# Differential expression analysis of yeast cells exposed or not to alpha factor

Hernan Lorenzi

## Load libraries

```
library(tidyverse)
library(DESeq2)
library(cowplot)
library(ggpubr)
library("RColorBrewer")
library(pheatmap)
library(ggsci)
library(BSgenome)
library("org.Sc.sgd.db")
library("AnnotationDbi")
library(EnhancedVolcano)
```

## Load useful functions

```
source("01_aux_rnaseq_functions.R")
```

## Load data

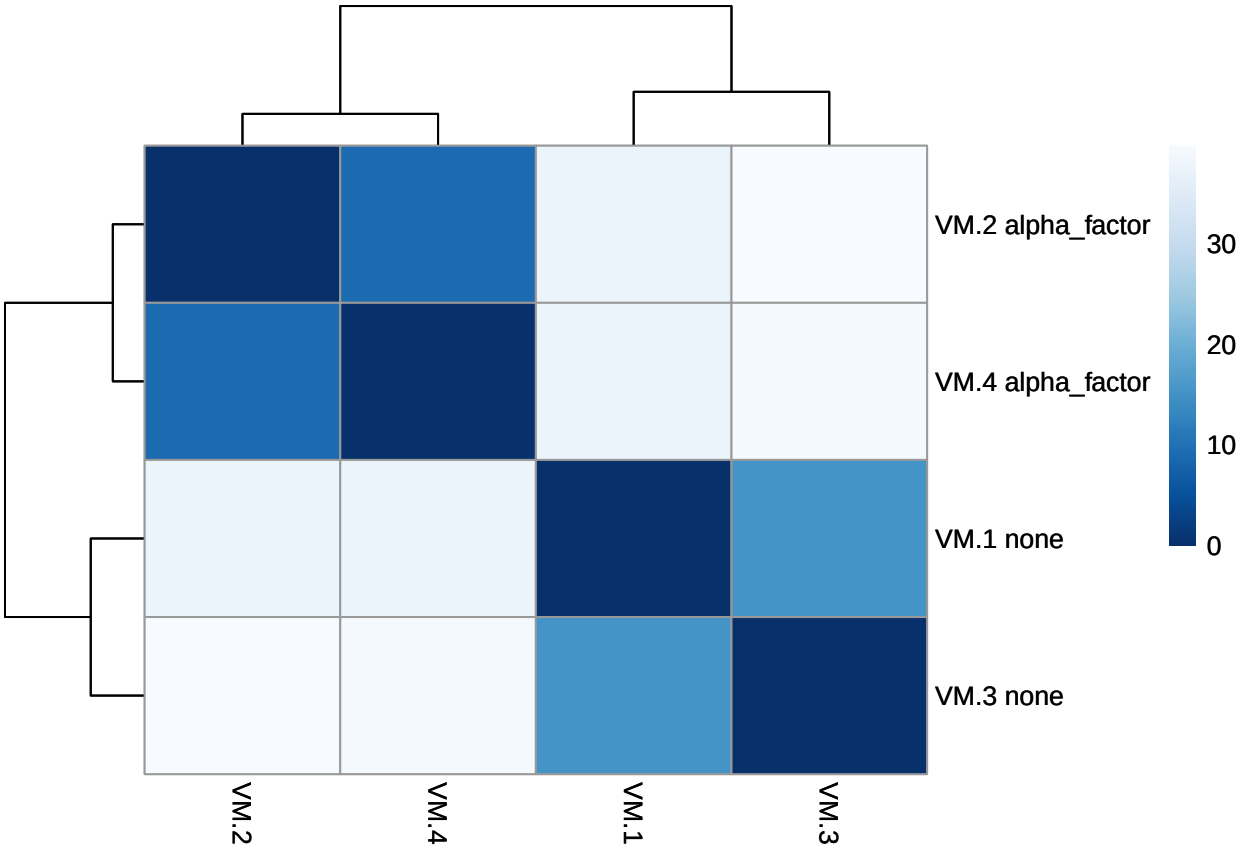
```
# Import read counts table
read_counts <- read.table(file = "data/read_counts_all.txt", header = TRUE, row.names = 1,
    sep = "\t")

# round read counts to the closest integer
read_counts <- round(read_counts, digits = 0)

# Read metadata table
metadata <- read.table(file = "data/metadata.txt", header = TRUE, row.names = 1,
    sep = "\t")

# Change metadata row names to match read_count's column names
rownames(metadata) <- colnames(read_counts)
```

```
# Sort tables so metadata and read counts match order
read_counts <- read_counts[, match(rownames(metadata), colnames(read_counts))]

# include total read counts in metadata
metadata$read_counts <- colSums(read_counts)

# include sample ids in metadata as a variable (column)
metadata$sample_id <- c("VM-1", "VM-2", "VM-3", "VM-4")
```

