

Preparing files for GSEA analysis

Abinet Joan

2024-04-22 14:33:22 +0200

Contents

Loading packages	1
Preparing gct and cls files for GSEA	1
Perilesionnal vs intralesional	1
Perilesionnal vs unaffected	2
Intralesionnal vs unaffected	3
Intralesionnal vs unaffected	3

Loading packages

```
suppressMessages(library(readxl))
suppressMessages(library(tidyr))
suppressMessages(library(ggplot2))
suppressMessages(library(dplyr))
```

Preparing gct and cls files for GSEA

The GSEA require 3 files that need to be generated from the sequencing data: A gct files describing the gene expression matrix and a cls files describing the metadata for each cells

A gmt files containing Molecular Signatures Database is directly downloaded from internet.

Perilesionnal vs intralesional

```
Q3 <- read_excel("../Data_DSPDA/Q3 - publication - V2.xlsx", sheet =
"TargetCountMatrix")
Q3 <- as.data.frame(Q3)
rownames(Q3) <- Q3$TargetName
Q3$TargetName <- NULL

Q3 <- Q3[,-c(1,5,9,12,13,17,18,23)]
```

```
# gct files
avg = as.data.frame(Q3)
df = data.frame(NAME=rownames(avg), Description="NA", stringsAsFactors = F)
df = cbind(df, avg)
df = rbind(c("#1.2", rep(NA, ncol(df)-1)), c(nrow(df), ncol(df)-2, rep(NA, ncol(df)-2)),
colnames(df), df)
```

```
#write.table(df, file = "exprMat.gct", sep = "\t", row.names = F, col.names = F, na = "",
quote = F)
```

```
# cls files
annotation_sample <- c(rep("perilesional",10), rep("intralesional", 5), "perilesional")

no_tot = ncol(avg)
no_classes = c("perilesional","intralesional")
df = data.frame(matrix(ncol=no_tot))
df[1,] = c(no_tot, length(no_classes), 1, rep(NA, no_tot - 3))
line2 = "#"
line3 = as.vector(annotation_sample)
for (i in 1:length(no_classes)){
  line2 = c(line2, as.vector(unique(annotation_sample)[i]))
}
line2 = c(line2, rep(NA, no_tot - length(line2)))
df = rbind(df, line2, line3)
#write.table(df, file = "MetaData.cls", sep = "\t", row.names = F, col.names = F, na =
"", quote = F)
```

Perilesionnal vs unaffected

```
Q3 <- read_excel("../Data_DSPDA/Q3 - publication - V2.xlsx", sheet =
"TargetCountMatrix")
Q3 <- as.data.frame(Q3)
rownames(Q3) <- Q3$TargetName
Q3$TargetName <- NULL

Q3 <- Q3[,-c(1,5,13,17,16,19,20,21,22)]
```

```
# gct files
avg = as.data.frame(Q3)
df = data.frame(NAME=rownames(avg), Description="NA", stringsAsFactors = F)
df = cbind(df, avg)
df = rbind(c("#1.2", rep(NA, ncol(df)-1)), c(nrow(df), ncol(df)-2, rep(NA, ncol(df)-2)),
colnames(df), df)
write.table(df, file = "PerivsUnaffected/exprMat.gct", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)
```

```
# cls files
annotation_sample <- c(rep("perilesional",6), "Unaffected",
rep("perilesional",2),"Unaffected", rep("perilesional",2), rep("Unaffected",2),
"perilesional")

no_tot = ncol(avg)
no_classes = c("perilesional","intralesional")
df = data.frame(matrix(ncol=no_tot))
df[1,] = c(no_tot, length(no_classes), 1, rep(NA, no_tot - 3))
line2 = "#"
line3 = as.vector(annotation_sample)
for (i in 1:length(no_classes)){
  line2 = c(line2, as.vector(unique(annotation_sample)[i]))
}
```

```

}
line2 = c(line2, rep(NA, no_tot - length(line2)))
df = rbind(df, line2, line3)
# write.table(df, file = "PerivsUnaffected/MetaData.cls", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)

