19: Mini Project: Investigating Pertussis Resurgence

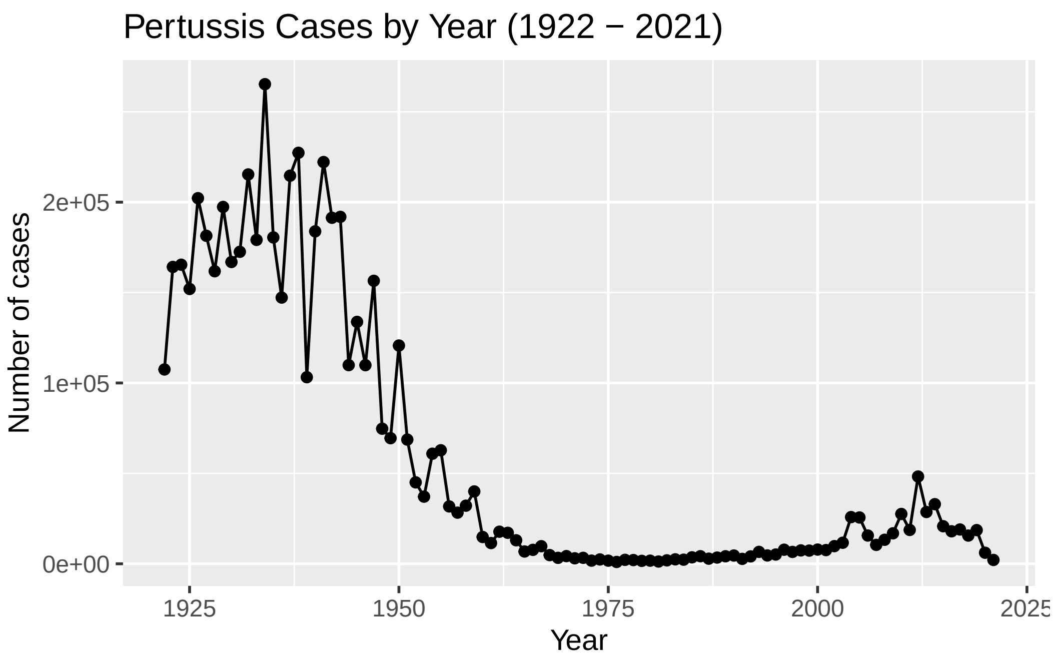
Nicolò (PID: 18109144)

1. Investigating pertussis cases by year

> Q1. With the help of the R "addin" package datapasta assign the CDC pertussis case number data to a data frame called cdc and use ggplot to make a plot of cases numbers over time.

```
cdc <- data.frame(
                                 Year = c(1922L,1923L,1924L,1925L,
                                          1926L,1927L,1928L,1929L,1930L,1931L,
                                          1932L,1933L,1934L,1935L,1936L,
                                          1937L,1938L,1939L,1940L,1941L,1942L,
                                          1943L,1944L,1945L,1946L,1947L,
                                          1948L,1949L,1950L,1951L,1952L,
                                          1953L,1954L,1955L,1956L,1957L,1958L,
                                          1959L,1960L,1961L,1962L,1963L,
                                          1964L,1965L,1966L,1967L,1968L,1969L,
                                          1970L,1971L,1972L,1973L,1974L,
                                          1975L,1976L,1977L,1978L,1979L,1980L,
                                          1981L,1982L,1983L,1984L,1985L,
                                          1986L,1987L,1988L,1989L,1990L,
                                          1991L,1992L,1993L,1994L,1995L,1996L,
                                          1997L,1998L,1999L,2000L,2001L,
                                          2002L,2003L,2004L,2005L,2006L,2007L,
                                          2008L,2009L,2010L,2011L,2012L,
                                          2013L,2014L,2015L,2016L,2017L,2018L,
                                          2019L,2020L,2021L),
         No..Reported.Pertussis.Cases = c(107473,164191,165418,152003,
                                          202210,181411,161799,197371,
                                          166914,172559,215343,179135,265269,
                                          180518,147237,214652,227319,103188,
                                          183866,222202,191383,191890,109873,
                                          133792,109860,156517,74715,69479,
                                          120718,68687,45030,37129,60886,
                                          62786,31732,28295,32148,40005,
                                          14809,11468,17749,17135,13005,6799,
                                          7717,9718,4810,3285,4249,3036,
                                          3287,1759,2402,1738,1010,2177,2063,
                                          1623,1730,1248,1895,2463,2276,
                                          3589,4195,2823,3450,4157,4570,
                                          2719,4083,6586,4617,5137,7796,6564,
                                          7405,7298,7867,7580,9771,11647,
                                          25827,25616,15632,10454,13278,
                                          16858,27550,18719,48277,28639,32971,
                                          20762,17972,18975,15609,18617,
                                          6124,2116)
       )
```

