submission2

Antara Sengupta

2024-08-08

R Markdown

```
# Loading in all required packages
library(readr)
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(ggplot2)
library(tibble)
library(tidyr)
library(dplyr)
```

Including Plots

```
# loading in my cleaned dataset from submission 1
all_data <- read.csv('all_data.csv')
```

Building a function to create the plots I made for Presentation 1, that takes the following input: (1) the name of the data frame, (2) a list of 1 or more gene names, (3) 1 continuous covariate, and (4) two categorical covariates. A lot of this code is re used from the plots I made in submission 1.

```
# Loading necessary libraries
library(ggplot2)
library(dplyr)
library(tidyr)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v purrr     1.0.2
## v lubridate 1.9.2     v stringr   1.5.0
## -- 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 errors
```

```r
# Using rlang package for the sym function that is required to make a function that can take parameters
# properly access the columns
# had issues with this functionality but did research and found the resource below
#https://stackoverflow.com/questions/57136322/what-does-the-operator-mean-in-r-particularly-in-the-cont
library(rlang)
```

```
##
## Attaching package: 'rlang'
##
## The following objects are masked from 'package:purrr':
##
##     %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
##     flatten_raw, invoke, splice
```

```r
# Define the function
# a lot of the code below are plots that I used from submission 1
plot_genes <- function(df, gene_name, continuous_covariate,categorical_covariate1,categorical_covariate

