

2020 Ab titer data: Median (at Day 0) based normalisation

Reading dataset from experimental datafile

```
setwd("/home/pramod/Documents/GitHub/2021-Ab-titer-data-normalisation")
data_2020_metadata <- read_csv("metadata_cmipb_2020.csv")

## Rows: 195 Columns: 1

## -- Column specification -----
## Delimiter: ","
## chr (1): specimen_id subject_id visit Timepoint infancy_vac

##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

json_subject <- fromJSON("subject.json")
json_specimen <- fromJSON("specimen.json")
subject_specimen <- left_join(json_specimen, json_subject) %>%
  dplyr::select(specimen_id, subject_id, planned_day_relative_to_boost)

## Joining, by = "subject_id"

titers_2020 <- read_csv("2020LD_ab_titer.csv") %>%
  left_join(subject_specimen) %>%
  mutate(
    antigen = replace(antigen, antigen == '1% PFA PT', 'PT'),
    isotype_antigen = paste0(isotype, "_", antigen, "_", unit),
    ab_titer_original = ab_titer, ## Keeping ab)titer_original for final dataframe
  )

## Rows: 31520 Columns: 7

## -- Column specification -----
## Delimiter: ","
## chr (3): isotype, antigen, unit
## dbl (3): specimen_id, ab_titer, lower_limit_of_detection
## lgl (1): is_antigen_specific

##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

## Joining, by = "specimen_id"
```

Count NA and below LOD samples per Isotype and Antigen pair

Note features with more than 80% missing values

We haven't removed them from pre-processing

```
count_samples <- titers_2020 %>%
  group_by(isotype_antigen) %>%
  summarise(count_samples = n())

count_0 <- titers_2020 %>%
  group_by(isotype_antigen) %>%
  filter(is.na(ab_titer) == TRUE | ab_titer < lower_limit_of_detection) %>%
  summarise(count_0 = n(), percentage_0 = (count_0/394)* 100) %>%
  arrange(desc(percentage_0))

count_0
```

```
## # A tibble: 73 x 3
##   isotype_antigen count_0 percentage_0
##   <chr>          <int>      <dbl>
## 1 IgE_OVA_IU/ML    394        100
## 2 IgG4_PD1_IU/ML   394        100
## 3 IgE_Measles_IU/ML 393        99.7
## 4 IgE_PD1_IU/ML    389        98.7
## 5 IgG4_Measles_IU/ML 389        98.7
## 6 IgE_ACT_IU/ML     387        98.2
## 7 IgE_BETV1_IU/ML   381        96.7
## 8 IgG2_BETV1_IU/ML   380        96.4
## 9 IgG3_BETV1_IU/ML   380        96.4
## 10 IgE_FHA_IU/ML    379        96.2
## # ... with 63 more rows
```

```
remove_features <- count_0[count_0$percentage_0 > 80,]$isotype_antigen
#titters_2020_feature_removed <- titters_2020[!titters_2020$isotype_antigen %in% remove_features, ]
```

Replace not detected values with lower_limit_of_detection

```
titters_2020_feature_removed = titters_2020
titters_2020_feature_removed_lod <- titters_2020_feature_removed %>%
  mutate(
    lower_limit_of_detection = if_else(isotype_antigen == "IgE_Total_UG/ML", 2.09613325980121, lower_limit_of_detection),
    ab_titer = if_else(ab_titer < lower_limit_of_detection, lower_limit_of_detection, ab_titer)
  )

## QC for LOD outliers
## No Outliers detected
lod_outlier <- titters_2020_feature_removed_lod %>%
  group_by(isotype_antigen) %>%
  #slice_min(MFI_normalized, n=2) %>%
```

```
#arrange(desc(MFI_normalized)) %>%
summarise(min = min(ab_titer), lod = lower_limit_of_detection, fc_min_max = lower_limit_of_detection/
filter(fc_min_max > 3)
```

'summarise()' has grouped output by 'isotype_antigen'. You can override using the '.groups' argument

Identifying outlier ab titers.

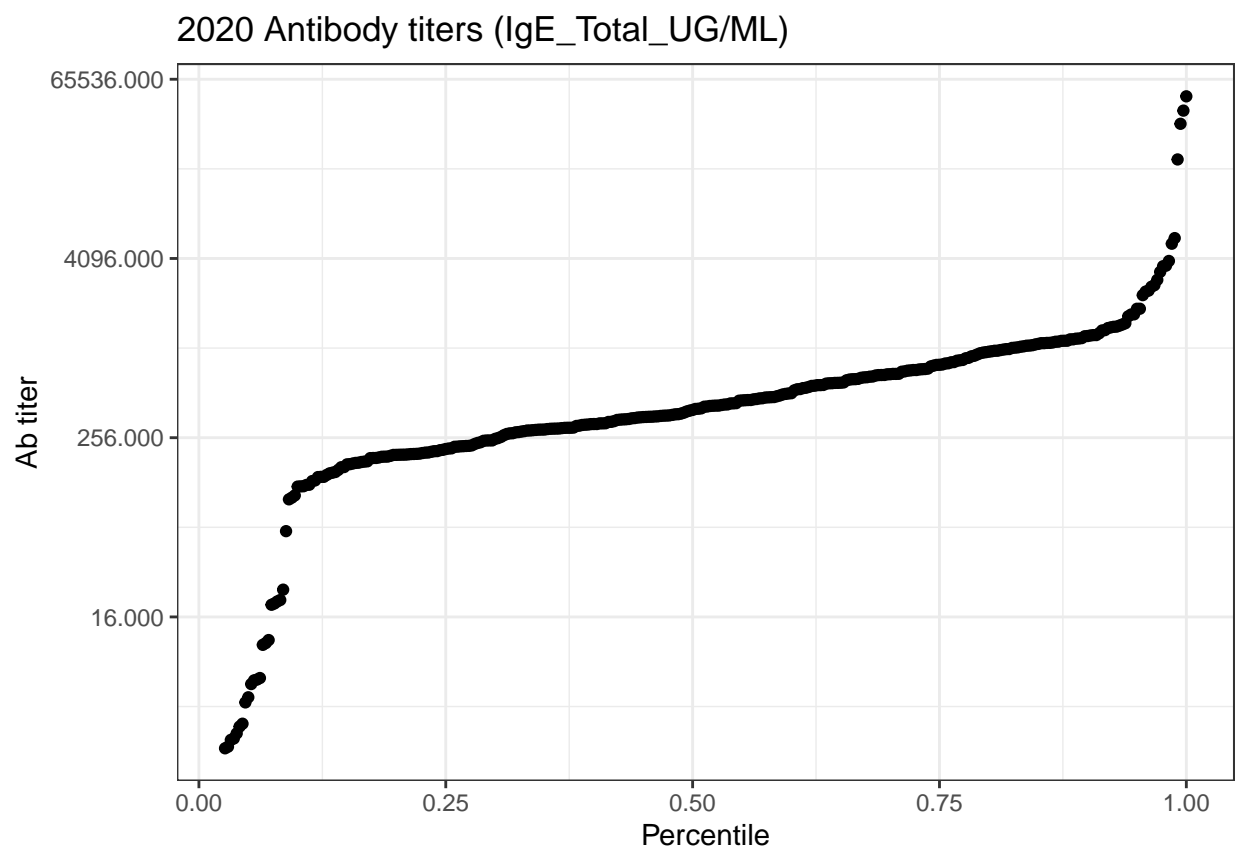
```
## If second_max_value/max_value > 3 then set max_value = LOD
top_outlier <- titers_2020_feature_removed_lod %>%
  group_by(isotype_antigen) %>%
  slice_max(ab_titer, n=2) %>%
  arrange(desc(ab_titer)) %>%
  summarise(fc_min_max = max(ab_titer)/min(ab_titer)) %>%
  filter(fc_min_max > 3)

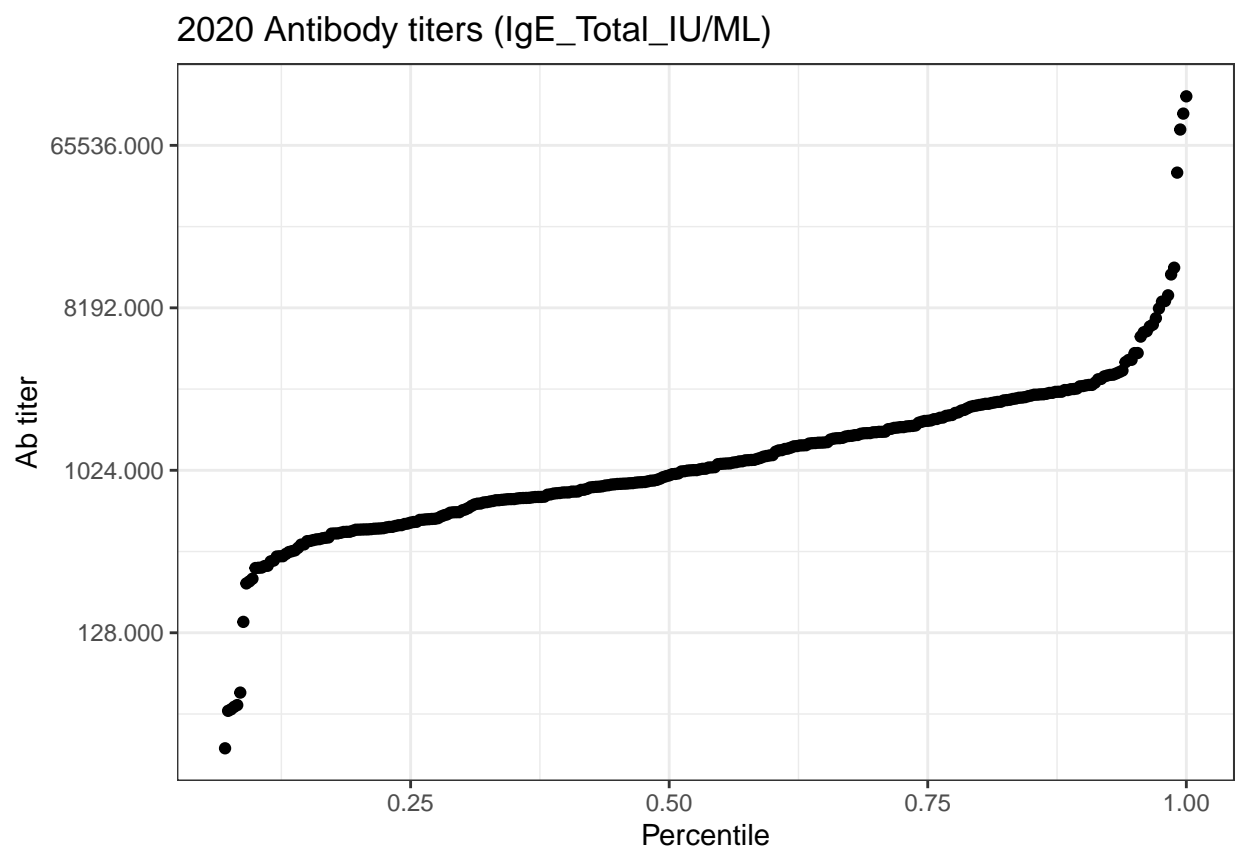
## If first_value/LOD > 3 then set LOD as first_value
bottom_outlier <- titers_2020_feature_removed_lod %>%
  group_by(isotype_antigen) %>%
  slice_min(ab_titer, n=2) %>%
  arrange(desc(ab_titer)) %>%
  summarise(fc_min_max = max(ab_titer)/min(ab_titer)) %>%
  filter(fc_min_max > 2)

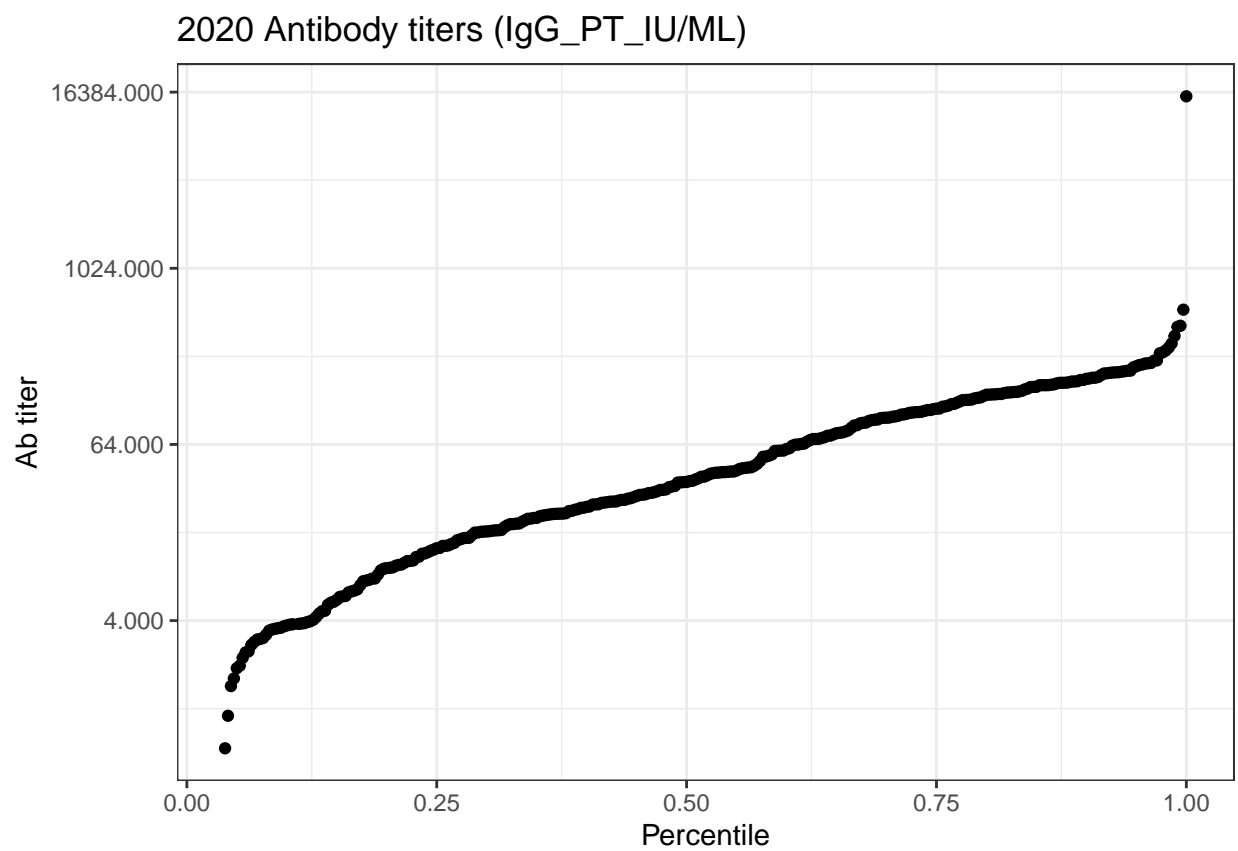
## Plotting cumulative distribution to identify outliers.
#Our transformation function
scaleFUN <- function(x) sprintf("%.3f", x)

#for(select_ia in c('IgE_Total'))
for(select_ia in unique(titers_2020_feature_removed_lod$isotype_antigen))
{
  plot1 <- titers_2020_feature_removed_lod %>%
    mutate(subject_id = as.factor(subject_id)) %>%
    filter(isotype_antigen == select_ia, planned_day_relative_to_boost < 50) %>%
    arrange(desc(ab_titer)) %>%
    ggplot(aes(y=ab_titer)) + stat_ecdf(geom = "point", pad = FALSE) +
    labs(y = "Ab titer", x = "Percentile")+ ggtitle(paste0("2020 Antibody titers (" , select_ia, ")")) +
    scale_y_continuous(trans = 'log2', labels=scaleFUN)

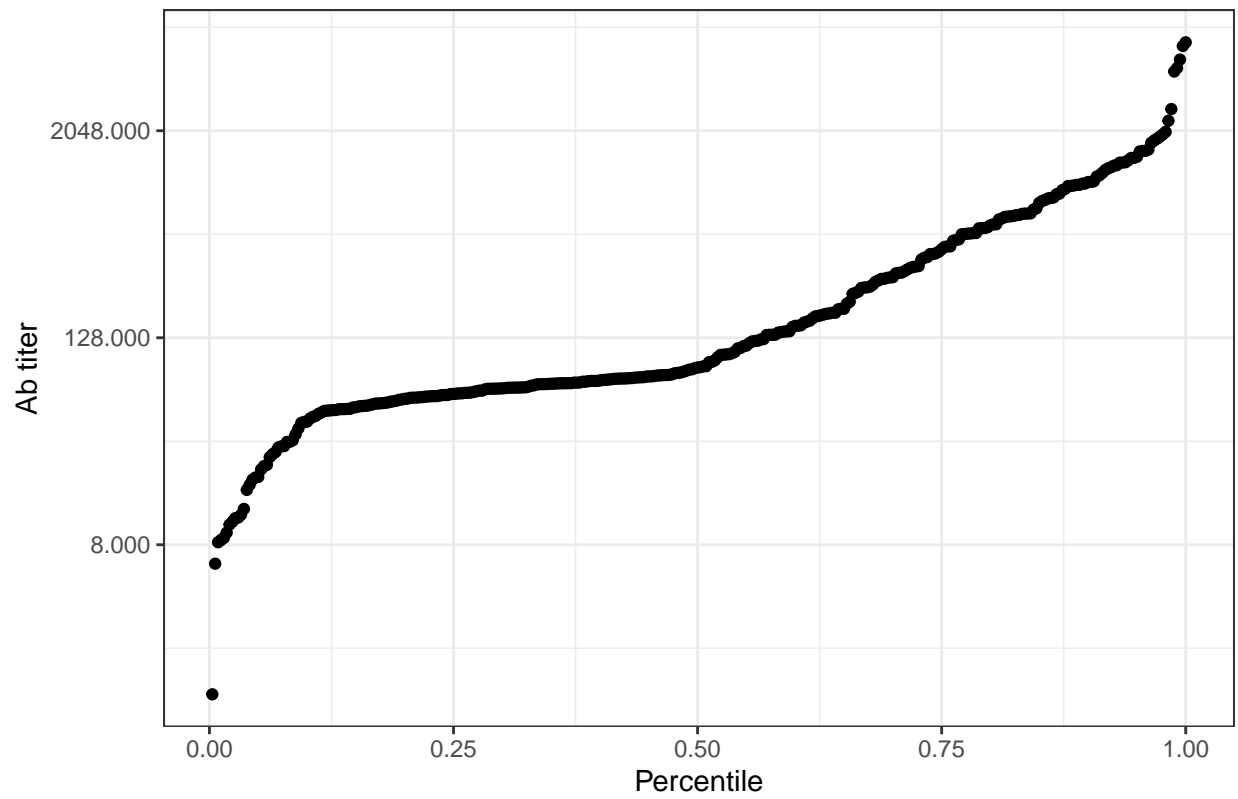
  plot(plot1)
}
```