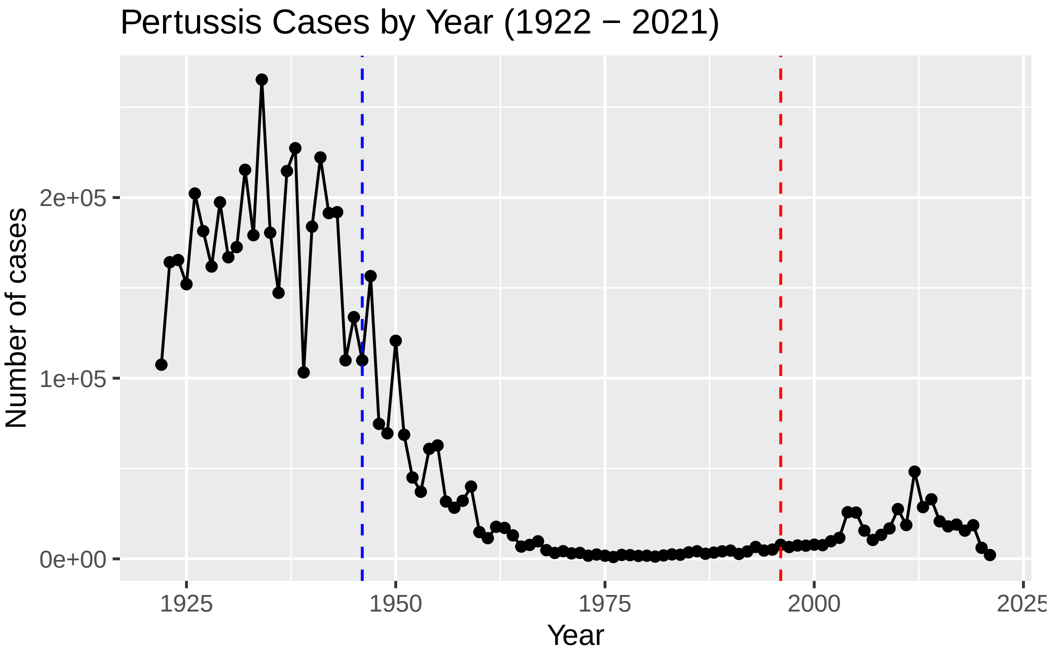
```
library(ggplot2)

ggplot(cdc) +
  aes(Year, No..Reported.Pertussis.Cases) +
  geom_point() +
  geom_line() +
  labs(x = "Year", y = "Number of cases", title = "Pertussis Cases by Year (1922 - 2021)")
```



2. A tale of two vaccines (wP & aP)

Q2. Using the ggplot geom_vline() function add lines to your previous plot for the 1946 introduction of the wP vaccine and the 1996 switch to aP vaccine (see example in the hint below). What do you notice?

```r
ggplot(cdc) +
  aes(Year, No..Reported.Pertussis.Cases) +
  geom_point() +
  geom_line() +
  labs(x = "Year", y = "Number of cases", title = "Pertussis Cases by Year (1922 - 2021)")
  geom_vline(xintercept = 1946, color = "blue", linetype = "dashed") +
  geom_vline(xintercept = 1996, color = "red", linetype = "dashed")
```



> Q3. Describe what happened after the introduction of the aP vaccine? Do you have a possible explanation for the observed trend?

Pertussis cases started rising again. One possible explanation is that less people might be getting vaccinated recently.

