

Scientific Accountability

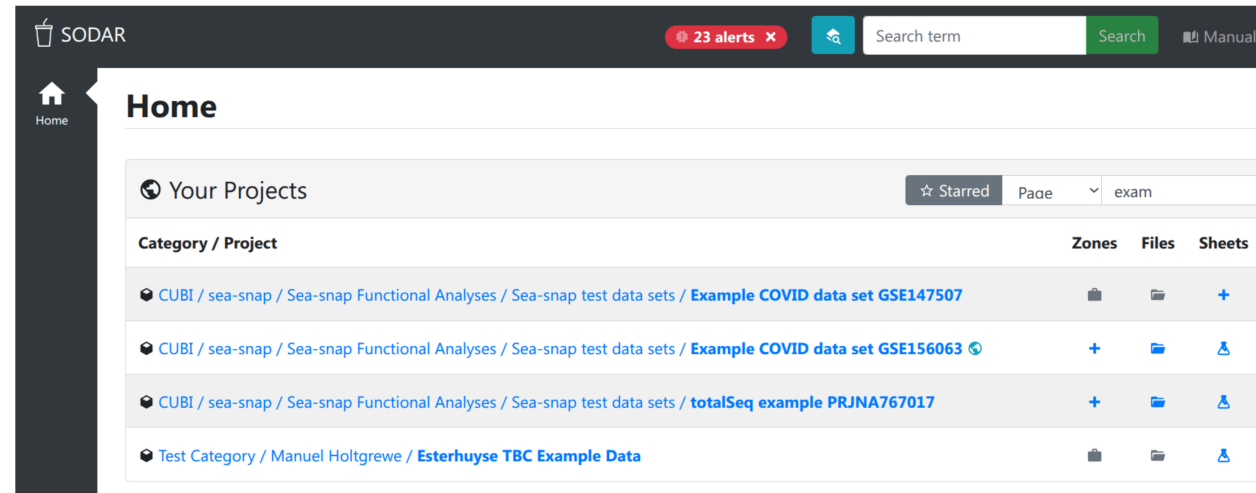
Generating reproducible reports with Shiny apps

January Weiner 

Core Unit for Bioinformatics, BIH@Charité

The CUBI environment

- Clinical and translational research
- Large university hospital and related institutions (BIH, MDC, Charité)



SODAR - platform for managing
multioning study data &
meta-data

The CUBI environment

- Clinical and translational research
- Large university hospital and related institutions (BIH, MDC, Charité)

Home

KIOSC Beta

Home / CUBI / sea-snap / Sea-snap Functional Analyses / Sea-snap test data sets / Example COVID data set GSE156063

