

Scoring cells CLEAN

Abinet Joan

2023-03-14 11:10:56 +0100

This code is supposed to be a cleaner version of “scoring_cells.rmd”

```
suppressMessages(library(dplyr))
suppressMessages(library(patchwork))
suppressMessages(library(ggplot2))
suppressMessages(library(DESeq2))
suppressMessages(library(readxl))
suppressMessages(library("pheatmap"))
suppressMessages(library("RColorBrewer"))
suppressMessages(library(singscore))
suppressMessages(library(GSEABase))
```

Loading transptomic spatial data and generating deseq2 object

```
## converting counts to integer mode

## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in
## design formula are characters, converting to factors

## the design formula contains one or more numeric variables with integer values,
## specifying a model with increasing fold change for higher values.
## did you mean for this to be a factor? if so, first convert
## this variable to a factor using the factor() function
```

load function from GetSignatureScore

```
source("GetSignatureScore.R")
```

function to display score in a barplot

```
visualise_score <- function(features.scores.use) {

  data_plot <- cbind(colData(dds_Q3),as.vector(features.scores.use))
  data_plot <- data_plot[order(data_plot$Position),]

  barplot(data_plot$`as.vector(features.scores.use)` ,
          col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
          ,rep("blue",8), rep("pink",2)),
```

```

names.arg =
c("1_1", "2_1", "1_2", "2_2", "1_6", "1_9", "2_3", "2_8", "1_3", "1_5", "1_8", "2_12",
  "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7",
  "1_12", "2_9"),

cex.names = 0.3

)
}

```

Get score Motro

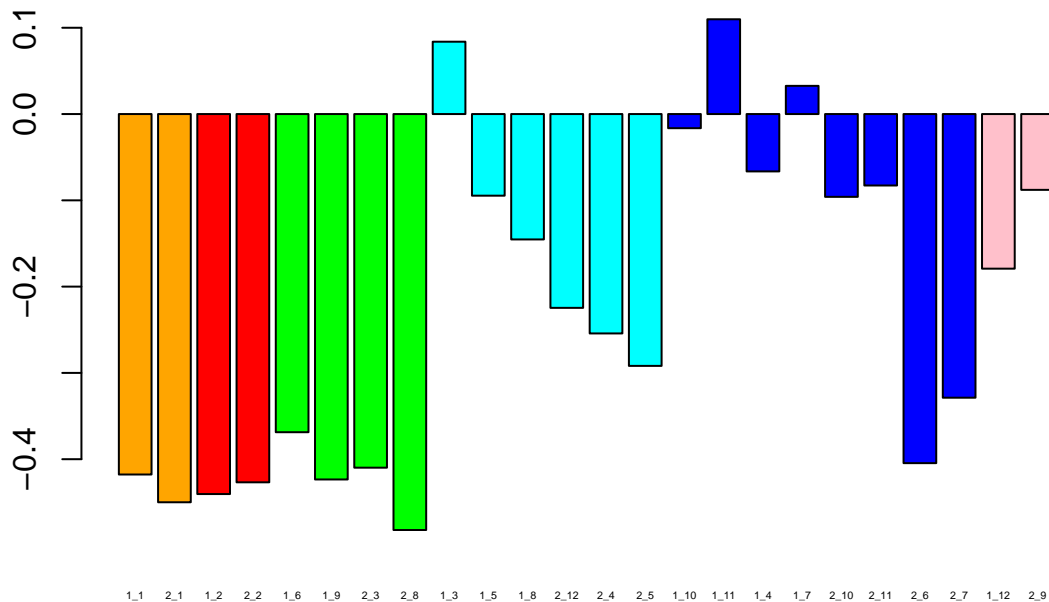
```

Motro_DE <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Project_cecilia/4G0_Scenic/signature_motro_DE.csv")
Motro_DE <- Motro_DE[order(Motro_DE$avg_log2FC, decreasing = T),]
Top20_Motro <- Motro_DE$X[1:20]

features.scores.use <- GetSignatureScore(dds_Q3, Top20_Motro)
new_row <- c("Motro", features.scores.use)

visualise_score(features.scores.use)

