Biomarker Visualizations

Jordan Lo

2022-11-30

# Enables us to read excel files  
library("readxl")  
# Allows us to do data processing and clean-up  
library("tidyverse")

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.0 ✔ purrr 0.3.5   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.3 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

# Enables us to write to an excel file  
library("writexl")  
# GGally alows us to use their pairs funtion  
library("GGally")

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

# ggplot2 lets us make pretty graphs  
library("ggplot2")  
# ggpubr   
library("ggpubr")

bio\_clinical <- read\_excel("C:\\Users\\notjello\\Documents\\Github\\Biomarker-Master-Sheet\\Biomarker\_Assays\_and\_Clinical\_Data\_20221122.xlsx")  
  
# creates new columns for the log transformations of the biomarker concentrations   
bio\_clinical$log\_ang2 <- log(bio\_clinical$`urine\_ang-2\_V1 (pg/mL)`)  
bio\_clinical$log\_il18 <- log(bio\_clinical$`urine\_il18\_V1 (pg/mL)`)  
bio\_clinical$log\_ngal <- log(bio\_clinical$`urine\_ngal\_V1 (pg/mL)`)  
bio\_clinical$log\_sfas <- log(bio\_clinical$`urine\_sfas\_V1 (pg/mL)`)  
  
# converts covid19 diagnosis to 0 for neg and 1 for pos  
bio\_clinical['COVID19'][bio\_clinical['COVID19'] == 'pos'] <- "1"  
bio\_clinical['COVID19'][bio\_clinical['COVID19'] == 'Pos'] <- "1"  
bio\_clinical['COVID19'][bio\_clinical['COVID19'] == 'neg'] <- "0"  
  
bio\_clinical$COVID19<- as.numeric(as.character(bio\_clinical$COVID19))  
  
# converts aki\_hosp from numeric into a character for better visualization  
bio\_clinical$aki\_hosp <- as.character(bio\_clinical$aki\_hosp)  
  
  
# view(bio\_clinical)

# creates a pairs plot of the biomarker concentration data using ggpairs  
bio\_clinical %>%  
 select(c(`log\_ang2`, `log\_il18`, `log\_ngal`, `log\_sfas`, )) %>% # selects the data used  
 ggpairs(labels = letters[1:ncol(bio\_clinical)], # creates the actual plot  
 lower = list(continuous=wrap("smooth"))) +   
 theme\_bw() + # color scheme of plot  
 ggtitle("Correlation Data of Biomarker Concentrations") + # creates a title  
 theme(plot.title = element\_text(hjust = 0.5)) # centers titel

## Warning in warn\_if\_args\_exist(list(...)): Extra arguments: "labels" are being  
## ignored. If these are meant to be aesthetics, submit them using the 'mapping'  
## variable within ggpairs with ggplot2::aes or ggplot2::aes\_string.

## Warning: Removed 50 rows containing non-finite values (`stat\_density()`).

## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 50 rows containing missing values  
  
## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 50 rows containing missing values  
  
## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 50 rows containing missing values

## Warning: Removed 50 rows containing non-finite values (`stat\_smooth()`).

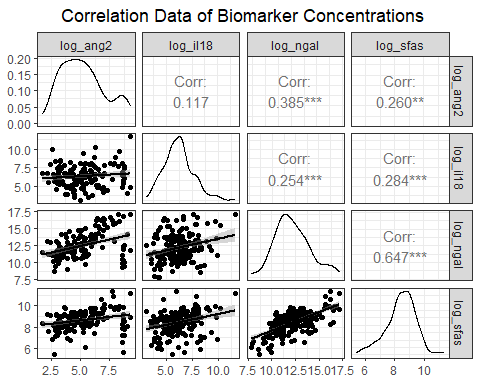
## Warning: Removed 50 rows containing missing values (`geom\_point()`).