  # plotting histogram of gene expression
  data <- df %>%
    filter(gene == gene_name) %>%
    select(participant_id, gene,expression,
           continuous_covariate, categorical_covariate1,
           categorical_covariate2)  %>%
    mutate(
      expression = as.numeric(expression),
      continuous_covariate = as.numeric(!!sym(continuous_covariate))  # Convert to numeric
    )

  # building histogram to show gene expression
  expression_hist <- ggplot(data, aes(x =as.numeric(expression))) +
    geom_histogram(binwidth = 0.5, color = "black", fill = "blue") +
    labs(title = sprintf("Histogram of Gene Expression for %s", gene_name), # using sprintf to add gene
         x = "Expression",
         y = "Frequency") +
    theme_minimal() #choosing preferred theme
  print(expression_hist)

  # creating a scatterplot that observes expression vs. a continuous covariate
  scatter_plot <- ggplot(data, aes(x = expression, y = continuous_covariate)) +
    geom_point(color = "purple",size = 3)  + #increasing size of the data points on the graph
    geom_smooth(method = "lm", se = FALSE, color = "#33FFF7",linetype = "dashed") +
    labs(title = sprintf("Exploring %s Expression vs. %s in Human Subjects",gene_name, continuous_covari
       x= sprintf("%s Expression", gene_name) ,
       y = continuous_covariate) +
    theme_bw() # changing to preferred theme
  print(scatter_plot)

  # boxplot of gene expression vs. two categorical covariates
  categorical_boxplot<- ggplot(data, aes(x = !!sym(categorical_covariate1), y = as.numeric(expression),