```



Dataframe

```
cell_score <- data.frame(new_row)
```

get score with singscore

```
Norm_exprMat <- assay(rlogQ3)
```

```

rankData <- rankGenes(Norm_exprMat)

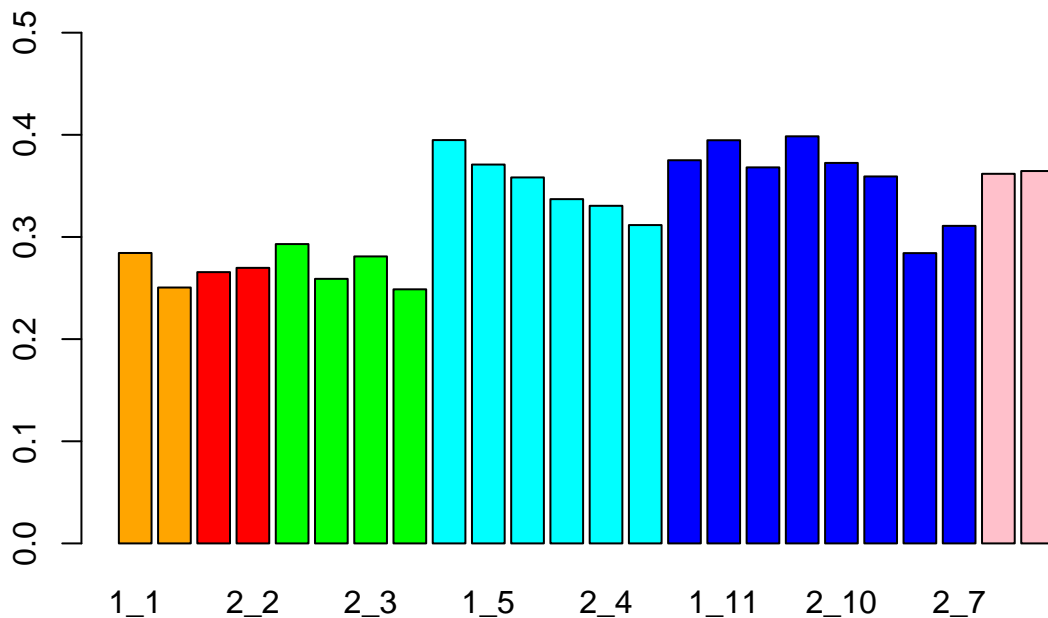
scoredf <- simpleScore(rankData, upSet = Top20_Motro)

scoredf <- cbind(colData(rlogQ3)$Position,scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]

barplot(scoredf$TotalScore,
        col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
        ,rep("blue",8), rep("pink",2)),

        names.arg =
        c("1_1","2_1","1_2","2_2","1_6","1_9","2_3","2_8","1_3","1_5","1_8","2_12",
        "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
        "2_9"), ylim = c(0,0.5)
)

```



```

Motro_DE <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Project_cecilia/4G0_Scenic/signature_motro_DE.csv")
Motro_DE <- Motro_DE[order(Motro_DE$avg_log2FC, decreasing = T),]
Top20_Motro <- Motro_DE$X[1:20]

Norm_exprMat <- counts(dds_Q3, normalized=T)

rankData <- rankGenes(Norm_exprMat)

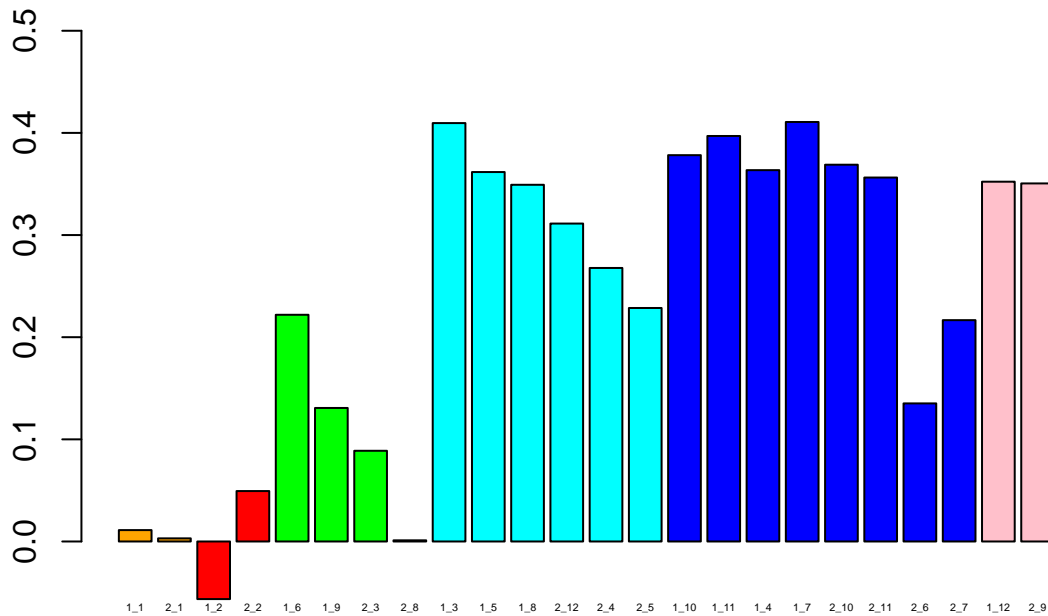
scoredf <- simpleScore(rankData, upSet = Top20_Motro)
new_row <- c("Motro", scoredf$TotalScore)

scoredf <- cbind(colData(rlogQ3)$Position,scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]

barplot(scoredf$TotalScore,
        col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
        ,rep("blue",8), rep("pink",2)),