Example COVID data set GSE156063 ☆ Example project on Covid

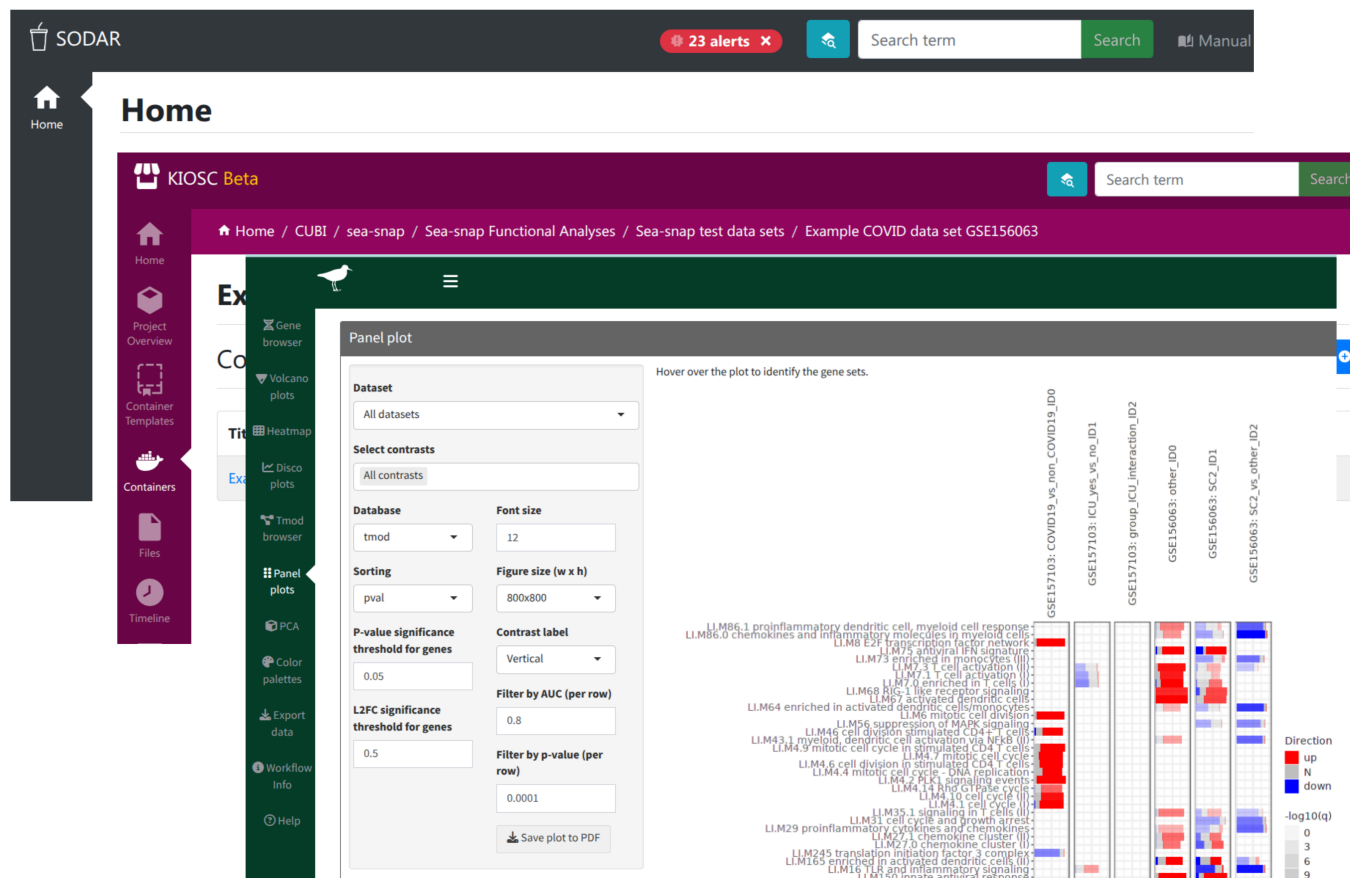
Containers

Title	Repository:Tag	Date Created	State
Example COVID data sets	ghcr.io/bihealth/seapiper:0.6.9	Oct. 5, 2021, 8:28 p.m.	running

kiosc - platform for apps in
docker containers

The CUBI environment

- Clinical and translational research
- Large university hospital and related institutions (BIH, MDC, Charité)



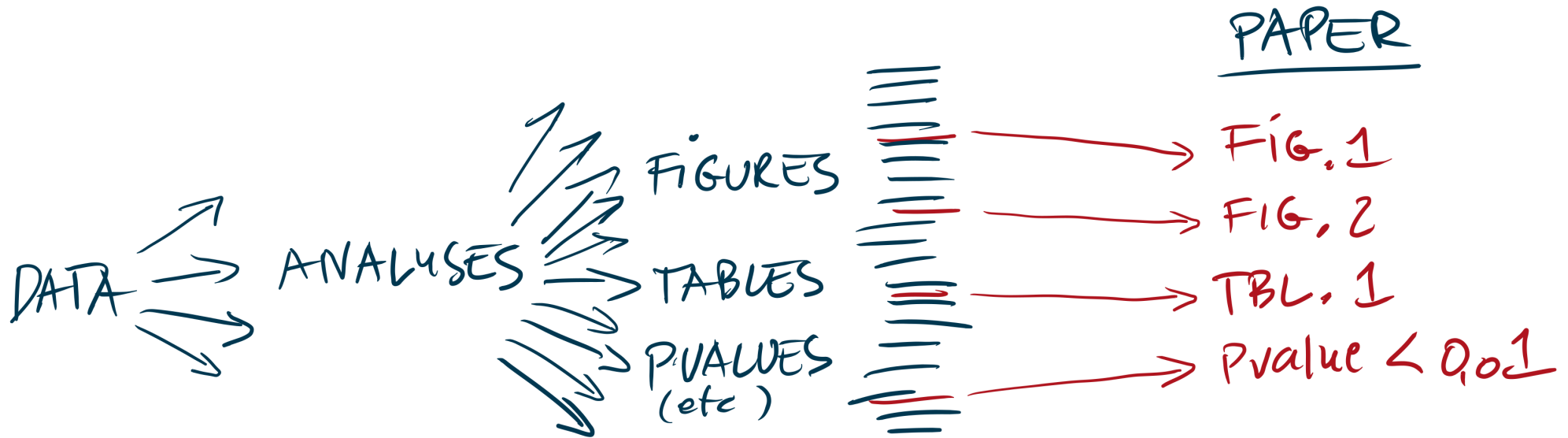
seaPiper - interactive pipeline explorer

