Heatmap RNAseq

2023-11-04

This script generates heatmaps from RNA sequencing (RNAseq) data, applying several data transformations and customizations. Here's an overview of the main steps:

1. Load Libraries
2. Data Import and Preparation: RNAseq data is read from a CSV file, and transformations are applied: Log2 and Z-score Transformation.
3. Heatmap Generation and Saving as PDF and PNG files with different dimensions to accommodate various presentation needs.
4. Troubleshooting: Commands are included to detect and locate any NA (missing) values, addressing common data issues that may prevent heatmap generation.

Based on the following script online: https://davetang.github.io/muse/pheatmap.html

Load necessary libraries

```
library(pheatmap)
library(tidyverse)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
## Warning: package 'stringr' was built under R version 4.2.3
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(ggplotify)
library(heatmaply)
```

```
## Loading required package: plotly
##
## Attaching package: 'plotly'
##
## The following object is masked from 'package:ggplot2':
##
##     last_plot
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following object is masked from 'package:graphics':
```

```
##
##     layout
##
## Loading required package: viridis
## Loading required package: viridisLite
##
## ======================
## Welcome to heatmaply version 1.5.0
##
## Type citation('heatmaply') for how to cite the package.
## Type ?heatmaply for the main documentation.
##
## The github page is: https://github.com/talgalili/heatmaply/
## Please submit your suggestions and bug-reports at: https://github.com/talgalili/heatmaply/issues
## You may ask questions at stackoverflow, use the r and heatmaply tags:
##   https://stackoverflow.com/questions/tagged/heatmaply
## ======================
```

Read the data table

```
mat<-read.csv("/Users/jeffreyreina/Documents/Salk/RNAseq MDA-MB-231 results/03.Result_X202SC23073852-Z0
```

Transform data to log2 scale

```
mattransformed<-log2(mat+1) #+1 to avoid errors with 0 values
```

Transform data to z-scores

```
cal_z_score <- function(x){
  (x - mean(x)) / sd(x)
}

matzscore <- t(apply(mattransformed, 1, cal_z_score))
```

Define sample groups for annotation

```
dfh<-data.frame(sample=as.character(colnames(mat)),group="Treatment")%>%
                column_to_rownames("sample")
dfh$group<-ifelse(rownames(dfh) %in% c("KO1","KO2","KO3"),
                "KO","WT")
dfh
```

```
##     group
## KO1    KO
## KO2    KO
## KO3    KO
## WT1    WT
## WT2    WT
## WT3    WT
```

Choose color palette for heatmap

```
my_palette <- colorRampPalette(c("green", "black", "red"))(n = 9999)
```

Save heatmap as a PDF

```
pdf("Heatmap2.pdf",width=4.4,height=6.3)

pheatmap(matzscore, color = my_palette, scale = "row", show_rownames = FALSE, annotation_col = dfh, ann
dev.off()
```

```
## pdf
##   3
```

Save heatmap as a PNG

```
png("pheatmap_5.png", res=300, width=7, height=4.5, unit="in")

pheatmap(matzscore, color = my_palette, scale = "row", show_rownames = FALSE, annotation_col = dfh, ann
dev.off()
```

```
## pdf
##   3
```

Save heatmap with exchanged width and height in PNG

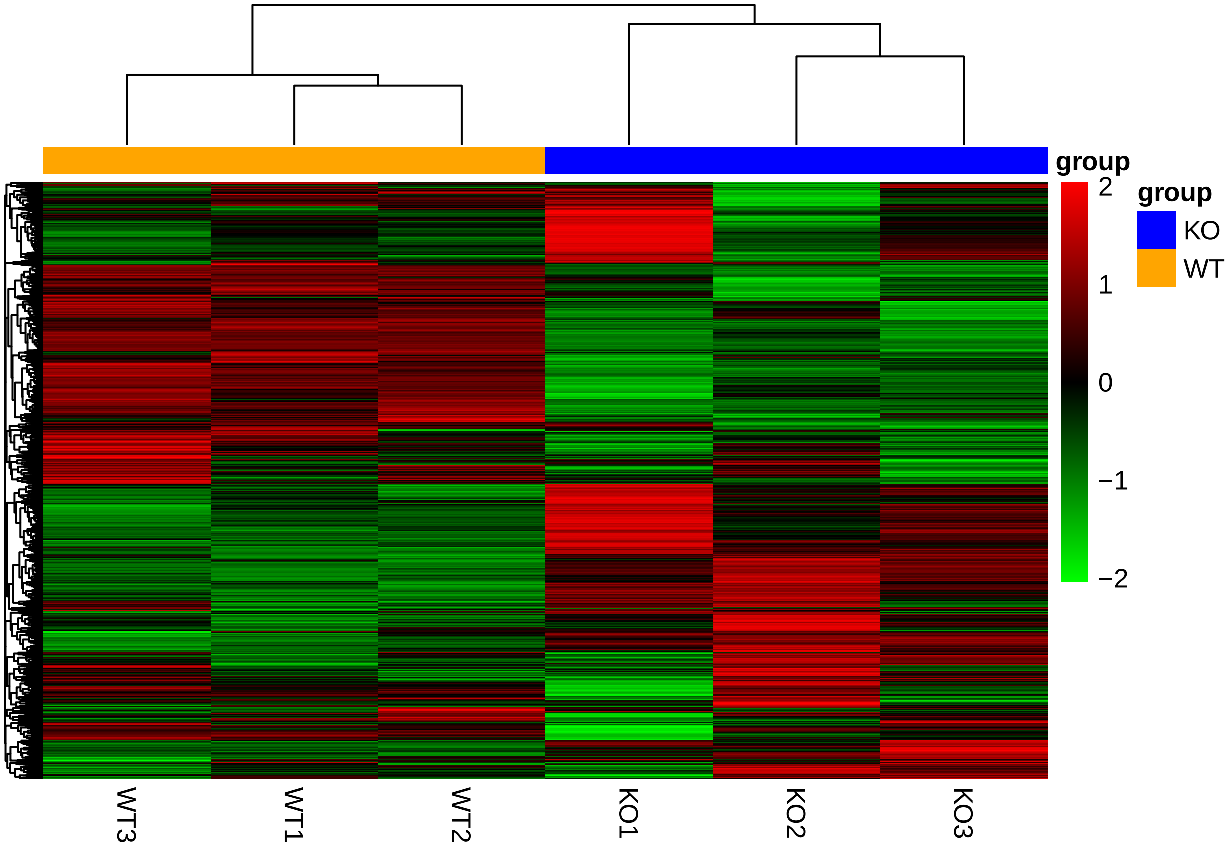
```
png("pheatmap_2.png", res=300, width=4.5, height=7, unit="in")

pheatmap(matzscore, color = my_palette, scale = "row", show_rownames = FALSE, annotation_col = dfh, ann
dev.off()
```

```
## pdf
##   3
```

Save heatmap as an R object

```
hm_ph <-pheatmap(matzscore, color = my_palette, scale = "row", show_rownames = FALSE, annotation_col = 
```



“ “