Ziva Analysis

Gabrielle Kosoy

6/7/2021

1. First you will import the libraries you need

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(drc)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

##
## 'drc' has been loaded.

## Please cite R and 'drc' if used for a publication,

## for references type 'citation()' and 'citation('drc')'.

##
## Attaching package: 'drc'

## The following objects are masked from 'package:stats':
##
##     gaussian, getInitial
```

```
library(readxl)
```

2. Next you will set your working directory and import the files you will use from it. Examples of how to write out your file names are in parentheses. Note the use of "", .csv, .xlsx, and the direction of slashes / .

```
# working file location
setwd("C:/Users/15183/Documents/Ziva data/Feb_8_2021_141_Day0,28")

# open Ziva data file, and choose 60 or 140ms exposure data
Ziva_data <- read.csv(file = "segresults_BY_IMAGE_SEGMENT_419.csv") %>%
    filter(Exposure_MS == 60)

# make excel file of your dilution/concentration information and read
# here
dilutions_1 <- read_xlsx("doses_141_day0.xlsx")

dilutions_2 <- read_xlsx("doses_141_day28.xlsx")
```

3. Run a function named "Filetered data" that will reduce your data to what you want to investigate. Then note the example to run this function for a sample serum for Day 0 and Day 28 for the SARS-CoV-2 protein. This function takes the inputs, data file (df), chips to analyze (well_names), protein to analyze (protein_name)

```
filtered_data <- function(df, well_names, protein_name) {
    processed_df <- df %>%
        filter(Well_Name %in% well_names) %>%
        filter(Protein_Name == protein_name) %>%
        return(processed_df)
}
```

1's and 2's are Day 0 replicates, 3's and 4's are Day 28 replicates. Chip "A" is my blank chip. Also run CytC for day 0 and day 28 as your internal control for non-specific binding. You will notice that D4 was omitted, you can omit chips that did not run properly on the machine by not including them for analysis.

```
RBD_Day0 <- filtered_data(Ziva_data, list("B1", "C1", "D1", "E1", "F1",
    "G1", "H1", "A1", "B2", "C2", "D2", "E2", "F2", "G2", "H2", "A2"),
    "SARS-CoV-2 RBD")
RBD_Day28 <- filtered_data(Ziva_data, list("B3", "C3", "D3", "E3", "F3",
    "G3", "H3", "A3", "B4", "C4", "E4", "F4", "G4", "H4", "A4"), "SARS-CoV-2 RBD")
CytC_Day0 <- filtered_data(Ziva_data, list("B1", "C1", "D1", "E1", "F1",
    "G1", "H1", "A1", "B2", "C2", "D2", "E2", "F2", "G2", "H2", "A2"),
    "cytC")
CytC_Day28 <- filtered_data(Ziva_data, list("B3", "C3", "D3", "E3", "F3",
    "G3", "H3", "A3", "B4", "C4", "E4", "F4", "G4", "H4", "A4"), "cytC")
```

4. Now we will get mean, and standard deviation information to make our graphs. We will run a function called "info by protein," it will create a table with the values you want to plot. We will use this function to get information about our reduced data for just SARS-CoV-2 RBD and CytC.

```
info_by_protein <- function(df, dilution) {
    df_info <- df %>%
        group_by(Well_Name) %>%
        summarize(mean_thickness = mean(Raw_Thickness), std_thickness = sd(Raw_Thickness),
            count = n(), sem_thickness = std_thickness/(sqrt(count)), .groups = "drop") %>%
        as.data.frame() %>%
        mutate(dose = dilution) %>%
        return(df_info)
}
#-8 is to excluse chip D4 from your dilutions
graph_RBD_1 <- info_by_protein(RBD_Day0, dilutions_1$`SARS2 RBD`)
graph_RBD_2 <- info_by_protein(RBD_Day28, dilutions_2$`SARS2 RBD`[-8])
info_cytC_1 <- info_by_protein(CytC_Day0, dilutions_1$`SARS2 RBD`)
info_cytC_2 <- info_by_protein(CytC_Day28, dilutions_2$`SARS2 RBD`[-8])
```

5. Now we will use our CytC control to correct the mean and standard deviation for non-specific binding. We will insert this new information into new columns in our table, column seven "V7" will be our new mean, column "V8" will be our new standard deviation.