```

Intralesionnal vs unaffected

```

Q3 <- read_excel("../Data_DSPDA/Q3 - publication - V2.xlsx", sheet =
"TargetCountMatrix")
Q3 <- as.data.frame(Q3)
rownames(Q3) <- Q3$TargetName
Q3$TargetName <- NULL

Q3 <- Q3[,c(9,12,16,18,19,20,21,22,23)]

# gct files
avg = as.data.frame(Q3)
df = data.frame(NAME=rownames(avg), Description="NA", stringsAsFactors = F)
df = cbind(df, avg)
df = rbind(c("#1.2", rep(NA, ncol(df)-1)), c(nrow(df), ncol(df)-2, rep(NA, ncol(df)-2)),
colnames(df), df)
# write.table(df, file = "IntravsUnaffected/exprMat.gct", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)

# cls files
annotation_sample <- c(rep("Unaffected",2), "intralesionnal","Unaffected",
rep("intralesionnal",4), "Unaffected")

no_tot = ncol(avg)
no_classes = c("Unaffected","intralesionnal")
df = data.frame(matrix(ncol=no_tot))
df[1,] = c(no_tot, length(no_classes), 1, rep(NA, no_tot - 3))
line2 = "#"
line3 = as.vector(annotation_sample)
for (i in 1:length(no_classes)){
  line2 = c(line2, as.vector(unique(annotation_sample)[i]))
}
line2 = c(line2, rep(NA, no_tot - length(line2)))
df = rbind(df, line2, line3)
#write.table(df, file = "IntravsUnaffected/MetaData.cls", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)

```

Intralesionnal vs unaffected

```

Q3 <- read_excel("../Data_DSPDA/Q3 - publication - V2.xlsx", sheet =
"TargetCountMatrix")
Q3 <- as.data.frame(Q3)
rownames(Q3) <- Q3$TargetName
Q3$TargetName <- NULL

Q3 <- Q3[,c(1,5,9,12,13,17,18,23)]

```

```
# gct files
avg = as.data.frame(Q3)
df = data.frame(NAME=rownames(avg), Description="NA", stringsAsFactors = F)
df = cbind(df, avg)
df = rbind(c("#1.2", rep(NA, ncol(df)-1)), c(nrow(df), ncol(df)-2, rep(NA, ncol(df)-2)),
colnames(df), df)
#write.table(df, file = "unaffectedvshealthy/exprMat.gct", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)
```

```
# cls files
annotation_sample <- c(rep("Healthy",2), rep("Unaffected",2), rep("Healthy",2),
rep("Unaffected",2))

no_tot = ncol(avg)
no_classes = c("Unaffected","Healthy")
df = data.frame(matrix(ncol=no_tot))
df[1,] = c(no_tot, length(no_classes), 1, rep(NA, no_tot - 3))
line2 = "#"
line3 = as.vector(annotation_sample)
for (i in 1:length(no_classes)){
  line2 = c(line2, as.vector(unique(annotation_sample)[i]))
}
line2 = c(line2, rep(NA, no_tot - length(line2)))
df = rbind(df, line2, line3)
#write.table(df, file = "unaffectedvshealthy/MetaData.cls", sep = "\t", row.names = F,
col.names = F, na = "", quote = F)
```

Then GSEA will be run and the results can be found in the directory 8.2-GSEA/output

The Dotplot are generated by the code in 8.2-GSEA/Plots/Creating_Dotplot.rmd

```
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 22.04.4 LTS
##
## Matrix products: default
## BLAS: /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblas-p-r0.3.20.so; LAPACK version 3.10.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
##  [3] LC_TIME=fr_BE.UTF-8      LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=fr_BE.UTF-8  LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=fr_BE.UTF-8     LC_NAME=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
## [11] LC_MEASUREMENT=fr_BE.UTF-8 LC_IDENTIFICATION=C
##
## time zone: Europe/Brussels
## tzcode source: system (glibc)
##
## attached base packages:
```

```
## [1] stats      graphics  grDevices utils      datasets  methods  base
##
## other attached packages:
## [1] dplyr_1.1.2  ggplot2_3.4.2 tidyr_1.3.0  readxl_1.4.2
##
## loaded via a namespace (and not attached):
## [1] vctrs_0.6.3      cli_3.6.1        knitr_1.43       rlang_1.1.1
## [5] xfun_0.39        purrr_1.0.1      generics_0.1.3   glue_1.6.2
## [9] colorspace_2.1-0 htmltools_0.5.5  scales_1.2.1     fansi_1.0.4
## [13] rmarkdown_2.23   grid_4.3.3       cellranger_1.1.0 munsell_0.5.0
## [17] evaluate_0.21    tibble_3.2.1     fastmap_1.1.1    yaml_2.3.7
## [21] lifecycle_1.0.3  compiler_4.3.3   pkgconfig_2.0.3  rstudioapi_0.14
## [25] digest_0.6.33    R6_2.5.1         tidyselect_1.2.0 utf8_1.2.3
## [29] pillar_1.9.0     magrittr_2.0.3   withr_2.5.0      gtable_0.3.3
## [33] tools_4.3.3
```