## Warning: Removed 50 rows containing non-finite values (`stat\_smooth()`).

## Warning: Removed 50 rows containing missing values (`geom\_point()`).

## Warning: Removed 50 rows containing non-finite values (`stat\_smooth()`).

## Warning: Removed 50 rows containing missing values (`geom\_point()`).

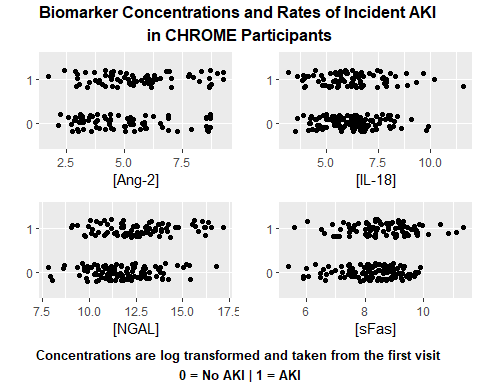


bio\_clinical\_NA <- bio\_clinical %>%  
 filter(aki\_hosp != "NA")  
  
# creates a jitter plot of the ang 2 levels and the presence of incident aki  
ang2\_jitter <-  
 ggplot(data = bio\_clinical\_NA, # selects dataframe  
 mapping = (aes(x = log\_ang2, y = aki\_hosp))) + # maps variables to x and y  
 geom\_jitter(width = 0, # sets parameters for the jitter  
 height = 0.2) +  
 labs(x = "[Ang-2]", # labels axes  
 y = " ")  
  
il18\_jitter <-  
 ggplot(data = bio\_clinical\_NA,   
 mapping = (aes(x = log\_il18, y = aki\_hosp))) +  
 geom\_jitter(width = 0,   
 height = 0.2) +  
 labs(x = "[IL-18]",  
 y = " ")  
  
sfas\_jitter <-  
 ggplot(data = bio\_clinical\_NA,   
 mapping = (aes(x = log\_sfas, y = aki\_hosp))) +  
 geom\_jitter(width = 0,   
 height = 0.2) +  
 labs(x = "[sFas]",  
 y = " ")  
  
ngal\_jitter <-  
 ggplot(data = bio\_clinical\_NA,   
 mapping = (aes(x = log\_ngal, y = aki\_hosp))) +  
 geom\_jitter(width = 0,   
 height = 0.2) +  
 labs(x = "[NGAL]",  
 y = " ")

# creates a 2x2 matrix of the four plots  
figure <- ggarrange(ang2\_jitter, il18\_jitter, ngal\_jitter, sfas\_jitter,  
 ncol = 2,   
 nrow = 2)

## Warning: Removed 50 rows containing missing values (`geom\_point()`).

# annotates the figure with a title and description  
annotate\_figure(figure,  
 top = text\_grob("Biomarker Concentrations and Rates of Incident AKI   
in CHROME Participants",  
 size = 12,  
 face = "bold"),  
 bottom = text\_grob("Concentrations are log transformed and taken from the first visit   
0 = No AKI | 1 = AKI",  
 size = 10,  
 face = "bold"))



ang2\_box <- ggplot(data = bio\_clinical\_NA, aes(x = log\_ang2, y = aki\_hosp)) +  
 geom\_boxplot()+  
 labs(x = "[Ang-2]",  
 y = " ")  
  
il18\_box <- ggplot(data = bio\_clinical\_NA, aes(x = log\_il18, y = aki\_hosp)) +  
 geom\_boxplot()+  
 labs(x = "[IL-18]",  
 y = " ")  
  
sfas\_box <- ggplot(data = bio\_clinical\_NA, aes(x = log\_sfas, y = aki\_hosp)) +  
 geom\_boxplot() +  
 labs(x = "[sFas]",  
 y = " ")  
  
ngal\_box <- ggplot(data = bio\_clinical\_NA, aes(x = log\_ngal, y = aki\_hosp)) +  
 geom\_boxplot()+  
 labs(x = "[NGAL]",  
 y = " ")  
  
figure <- ggarrange(ang2\_box, il18\_box, ngal\_box, sfas\_box,  
 ncol = 2,   
 nrow = 2)

## Warning: Removed 50 rows containing non-finite values (`stat\_boxplot()`).

annotate\_figure(figure,  
 top = text\_grob("Biomarker Concentrations and Rates of Incident AKI   
in CHROME Participants",  
 size = 12,  
 face = "bold"),  
 bottom = text\_grob("Concentrations are log transformed and taken from the first visit   
0 = No AKI | 1 = AKI",  
 size = 10,  
 face = "bold"))