```

```
names.arg =
c("1_1", "2_1", "1_2", "2_2", "1_6", "1_9", "2_3", "2_8", "1_3", "1_5", "1_8", "2_12",
"2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
"2_9"), ylim = c(0, 0.5),
cex.names = 0.3
)
```



Dataframe singscore

```
cell_singscore <- data.frame(new_row)
```

Get score AM

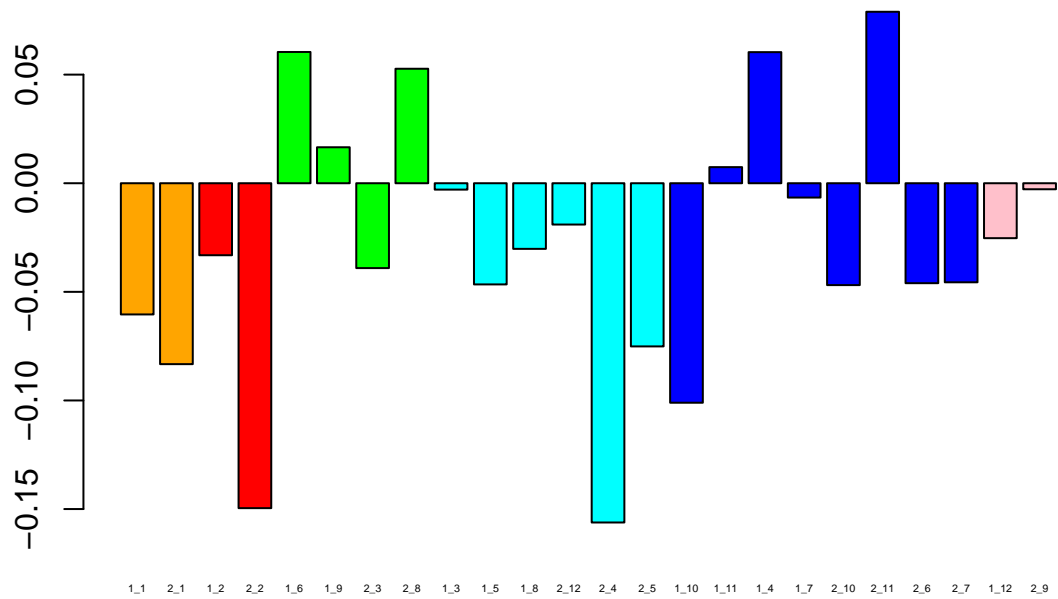
```
myeloid.markers <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Cécilia_project_2/1.2Visualise_feature/myeloids

myeloid.markers_AM <- myeloid.markers[myeloid.markers$cluster == 5,]$gene

# Ear2 and Kcnq1ot1 are not in the expression data
myeloid.markers_AM <- myeloid.markers_AM[-c(7,13)]

features.scores.use <- GetSignatureScore(dds_Q3, myeloid.markers_AM)
new_row <- c("AM", features.scores.use)
#cell_score <- cbind(cell_score, new_row)

visualise_score(features.scores.use)
```



```

Norm_exprMat <- counts(dds_Q3, normalized=T)
#Norm_exprMat <- assay(rlogQ3)

rankData <- rankGenes(Norm_exprMat)

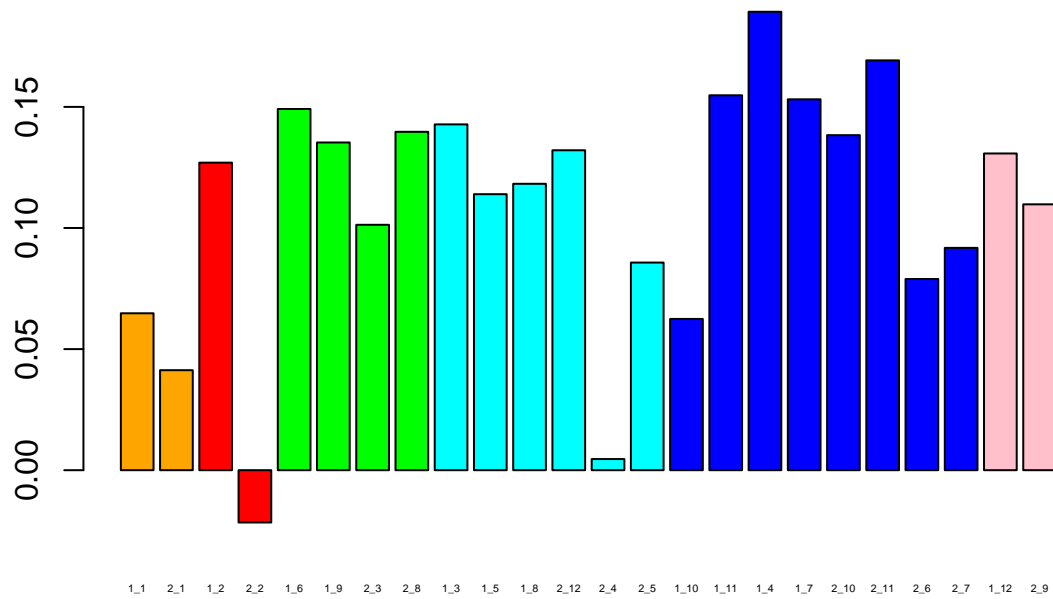
scoredf <- simpleScore(rankData, upSet = myeloid.markers_AM)
new_row <- c("AM", scoredf$TotalScore)
cell_singscore <- cbind(cell_singscore, new_row)

scoredf <- cbind(colData(rlogQ3)$Position, scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]

barplot(scoredf$TotalScore,
        col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
        ,rep("blue",8), rep("pink",2)),

        names.arg =
        c("1_1", "2_1", "1_2", "2_2", "1_6", "1_9", "2_3", "2_8", "1_3", "1_5", "1_8", "2_12",
        "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
        "2_9"), cex.names = 0.3
)