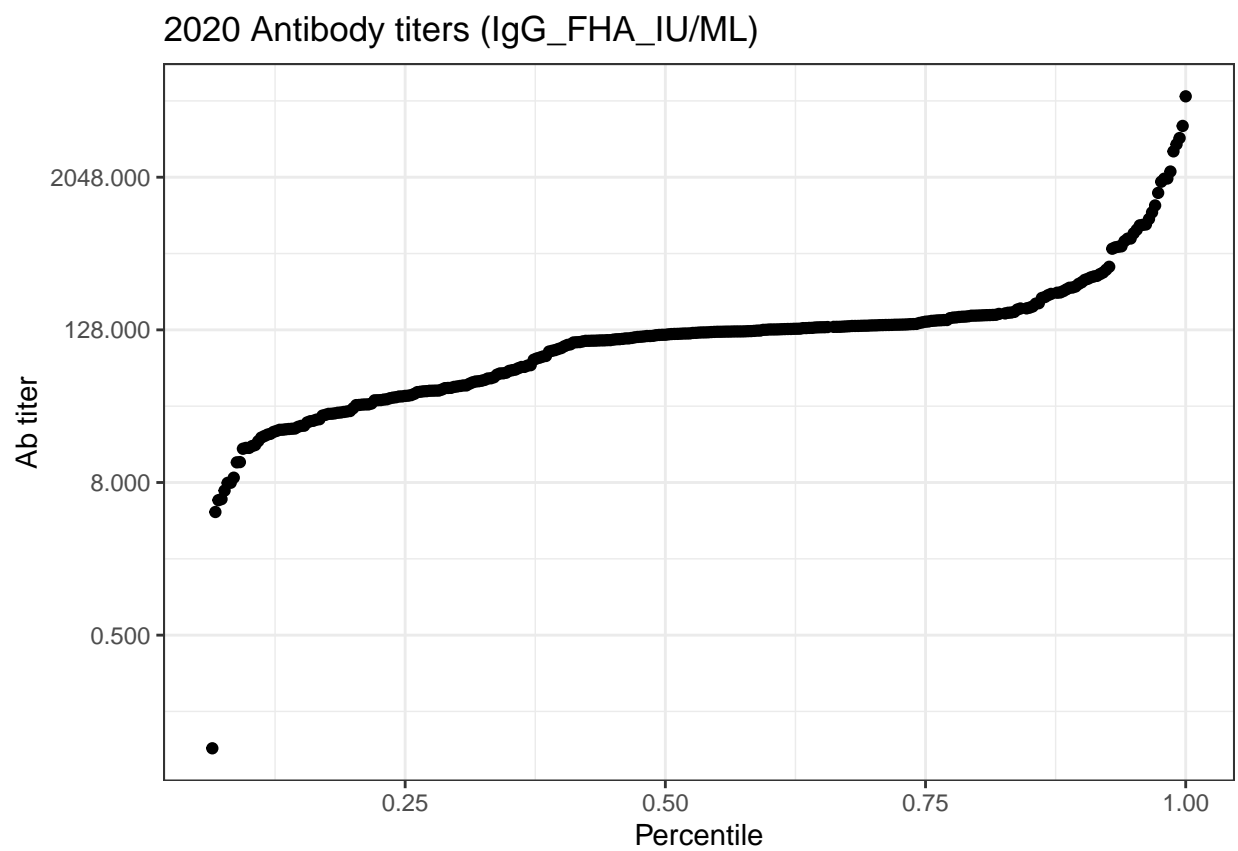


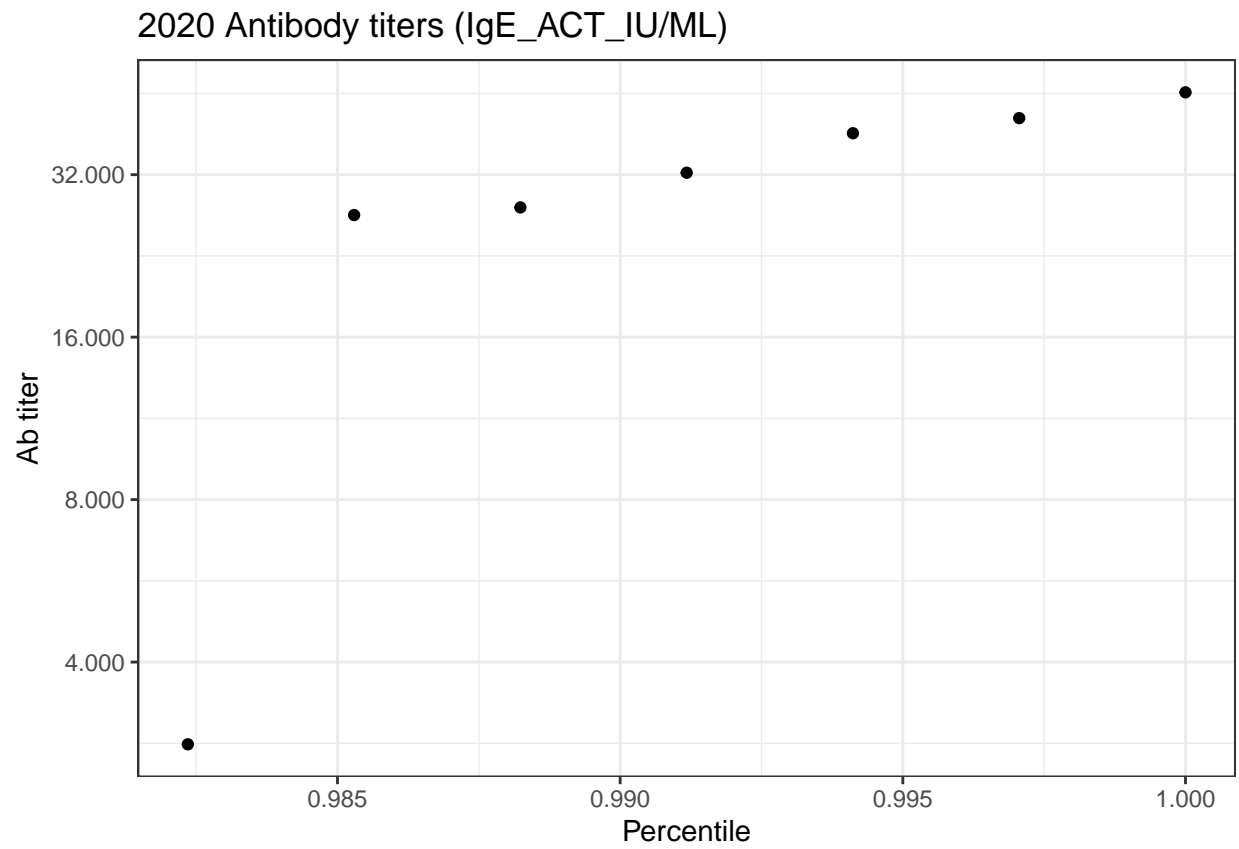


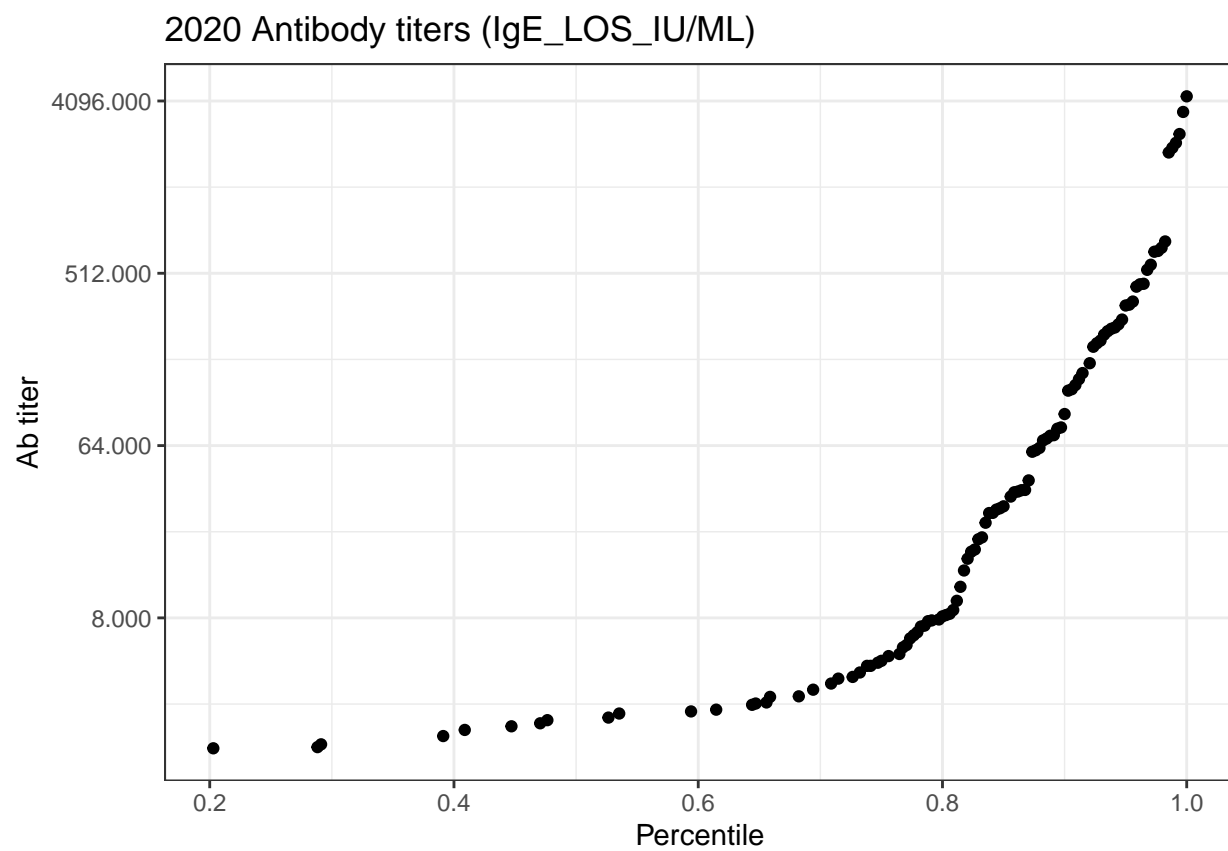


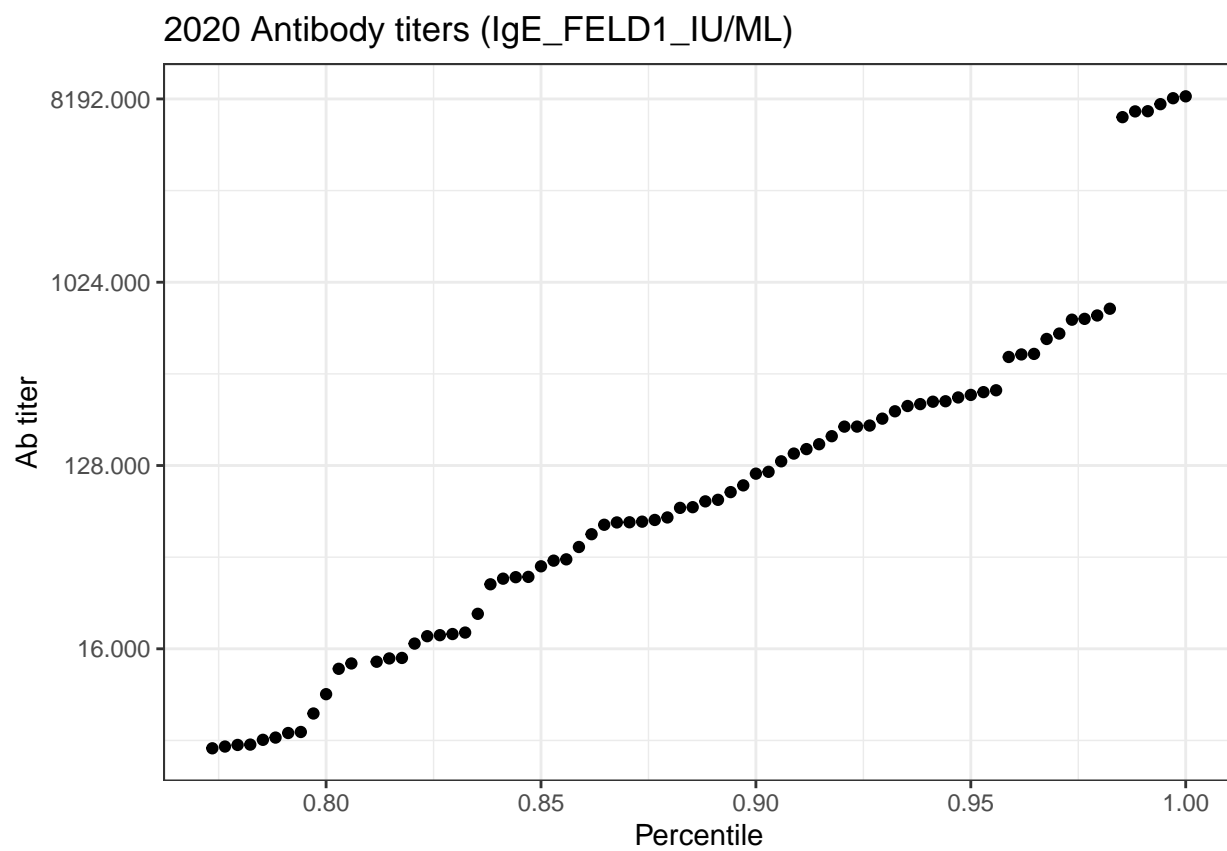
2020 Antibody titers (IgG_PRN_IU/ML)