                                         fill = !!sym(categorical_covariate2))) +
    geom_boxplot() +
    scale_x_discrete(labels = c("disease state: COVID-19" = "COVID-19 Positive",
                                "disease state: non-COVID-19" = "COVID-19 Negative"))+
```

```
    # changing the x values to visually cleaner/shorter ones
    labs(title = sprintf("Observing patterns of %s Expression by %s and %s",gene_name,categorical_covari
         x= categorical_covariate1 ,y = sprintf("%s Expression",gene_name), fill = categorical_covariat
    theme_bw() # changing to preferred theme
  print(categorical_boxplot)
}
```

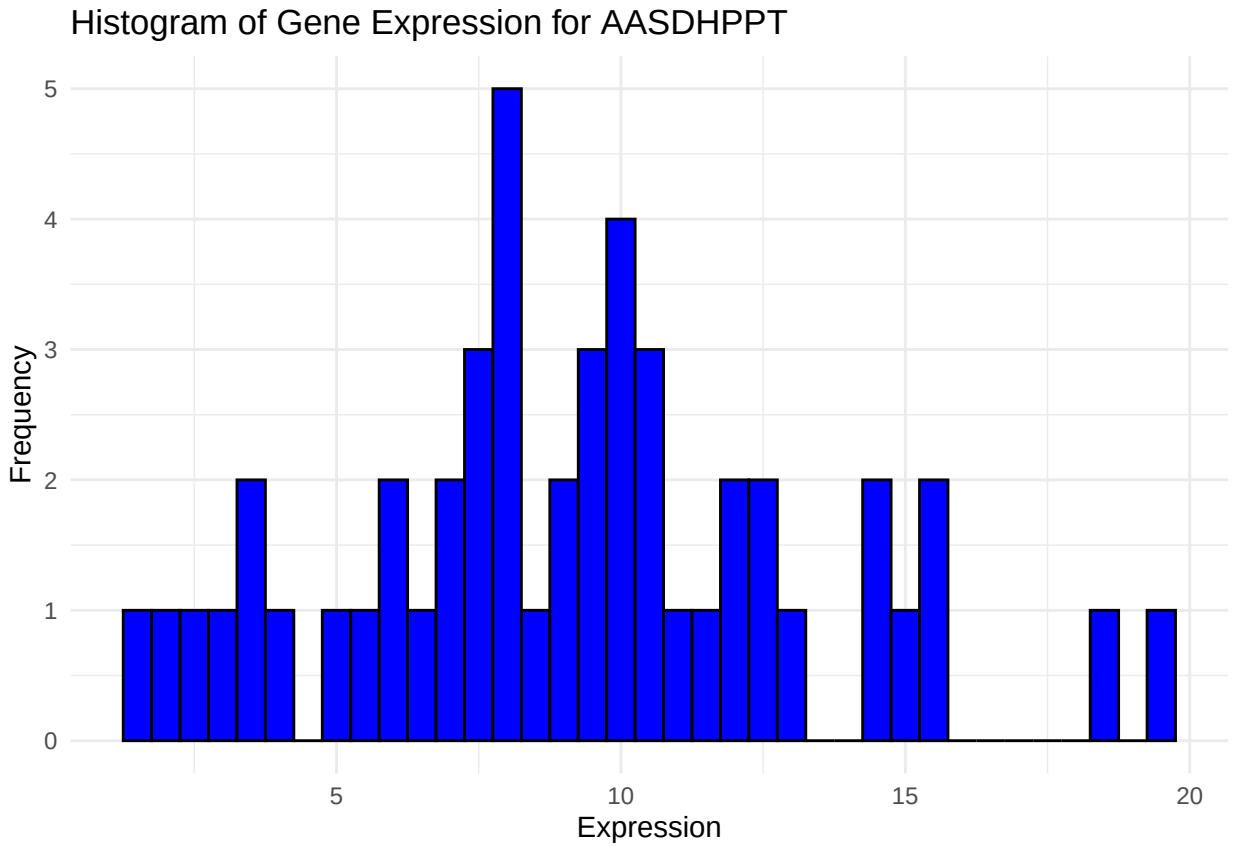
```
# List of genes to analyze
genes_to_analyze <- c("AASDHPPT", "ABCF2-H2BE1", "ABHD17C")  # choosing three genes to run my function