```



score neutro

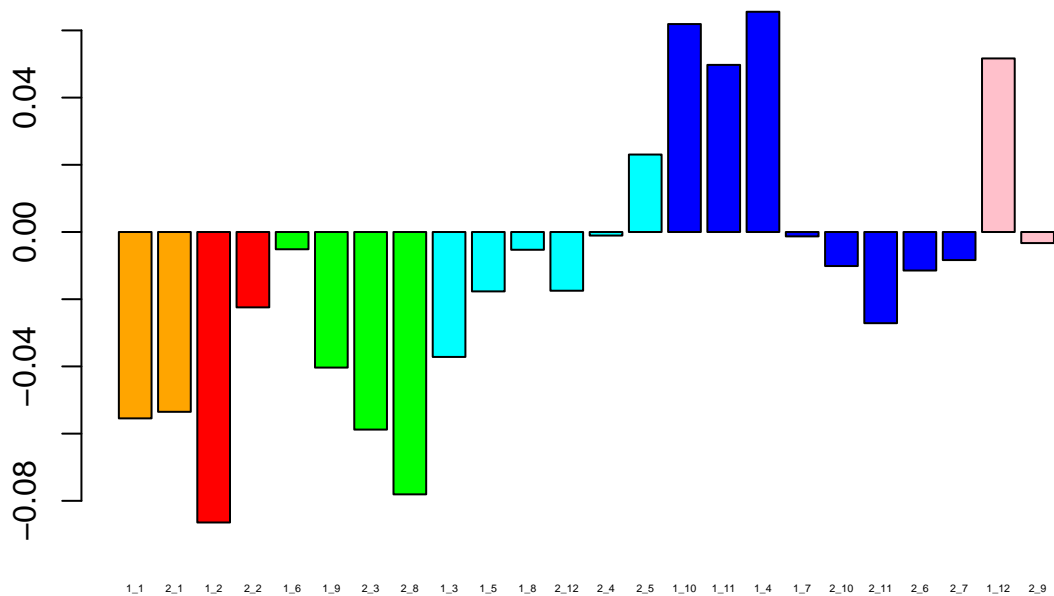
```
myeloid.markers <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Cécilia_project_2/1.2Visualise_feature/myeloids,

myeloid.markers_neut <- myeloid.markers[myeloid.markers$cluster == 6,]$gene

# Stfa2l1 and Cstdc4 are not in the expression data
myeloid.markers_neut <- myeloid.markers_neut[-c(5,8)]

features.scores.use <- GetSignatureScore(dds_Q3, myeloid.markers_neut)
new_row <- c("Neutro", features.scores.use)
cell_score <- cbind(cell_score, new_row)

visualise_score(features.scores.use)
```



```
which(rownames(Norm_exprMat) == "Slc7a11" )
```

```
## [1] 16071
```

```
Norm_exprMat <- counts(dds_Q3, normalized=T)
#Norm_exprMat <- assay(rlogQ3)
```