Reproducibility vs Accountability



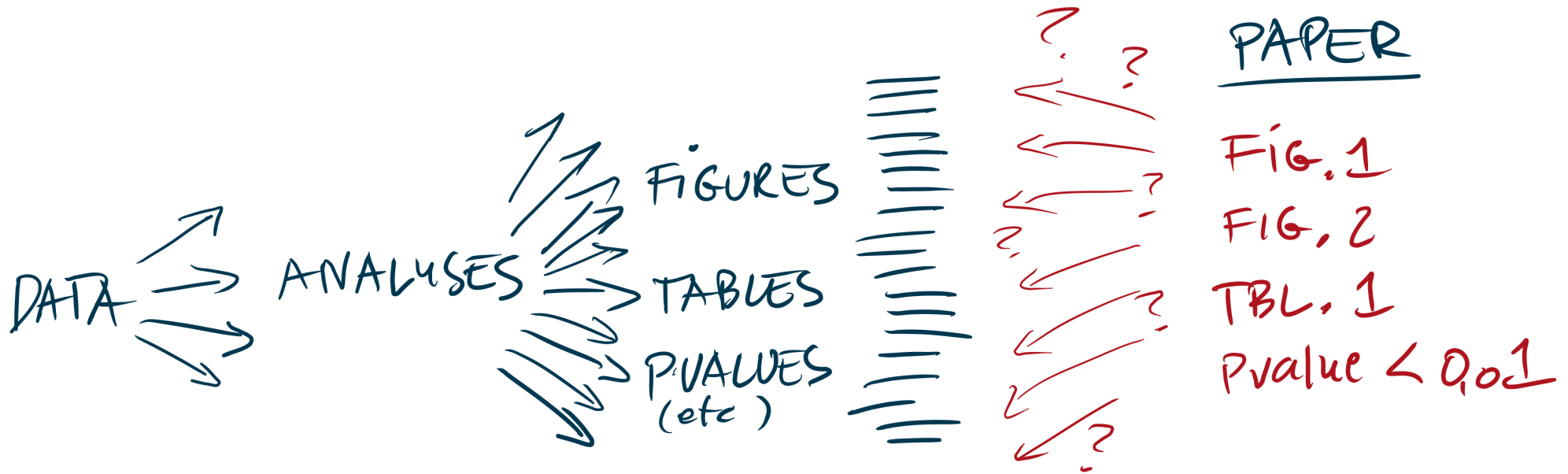
In a high-throughput bioinformatic project, a lot of analyses, models, figures, tables are generated.