3. Exploring CMI-PB data

The CMI-PB API returns JSON data

The CMI-PB API (like most APIs) sends responses in JSON format. To read these types of files into R we will use the `read_json()` function from the jsonlite package.

```
# Allows us to read, write and process JSON data
library(jsonlite)
```

Let's now read the main subject database table from the CMI-PB API.

```
subject <- read_json("https://www.cmi-pb.org/api/subject", simplifyVector = TRUE)
head(subject, 3)
```

```
  subject_id infancy_vac biological_sex              ethnicity  race
1          1          wP         Female Not Hispanic or Latino White
2          2          wP         Female Not Hispanic or Latino White
3          3          wP         Female                Unknown White
  year_of_birth date_of_boost      dataset
1    1986-01-01    2016-09-12 2020_dataset
2    1968-01-01    2019-01-28 2020_dataset
3    1983-01-01    2016-10-10 2020_dataset
```

> Q4. How may aP and wP infancy vaccinated subjects are in the dataset?

```
table(subject$infancy_vac)
```

```

aP wP 
60 58 
```

> Q5. How many Male and Female subjects/patients are in the dataset?

```
table(subject$biological_sex)
```

```

Female   Male 
    79     39 
```

> Q6. What is the breakdown of race and biological sex (e.g. number of Asian females, White males etc…)?

```
table(subject$race, subject$biological_sex)
```

```
                                           
                                            Female Male
  American Indian/Alaska Native                  0    1
  Asian                                         21   11
  Black or African American                      2    0
  More Than One Race                             9    2
  Native Hawaiian or Other Pacific Islander      1    1
  Unknown or Not Reported                       11    4
  White                                         35   20
```

Side-Note: Working with dates

```r
library(lubridate)
```

```
Warning: package 'lubridate' was built under R version 4.3.2


Attaching package: 'lubridate'

The following objects are masked from 'package:base':

    date, intersect, setdiff, union
```

> Q7. Using this approach determine (i) the average age of wP individuals, (ii) the average age of aP individuals; and (iii) are they significantly different?

```r
# age of wP individuals
wp_birth <- subject[which(subject$infancy_vac == "wP"),]$year_of_birth
round(summary(time_length( today() - ymd(wp_birth),  "years")))
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   28      31      35      36      39      56 
```

```r
# average age of aP individuals
ap_birth <- subject[which(subject$infancy_vac == "aP"),]$year_of_birth
round(summary(time_length( today() - ymd(ap_birth),  "years")))
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   21      26      26      26      27      30 
```

```r
x <- t.test(time_length(today() - ymd(wp_birth), "years"), time_length(today() - ymd(ap_bi
x$p.value
```

```
[1] 6.813505e-19
```

They are significantly different.

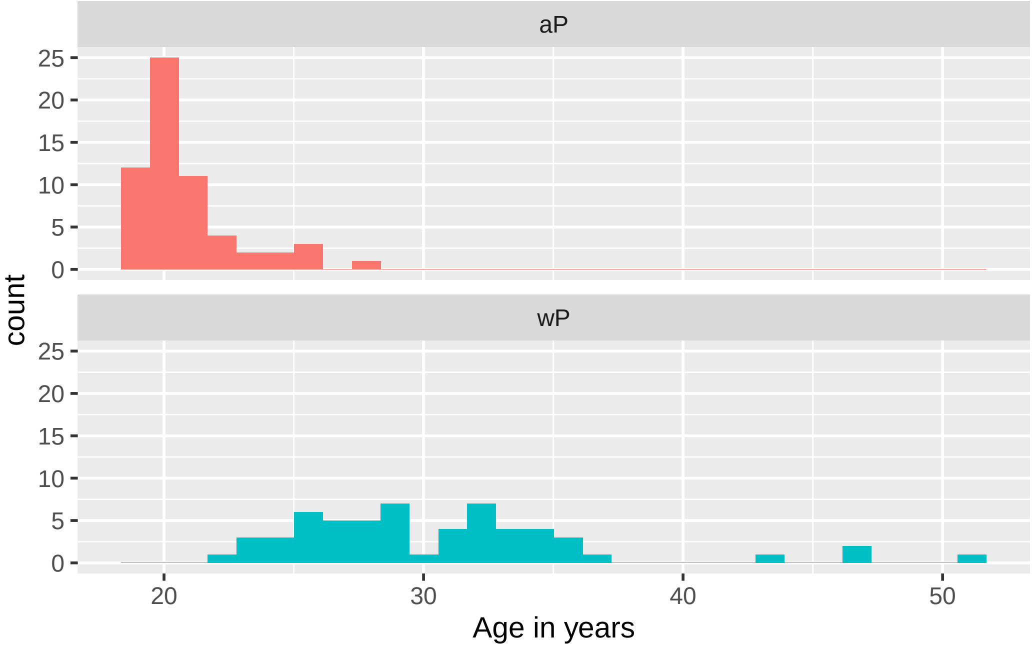
> Q8. Determine the age of all individuals at time of boost?

```
subject$age <- ymd(subject$date_of_boost) - ymd(subject$year_of_birth)
```