```
graph_RBD_1[, 7] <- graph_RBD_1$mean_thickness - info_cytC_1$mean_thickness
graph_RBD_1[, 8] <- sqrt(((graph_RBD_1$std_thickness)^2) + ((info_cytC_1$std_thickness)^2))
graph_RBD_2[, 7] <- graph_RBD_2$mean_thickness - info_cytC_2$mean_thickness
graph_RBD_2[, 8] <- sqrt(((graph_RBD_2$std_thickness)^2) + ((info_cytC_2$std_thickness)^2))
```

6. At this point you can look at your graph_RBD_1 file, or whatever you decided to name your table. If the points in V7 first decrease, then start increasing again (this is most likely to happen at low concentrations), this means your data points are on the left side of the curve. For more information about the left/right curve, see the intensity vs thickness plot. To correct for this you will need to switch the sign in front of those points to tell R which side of the curve they are on. The following code does this.

```
graph_RBD_1$V7[1:2] <- graph_RBD_1$V7[1:2] * -1  #changes your control (your 'A's')
graph_RBD_2$V7[1:2] <- graph_RBD_2$V7[1:2] * -1
# you can choose how many points to change in the square brackets
graph_RBD_1$V7[7:16] <- graph_RBD_1$V7[7:16] * -1
graph_RBD_2$V7[12:15] <- graph_RBD_2$V7[12:15] * -1
```

7. Now you want to get parameters to build a four parameter logistic from your data for Day 0 and Day 28. This code will automatically get you the parameters that R thinks is best.

```
RBD.m1 <- drm(V7 ~ dose, data = graph_RBD_1, fct = LL.4(fixed = c(NA, NA,
    NA, NA)))
RBD.m2 <- drm(V7 ~ dose, data = graph_RBD_2, fct = LL.4(fixed = c(NA, NA,
    NA, NA)))
parametersRBD_1 <- RBD.m1$coefficients
b = round(parametersRBD_1[1], digits = 2)
c = round(parametersRBD_1[2], digits = 2)
d = round(parametersRBD_1[3], digits = 2)
e = round(parametersRBD_1[4], digits = 2)
parametersRBD_2 <- RBD.m2$coefficients
b2 = round(parametersRBD_2[1], digits = 2)
```

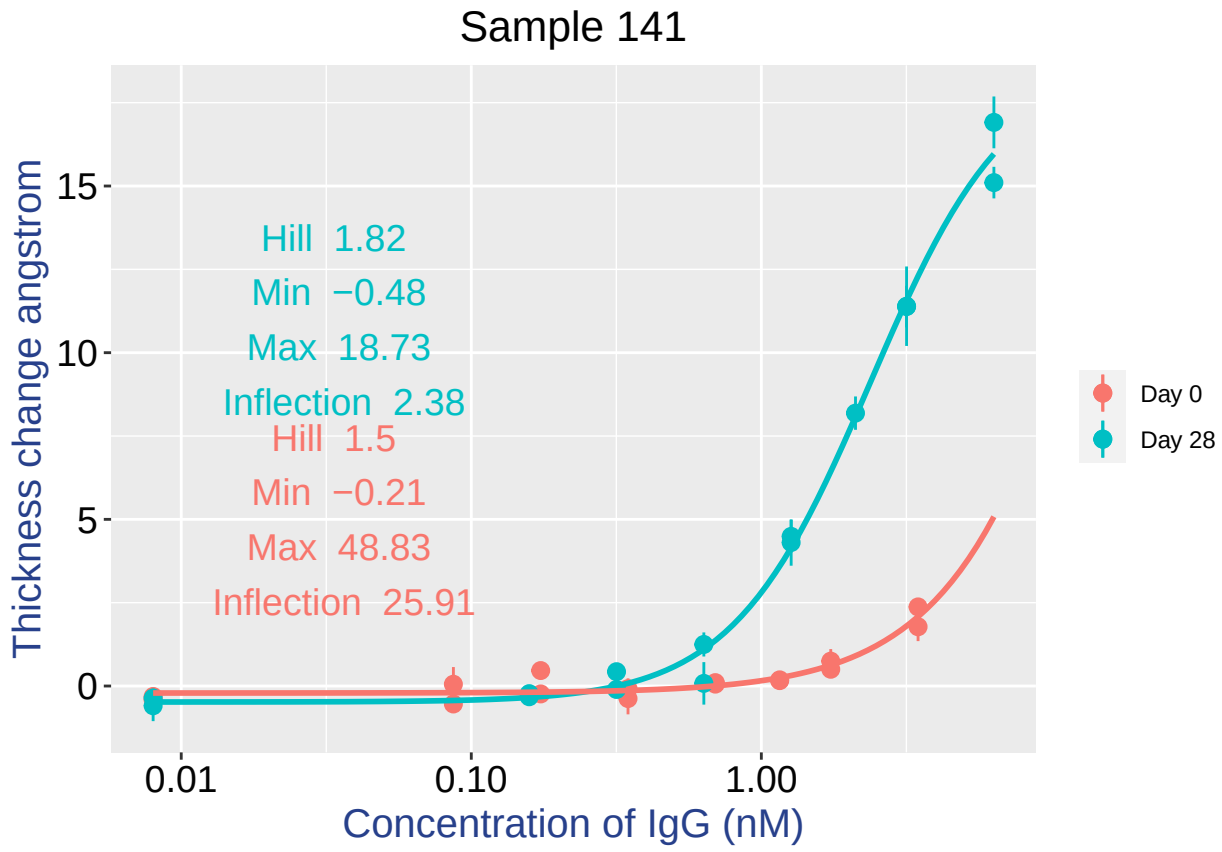
```
c2 = round(parametersRBD_2[2], digits = 2)
d2 = round(parametersRBD_2[3], digits = 2)
e2 = round(parametersRBD_2[4], digits = 2)
```

If you want to change the parameters to better fit the line, you can adjust like seen in the example below. Note that you set a value in the m1/m2 line, and you adjust that parameter in the code below it as well. Note that you need to change the value in square brackets to match which parameter it is, the way e= and e2= becomes parameter[3] if d and d2 are set by you. For the sake of this document, there are # before these lines so they won't run.