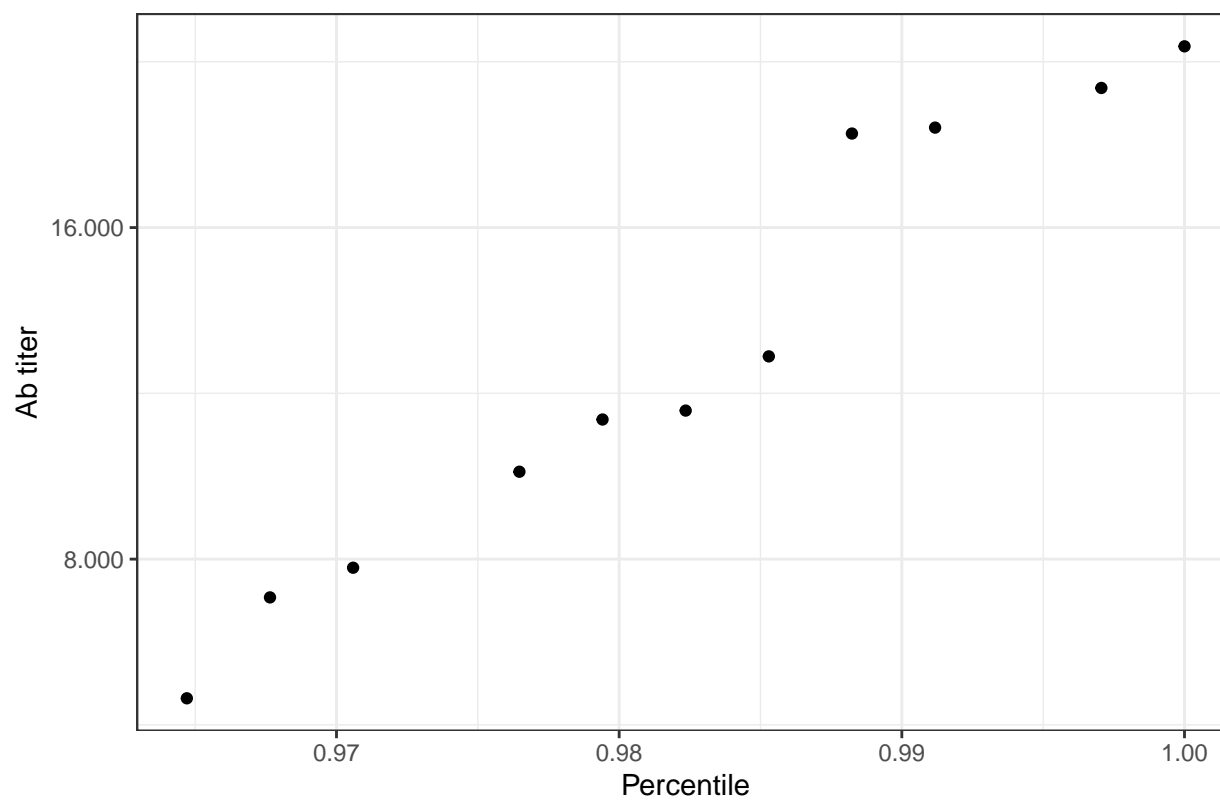


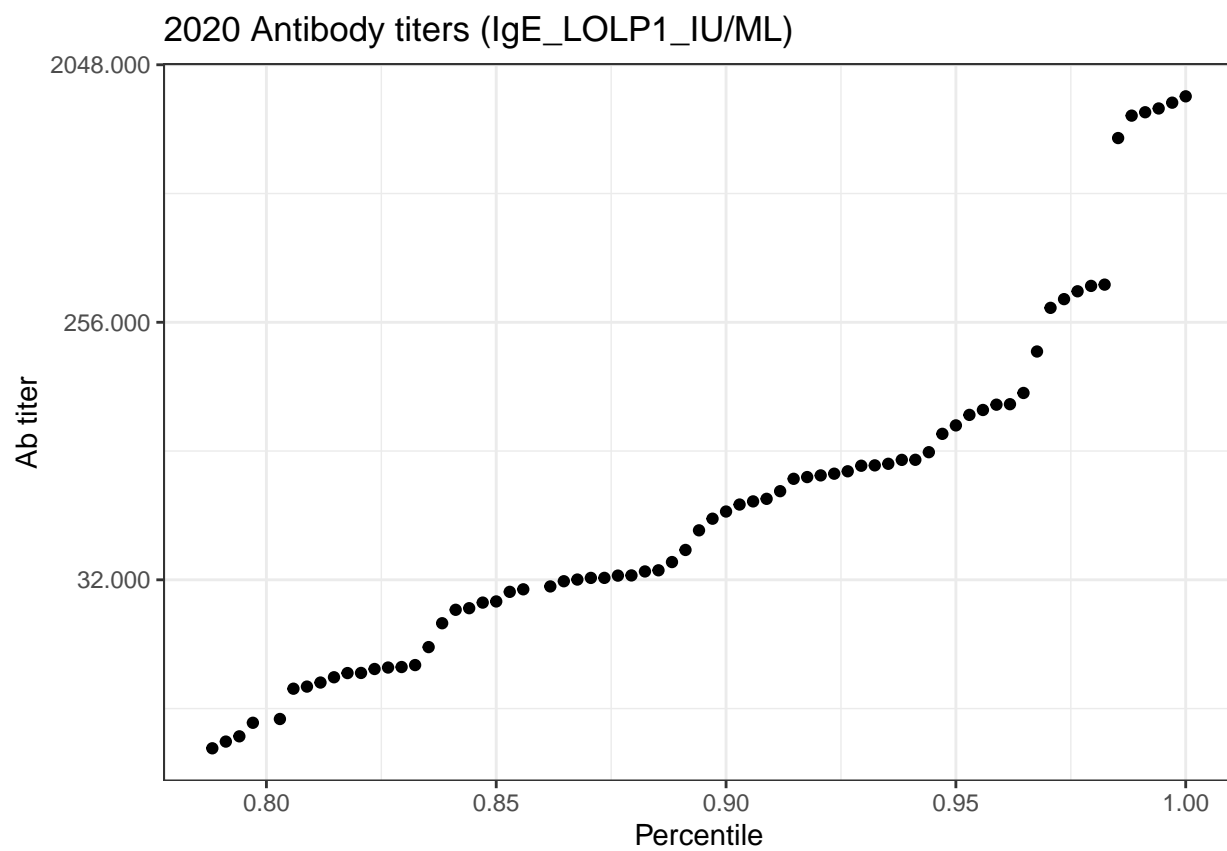




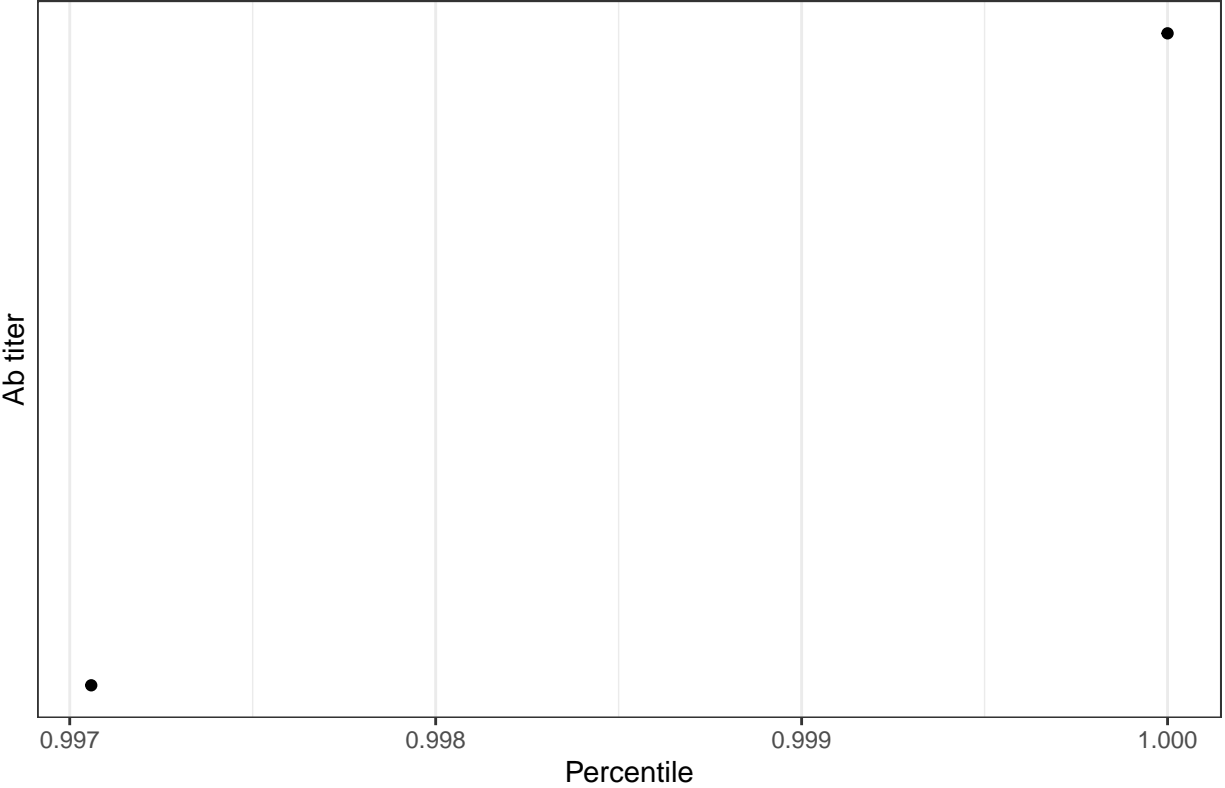


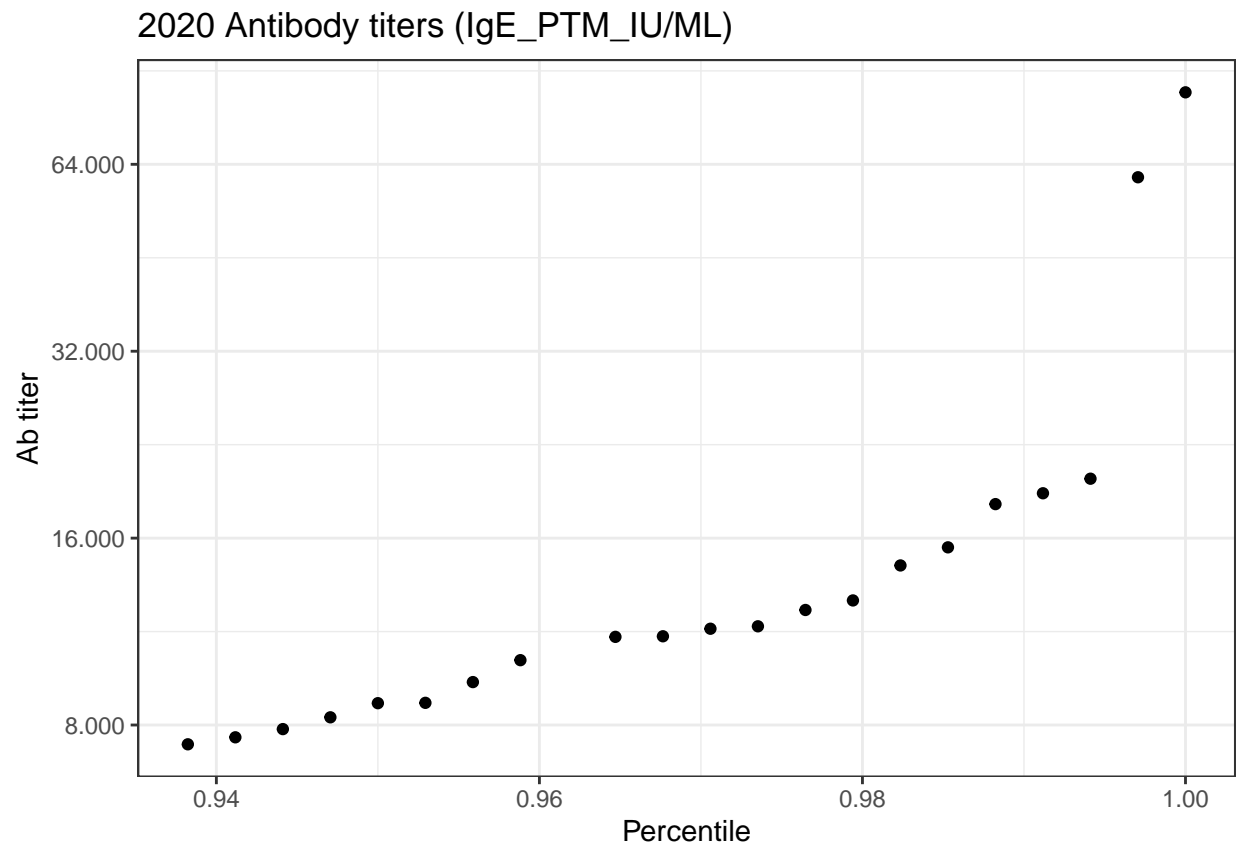
2020 Antibody titers (IgE_BETV1_IU/ML)

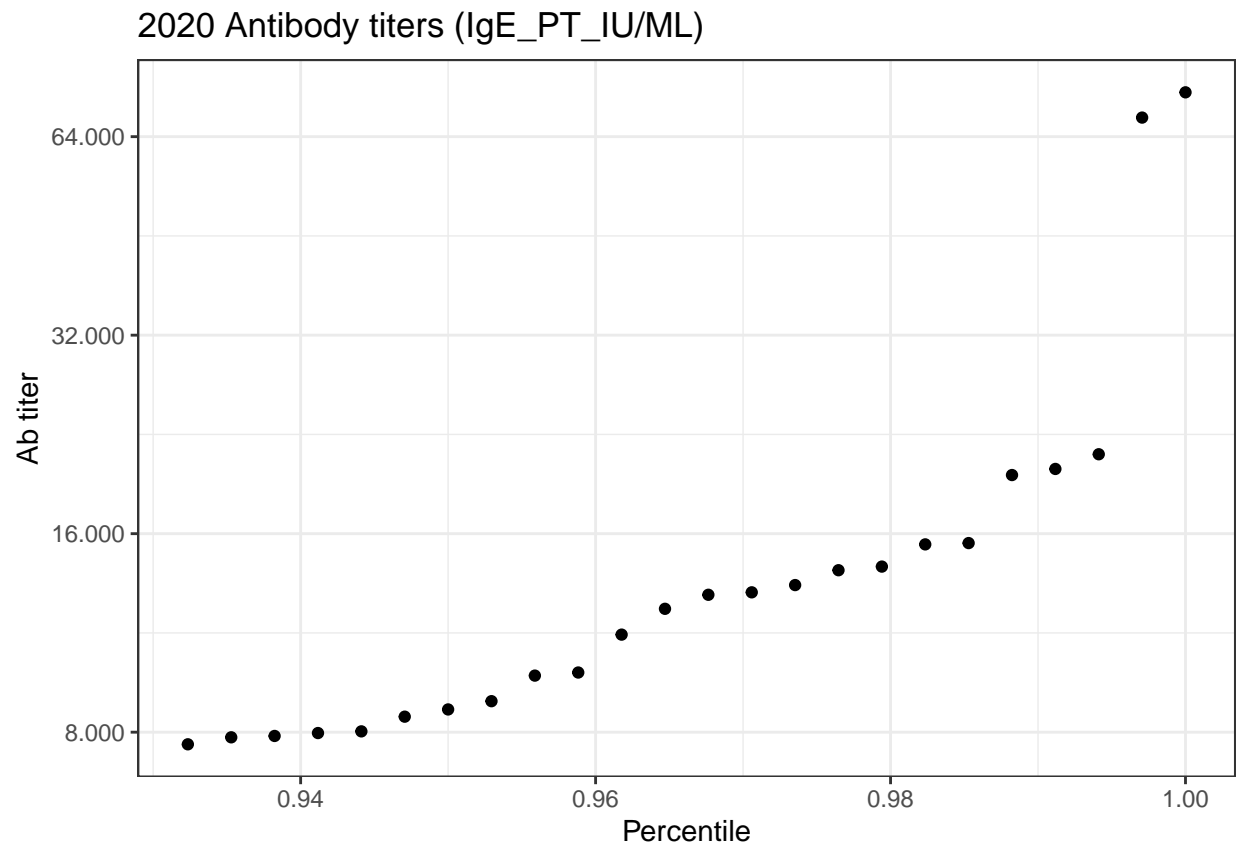


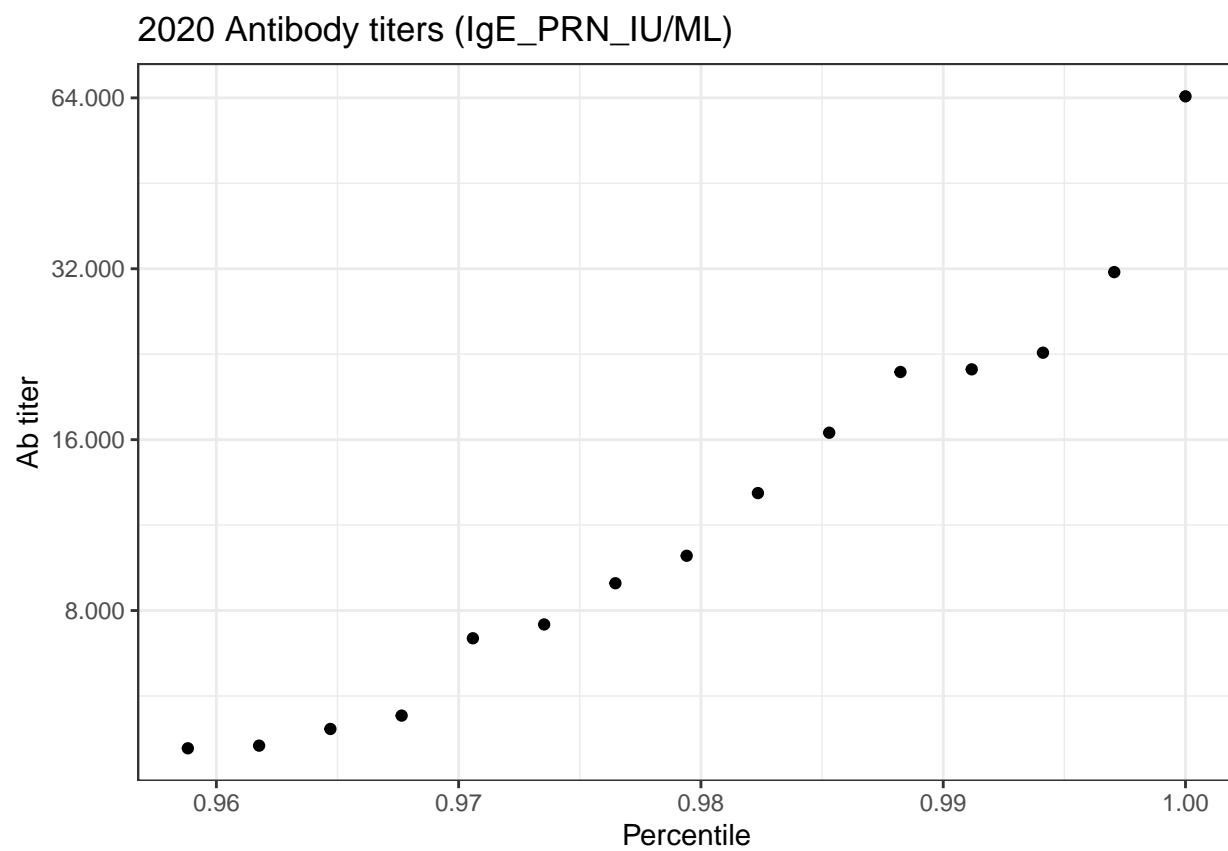


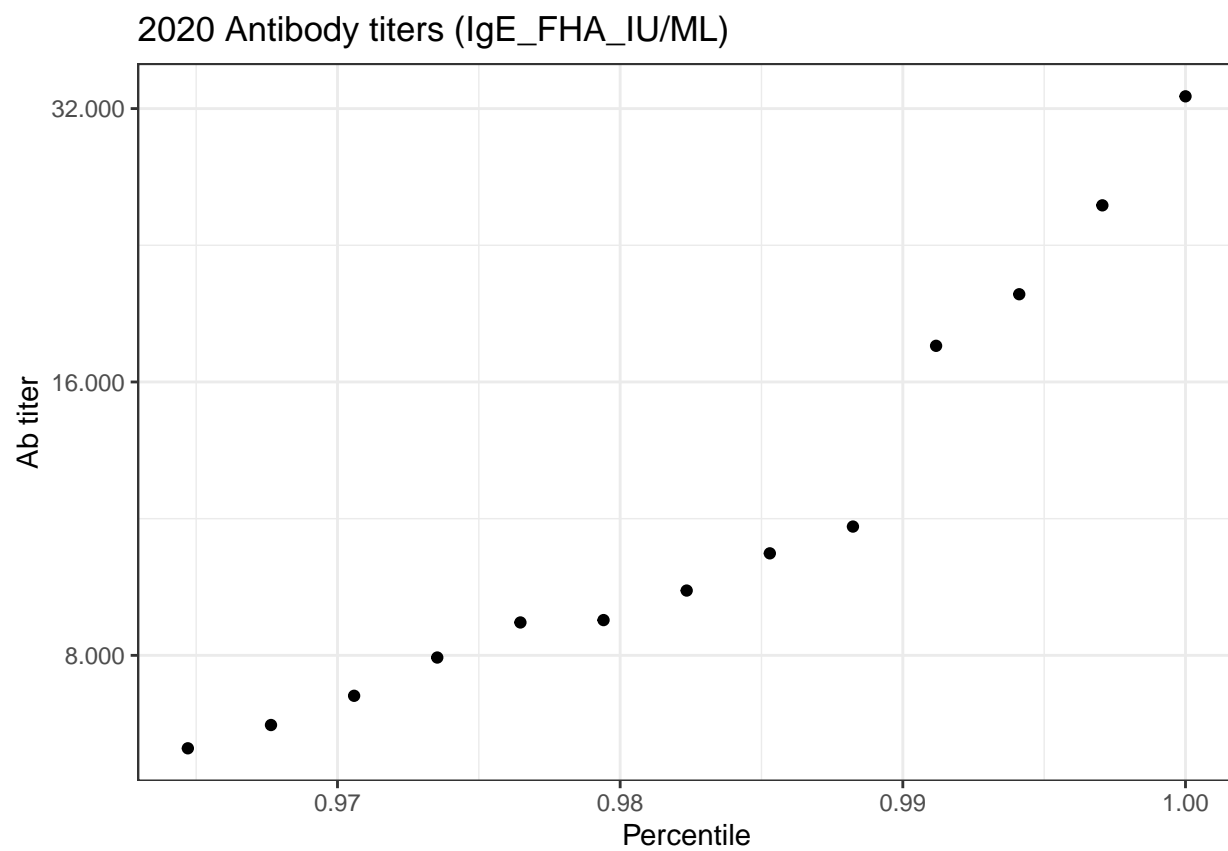
2020 Antibody titers (IgE_Measles_IU/ML)