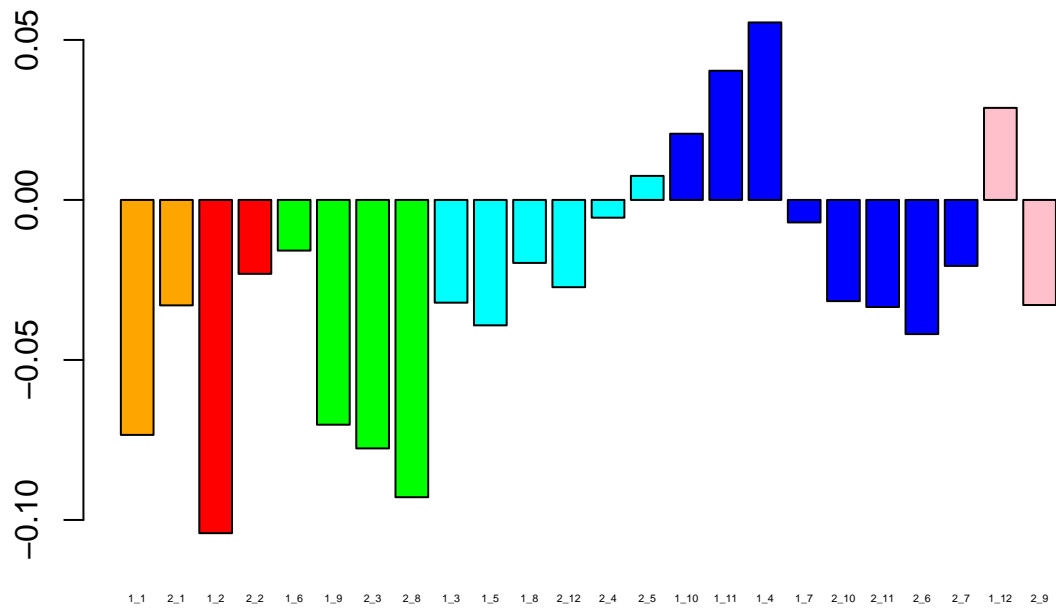
```
rankData <- rankGenes(Norm_exprMat)
```

```
scoredf <- simpleScore(rankData, upSet = myeloid.markers_neut)
new_row <- c("Neutro", scoredf$TotalScore)
cell_singscore <- cbind(cell_singscore, new_row)
```

```
scoredf <- cbind(colData(rlogQ3)$Position, scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]
```

```
barplot(scoredf$TotalScore,
  col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
  ,rep("blue",8), rep("pink",2)),

  names.arg =
  c("1_1","2_1","1_2","2_2","1_6","1_9","2_3","2_8","1_3","1_5","1_8","2_12",
  "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
  "2_9"), cex.names = 0.3
)
```



score cd64

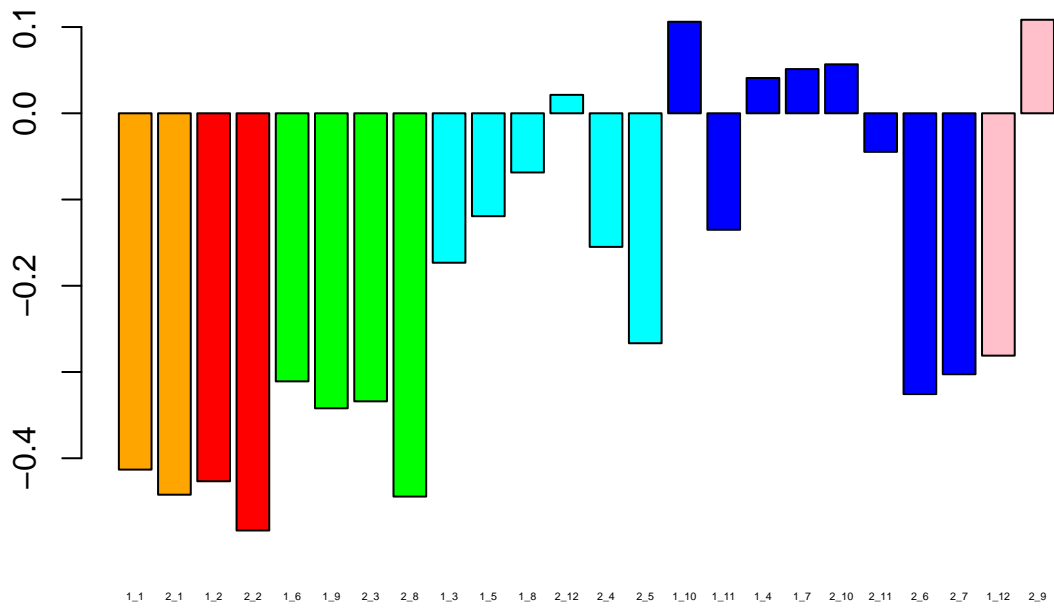
```
myeloid.markers <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Cécilia_project_2/1.2Visualise_feature/myeloids,

myeloid.markers_cd64 <- myeloid.markers[myeloid.markers$cluster == 4,]$gene

# Ly6c2 and Ifi204 are not in the expression data
myeloid.markers_cd64 <- myeloid.markers_cd64[-c(2,12)]

features.scores.use <- GetSignatureScore(dds_Q3, myeloid.markers_cd64)
new_row <- c("cd64", features.scores.use)
cell_score <- cbind(cell_score, new_row)

visualise_score(features.scores.use)
```

```
which(rownames(Norm_exprMat) == "Ms4a6d" )
```

```
## [1] 10345
```

```
Norm_exprMat <- counts(dds_Q3, normalized=T)
#Norm_exprMat <- assay(rlogQ3)
```