# Choosing the continuous and categorical covariates
continuous_covariate <- "ferritin.ng.ml."
categorical_covariate1 <- "disease_status"
categorical_covariate2 <- "sex"

# Call the function with the data and parameters
for (gene in genes_to_analyze) {
  # looping through all the genes and calling my function
  plot_genes(all_data, gene, continuous_covariate,categorical_covariate1,categorical_covariate2)
}
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(continuous_covariate)
##
##   # Now:
##   data %>% select(all_of(continuous_covariate))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

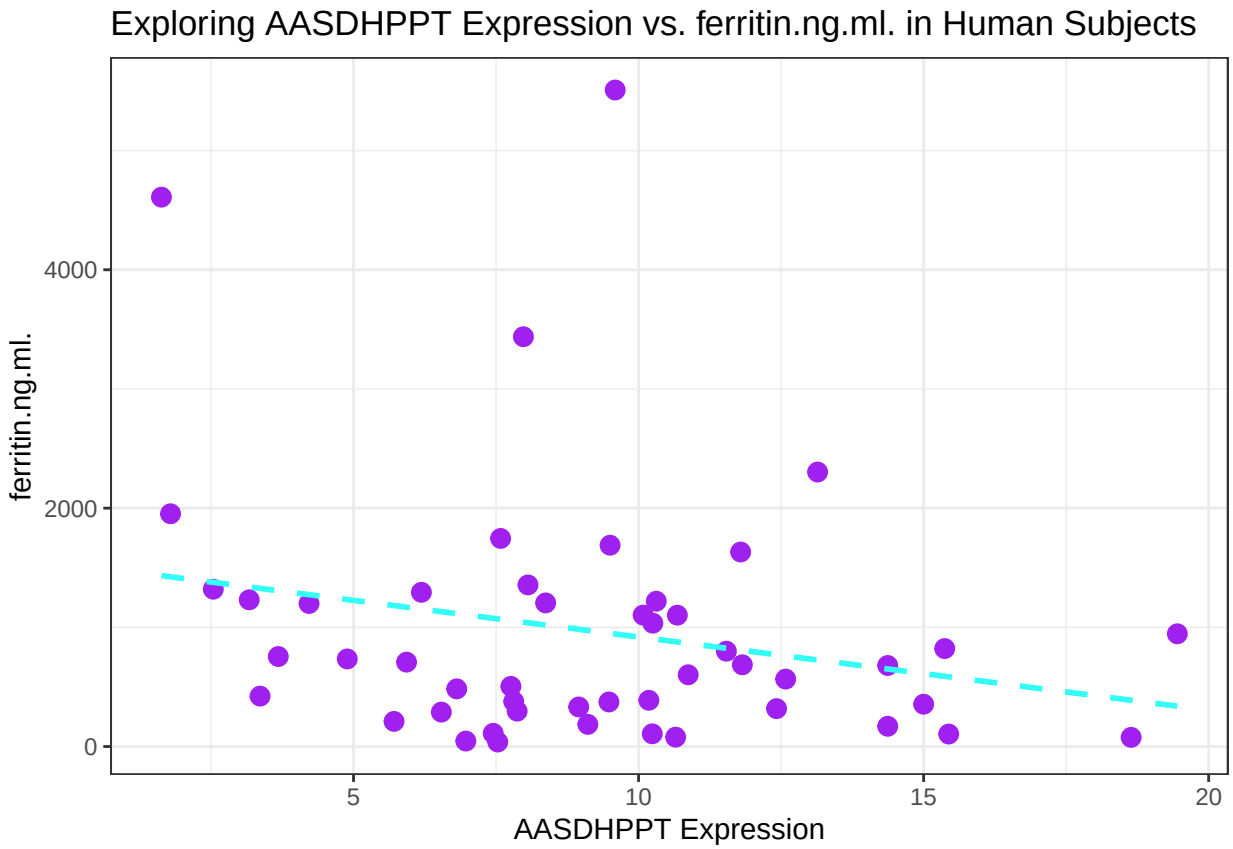
```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(categorical_covariate1)
##
##   # Now:
##   data %>% select(all_of(categorical_covariate1))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

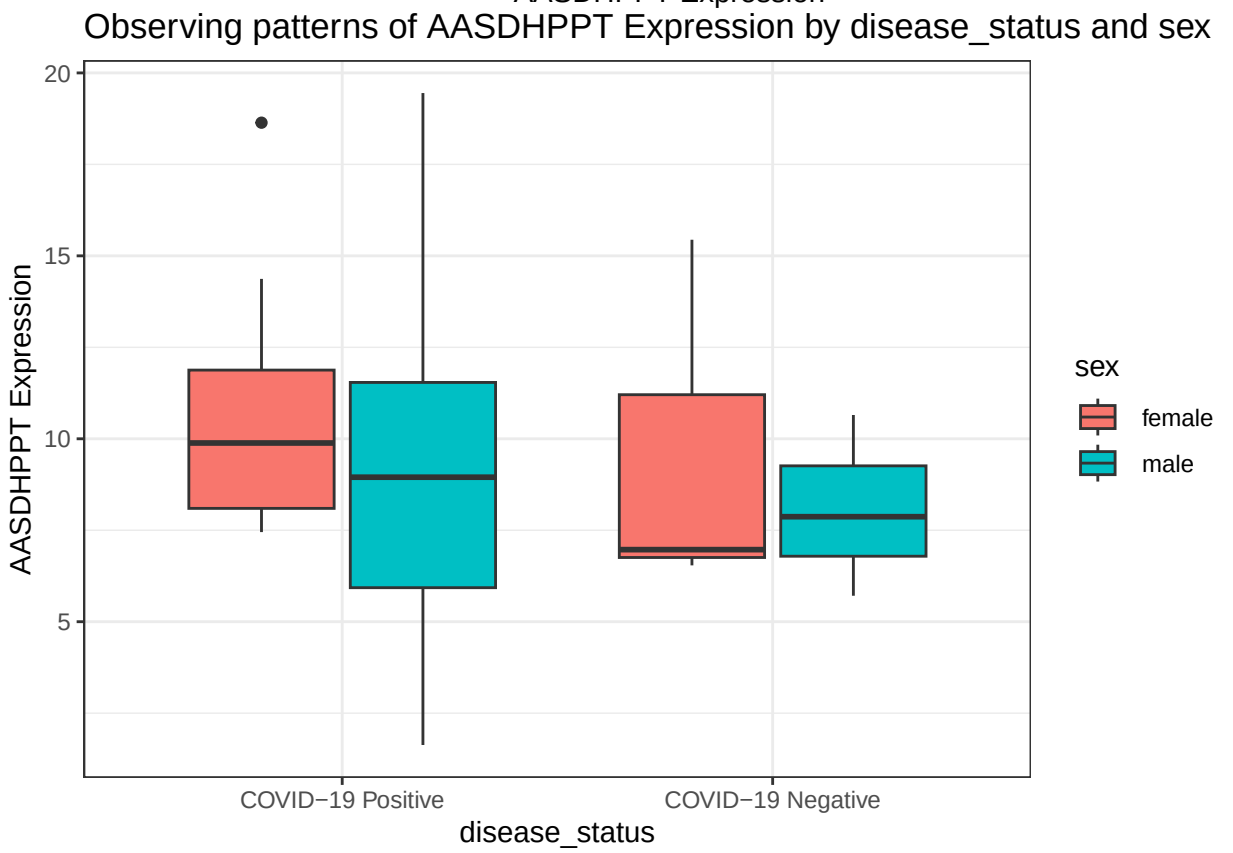
