R\_final\_V3\_FINAL\_SUBMISSION

2025-08-16

import

library(ggplot2) #import lib and data  
genes <- read.csv("QBS103\_GSE157103\_genes.csv", row.names = 1)  
metadata <- read.csv("QBS103\_GSE157103\_series\_matrix-1.csv")

redefining the plotting function from v2 to generate publication ready plots

plot\_all\_figures <- function(metadata, gene\_name, cont\_var, cat\_var1, cat\_var2) {  
   
 temp <- metadata # make a copy  
 temp[[gene\_name]] <- as.numeric(genes[gene\_name, ])  
   
 # remove row wehre age is : and convert >89 to 90  
 #trim whitespace to be able to find :   
 #from https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/trimws  
 metadata <- metadata[trimws(metadata[[cont\_var]]) != ":", ]  
 temp[[cont\_var]][temp[[cont\_var]] == ">89"] <- "90"  
   
 # convert age to numeric  
 temp[[cont\_var]] <- as.numeric(temp[[cont\_var]])  
   
 # filter out rows with missing values  
 metadata\_clean <- temp[ !(is.na(temp[[cont\_var]]) | is.na(temp[[gene\_name]])), ]  
 metadata\_clean <- metadata\_clean[metadata\_clean[[cat\_var1]] != " unknown", ]  
  
  
 #histogram for gene expression   
 print(  
 ggplot(metadata\_clean, aes\_string(x = gene\_name)) +  
 geom\_histogram(bins = 50, fill = "pink", color = "black") +  
 scale\_x\_continuous(breaks = seq(0, 190, by = 10)) +  
 scale\_y\_continuous(breaks = seq(0, 10, by = 1)) +  
 labs(title = paste("Histogram of", gene\_name, "Expression"),  
 x = paste(gene\_name, "Expression Level"),  
 y = "Frequency")  
 )  
  
 #scatterplot  
 print(  
 ggplot(metadata\_clean, aes\_string(x = cont\_var, y = gene\_name)) +  
 scale\_x\_continuous(breaks = seq(10, 100, by = 10)) +  
 scale\_y\_continuous(breaks = seq(0, 200, by = 10)) +  
 geom\_point(size = 2, colour = "pink", alpha = 0.8) +  
 labs(title = paste(gene\_name, "Expression vs. Age"),  
 x = "Age (years)",  
 y = paste(gene\_name, "Expression Level"))  
 )  
  
 #boxplot  
 print(  
 ggplot(metadata\_clean, aes\_string(x = cat\_var2, y = gene\_name, fill = cat\_var1)) +  
 geom\_boxplot() +  
 scale\_fill\_manual(values = c(" female" = "pink", " male" = "deeppink")) +  
 labs(title = paste(gene\_name, "Expression by ICU Status and Sex" ),  
 x = "ICU Status",  
 y = paste(gene\_name, "Expression Level"),  
 fill = "Sex") +  
 scale\_y\_continuous(breaks = seq(0, 200, by = 20))  
 )  
}

Generate summary statistics table

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tableone)  
  
#extract my genes row from the gene df and add to metadata  
metadata$ABHD5 <- as.numeric(genes["ABHD5", ])  
metadata$ventilator.free\_days <- as.numeric(metadata$ventilator.free\_days)  
metadata$ferritin.ng.ml. <- as.numeric(metadata$ferritin.ng.ml.)

## Warning: NAs introduced by coercion

metadata$age <- as.numeric(metadata$age)

## Warning: NAs introduced by coercion

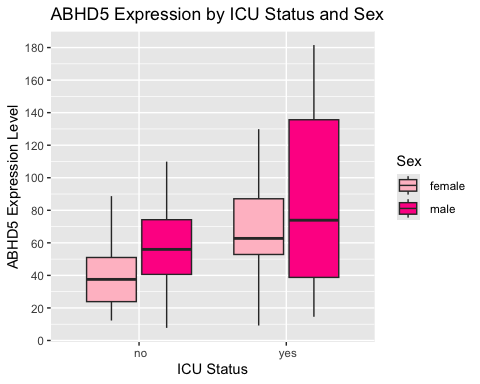
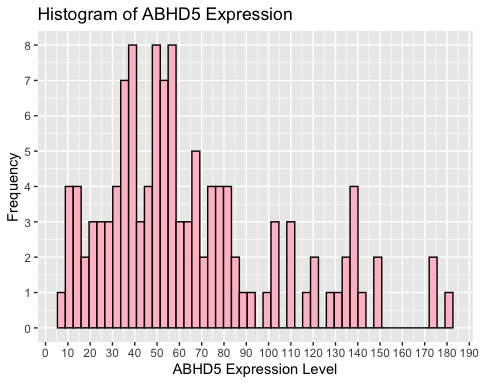
#clean  
metadata\_clean <- metadata %>%  
 filter(!is.na(age) & !is.na(ferritin.ng.ml.) & !is.na(ventilator.free\_days) &  
 !is.na(sex) & !is.na(icu\_status))  
  
contVars <- c("age", "ventilator.free\_days", "ferritin.ng.ml.")  
catVars <- c("sex", "mechanical\_ventilation", "icu\_status")  
  
vars <- c(contVars, catVars)  
  
table1 <- CreateTableOne(  
 vars = vars,  
 strata = "icu\_status", # stratify by icu status  
 data = metadata\_clean,  
 factorVars = catVars  
)  
  
  
print(table1,  
 nonnormal = c("ferritin.ng.ml."),   
 quote = FALSE,  
 noSpaces = TRUE,  
 test = TRUE)