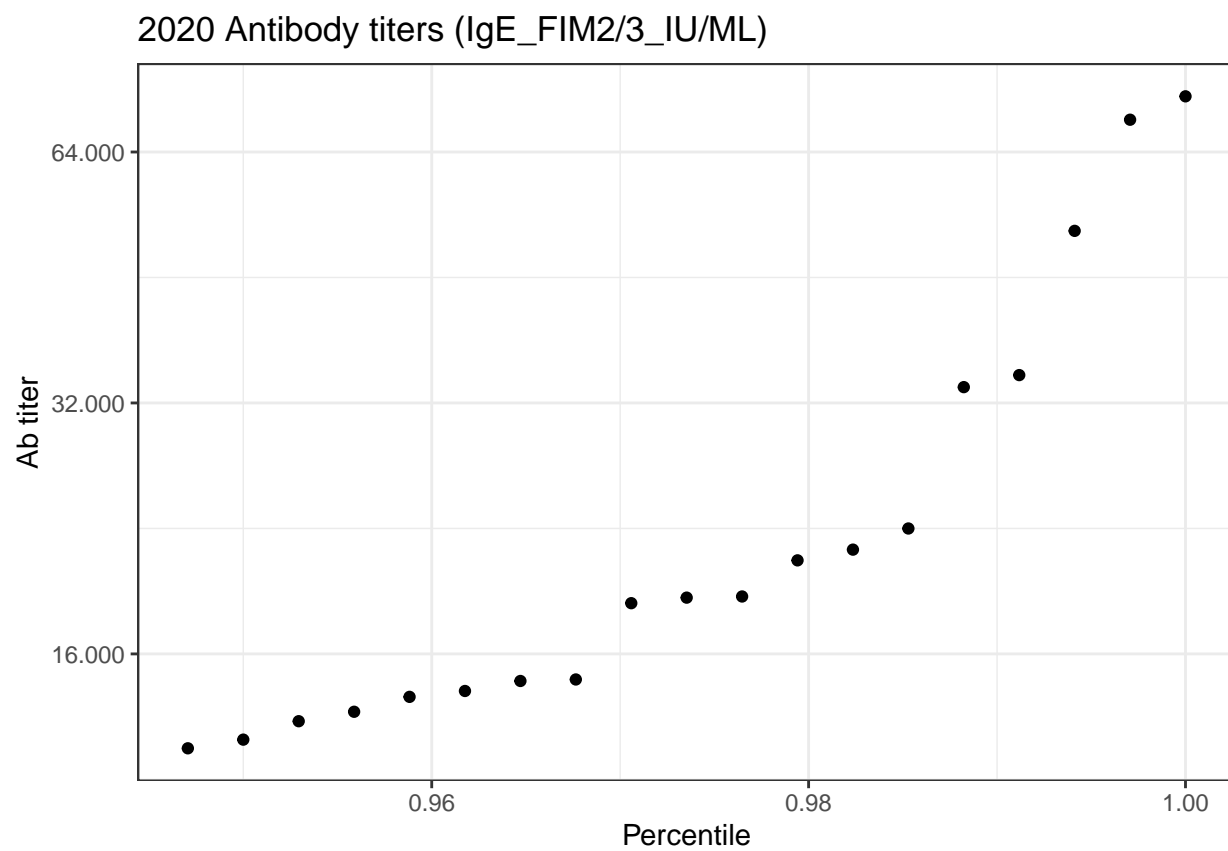




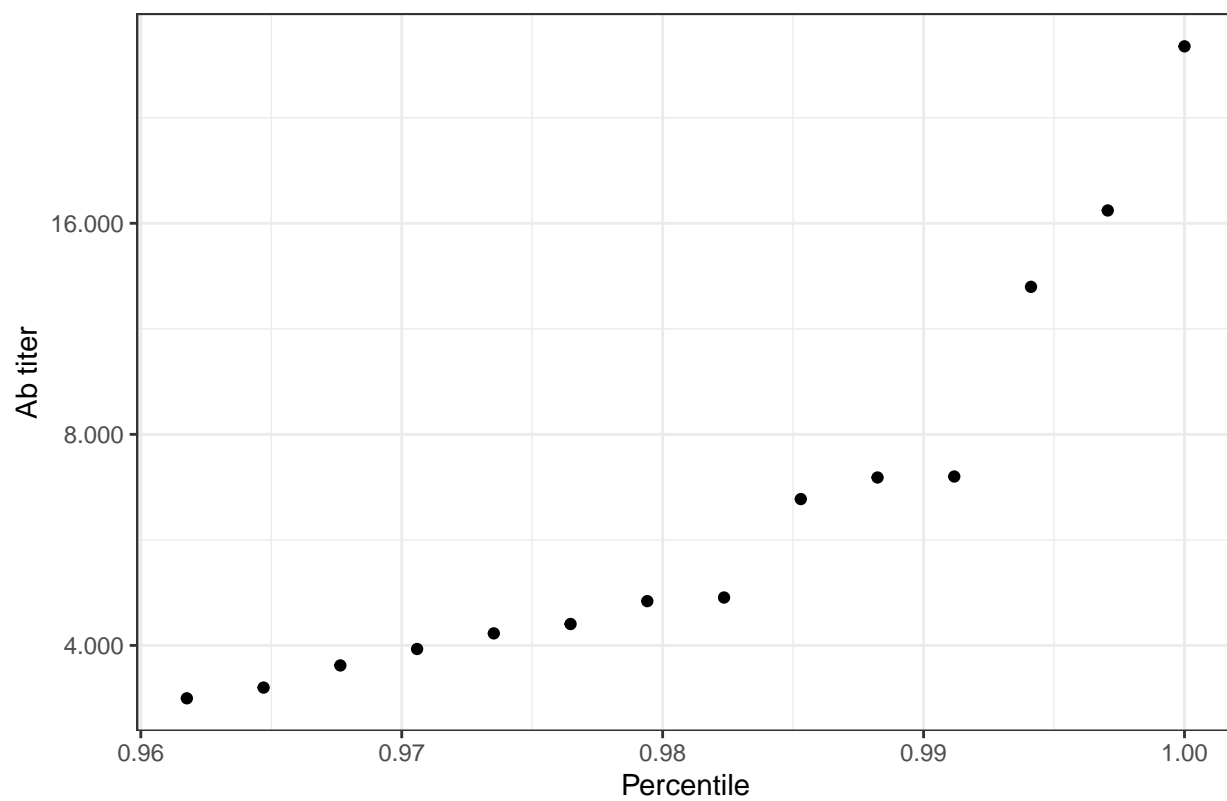


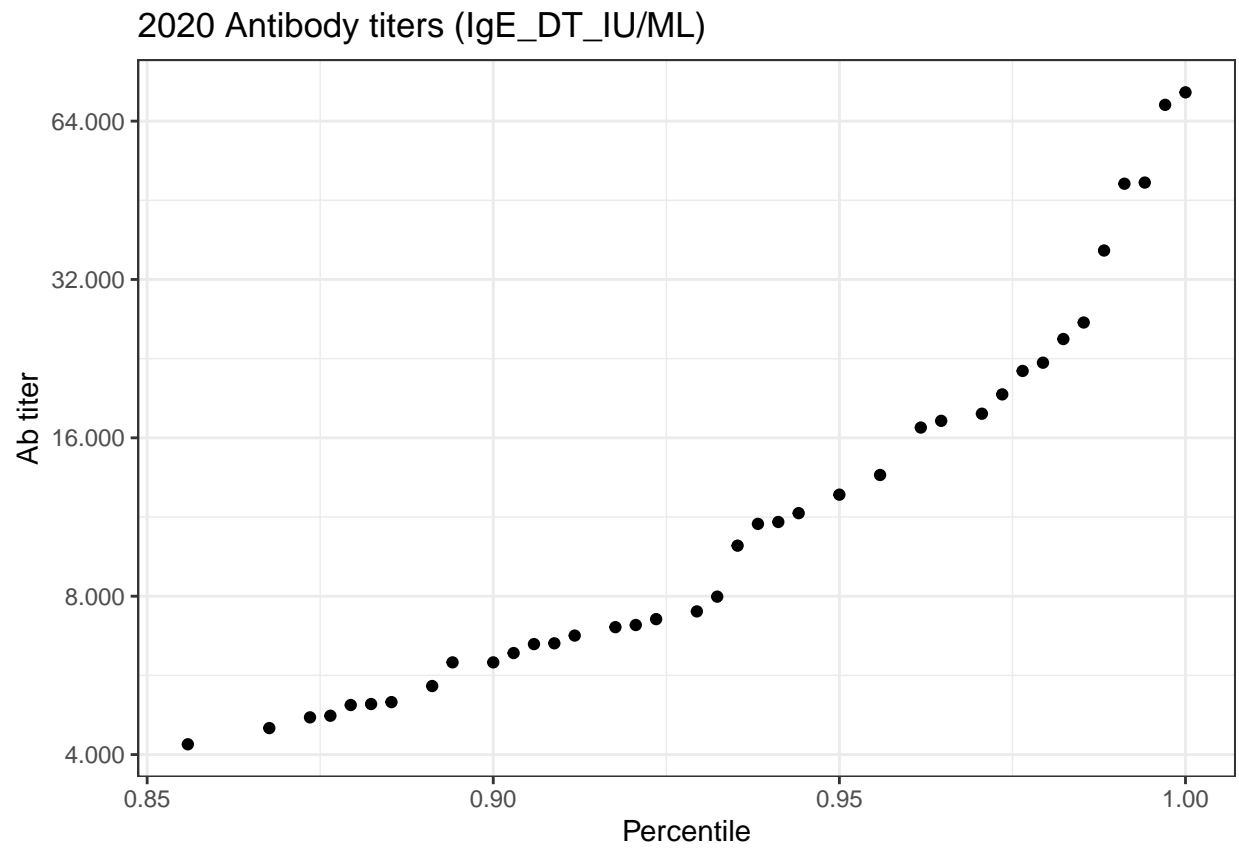




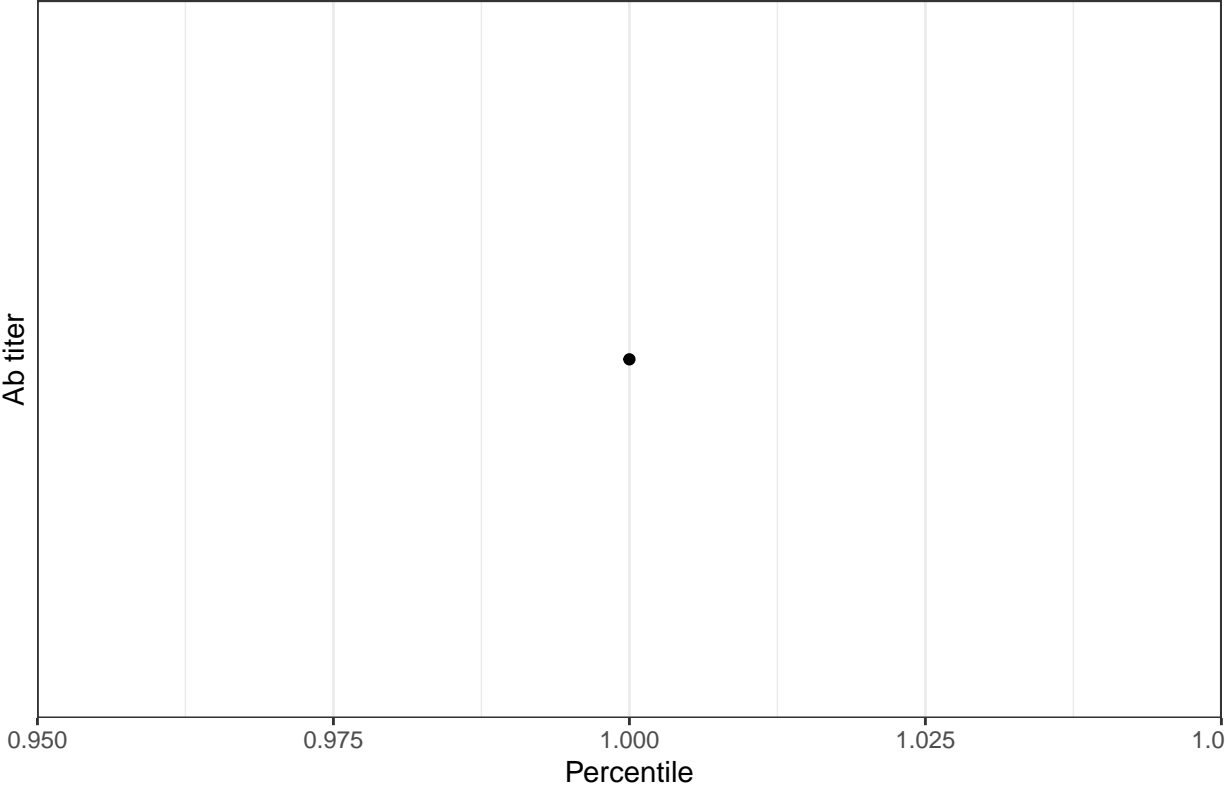


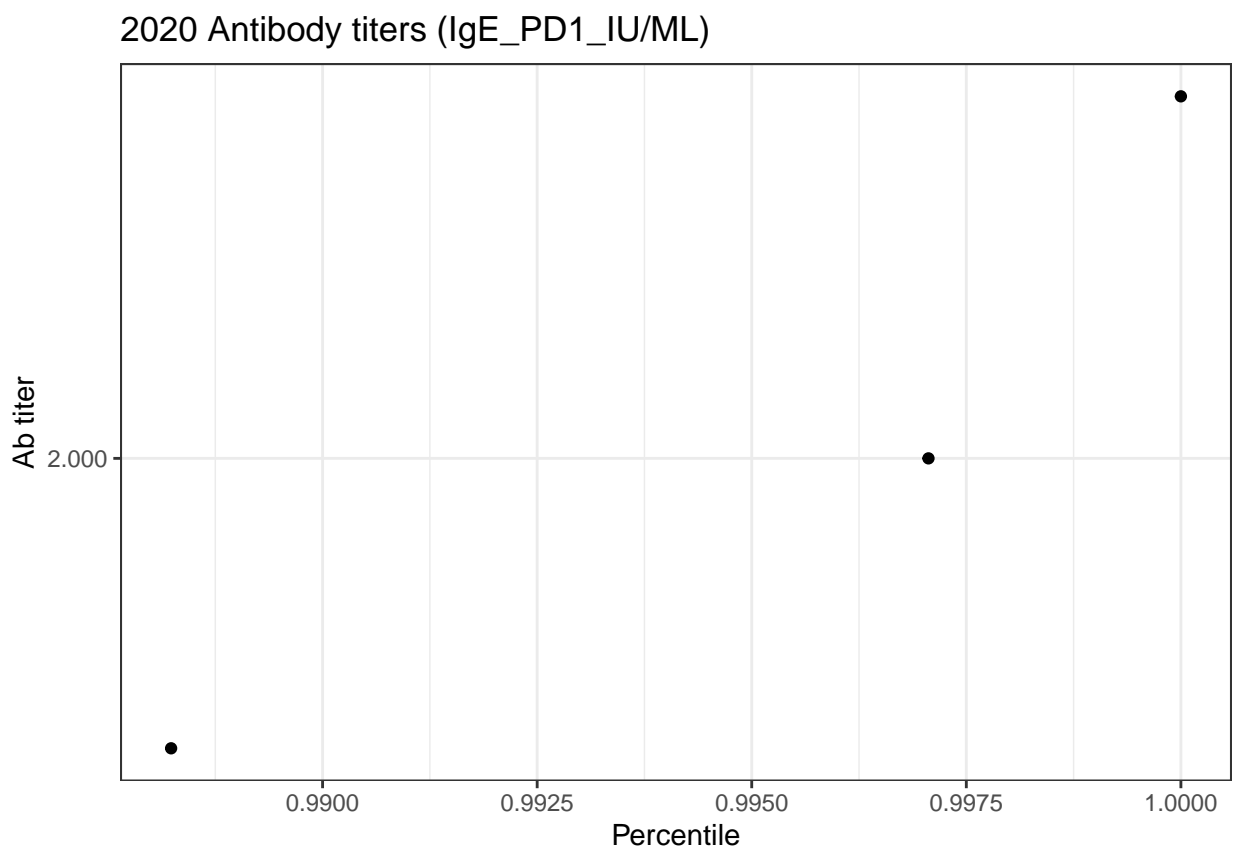
2020 Antibody titers (IgE_TT_IU/ML)



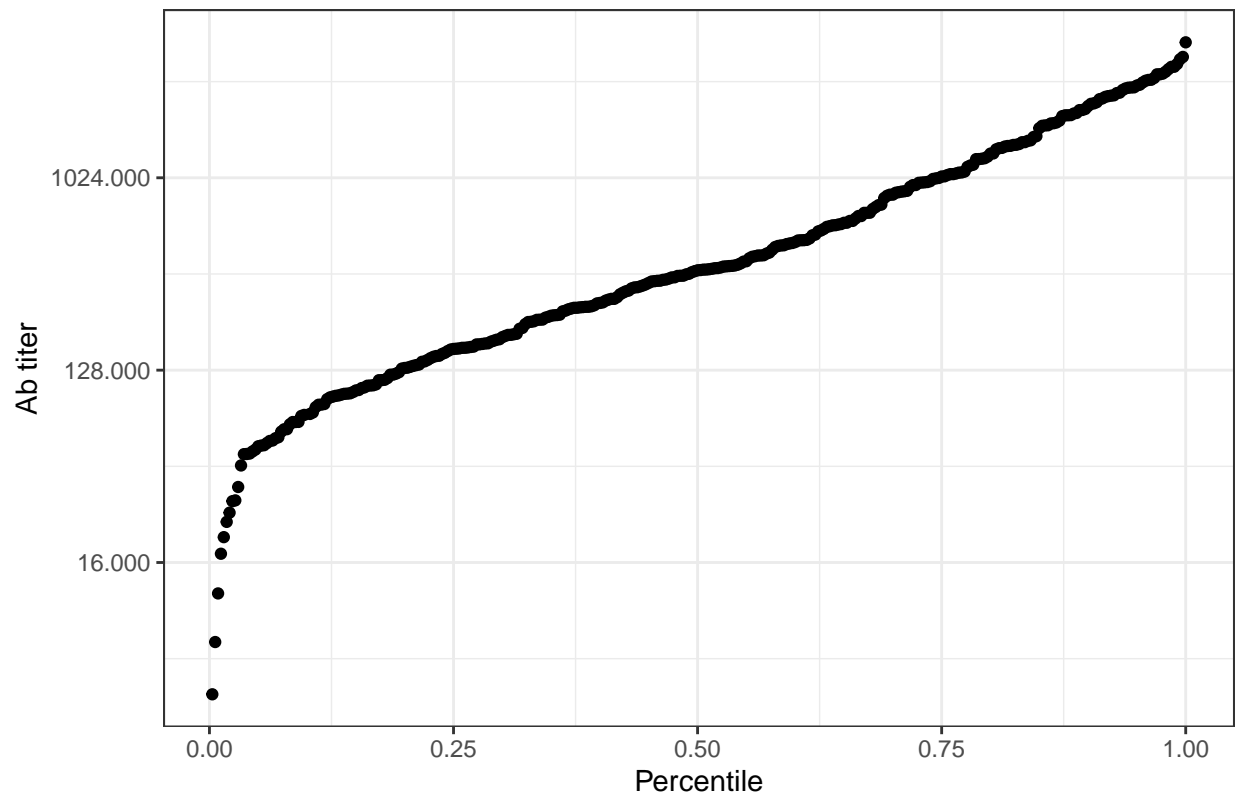


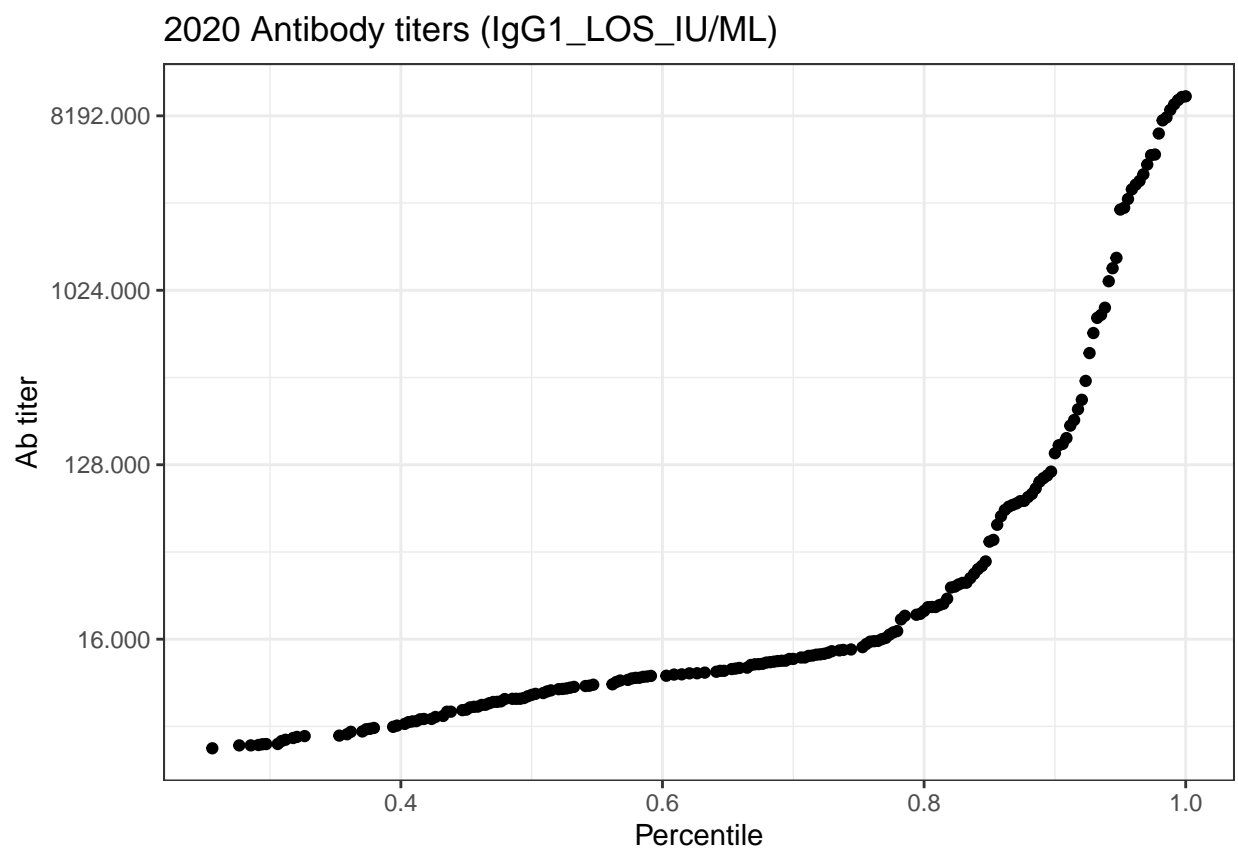
2020 Antibody titers (IgE_OVA_IU/ML)