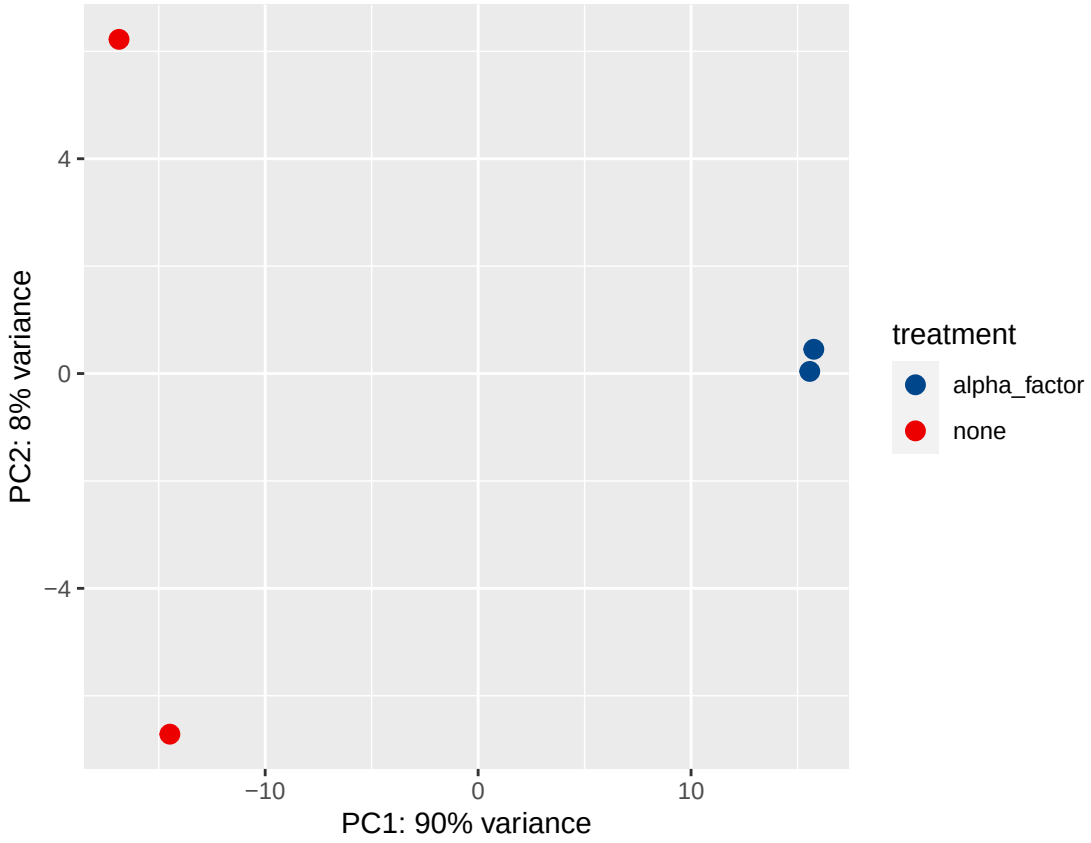
## DE analysis with DESeq2

```
dir.create(path = "./Plots", showWarnings = FALSE)

# Adding read_depth in design to control for read_depth
dds <- DESeqDataSetFromMatrix(countData = read_counts,
                              colData = metadata,
                              design = ~ treatment)


# Plot total reads per sample using barchar
p <- ggbarplot(data = metadata,
          x = "sample_id",
          y = "read_counts",
          x.text.angle = 90,
          fill = "treatment",
          title = "Total read counts per sample",
          ylab = "Read counts",
          sort.by.groups = TRUE,
          palette = c("red","orange"), #"jco",
          sort.val = "asc")

ggsave2("Plots/barplot_read_counts_per_sample.pdf", plot = p)

# Normalize counts
dds.vst <- vst(dds, blind=FALSE)

# Keep genes with at least 20 reads total across samples
keep <- rowSums(as.data.frame(dds.vst@assays@data@listData)) >= 20
dds.vst <- dds.vst[keep,]

# Calculate distances between samples
sampleDists <- dist(t(assay(dds.vst)))

