.GF
0.5
0.4
0.3
0.2
0.1
0