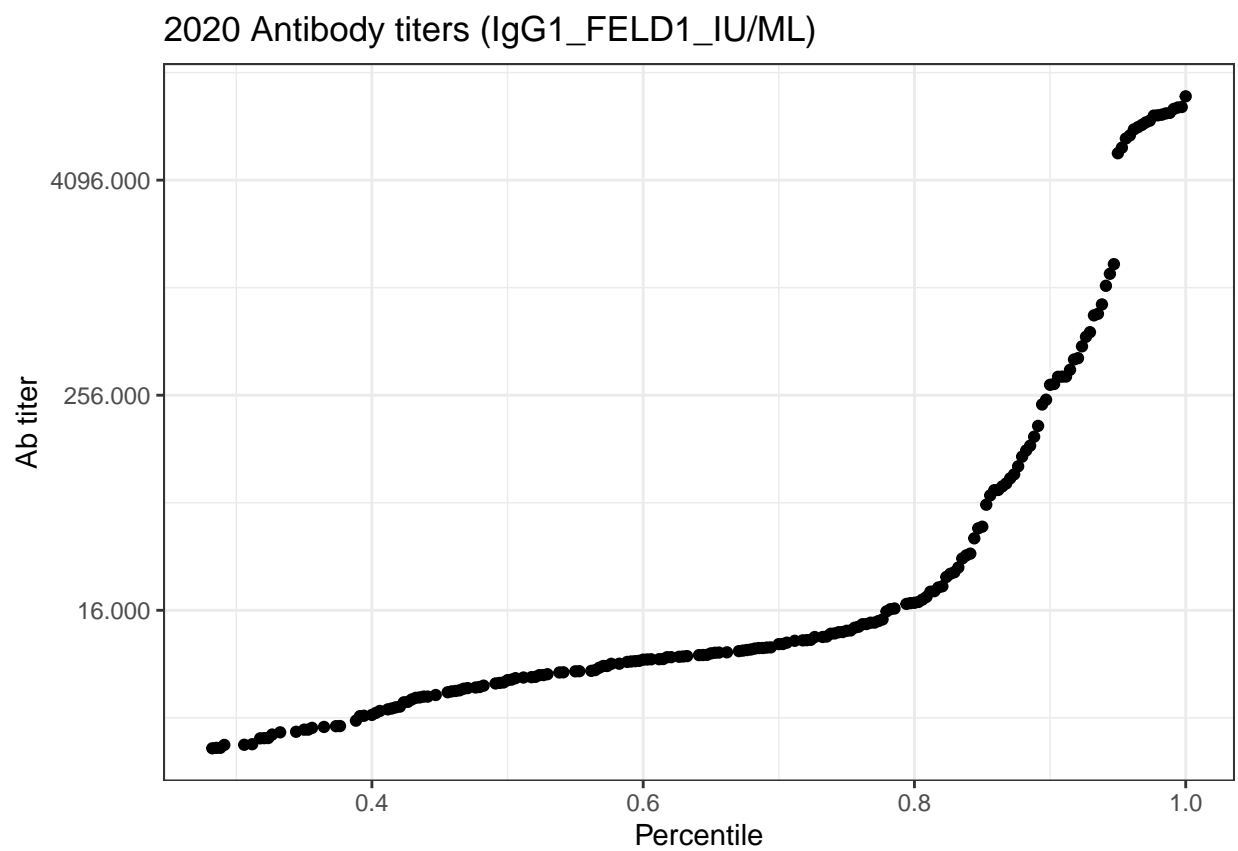


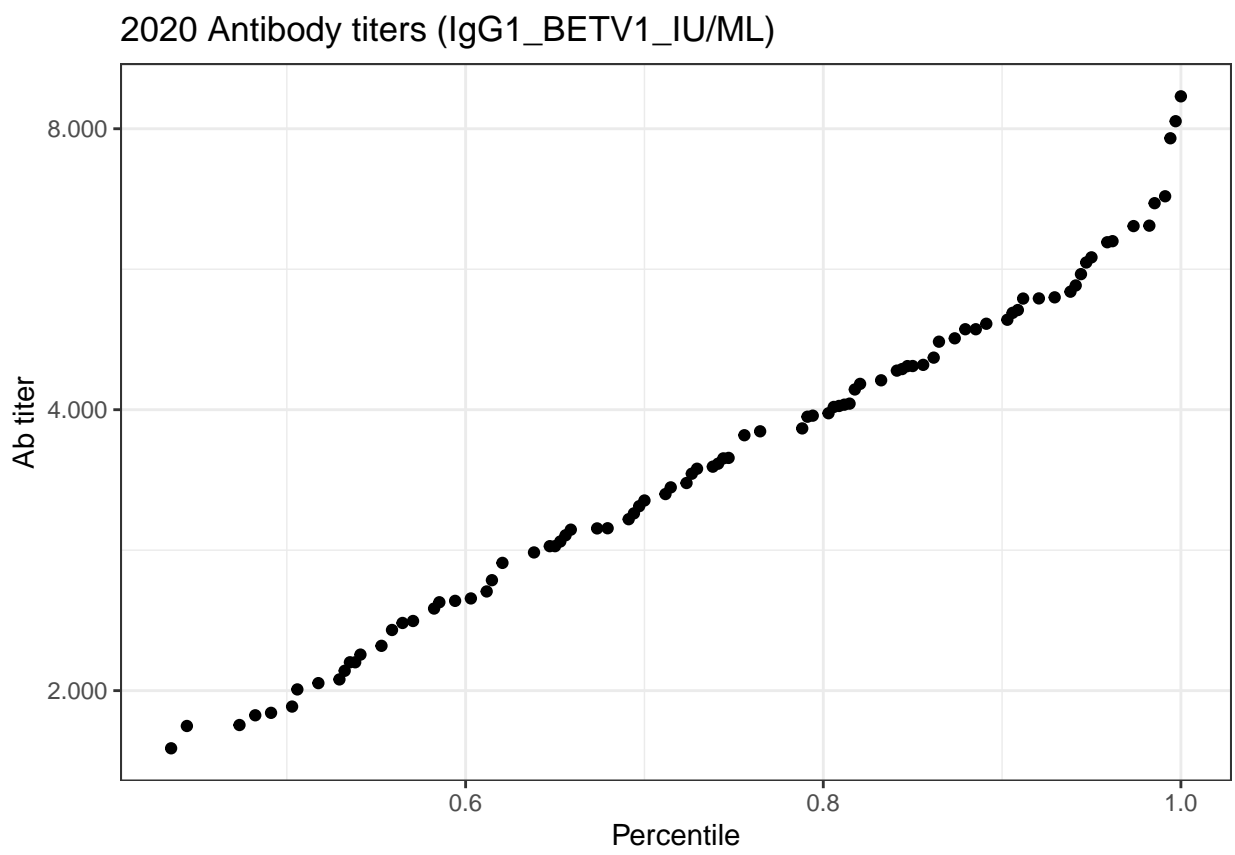


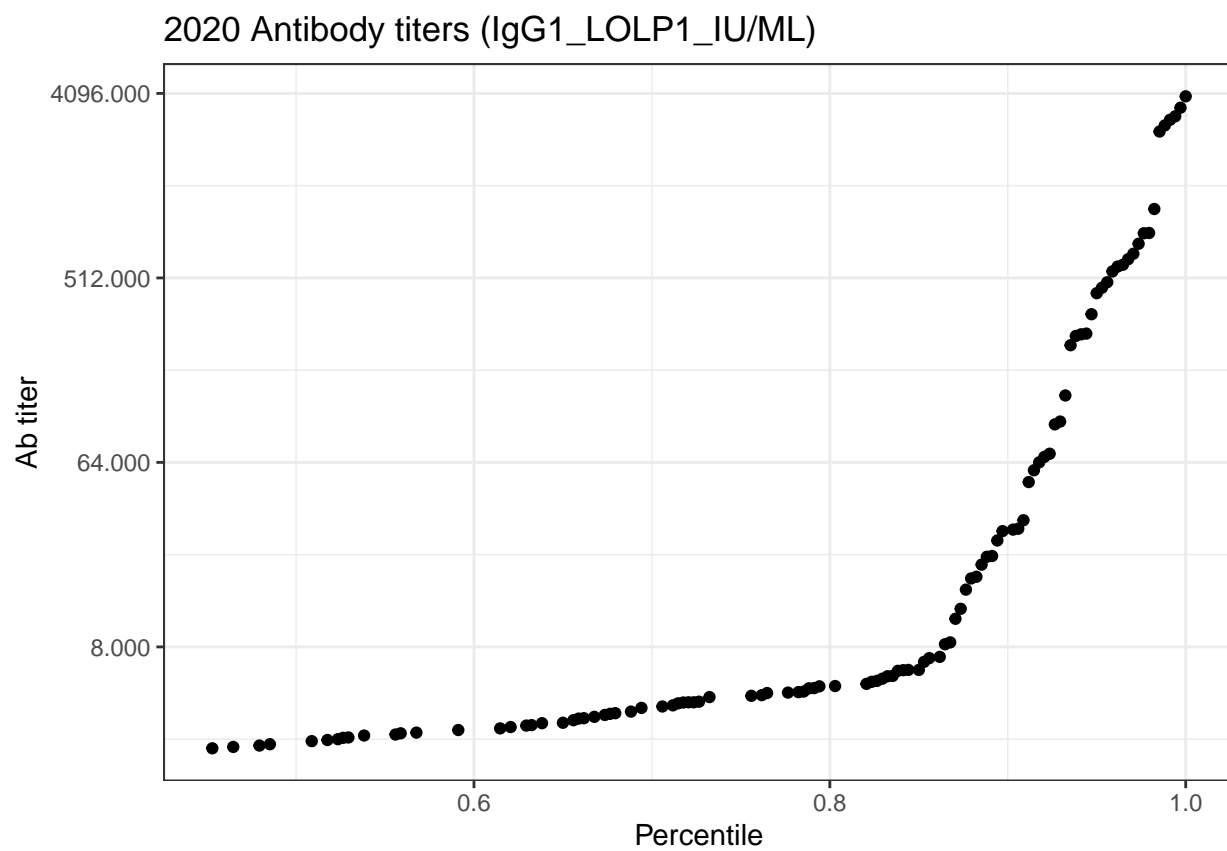
2020 Antibody titers (IgG1_ACT_IU/ML)