```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(categorical_covariate2)
##
##   # Now:
##   data %>% select(all_of(categorical_covariate2))
##
```

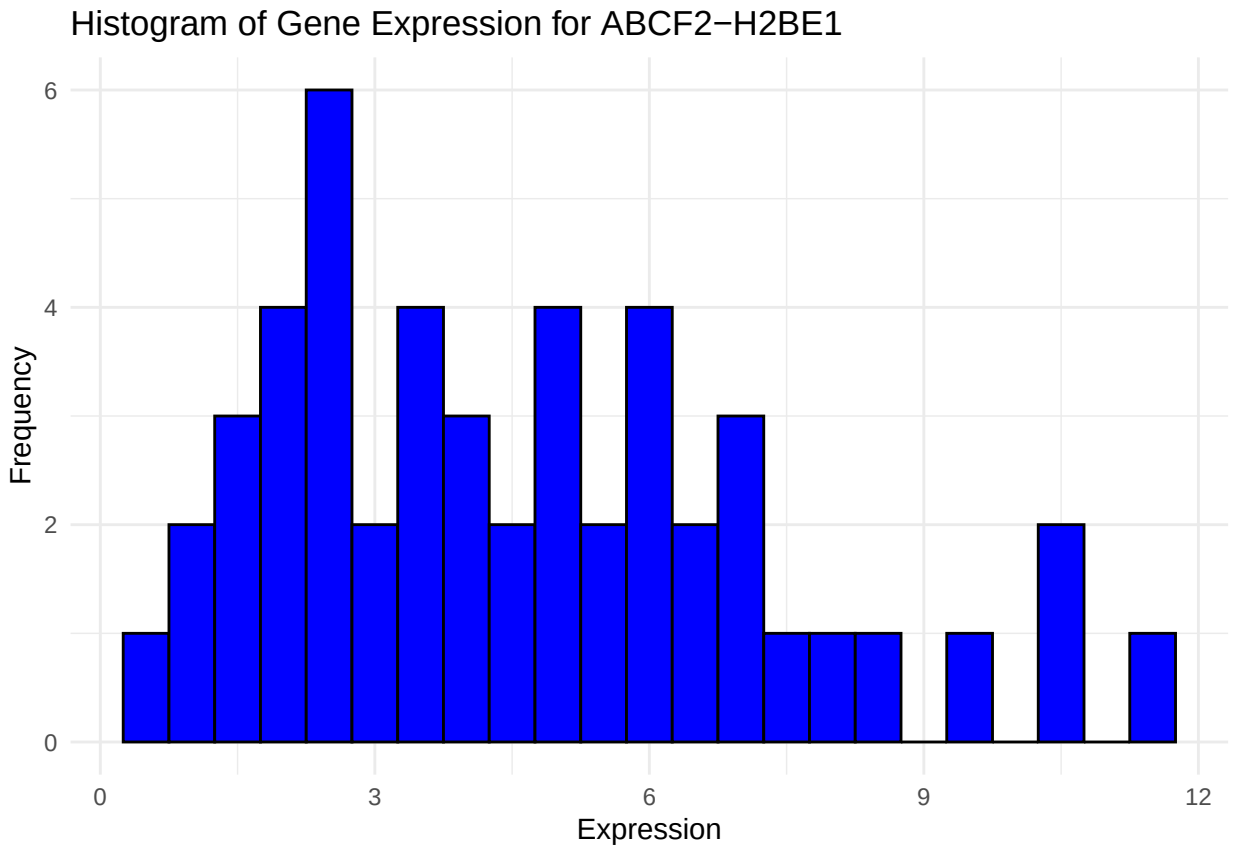
```
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



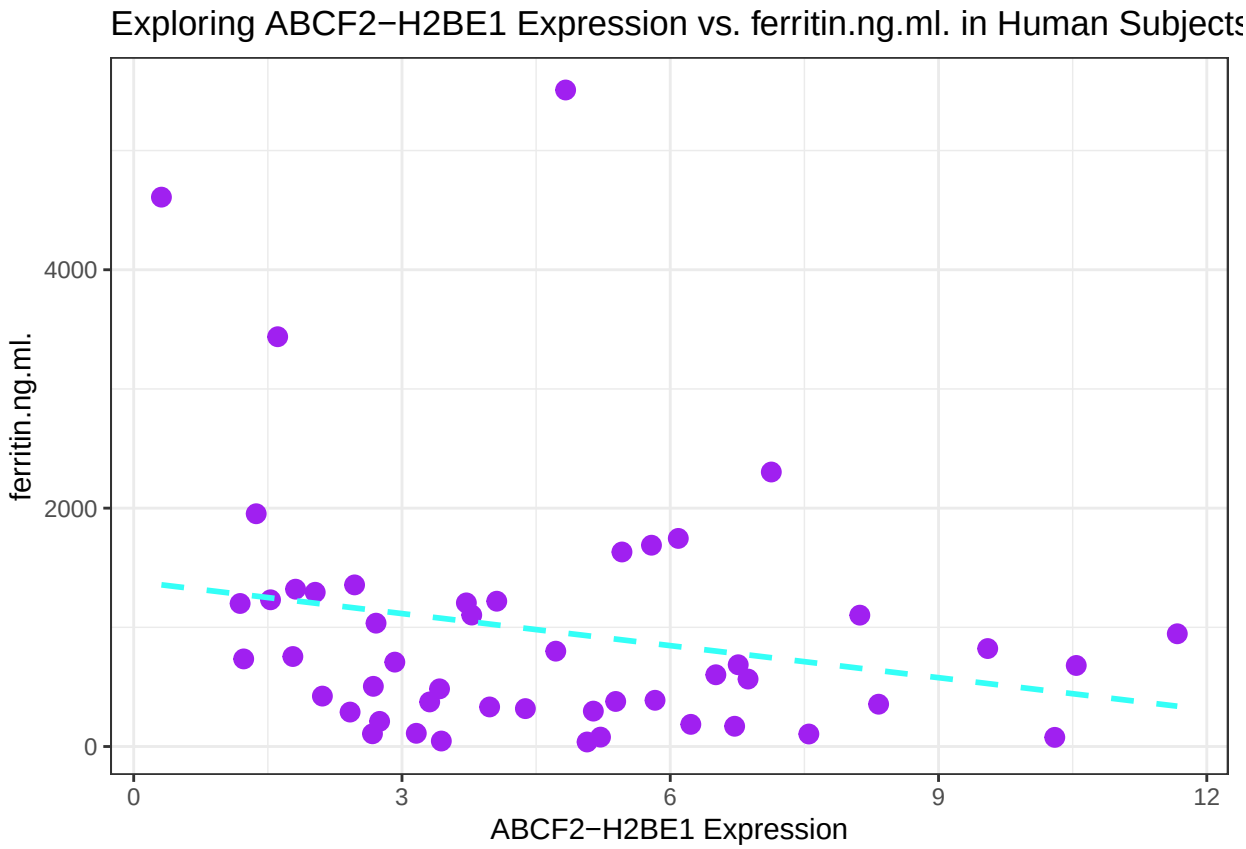

```
## `geom_smooth()` using formula = 'y ~ x'
```

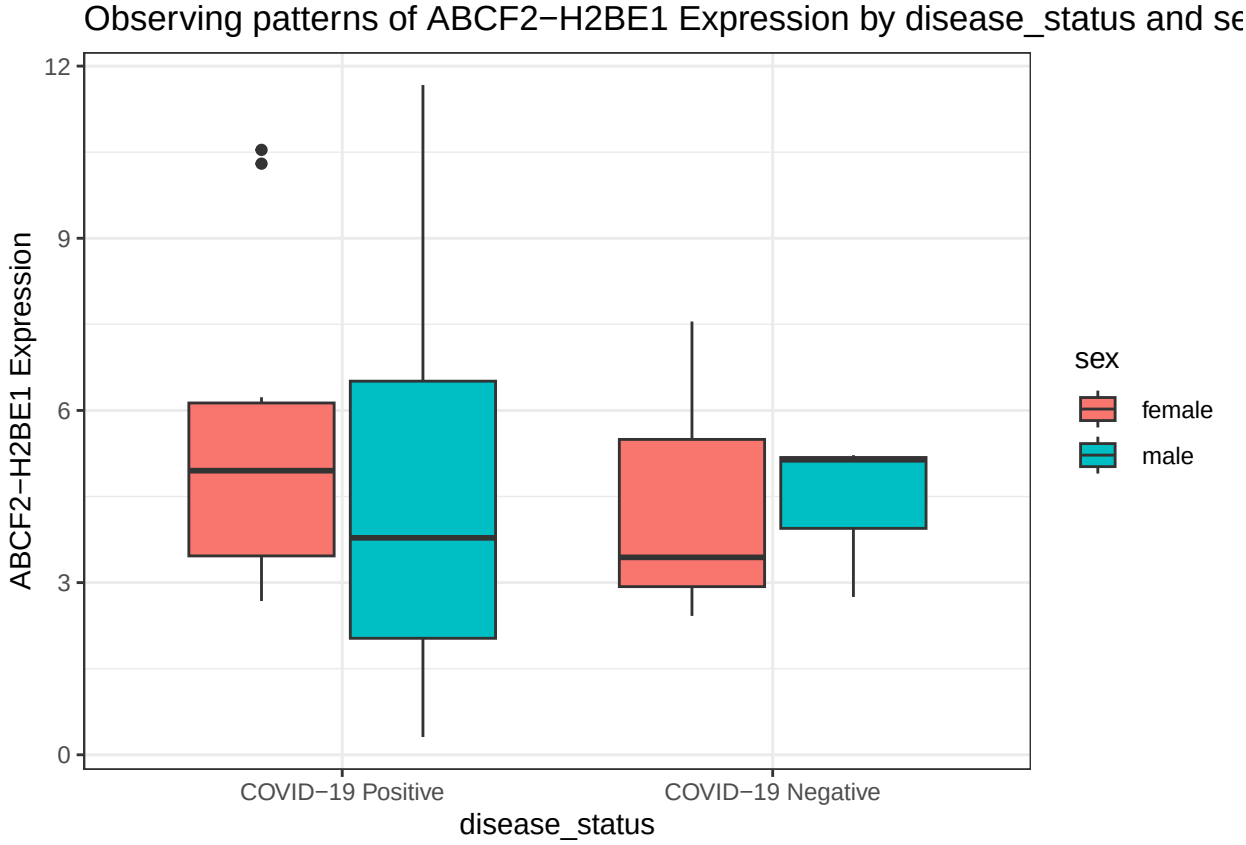

Exploring AASDHPPT Expression vs. ferritin.ng.ml. in Human Subjects
ferritin.ng.ml.
4000
2000
0
5
10
15
20
AASDHPPT Expression

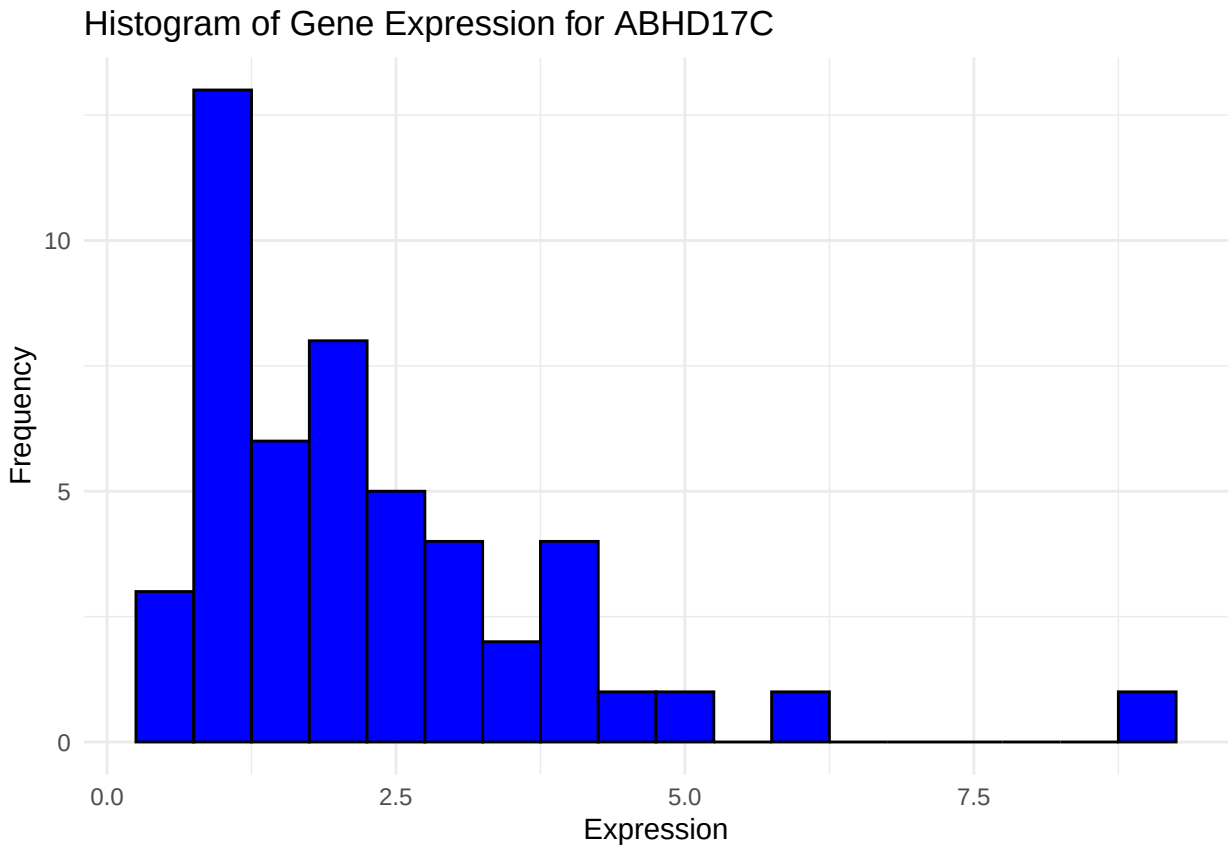

Observing patterns of AASDHPPT Expression by disease_status and sex
AASDHPPT Expression
20
15
10
5
COVID−19 Positive
COVID−19 Negative
disease_status
sex
female