Reproducibility vs Accountability



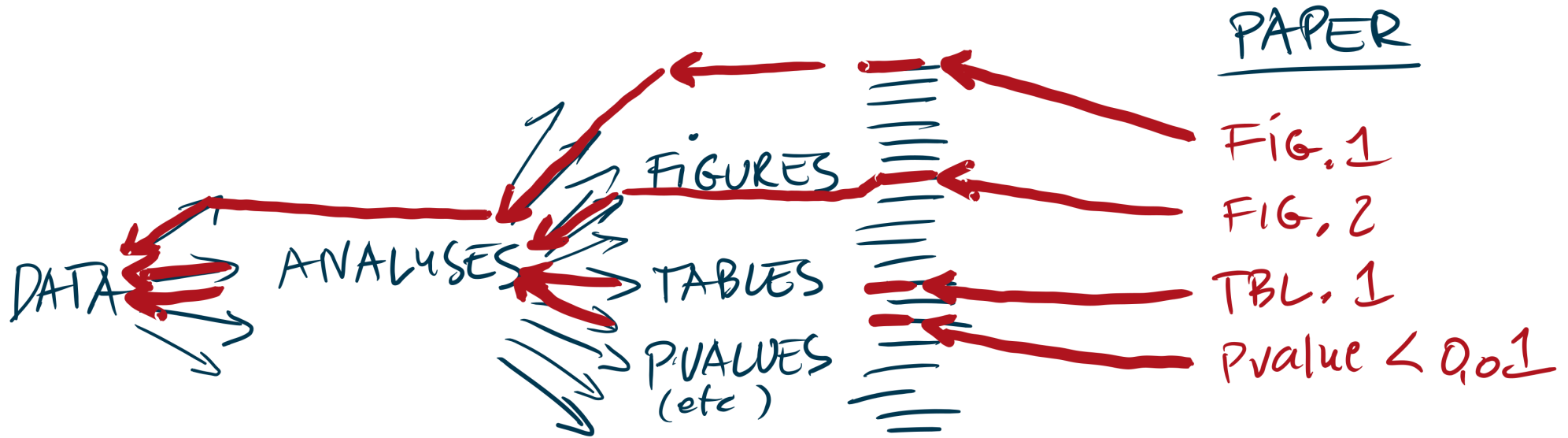
Only some of them make it to the final paper.

Reproducibility vs Accountability



When you read a paper, how do you link the figures back to the data?

Reproducibility vs Accountability



We know how to do it: use R! (and Rmarkdown, Quarto, etc.) In practice, what is reproducible, is often not accountable.

How to turn interactive work into reproducible reports?

The Bioshmods project

```

1 library(bioshmods)
2 data(C19)
3
4 ui <- fluidPage(
5   fluidRow(selectizeInput("id", label="Search for a gene",
6     choices=NULL),
7   fluidRow(geneBrowserPlotUI("gplot", TRUE))
8   ))
9
10 server <- function(input, output, session) {
11   gene_id <- reactiveValues()
12   updateSelectizeInput(session, "id", choices=C19$annotation$SYMBOL)
13
14   # translate symbol to primary ID
15   observeEvent(input$id, {
16     nn <- match(input$id, C19$annotation$SYMBOL)
17     gene_id$id <- C19$annotation$PrimaryID[ nn ]
18   })
19
20   geneBrowserPlotServer("gplot", gene_id=gene_id,
21     covar=C19$covariates,
22     exprs=C19$expression,
23     annot=C19$annotation,
24     cntr=C19$contrasts
25   )
26 }
27 shinyApp(ui, server)