# Plot inter-sample distances
old.par <- par(no.readonly=T)

sampleDistMatrix <- as.matrix(sampleDists)
rownames(sampleDistMatrix) <- paste(rownames(sampleDistMatrix), dds.vst$treatment)
#colnames(sampleDistMatrix) <- NULL
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)
```

```
p.hm <- pheatmap(sampleDistMatrix,
         clustering_distance_rows=sampleDists,
         clustering_distance_cols=sampleDists,
         col=colors)

ggsave2(filename = "./Plots/heat_map.pdf", plot = p.hm)
p.hm
```




```
# PCA
pcaData <- plotPCA(dds.vst, intgroup=c("treatment"), returnData=TRUE)
percentVar <- round(100 * attr(pcaData, "percentVar"))
y.coords = c(min(pcaData$PC1, pcaData$PC2), max(pcaData$PC1, pcaData$PC2))
x.coords = y.coords
p1 <- ggplot(pcaData, aes(PC1, PC2, color=treatment)) +
  geom_point(size=3) + scale_color_lancet() +
  xlab(paste0("PC1: ",percentVar[1],"% variance")) +
  ylab(paste0("PC2: ",percentVar[2],"% variance")) +
  coord_fixed(ratio = (max(pcaData$PC1)-min(pcaData$PC1))/(max(pcaData$PC2)-min(pcaData$PC2)))

ggsave("Plots/pca_by_group.pdf", plot = p1)
p1
```