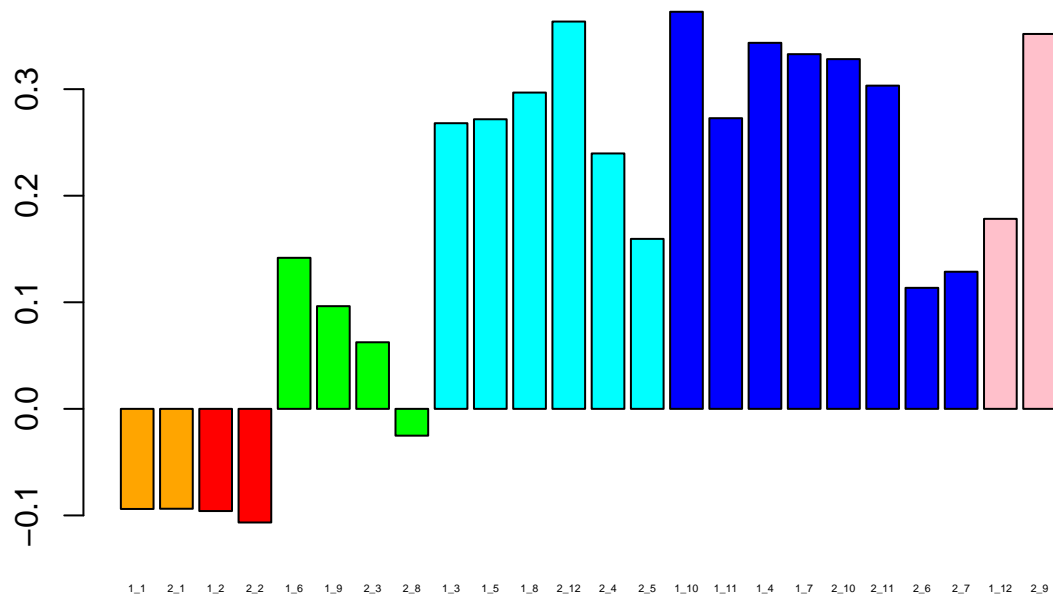
```
rankData <- rankGenes(Norm_exprMat)
```

```
scoredf <- simpleScore(rankData, upSet = myeloid.markers_cd64)
new_row <- c("cd64", scoredf$TotalScore)
cell_singscore <- cbind(cell_singscore, new_row)
```

```
scoredf <- cbind(colData(rlogQ3)$Position, scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]
```

```
barplot(scoredf$TotalScore,
  col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
  ,rep("blue",8), rep("pink",2)),

  names.arg =
  c("1_1", "2_1", "1_2", "2_2", "1_6", "1_9", "2_3", "2_8", "1_3", "1_5", "1_8", "2_12",
  "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
  "2_9"), cex.names = 0.3
)
```