## Stratified by icu\_status  
## no   
## n 48   
## age (mean (SD)) 58.96 (18.00)   
## ventilator.free\_days (mean (SD)) 26.73 (5.68)   
## ferritin.ng.ml. (median [IQR]) 406.00 [187.75, 905.75]  
## sex = male (%) 24 (50.0)   
## mechanical\_ventilation = yes (%) 3 (6.2)   
## icu\_status = yes (%) 0 (0.0)   
## Stratified by icu\_status  
## yes p test   
## n 59   
## age (mean (SD)) 64.05 (13.38) 0.096   
## ventilator.free\_days (mean (SD)) 14.22 (11.82) <0.001   
## ferritin.ng.ml. (median [IQR]) 685.00 [325.00, 1212.00] 0.066 nonnorm  
## sex = male (%) 37 (62.7) 0.261   
## mechanical\_ventilation = yes (%) 43 (72.9) <0.001   
## icu\_status = yes (%) 59 (100.0) <0.001

plot my gene from the first assignment: ABDH5

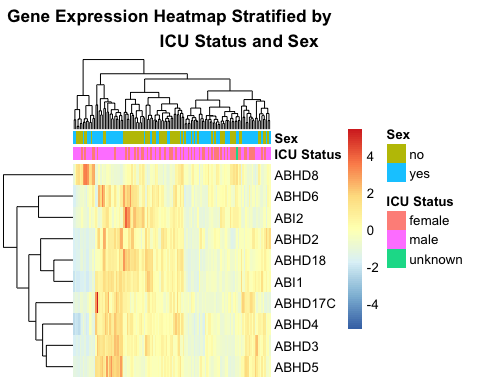
genes\_to\_plot <- c("ABHD5")  
#plot  
for (gene in genes\_to\_plot) {  
 plot\_all\_figures(metadata, gene\_name = gene, cont\_var = "age", cat\_var1 = "sex"  
 , cat\_var2 = "icu\_status")  
}

## Warning: `aes\_string()` was deprecated in ggplot2 3.0.0.  
## ℹ Please use tidy evaluation idioms with `aes()`.  
## ℹ See also `vignette("ggplot2-in-packages")` for more information.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.



generating a heatmap

library(pheatmap)  
  
# 10 genes   
gene\_list <- c("ABHD17C", "ABHD18", "ABHD2" , "ABHD3", "ABHD4",  
 "ABHD5", "ABHD6", "ABHD8", "ABI1", "ABI2")  
  
# expression matrix  
expr\_mat <- as.matrix(apply(genes[gene\_list, , drop = FALSE], 2, as.numeric))   
#extract the 10 genes i picked  
rownames(expr\_mat) <- gene\_list #labels with gene names  
  
# tracking bars  
bars <- data.frame(  
 sex = factor(trimws(as.character(metadata$sex))),  
 icu\_status = factor(trimws(as.character(metadata$icu\_status)))  
)  
  
#change names of labels  
colnames(bars) <- c("ICU Status", "Sex")  
  
#how to use matrix()  
#https://www.datamentor.io/r-programming/matrix#google\_vignette  
  
#make tracking bars corresponding to icu status and sex   
rownames(bars) <- colnames(expr\_mat) # making sure things align  
  
#how to use pheatmap   
#https://biostatsquid.com/step-by-step-heatmap-tutorial-with-pheatmap/  
# heatmap  
pheatmap(expr\_mat,#data from expression matrix   
 scale = "row",  
 main = paste( "Gene Expression Heatmap Stratified by   
 ICU Status and Sex" ),  
 annotation\_col = bars, #add tracking bars  
 cluster\_rows = TRUE,clustering\_distance\_rows = "euclidean", #cluster   
 cluster\_cols = TRUE, #cluster  
 show\_colnames = FALSE) #hide sample names

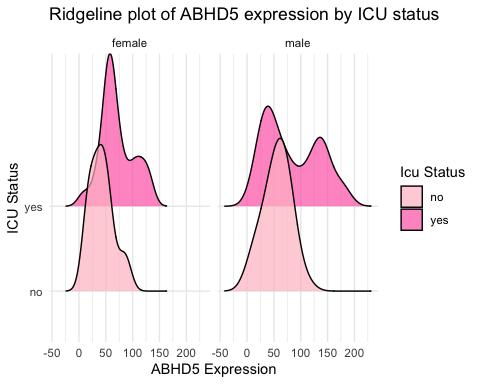


generating a novel plot

library(ggplot2)  
library(ggridges)  
df <- data.frame( #make df  
 expr = as.numeric(genes["ABHD5", ]), #get gene row  
 icu\_status = factor(trimws(metadata$icu\_status), levels = #get icu status  
 c("no","yes")),  
 sex = factor(trimws(metadata$sex), levels = c("female","male"))  
 #get sex  
)  
df\_clean <- subset(df, !is.na(expr) & sex != "unknown" & !is.na(icu\_status))  
  
  
#plot   
#ridgeline plot = stacked density plot  
ggplot(df\_clean, aes(x = expr, y = icu\_status, fill = icu\_status)) +  
 geom\_density\_ridges(alpha = 0.7) +  
 facet\_wrap(~ sex) +  
 scale\_fill\_manual(values = c("no" = "pink", "yes" = "hotpink"),  
 name = "Icu Status") +  
   
 labs(  
 x = "ABHD5 Expression",  
 y = "ICU Status",  
 title = "Ridgeline plot of ABHD5 expression by ICU status"  
 ) +  
 theme\_minimal()

## Picking joint bandwidth of 11.1

## Picking joint bandwidth of 16.4



#analysis  
#In both sexes patients with ICU “yes” status show greater variability in  
#ABHD5 expression compared to ICU “no.”