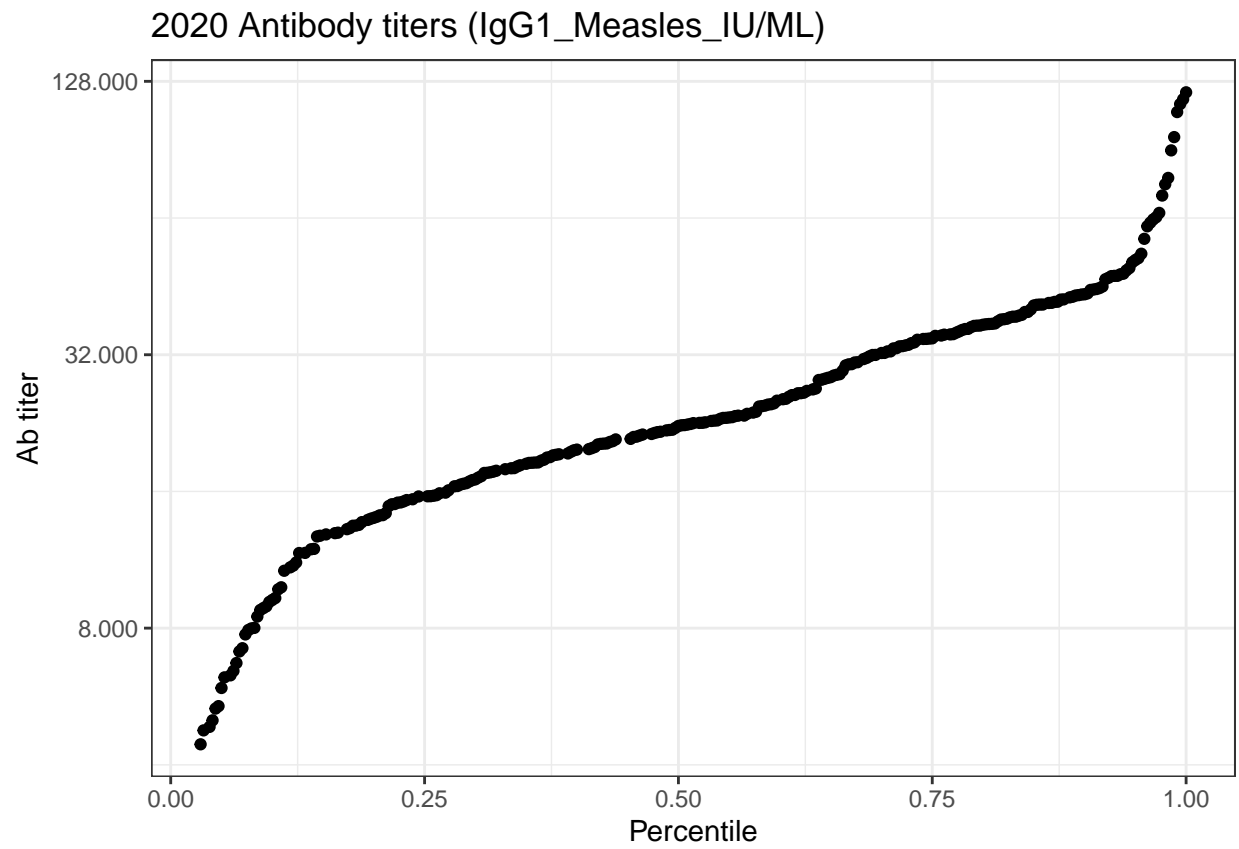


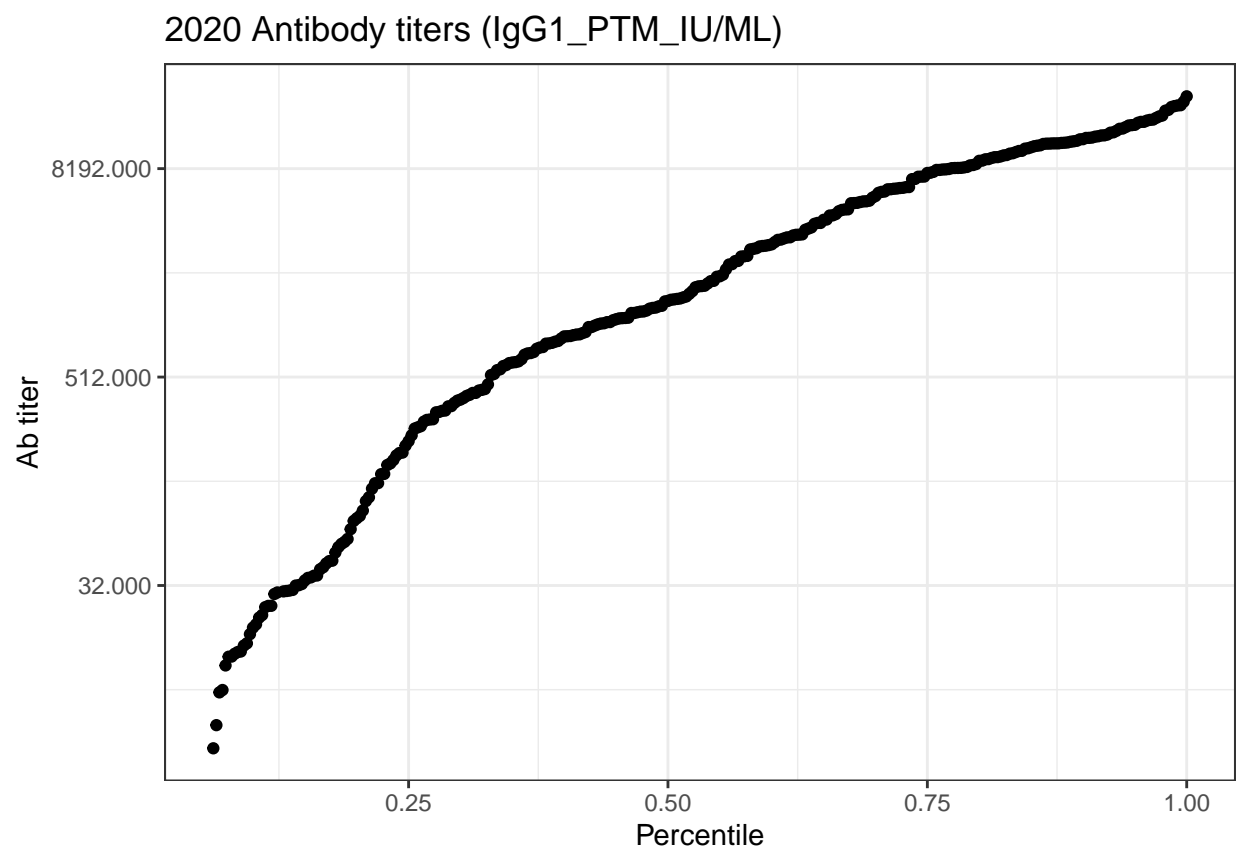


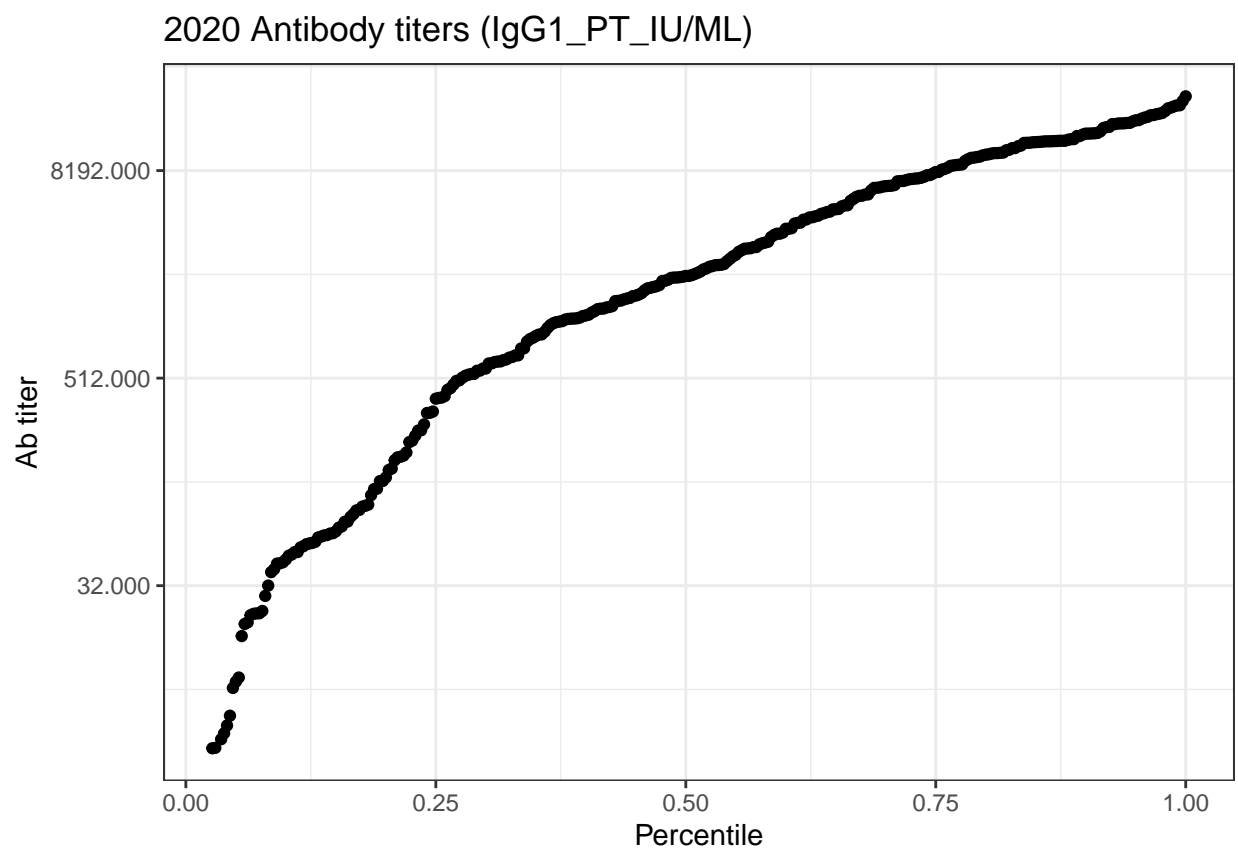


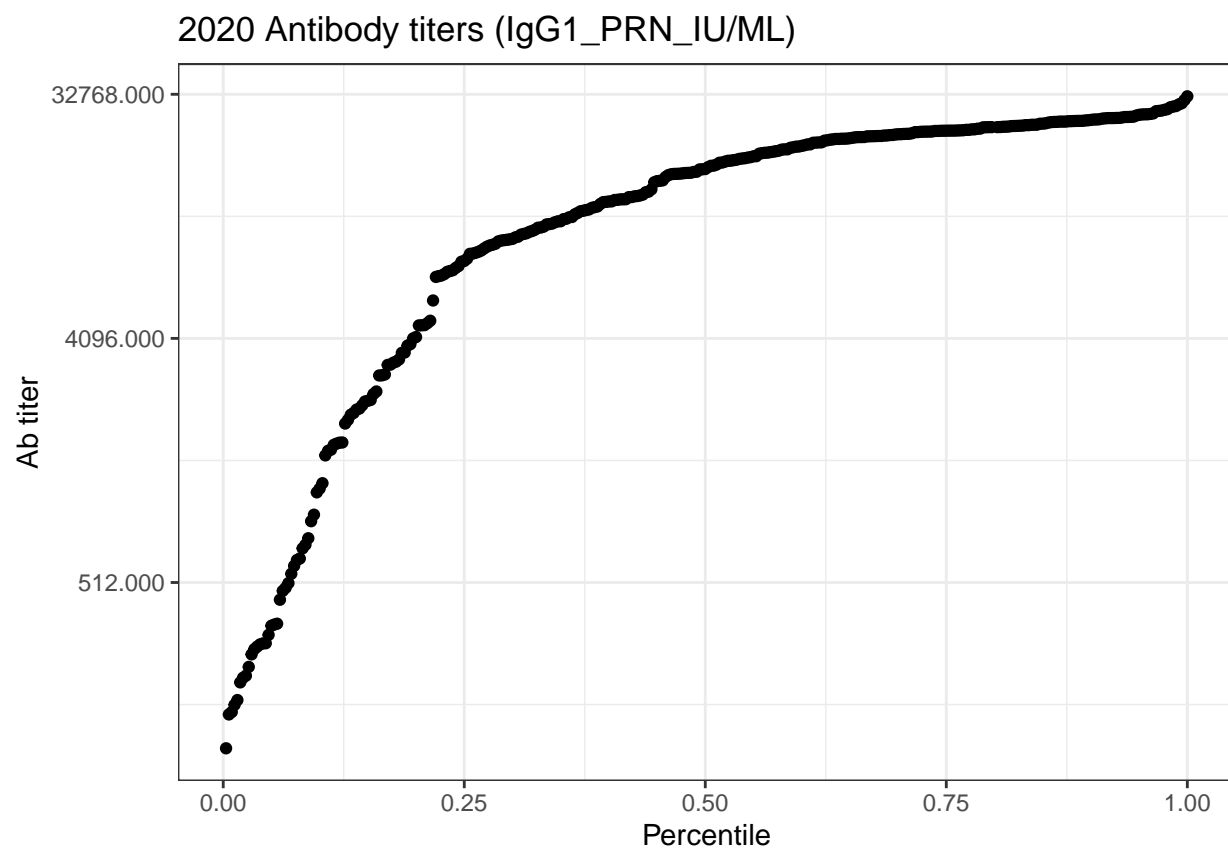


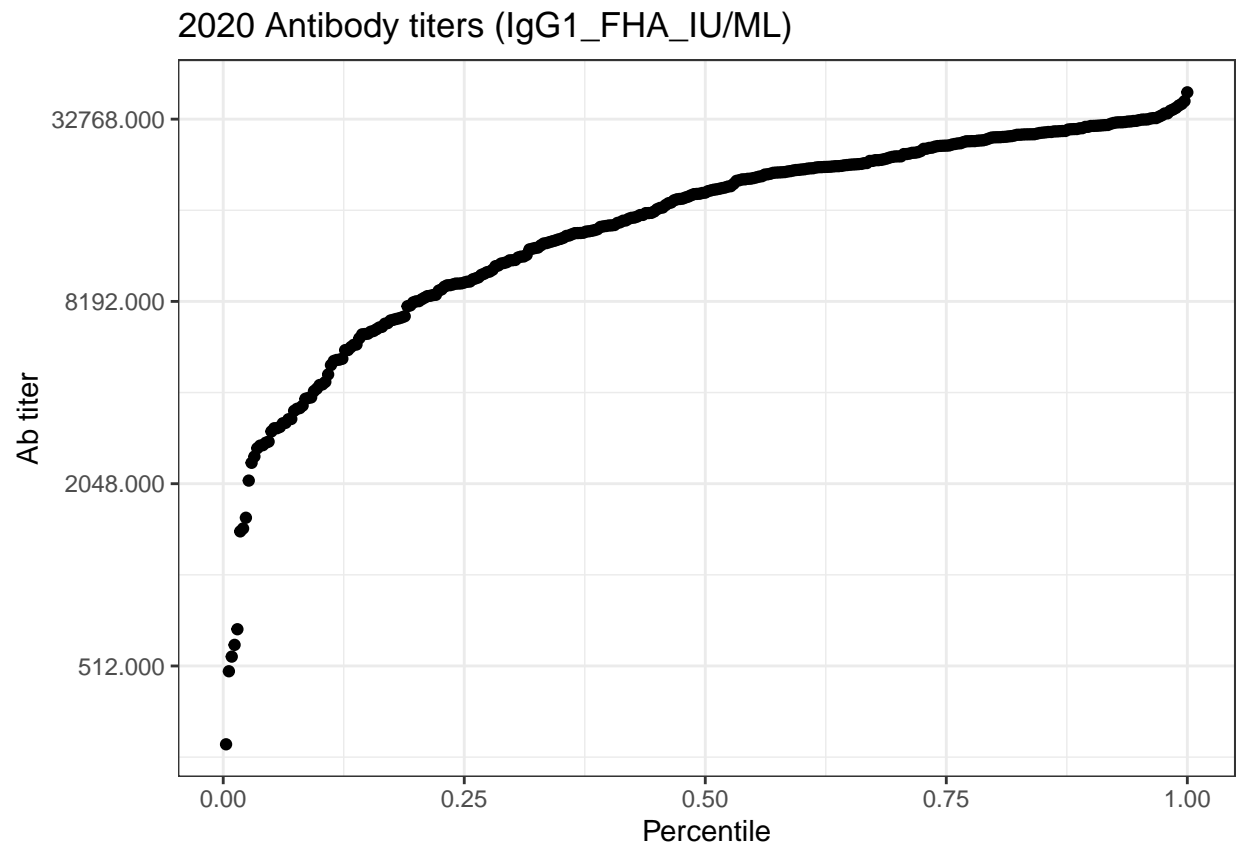


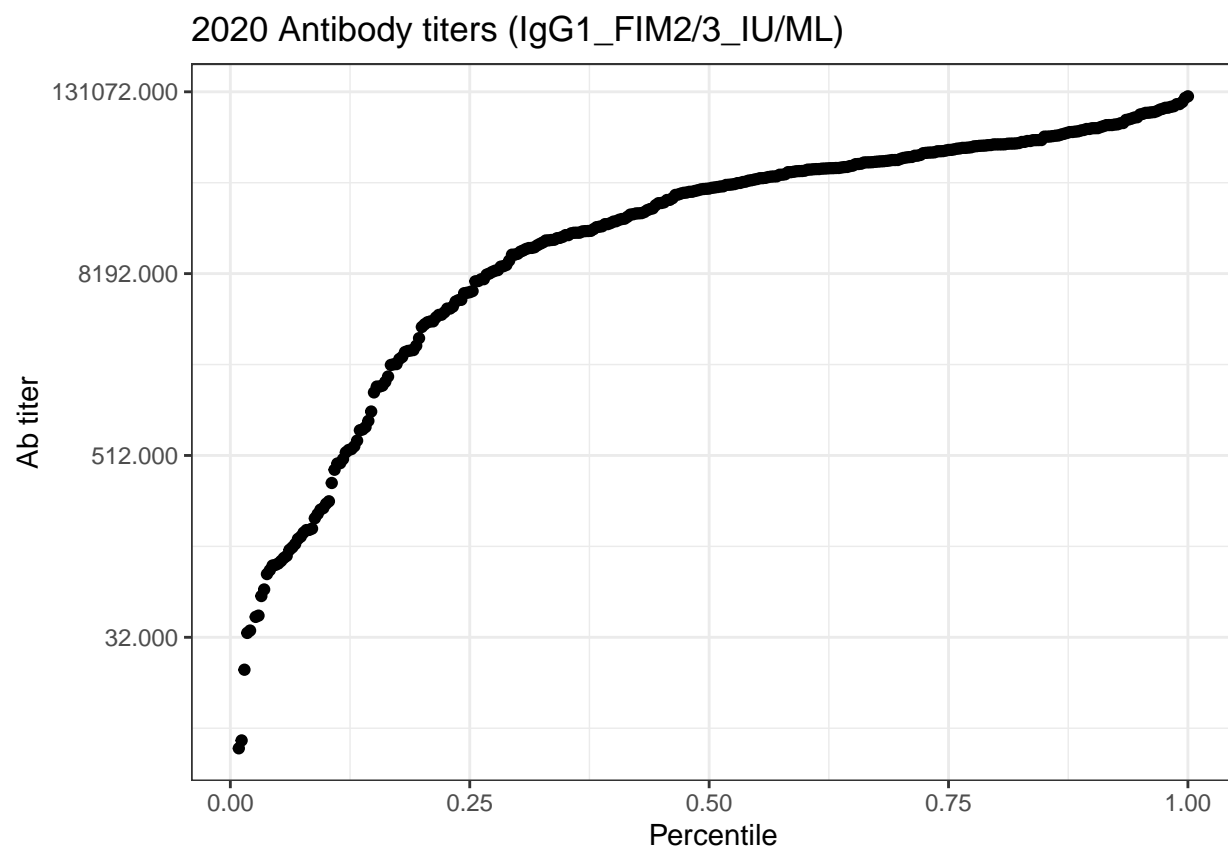