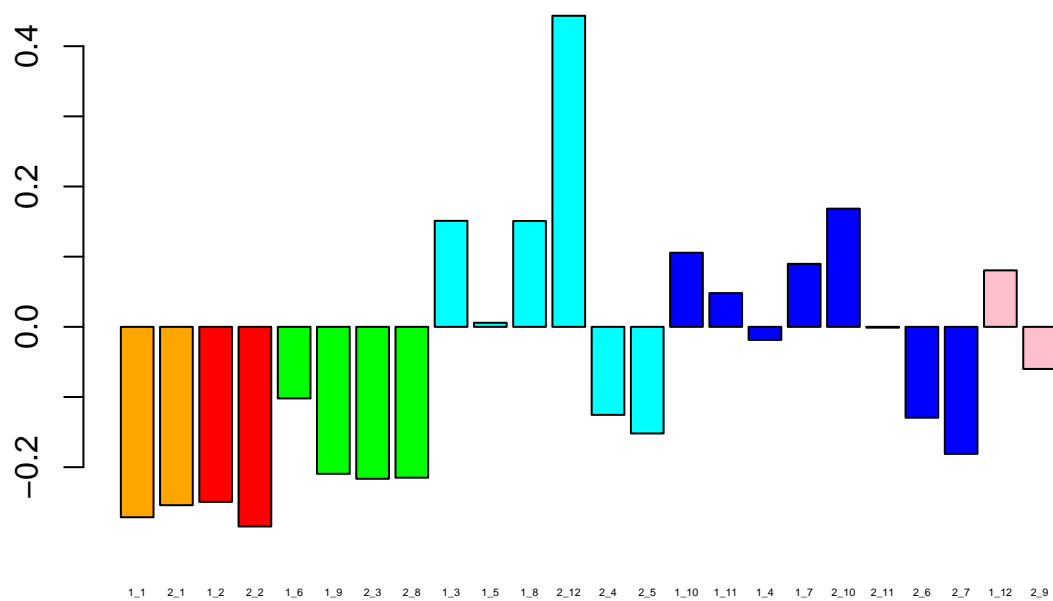
score CD206- IM

```
myeloid.markers <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Cécilia_project_2/1.2Visualise_feature/myeloids,

myeloid.markers_IM <- myeloid.markers[myeloid.markers$cluster == 0,]$gene

features.scores.use <- GetSignatureScore(dds_Q3, myeloid.markers_IM)
new_row <- c("CD206- IM", features.scores.use)
cell_score <- cbind(cell_score, new_row)

visualise_score(features.scores.use)
```



```

rankData <- rankGenes(Norm_exprMat)

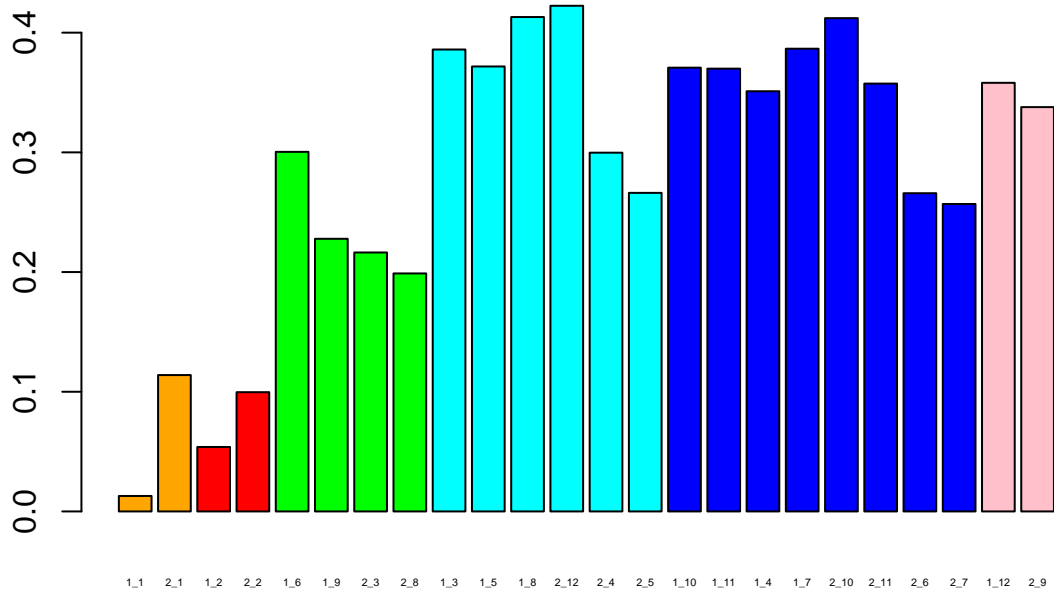
scoredf <- simpleScore(rankData, upSet = myeloid.markers_IM)
new_row <- c("CD206- IM", scoredf$TotalScore)
#cell_singscore <- cbind(cell_singscore, new_row)

scoredf <- cbind(colData(rlogQ3)$Position,scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]

barplot(scoredf$TotalScore,
        col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
        ,rep("blue",8), rep("pink",2)),

        names.arg =
        c("1_1","2_1","1_2","2_2","1_6","1_9","2_3","2_8","1_3","1_5","1_8","2_12",
        "2_4","2_5","1_10","1_11","1_4","1_7","2_10","2_11","2_6","2_7","1_12",
        "2_9"), cex.names = 0.3
)