```

The Bioshmods project

dataset
default

covariate
group

covariate
Expression

color by
N/A

symbol by
N/A

Link data points by
N/A

Categorical plot
box

Trellis by
N/A

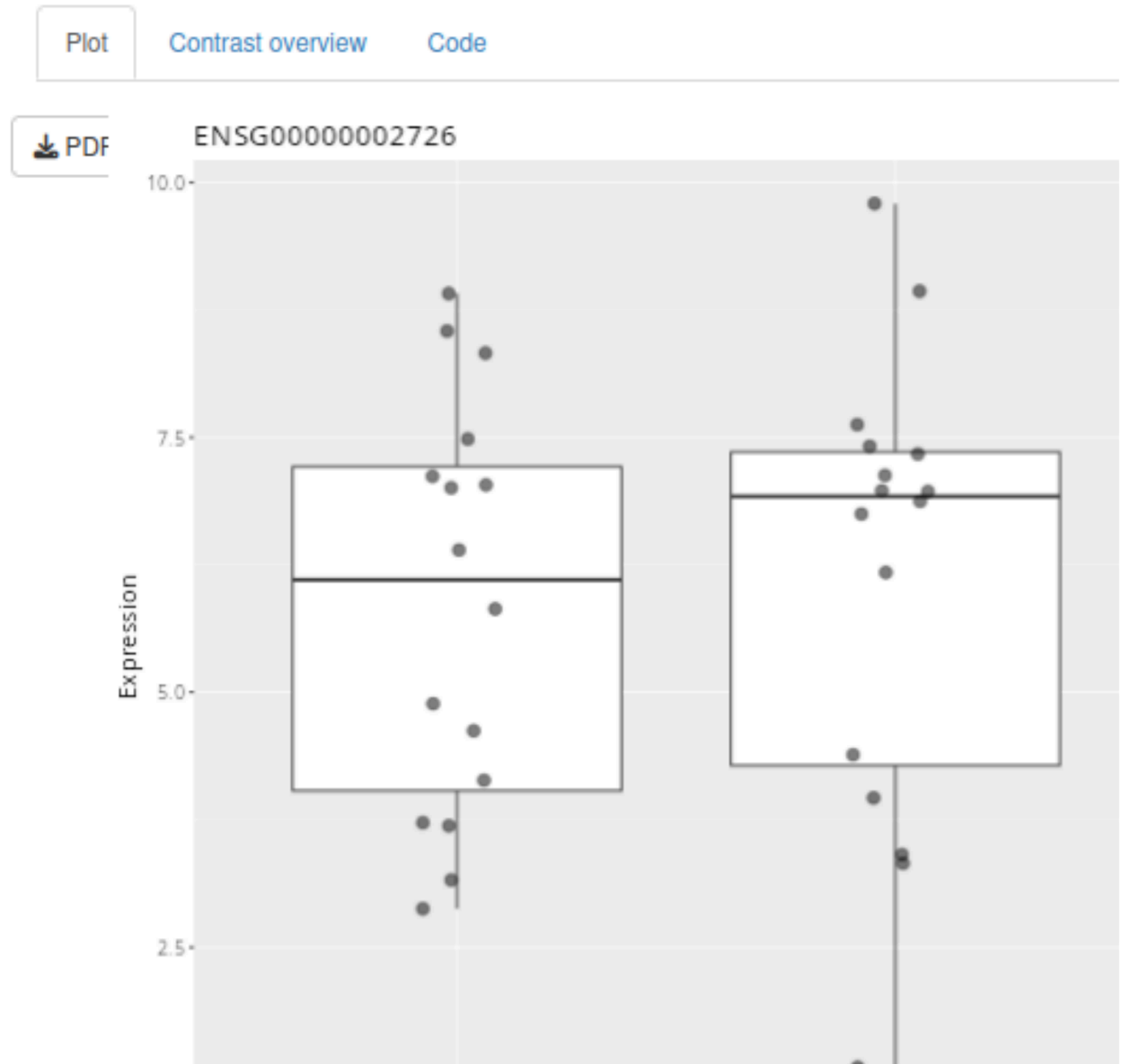
Trellis scales
fixed

Font size
14

☐ Transpose plot

Figure size (w x h)
600x600

Additional info:



The Bioshmods project

dataset

default ▼

covariate

group ▼

Link data points by

N/A ▼

covariate

Expression ▼

Categorical plot

box ▼

color by

N/A ▼

Trellis by

N/A ▼

symbol by

N/A ▼

Trellis scales

fixed ▼

Font size

14 ▼

☐ Transpose plot

Figure size (w x h)

600x600 ▼

Additional info:

Plot

Contrast overview

Code

```
library(ggplot2)

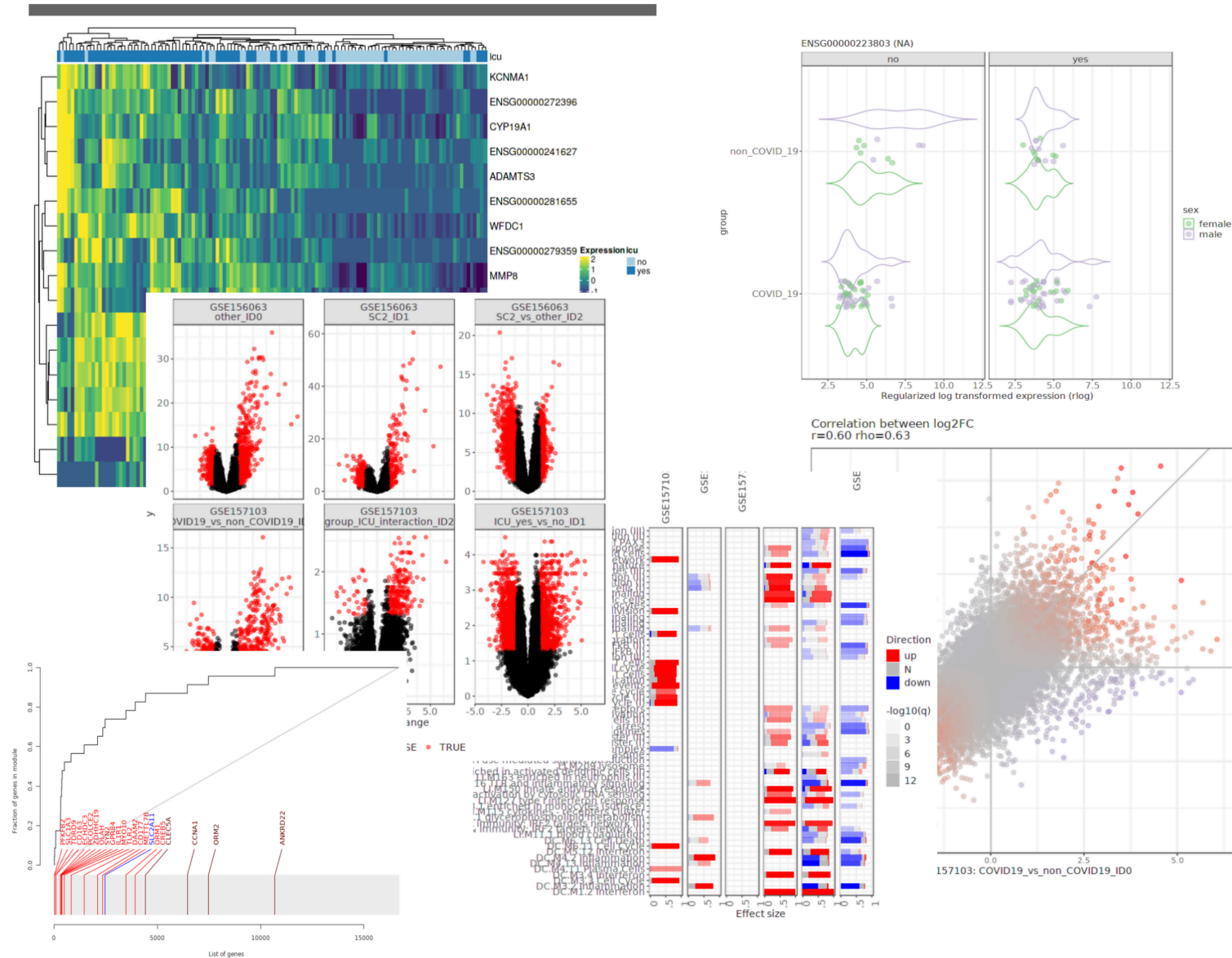
# Build the plotting data frame.
df <- data.frame(covar, Expression=exprs[id, ], check.names=FALSE)
colnames(df)[ncol(df)] <- expressionLabel

title <- sprintf("%s (%s)", id, annot[match(id, annot[[annot_id_col]]), ][[annot_symb_col]])

g <- ggplot(df, ggplot2::aes(x=.data[[xCovar]], y=.data[[yCovar]],
                           group=.data[[groupBy]], color=.data[[colorBy]])) +
  geom_boxplot(outlier.shape=NA) + ggplot2::geom_jitter(size=3, alpha=.5, width=.1) +
  geom_point(size=3) +
  ggtitle(title)

g
```


The Bioshmods project



Thank you

Links

- github.com/january3/useR2026
- github.com/bihealth/bioshmods

I'd love to hear your feedback!