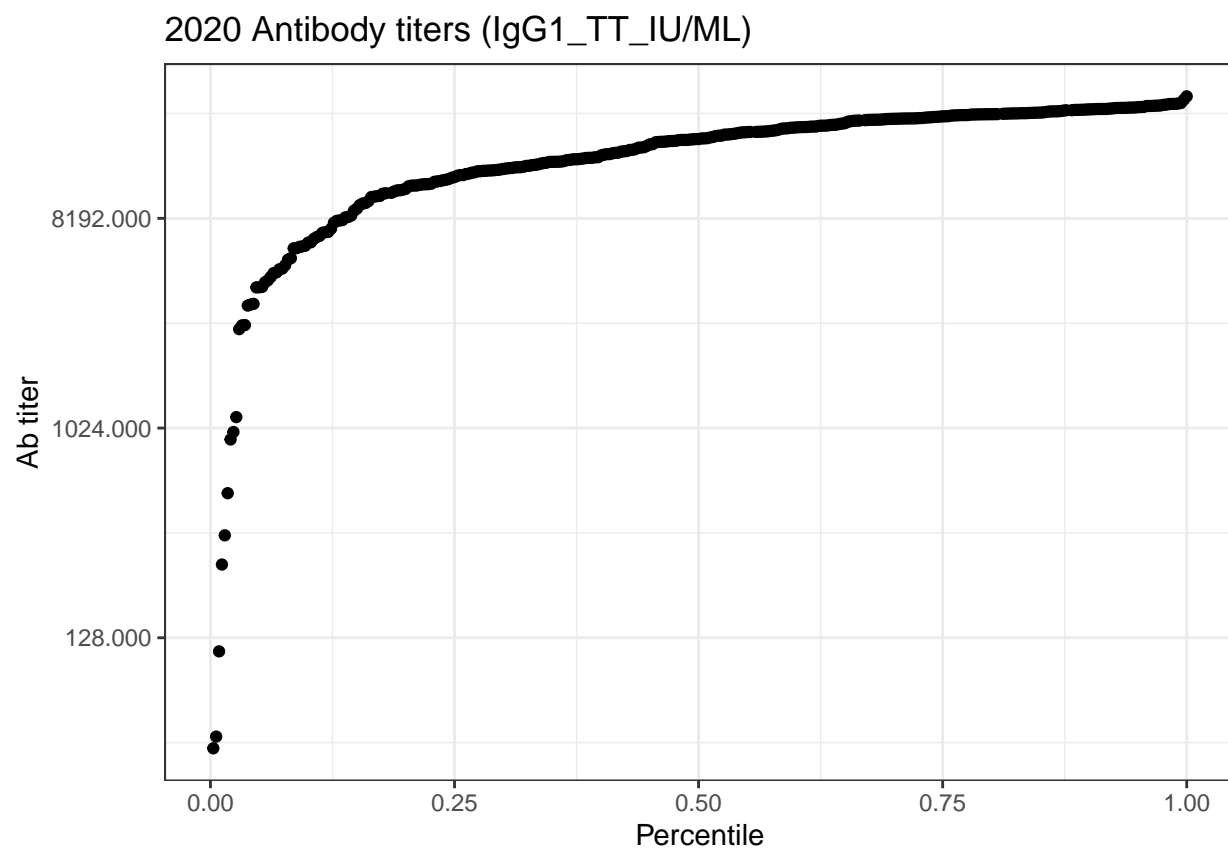


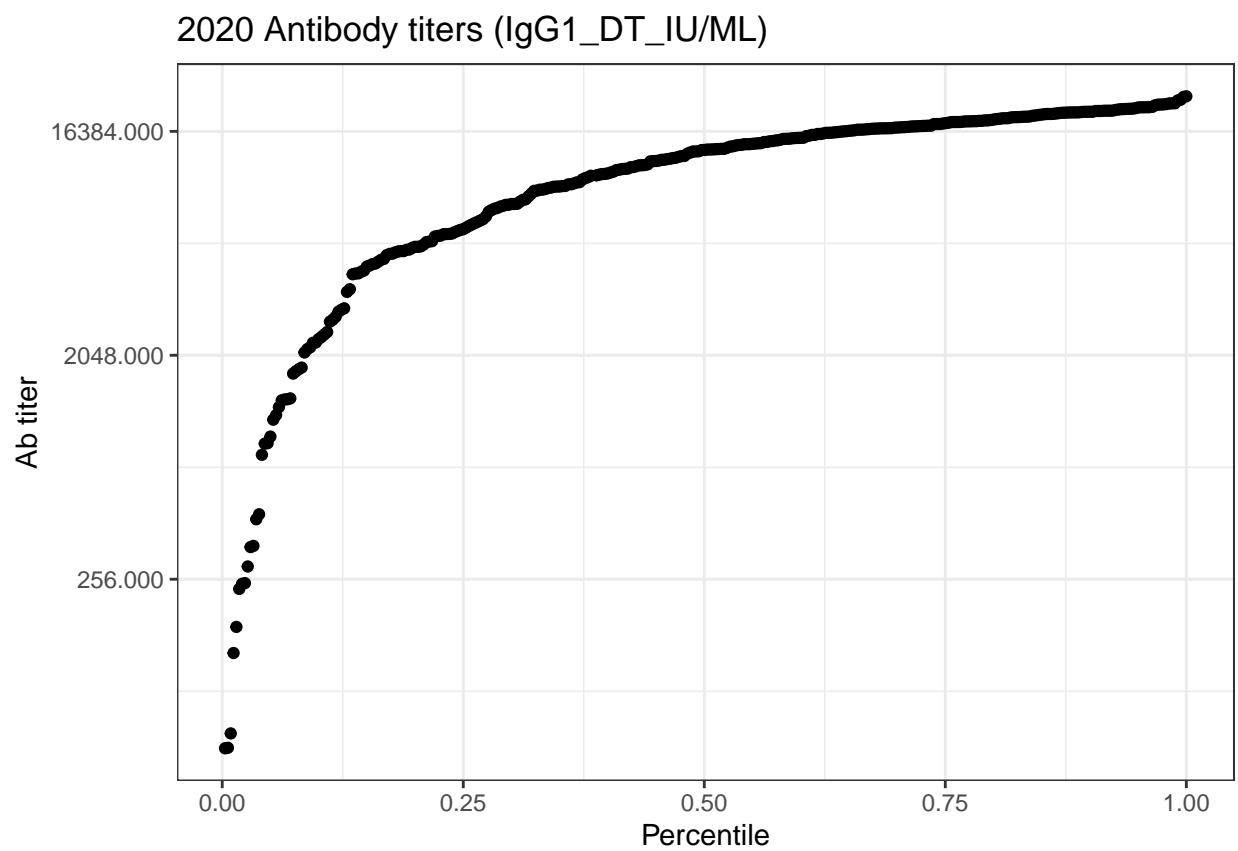




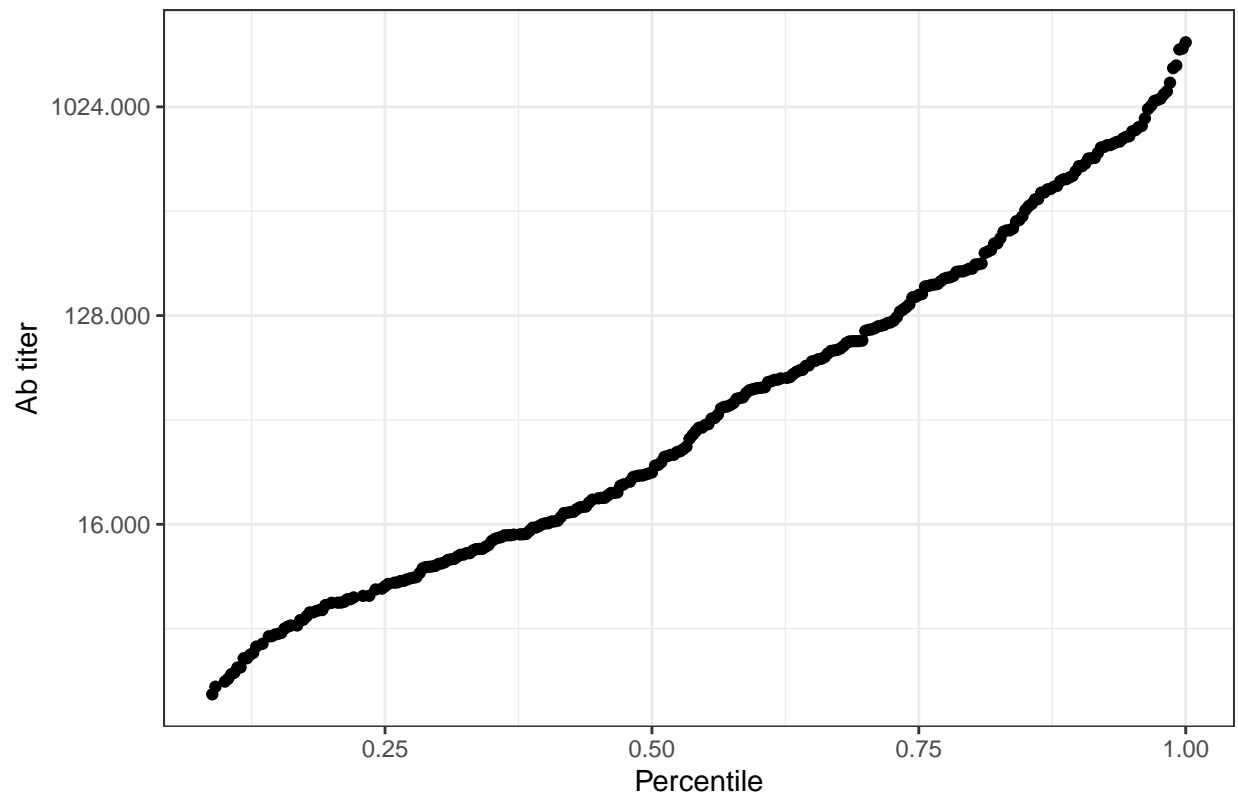




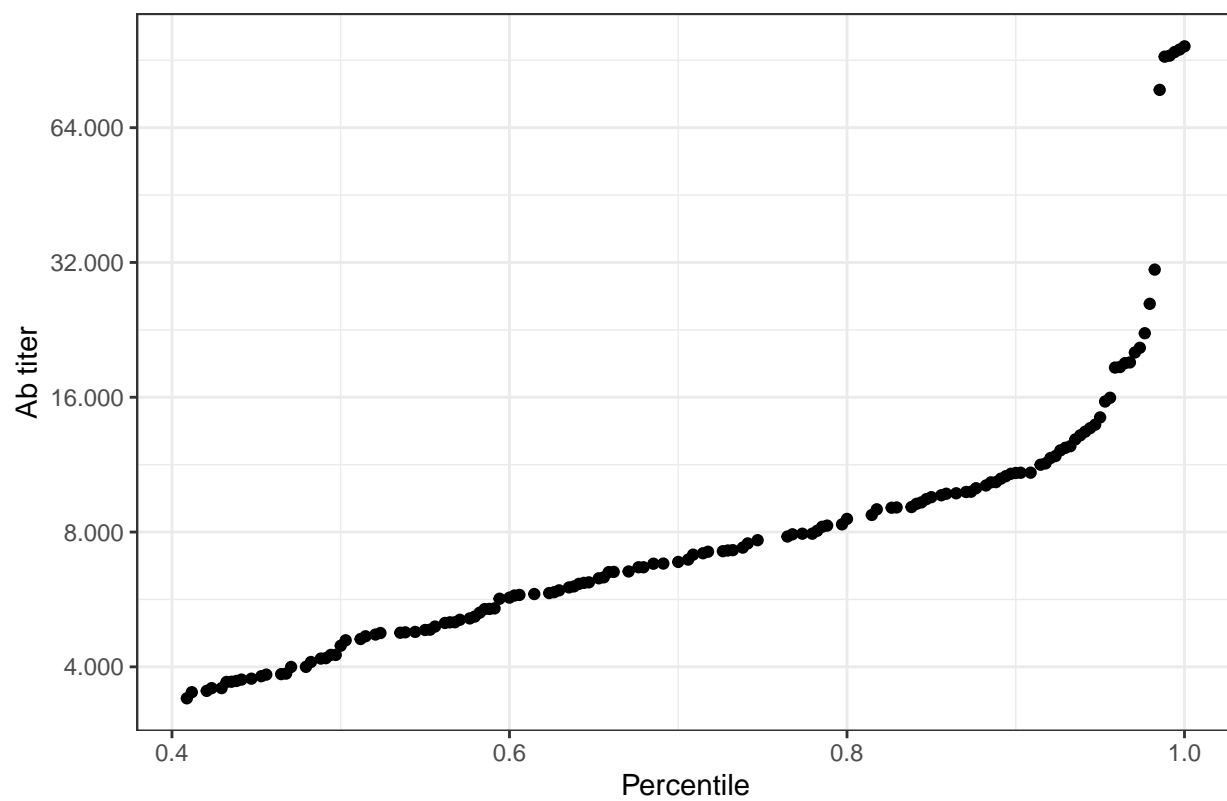


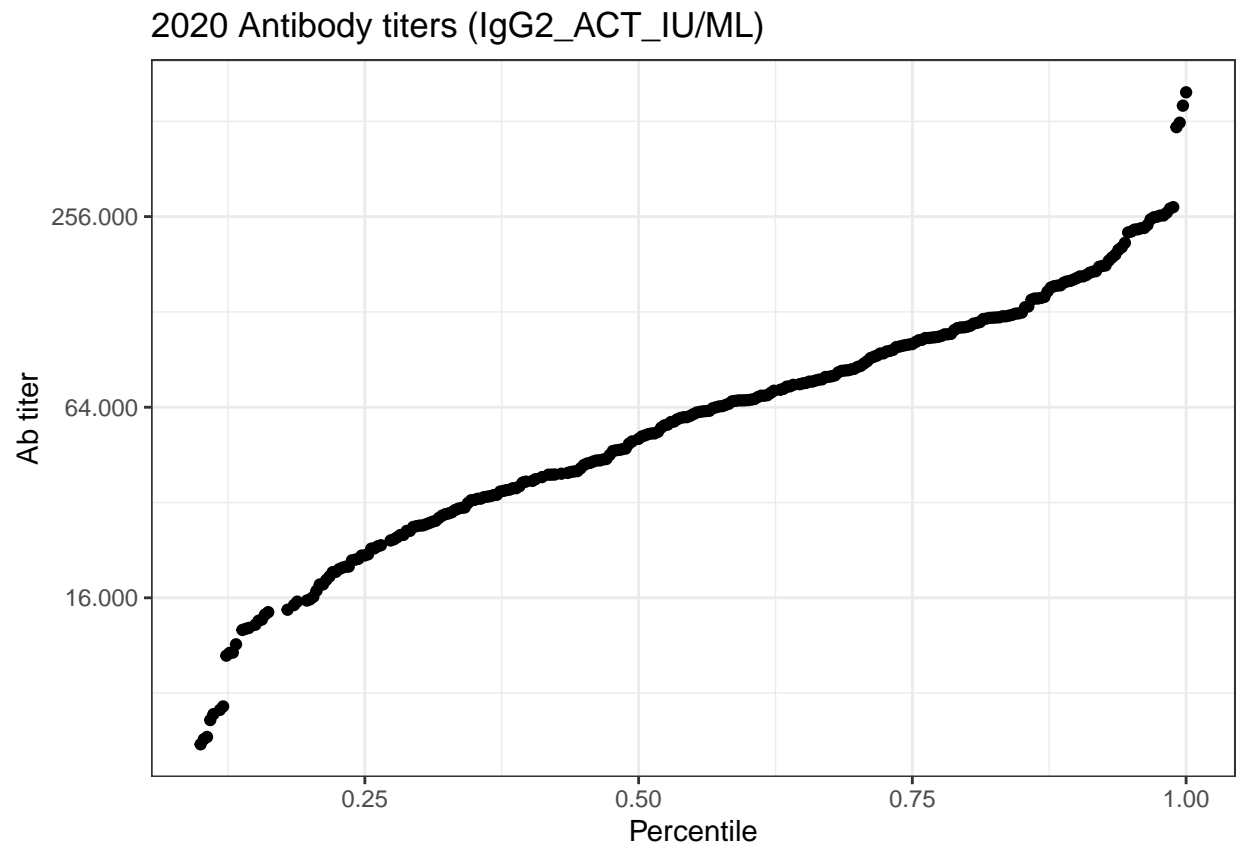


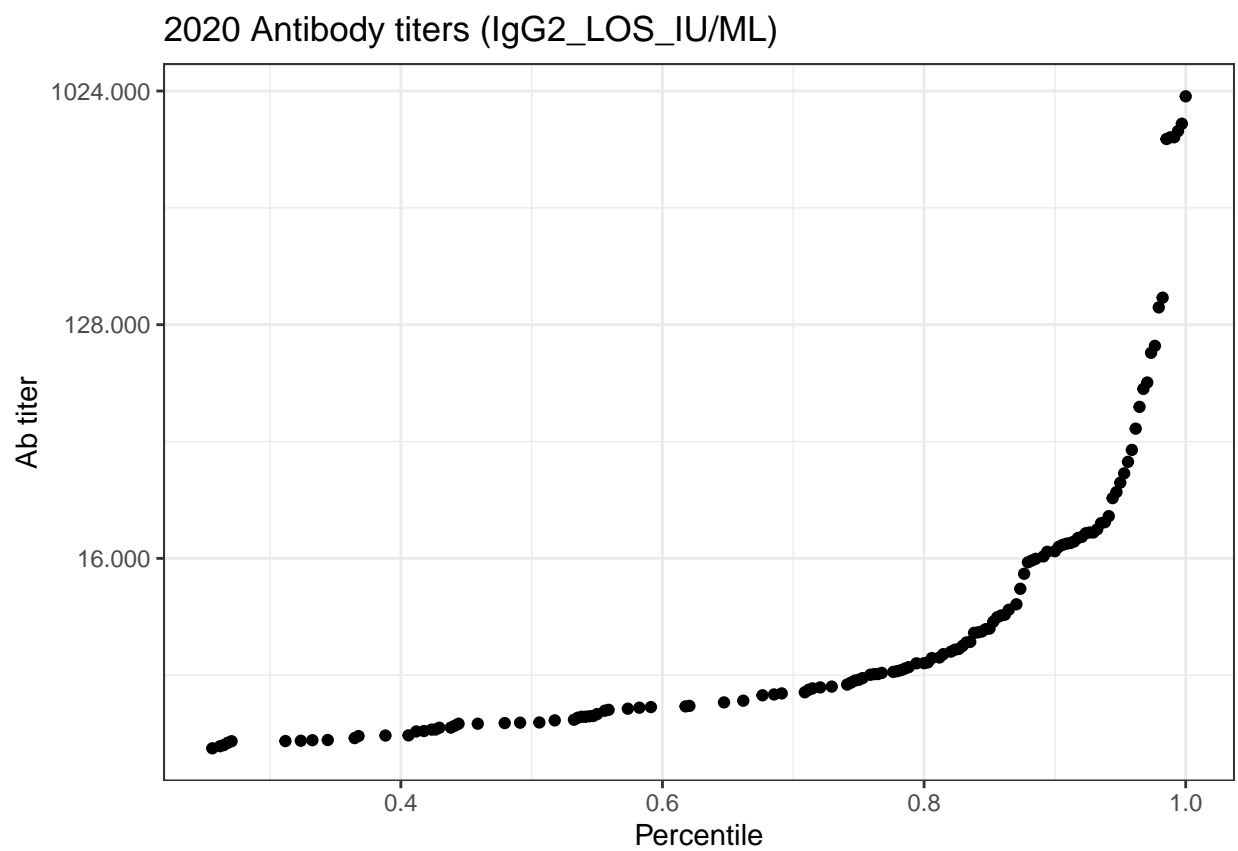
2020 Antibody titers (IgG1_OVA_IU/ML)