```



score CD206+ IM

```

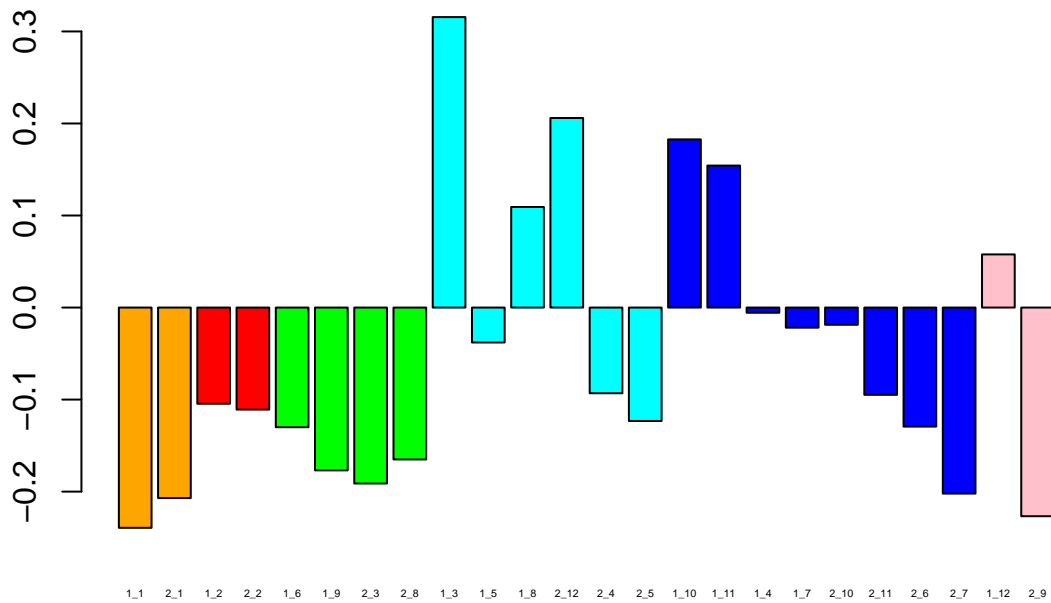
myeloid.markers <-
read.csv("/home/qiang/Documents/Travail_cell/Travail_V3/Cécilia_project_2/1.2Visualise_feature/myeloids.

myeloid.markers_CD206 <- myeloid.markers[myeloid.markers$cluster == 7,]$gene

features.scores.use <- GetSignatureScore(dds_Q3, myeloid.markers_CD206)
new_row <- c("CD206+ IM", features.scores.use)
#cell_score <- cbind(cell_score, new_row)

visualise_score(features.scores.use)

```



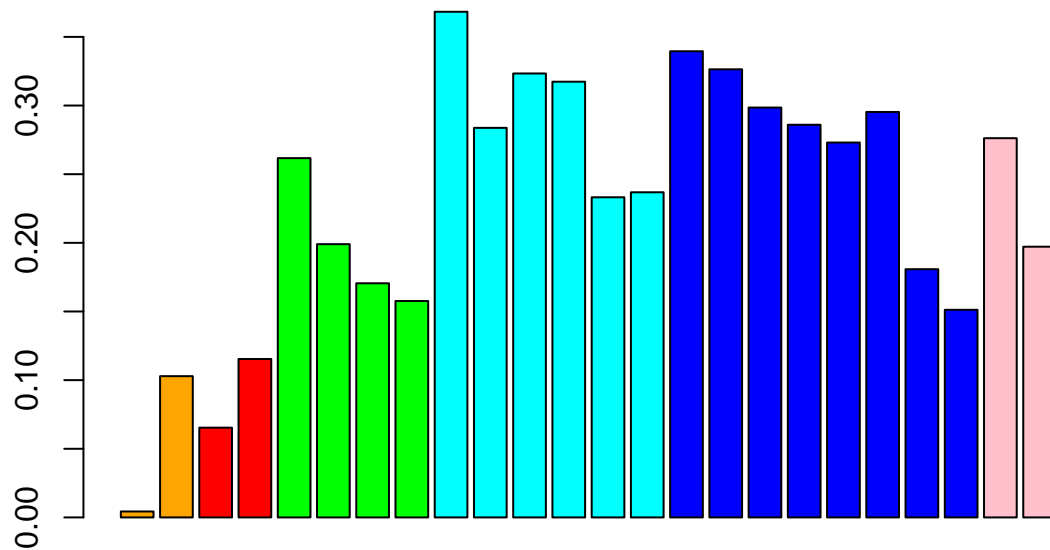
```
rankData <- rankGenes(Norm_exprMat)

scoredf <- simpleScore(rankData, upSet = myeloid.markers_CD206)
new_row <- c("CD206+ IM", scoredf$TotalScore)
#cell_singscore <- cbind(cell_singscore, new_row)

scoredf <- cbind(colData(rlogQ3)$Position,scoredf)
scoredf <- scoredf[order(scoredf$`colData(rlogQ3)$Position`),]

barplot(scoredf$TotalScore,
        col = c(rep("orange",2), rep("red",2) , rep("green",4), rep("cyan",6)
        ,rep("blue",8), rep("pink",2)),

        names.arg =
        c("1_1","2_1","1_2","2_2","1_6","1_9","2_3","2_8","1_3","1_5","1_8","2_12",
        "2_4", "2_5", "1_10", "1_11", "1_4", "1_7", "2_10", "2_11", "2_6", "2_7", "1_12",
        "2_9"), cex.names = 0.3
)
```



for module score # write csv file

```
colnames(cell_score) <- cell_score[1,]
cell_score <- cell_score[-1,]

rownames(cell_score) <- colData(dds_Q3)$SegmentDisplayName

#write.csv(cell_score, "cell_score.csv")
```

#write csv file for singscore

```
colnames(cell_singscore) <- cell_singscore[1,]
cell_singscore <- cell_singscore[-1,]

rownames(cell_singscore) <- colData(dds_Q3)$SegmentDisplayName

#write.csv(cell_singscore, "cell_singscore.csv")
```