Q9. With the help of a faceted boxplot or histogram (see below), do you think these two groups are significantly different?

```
ggplot(subject) +
  aes(time_length(age, "year"),
      fill=as.factor(infancy_vac)) +
  geom_histogram(show.legend=FALSE) +
  facet_wrap(vars(infancy_vac), nrow=2) +
  xlab("Age in years")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The wto groups almost don't overlap, so they are significantly different.

Joining multiple tables

Read the specimen and ab_titer tables into R and store the data as `specimen` and `titer` named data frames.

```r
# Complete the API URLs...
specimen <- read_json("https://www.cmi-pb.org/api/specimen", simplifyVector = TRUE)
titer <- read_json("https://www.cmi-pb.org/api/plasma_ab_titer", simplifyVector = TRUE)
```

To know whether a given `specimen_id` comes from an aP or wP individual we need to link (a.k.a. "join" or merge) our `specimen` and `subject` data frames. The excellent dplyr package (that we have used previously) has a family of `join()` functions that can help us with this common task:

> Q9. Complete the code to join `specimen` and `subject` tables to make a new merged data frame containing all specimen records along with their associated subject details:

```r
library(dplyr)
```

```
Warning: package 'dplyr' was built under R version 4.3.2
```

```

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
```

```r
meta <- inner_join(specimen, subject)
```

```
Joining with `by = join_by(subject_id)`
```

```r
dim(meta)
```

```
[1] 939  14
```

```r
head(meta)
```

```
  specimen_id subject_id actual_day_relative_to_boost
1           1          1                           -3
2           2          1                            1
3           3          1                            3
4           4          1                            7
5           5          1                           11
6           6          1                           32
  planned_day_relative_to_boost specimen_type visit infancy_vac biological_sex
1                             0         Blood     1          wP         Female
2                             1         Blood     2          wP         Female
3                             3         Blood     3          wP         Female
4                             7         Blood     4          wP         Female
5                            14         Blood     5          wP         Female
6                            30         Blood     6          wP         Female
               ethnicity  race year_of_birth date_of_boost      dataset
1 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
2 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
3 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
4 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
5 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
6 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
         age
1 11212 days
2 11212 days
3 11212 days
4 11212 days
5 11212 days
6 11212 days
```

> Q10. Now using the same procedure join `meta` with `titer` data so we can further analyze this data in terms of time of visit aP/wP, male/female etc.

```r
abdata <- inner_join(titer, meta)
```

```
Joining with `by = join_by(specimen_id)`
```

```r
dim(abdata)
```

```
[1] 41810    21
```

> Q11. How many specimens (i.e. entries in `abdata`) do we have for each `isotype`?

```
table(abdata$isotype)
```

```

 IgE  IgG IgG1 IgG2 IgG3 IgG4 
6698 3240 7968 7968 7968 7968 
```

> Q12. What are the different `$dataset` values in `abdata` and what do you notice about the number of rows for the most "recent" dataset?