# Run DE analysis

```
dir.create(path = "./DE", showWarnings = FALSE)

# Calculate DE for WT samples
dds$treatment <- relevel(dds$treatment, "none")
dds <- DESeq(dds)
resultsNames(dds)
```

```
## [1] "Intercept"                      "treatment_alpha_factor_vs_none"
```

```
# Using lfcShrink instead of results to reduce high Log2FC bias of genes with
# low expression
res_treatment_vs_control <- lfcShrink(dds, coef = "treatment_alpha_factor_vs_none",
    type = "ashr", )

# Replace NAs by 1s
res_treatment_vs_control$pvalue[is.na(res_treatment_vs_control$pvalue)] <- 1
res_treatment_vs_control$padj[is.na(res_treatment_vs_control$padj)] <- 1

summary(res_treatment_vs_control, alpha = 0.05)
```

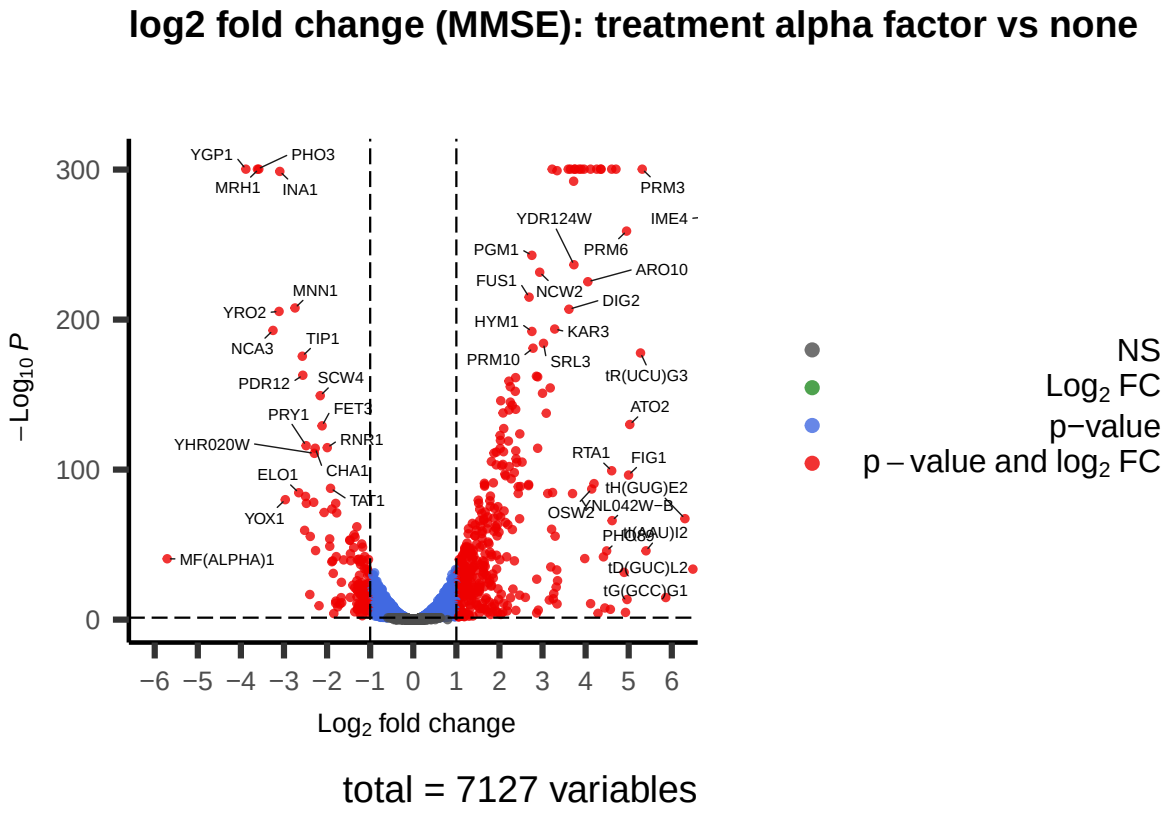
```
##
## out of 7043 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 1592, 23%
```

```
## LFC < 0 (down)     : 1591, 23%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 6)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

```
res_treatment_vs_control$gene_names <- get_gene_names_from_gene_ids(ensemble_ids = rownames(res_treatmen
    annotation_db = org.Sc.sgd.db, look_for = "ENSEMBL", fetch = "GENENAME")

# Sort result table based on adj.p (ascending)
res_treatment_vs_control <- res_treatment_vs_control[order(res_treatment_vs_control$padj,
    decreasing = F), ]
# Save DE results
write.table(x = as.data.frame(res_treatment_vs_control), file = "./DE/DE_treatment_vs_control.txt",
    sep = "\t", col.names = NA)
```

## Plot volcano plots

```
generate_volcano_plot(res.tmp = res_treatment_vs_control, my_file_name = "volcano_plot.pdf")
```