male



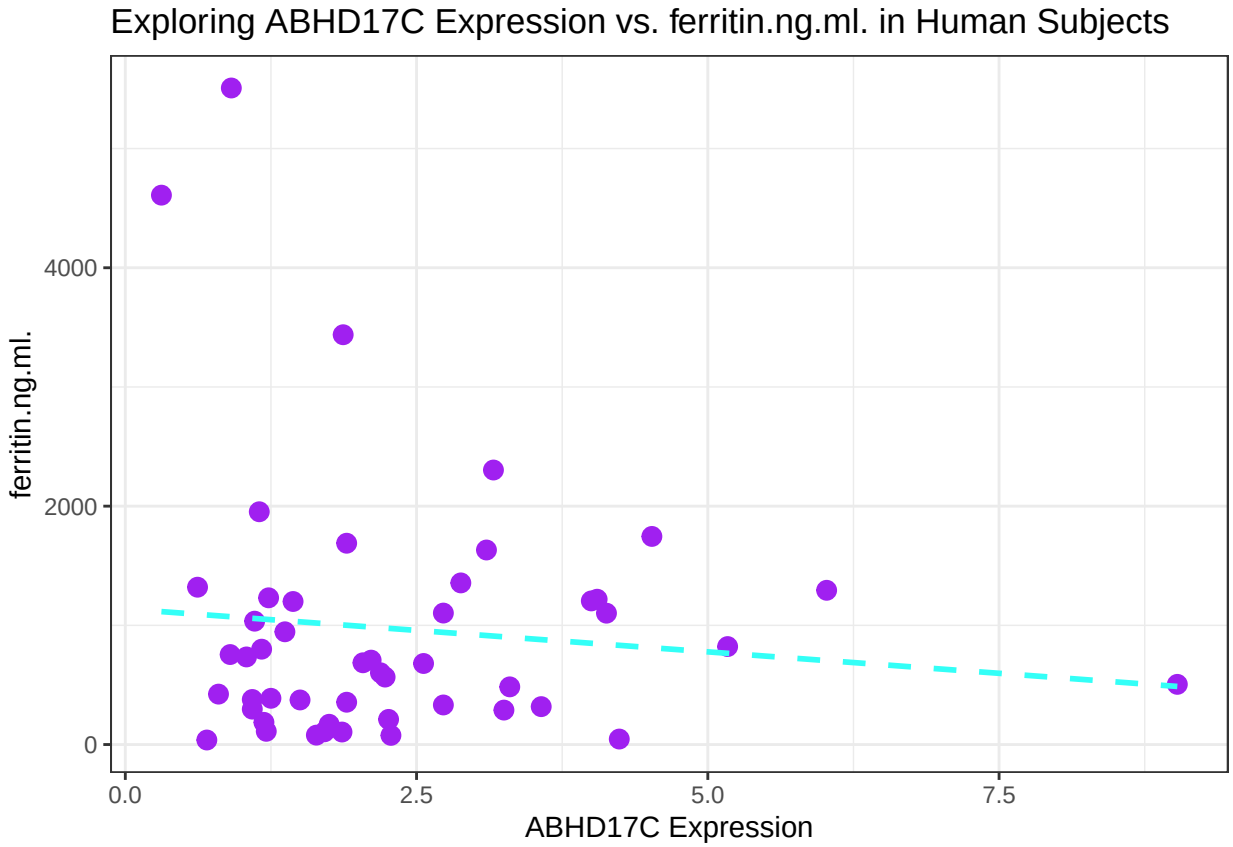

```
## `geom_smooth()` using formula = 'y ~ x'
```

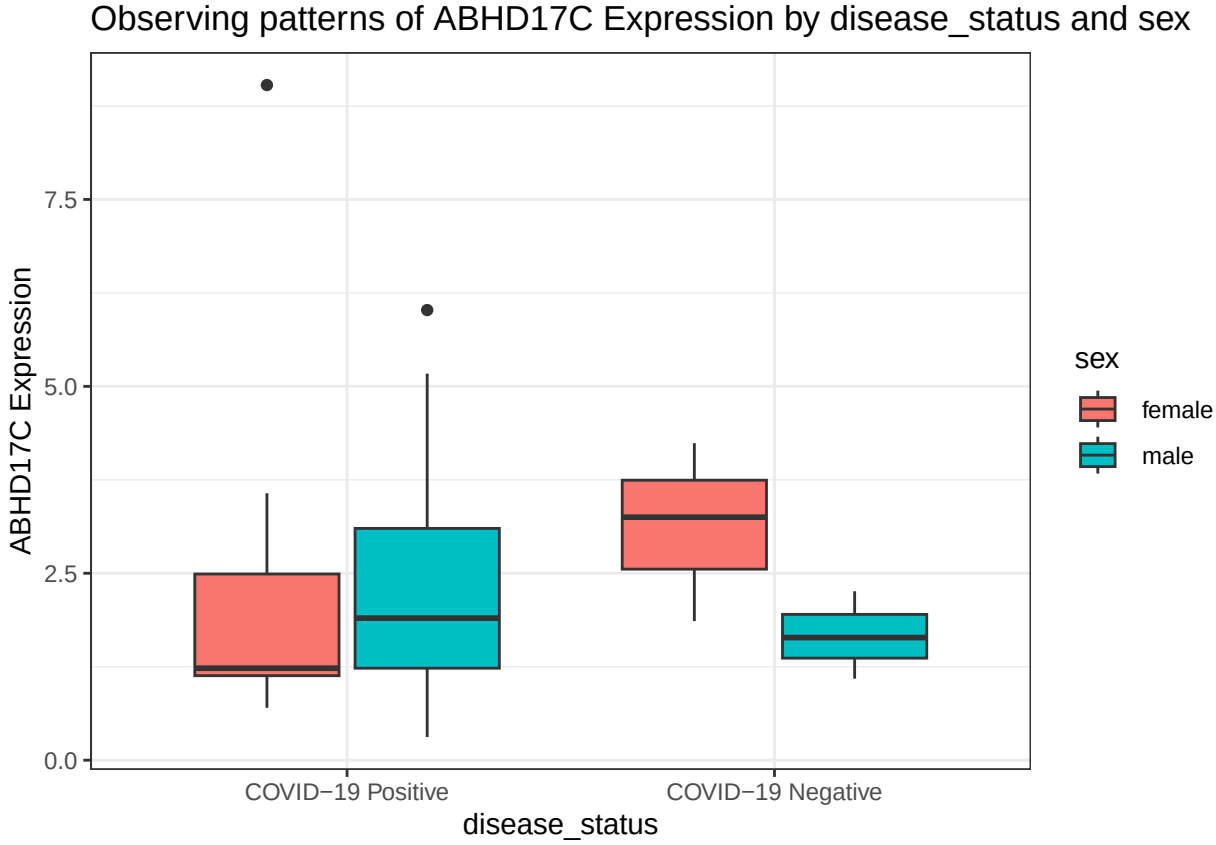

Exploring ABCF2−H2BE1 Expression vs. ferritin.ng.ml. in Human Subjects
ferritin.ng.ml.
4000
2000
0
0
3
6
9
12
ABCF2−H2BE1 Expression


Observing patterns of ABCF2−H2BE1 Expression by disease_status and se
ABCF2−H2BE1 Expression
12
9
6
3
0
COVID−19 Positive
COVID−19 Negative
disease_status
sex
female
male




```
## `geom_smooth()` using formula = 'y ~ x'
```


Exploring ABHD17C Expression vs. ferritin.ng.ml. in Human Subjects
ferritin.ng.ml.
4000
2000
0
0.0
2.5
5.0
7.5
ABHD17C Expression


Observing patterns of ABHD17C Expression by disease_status and sex
ABHD17C Expression
7.5
5.0
2.5
0.0
COVID−19 Positive
COVID−19 Negative
disease_status
sex
female
male