```
# RBD.m1 <- drm(V7 ~ dose, data = graph_RBD_1, fct = LL.4(fixed =
# c(NA,NA,45,NA))) RBD.m2 <- drm(V7 ~ dose, data = graph_RBD_2, fct =
# LL.4(fixed = c(NA,NA,45,NA))) parametersRBD_1 <-
# RBD.m1$coefficients; b = round(parametersRBD_1[1], digits = 2); c =
# round(parametersRBD_1[2], 2); d = 45; e = round(parametersRBD_1[3],
# digits = 2) parametersRBD_2 <- RBD.m2$coefficients; b2 =
# round(parametersRBD_2[1], digits = 2); c2 =
# round(parametersRBD_2[2], 2); d2 = 45; e2 =
# round(parametersRBD_2[3], digits = 2)
```

8. Finally! You can graph the corrected mean and standard deviation for your serum at Day 0 and Day 28 of your protein of choice (example here has been for SARS-CoV-2 RBD). You can change the title of the graph next to ggtitle where it says "Sample 141". In the same line you can change your axes titles. In annotate you can adjust the location of your parameters written in your graph, which will change based on the maximum y data point.

```
ggplot(data = NULL) + ggtitle(paste("Sample 141")) + xlab("Concentration of IgG (nM)") +
    ylab("Thickness change angstrom") + theme(plot.title = element_text(hjust = 0.5)) +
    theme(plot.title = element_text(size = 16)) + geom_pointrange(size = 0.5,
    data = graph_RBD_1, mapping = aes(color = "Day 0", x = dilutions_1$`SARS2 RBD`,
        y = V7, ymin = V7 - V8, ymax = V7 + V8)) + geom_pointrange(size = 0.5,
    data = graph_RBD_2, mapping = aes(color = "Day 28", x = dilutions_2$`SARS2 RBD`[-8],
        y = V7, ymin = V7 - V8, ymax = V7 + V8)) + scale_x_log10() + theme(legend.title = element_blank
    theme(axis.title.x = element_text(colour = "royalblue4", size = 15)) +
    theme(axis.title.y = element_text(colour = "royalblue4", size = 15)) +
    theme(axis.text.x = element_text(colour = "grey1", size = 14)) + theme(axis.text.y = element_text(c
    size = 14)) + geom_function(fun = function(x) c2 + ((d2 - c2)/abs(1 +
    ((x/e2)^b2))), colour = "#00BFC4", size = 1) + geom_function(fun = function(x) c +
    ((d - c)/abs(1 + ((x/e)^b))), colour = "#F8766D", size = 1) + annotate(colour = "#00BFC4",
    "text", size = 5, x = 0.035, y = 11, label = paste("Hill ", -b2, "\n",
        "Min ", c2, "\n", "Max ", d2, "\n", "Inflection ", e2)) + annotate(colour = "#F8766D",
    "text", size = 5, x = 0.035, y = 5, label = paste("Hill ", -b, "\n",
        "Min ", c, "\n", "Max ", d, "\n", "Inflection ", e))
```


Sample 141
Thickness change angstrom
15
10
5
0
Hill 1.82
Min −0.48
Max 18.73
Inflection 2.38
Hill 1.5
Min −0.21
Max 48.83
Inflection 25.91
0.01
0.10
1.00
Concentration of IgG (nM)
Day 0
Day 28