```
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] EnhancedVolcano_1.12.0      ggrepel_0.9.1
##  [3] org.Sc.sgd.db_3.14.0        AnnotationDbi_1.56.2
##  [5] BSgenome_1.62.0             rtracklayer_1.54.0
##  [7] Biostrings_2.62.0           XVector_0.34.0
##  [9] ggsci_2.9                   pheatmap_1.0.12
## [11] RColorBrewer_1.1-3          ggpubr_0.4.0
## [13] cowplot_1.1.1               DESeq2_1.34.0
## [15] SummarizedExperiment_1.24.0 Biobase_2.54.0
## [17] MatrixGenerics_1.6.0        matrixStats_0.62.0
## [19] GenomicRanges_1.46.1        GenomeInfoDb_1.30.1
## [21] IRanges_2.28.0              S4Vectors_0.32.4
## [23] BiocGenerics_0.40.0         forcats_0.5.1
## [25] stringr_1.4.0               dplyr_1.0.9
## [27] purrr_0.3.4                 readr_2.1.2
## [29] tidyr_1.2.0                 tibble_3.1.7
## [31] ggplot2_3.3.6               tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##   [1] readxl_1.4.0             backports_1.4.1          systemfonts_1.0.4
##   [4] splines_4.1.1            BiocParallel_1.28.3      digest_0.6.29
##   [7] invgamma_1.1             htmltools_0.5.2          SQUAREM_2021.1
##  [10] fansi_1.0.3              magrittr_2.0.3           memoise_2.0.1
##  [13] tzdb_0.3.0               annotate_1.72.0          modelr_0.1.8
##  [16] extrafont_0.18           extrafontdb_1.0          colorspace_2.0-3
##  [19] blob_1.2.3               rvest_1.0.2              textshaping_0.3.6
##  [22] haven_2.5.0              xfun_0.31                crayon_1.5.1
##  [25] RCurl_1.98-1.7           jsonlite_1.8.0           genefilter_1.76.0
##  [28] survival_3.2-11          glue_1.6.2               gtable_0.3.0
##  [31] zlibbioc_1.40.0          DelayedArray_0.20.0      proj4_1.0-11
##  [34] car_3.1-0                Rttf2pt1_1.3.10          maps_3.4.0
##  [37] abind_1.4-5              scales_1.2.0             DBI_1.1.3
##  [40] rstatix_0.7.0            Rcpp_1.0.9               xtable_1.8-4
##  [43] bit_4.0.4                truncnorm_1.0-8          httr_1.4.3
##  [46] ellipsis_0.3.2           farver_2.1.1             pkgconfig_2.0.3
##  [49] XML_3.99-0.10            dbplyr_2.2.1             locfit_1.5-9.5
```

```
##  [52] utf8_1.2.2              labeling_0.4.2          tidyselect_1.1.2
##  [55] rlang_1.0.3             munsell_0.5.0           cellranger_1.1.0
##  [58] tools_4.1.1             cachem_1.0.6            cli_3.3.0
##  [61] generics_0.1.3          RSQLite_2.2.14          broom_1.0.0
##  [64] evaluate_0.15           fastmap_1.1.0           yaml_2.3.5
##  [67] ragg_1.2.2              knitr_1.39              bit64_4.0.5
##  [70] fs_1.5.2                KEGGREST_1.34.0         ash_1.0-15
##  [73] formatR_1.12            ggrastr_1.0.1           xml2_1.3.3
##  [76] compiler_4.1.1          rstudioapi_0.13         beeswarm_0.4.0
##  [79] png_0.1-7               ggsignif_0.6.3          reprex_2.0.1
##  [82] geneplotter_1.72.0      stringi_1.7.6           highr_0.9
##  [85] ggalt_0.4.0             lattice_0.20-44         Matrix_1.3-4
##  [88] vctrs_0.4.1             pillar_1.7.0            lifecycle_1.0.1
##  [91] irlba_2.3.5             bitops_1.0-7            R6_2.5.1
##  [94] BiocIO_1.4.0            KernSmooth_2.23-20      vipor_0.4.5
##  [97] MASS_7.3-54             assertthat_0.2.1        rjson_0.2.21
## [100] withr_2.5.0             GenomicAlignments_1.30.0 Rsamtools_2.10.0
## [103] GenomeInfoDbData_1.2.7  parallel_4.1.1          hms_1.1.1
## [106] grid_4.1.1              rmarkdown_2.14          ashr_2.2-54
## [109] carData_3.0-5           mixsqp_0.3-43           lubridate_1.8.0
## [112] ggbeeswarm_0.6.0        restfulr_0.0.15
```