```
table(abdata$dataset)
```

```

2020_dataset 2021_dataset 2022_dataset 
       31520         8085         2205 
```

The rows in the most recent database are significantly smaller.

4. Examine IgG Ab titer levels

Now using our joined/merged/linked `abdata` dataset `filter()` for IgG `isotype`.

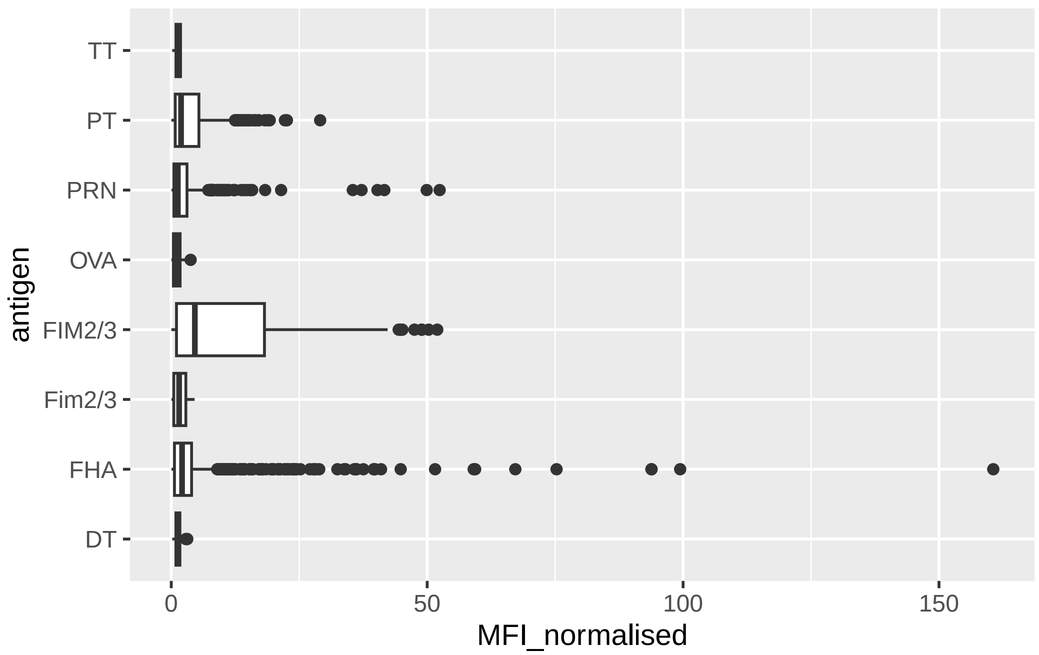
```
igg <- abdata %>% filter(isotype == "IgG")
head(igg)
```

```
  specimen_id isotype is_antigen_specific antigen        MFI MFI_normalised
1           1     IgG                TRUE      PT   68.56614       3.736992
2           1     IgG                TRUE     PRN  332.12718       2.602350
3           1     IgG                TRUE     FHA 1887.12263      34.050956
4          19     IgG                TRUE      PT   20.11607       1.096366
5          19     IgG                TRUE     PRN  976.67419       7.652635
6          19     IgG                TRUE     FHA   60.76626       1.096457
   unit lower_limit_of_detection subject_id actual_day_relative_to_boost
1 IU/ML                 0.530000          1                           -3
2 IU/ML                 6.205949          1                           -3
3 IU/ML                 4.679535          1                           -3
4 IU/ML                 0.530000          3                           -3
5 IU/ML                 6.205949          3                           -3
6 IU/ML                 4.679535          3                           -3
```

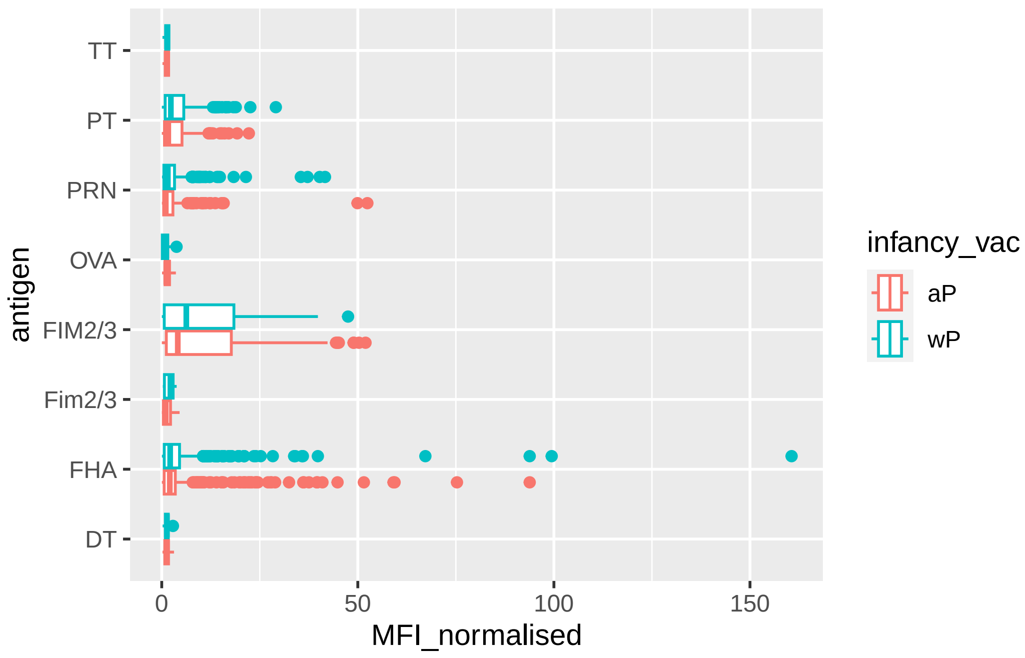
```
  planned_day_relative_to_boost specimen_type visit infancy_vac biological_sex
1                             0         Blood     1          wP         Female
2                             0         Blood     1          wP         Female
3                             0         Blood     1          wP         Female
4                             0         Blood     1          wP         Female
5                             0         Blood     1          wP         Female
6                             0         Blood     1          wP         Female
               ethnicity  race year_of_birth date_of_boost      dataset
1 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
2 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
3 Not Hispanic or Latino White    1986-01-01    2016-09-12 2020_dataset
4                Unknown White    1983-01-01    2016-10-10 2020_dataset
5                Unknown White    1983-01-01    2016-10-10 2020_dataset
6                Unknown White    1983-01-01    2016-10-10 2020_dataset
         age
1 11212 days
2 11212 days
3 11212 days
4 12336 days
5 12336 days
6 12336 days
```

Q13. Complete the following code to make a summary boxplot of Ab titer levels (MFI) for all antigens:

```
ggplot(igg) +
  aes(MFI_normalised, antigen) +
  geom_boxplot()
```



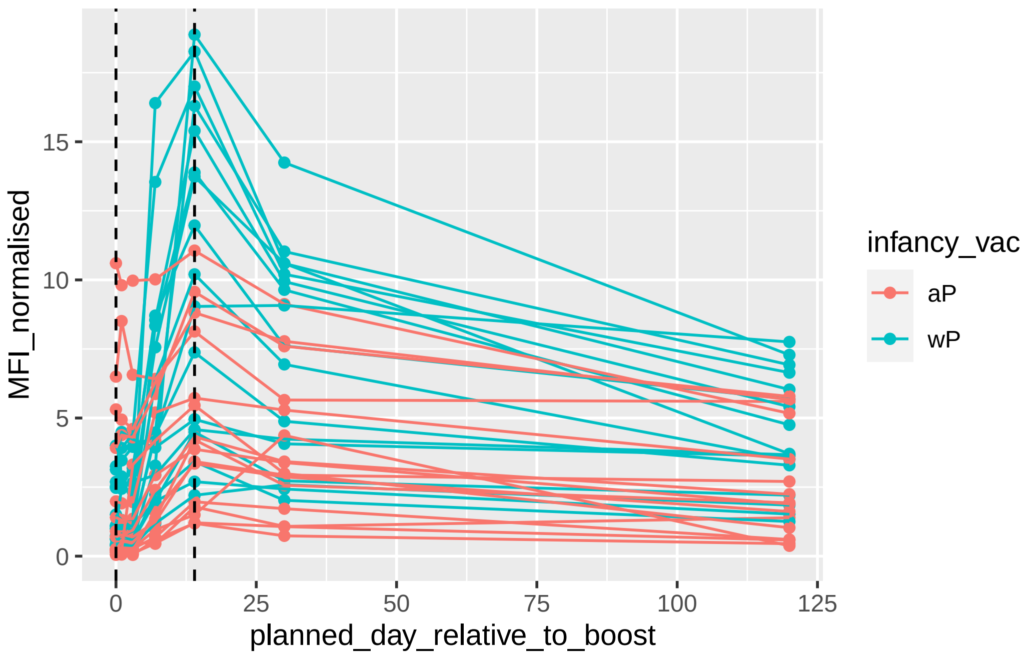
```
ggplot(igg) +
  aes(MFI_normalised, antigen, col=infancy_vac) +
  geom_boxplot()
```



Q15. Filter to pull out only two specific antigens for analysis and create a boxplot for each. You can chose any you like. Below I picked a "control" antigen ("OVA", that is not in our vaccines) and a clear antigen of interest ("PT", Pertussis Toxin, one of the key virulence factors produced by the bacterium B. pertussis).

Focus in on the IgG to PT antigen in the 2021 dataset:

```
igg.pt <- igg %>% filter(antigen == "PT", dataset == "2021_dataset")

ggplot(igg.pt) +
  aes(planned_day_relative_to_boost, MFI_normalised, col = infancy_vac, group = subject_id
  geom_point() +
  geom_line() +
  geom_vline(xintercept=0, linetype="dashed") +
  geom_vline(xintercept=14, linetype="dashed")
```


MFI_normalised
15
10
5
0
0
25
50
75
100
125
planned_day_relative_to_boost
infancy_vac
aP
wP