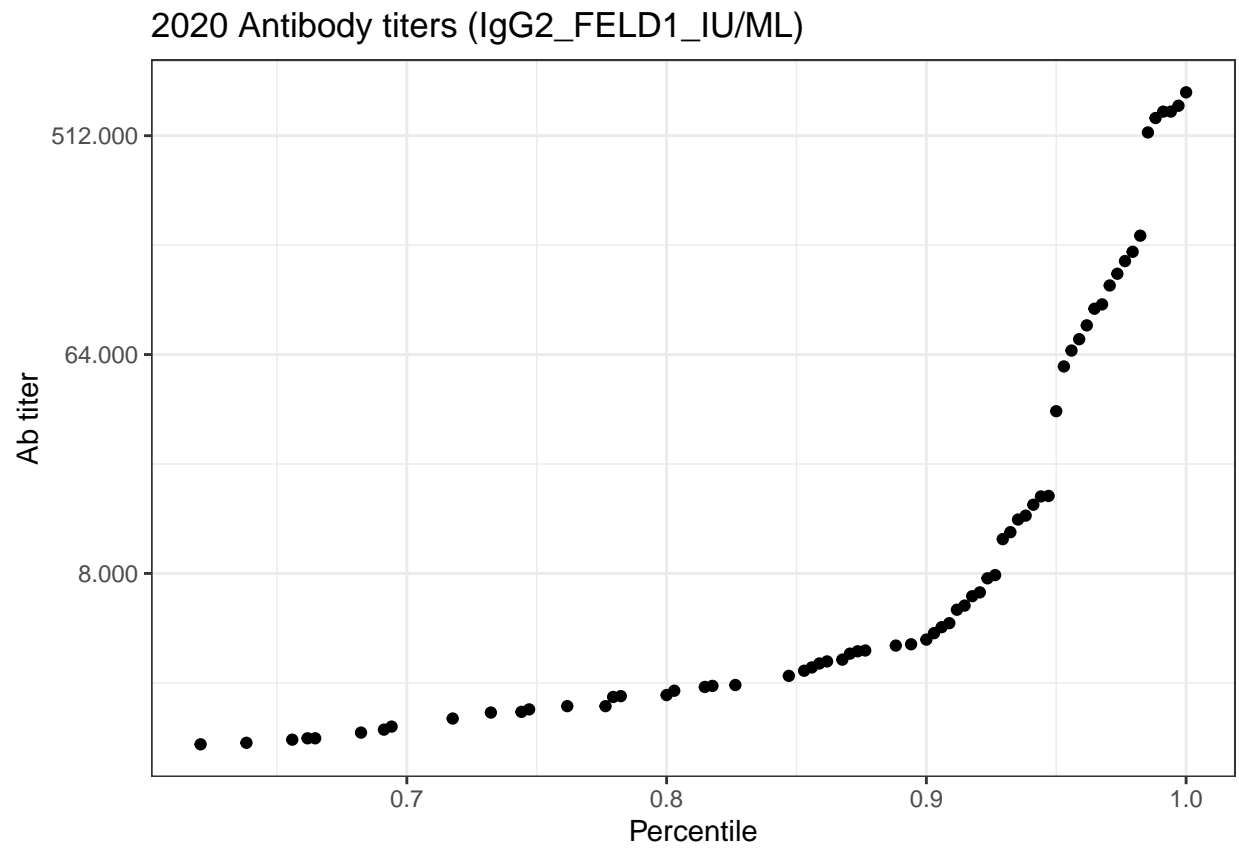


2020 Antibody titers (IgG1_PD1_IU/ML)

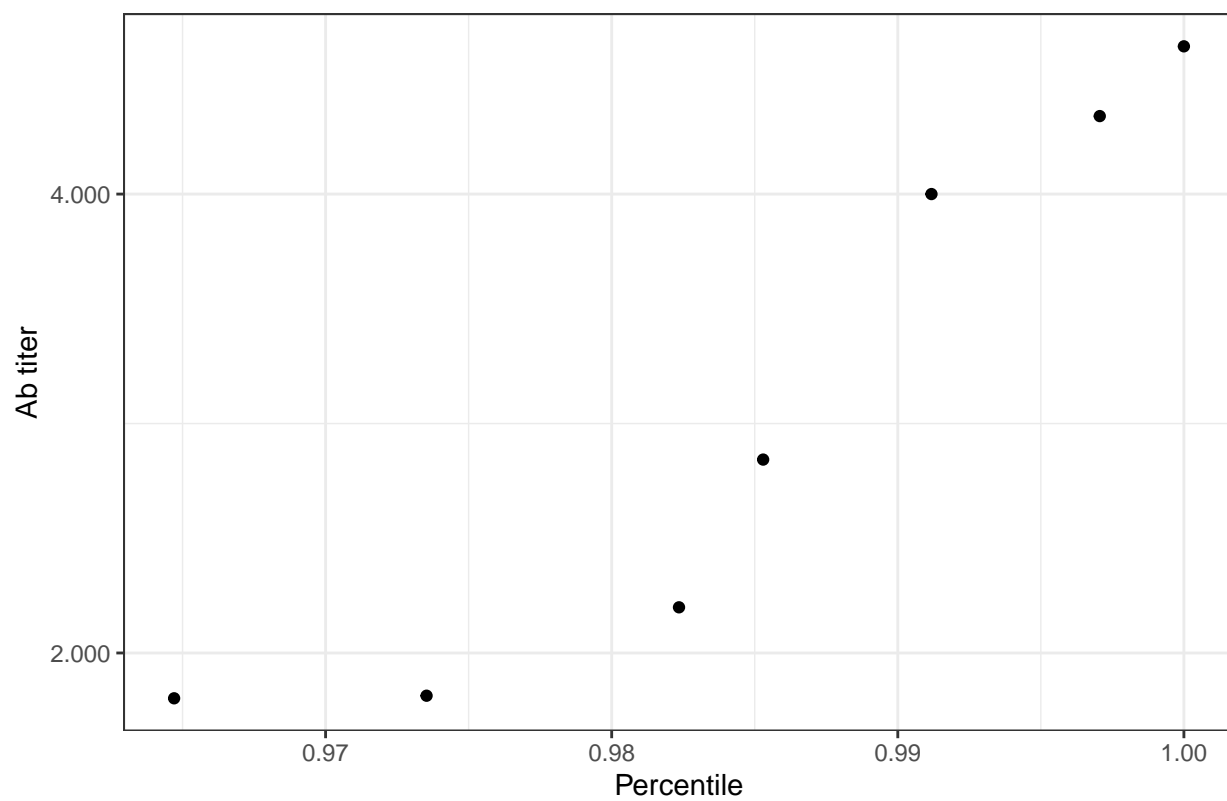


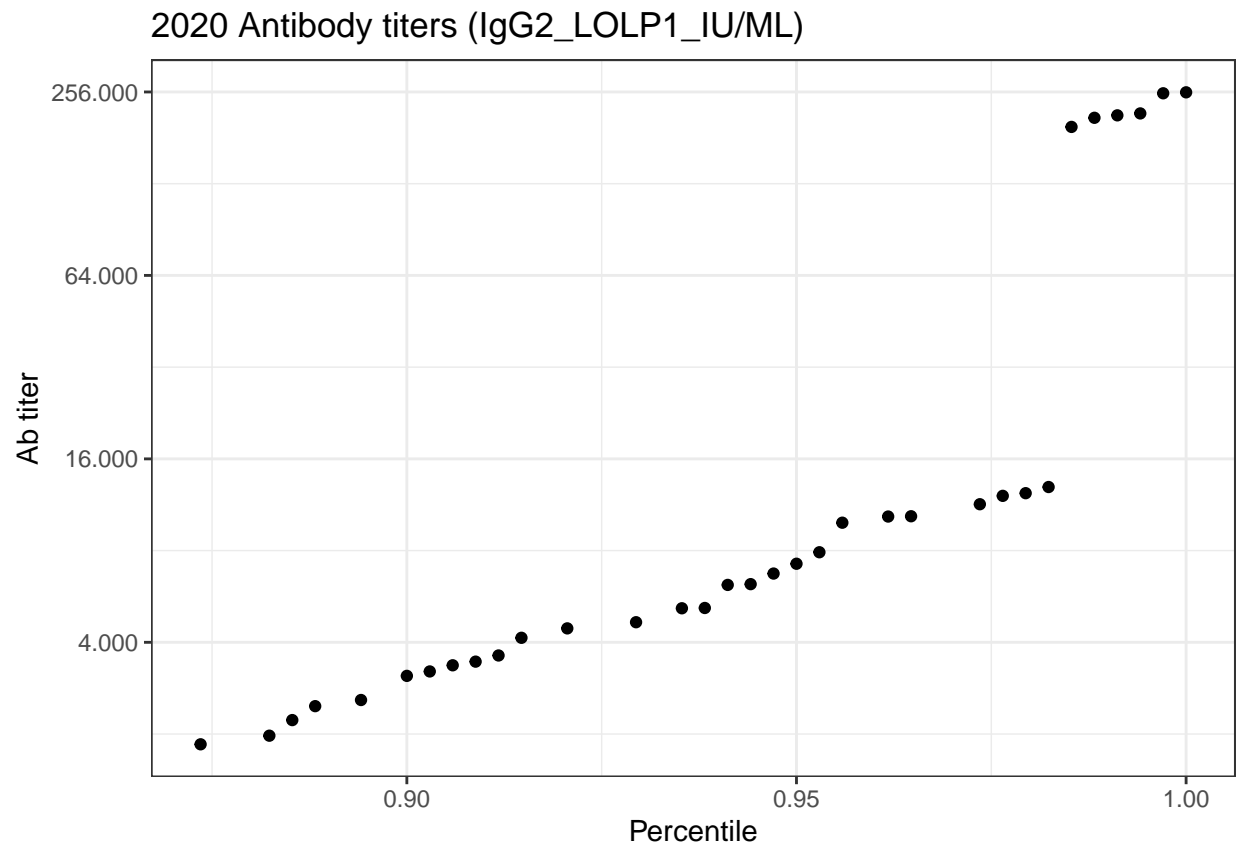


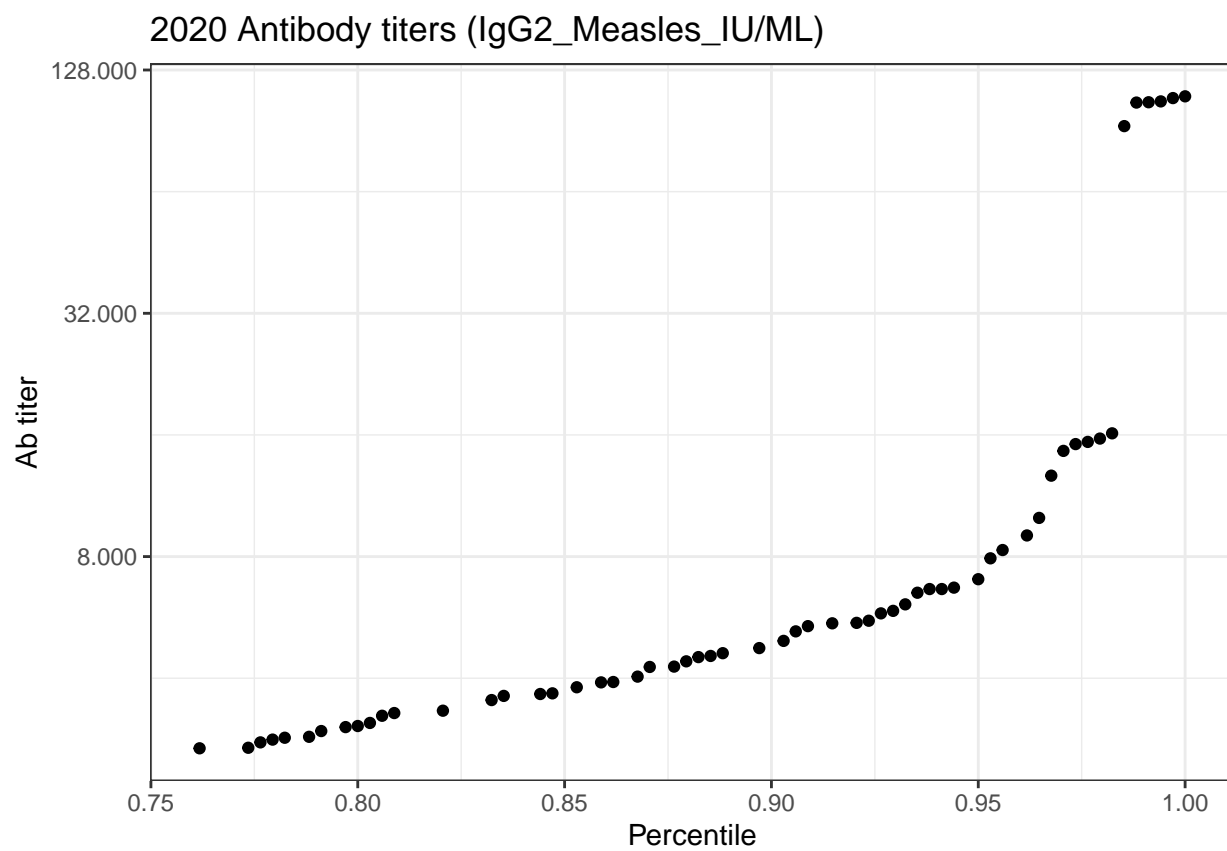


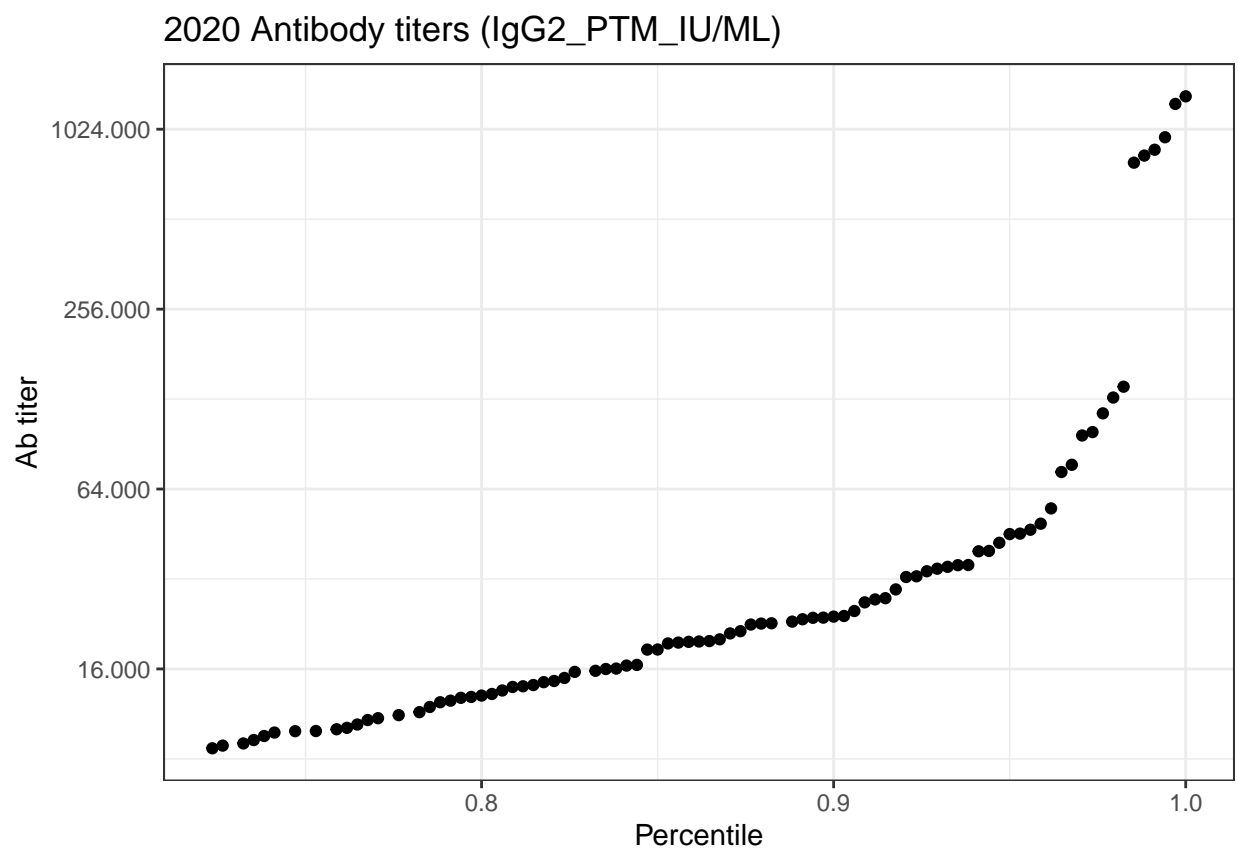


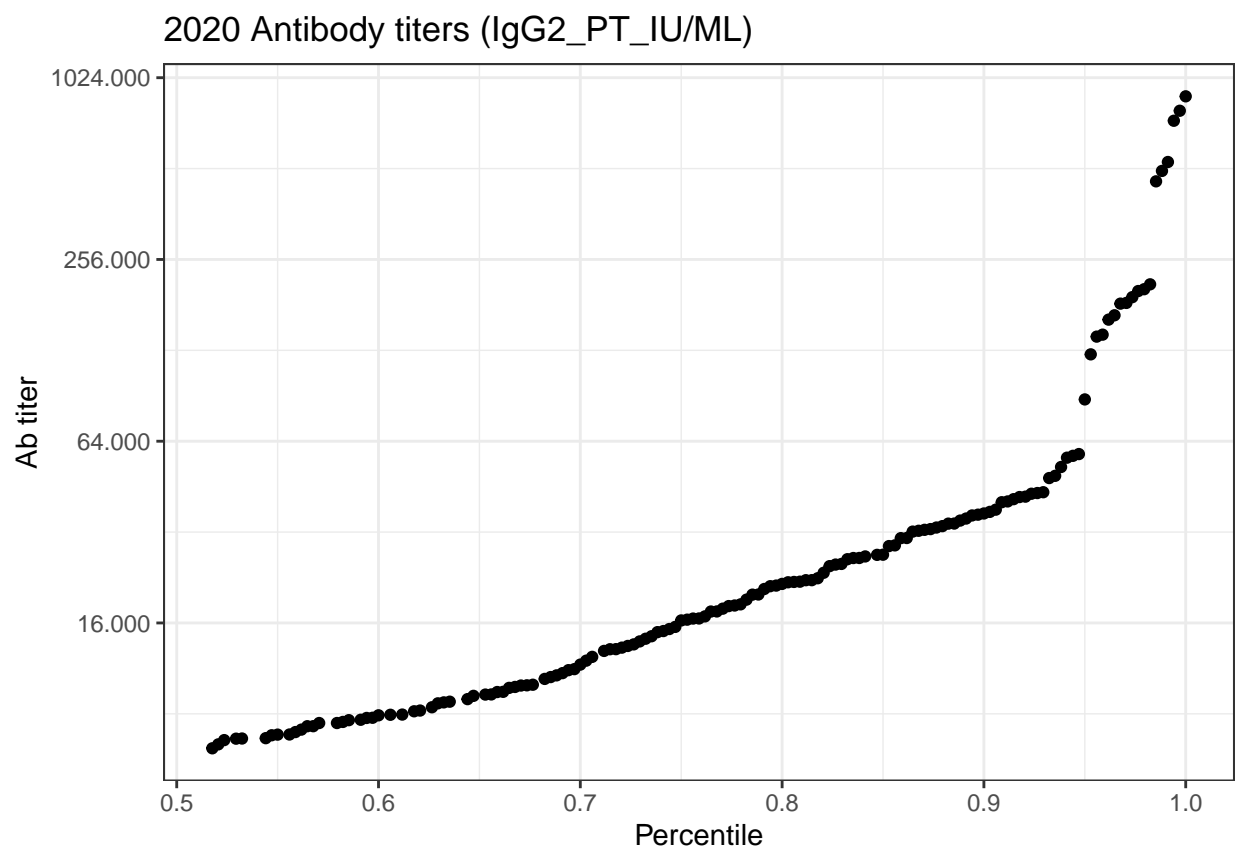
2020 Antibody titers (IgG2_BETV1_IU/ML)



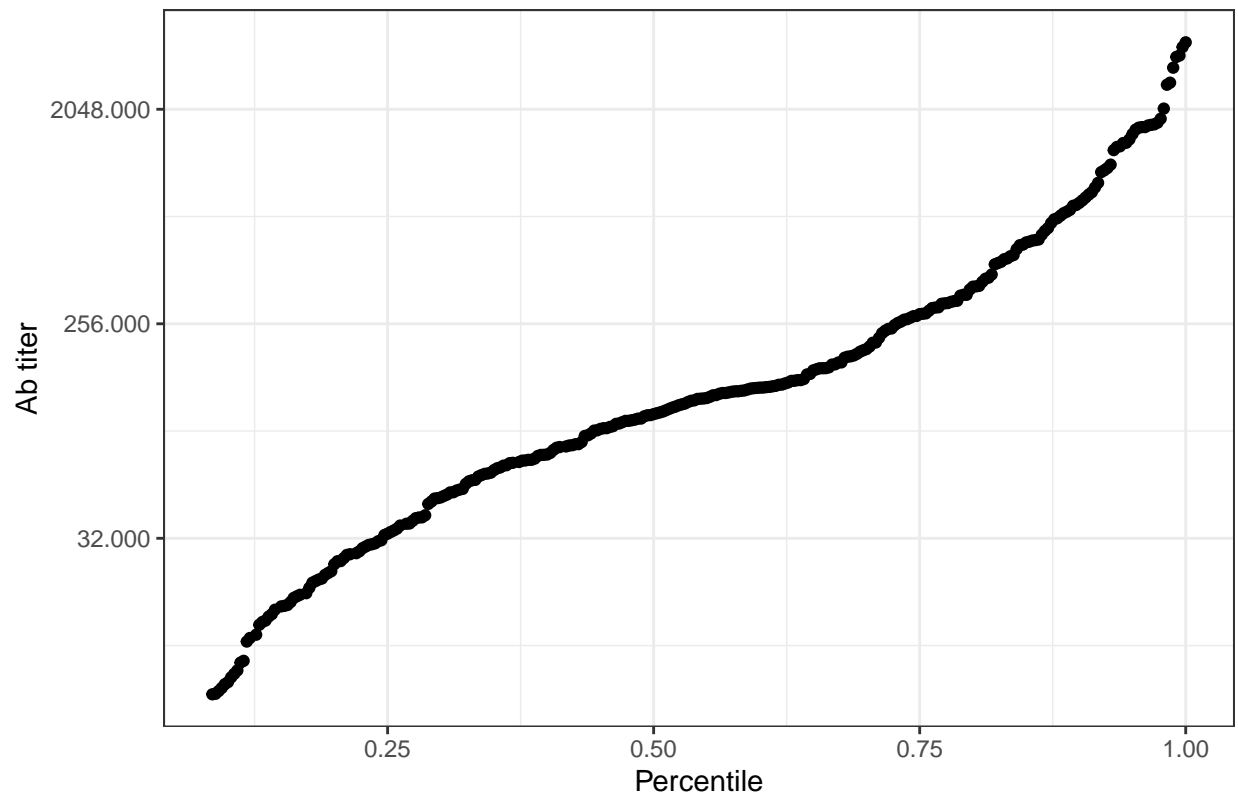


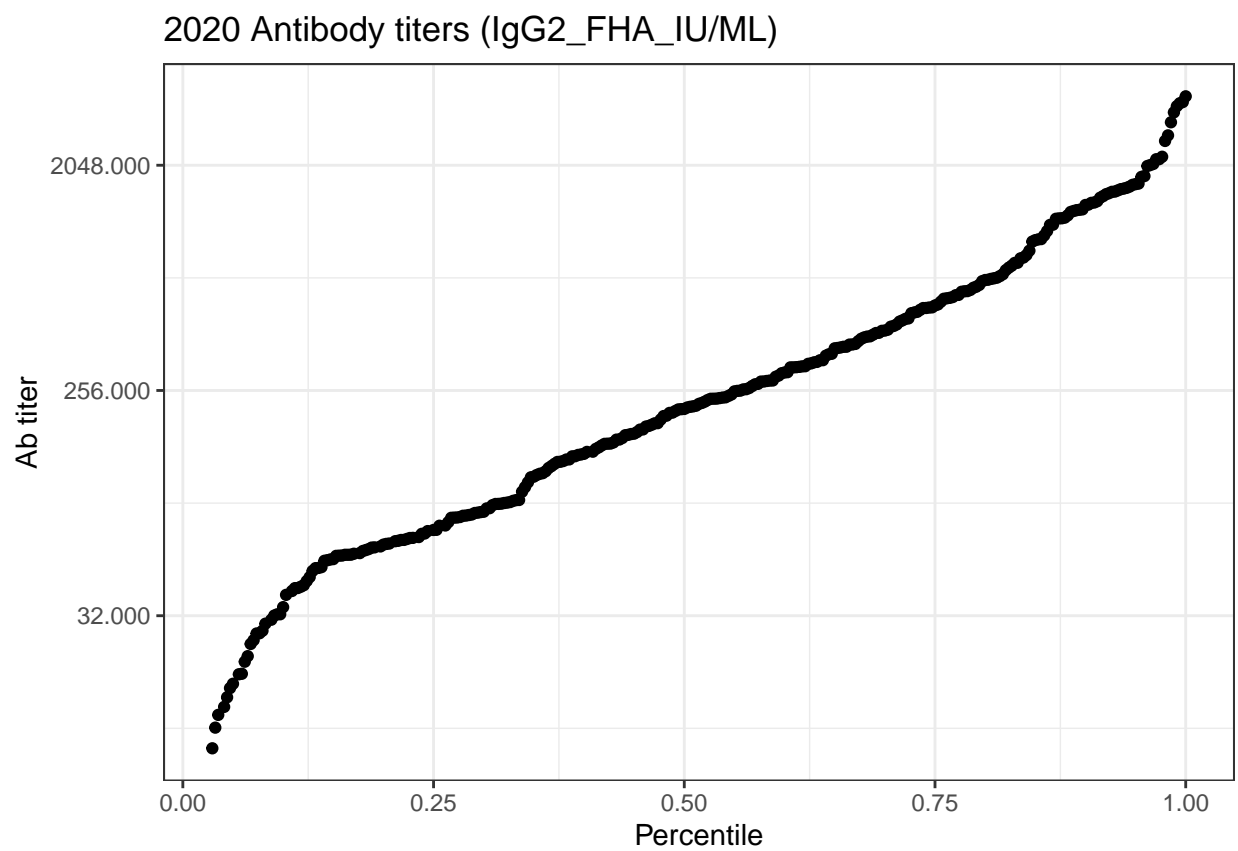


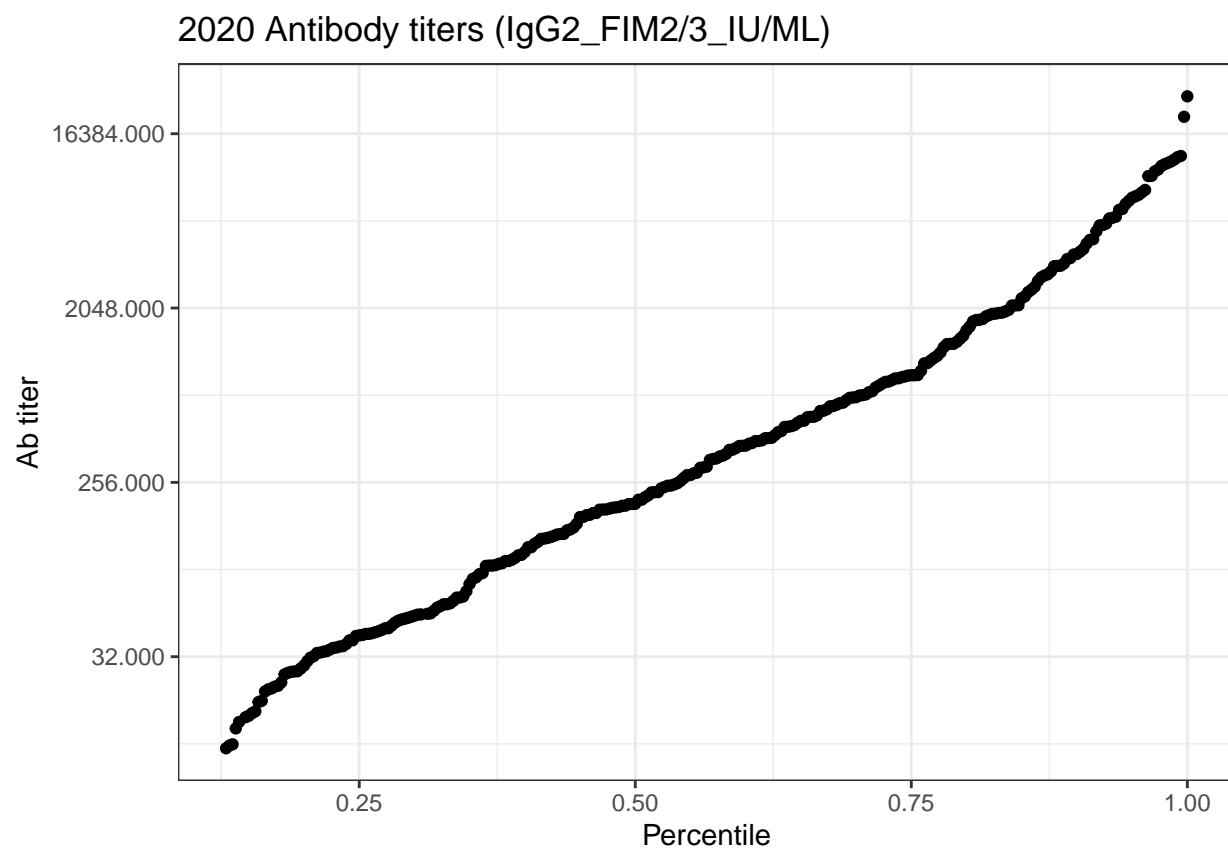


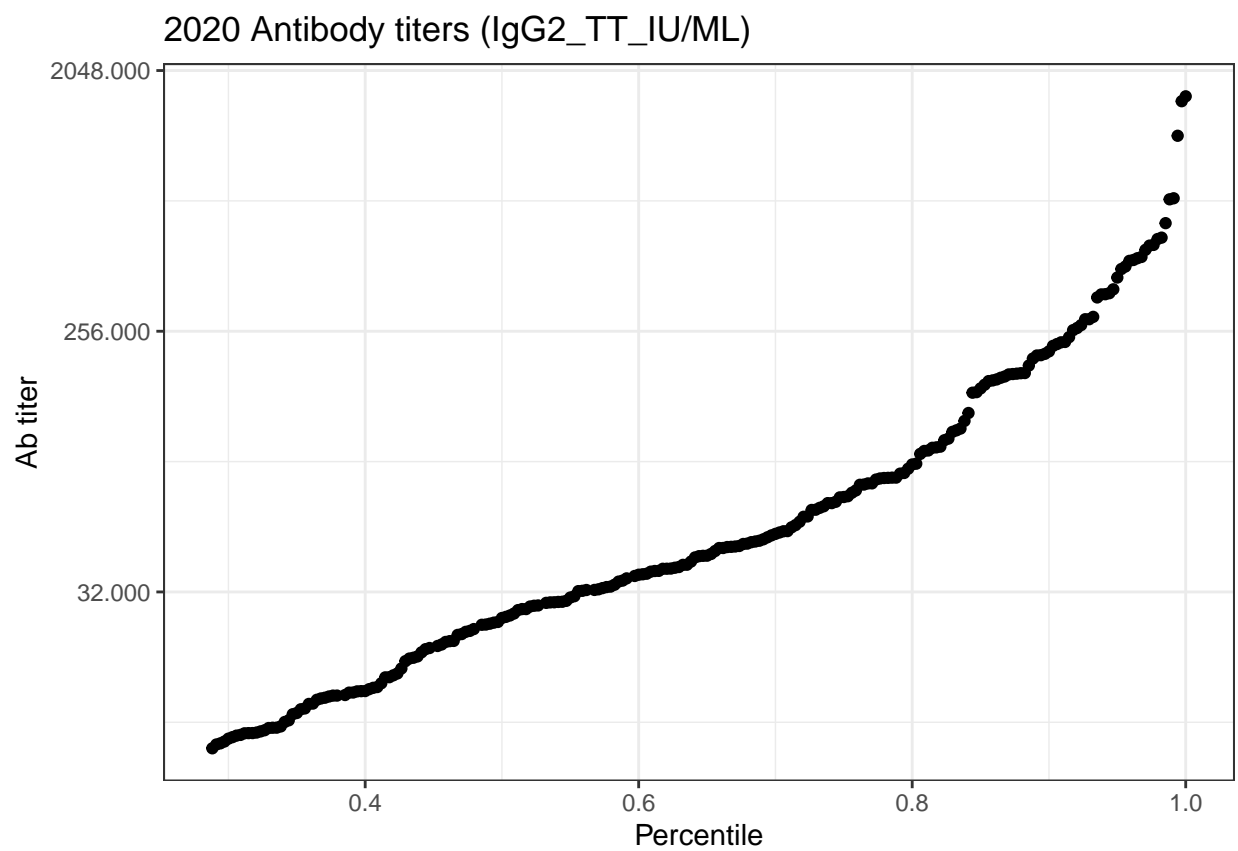


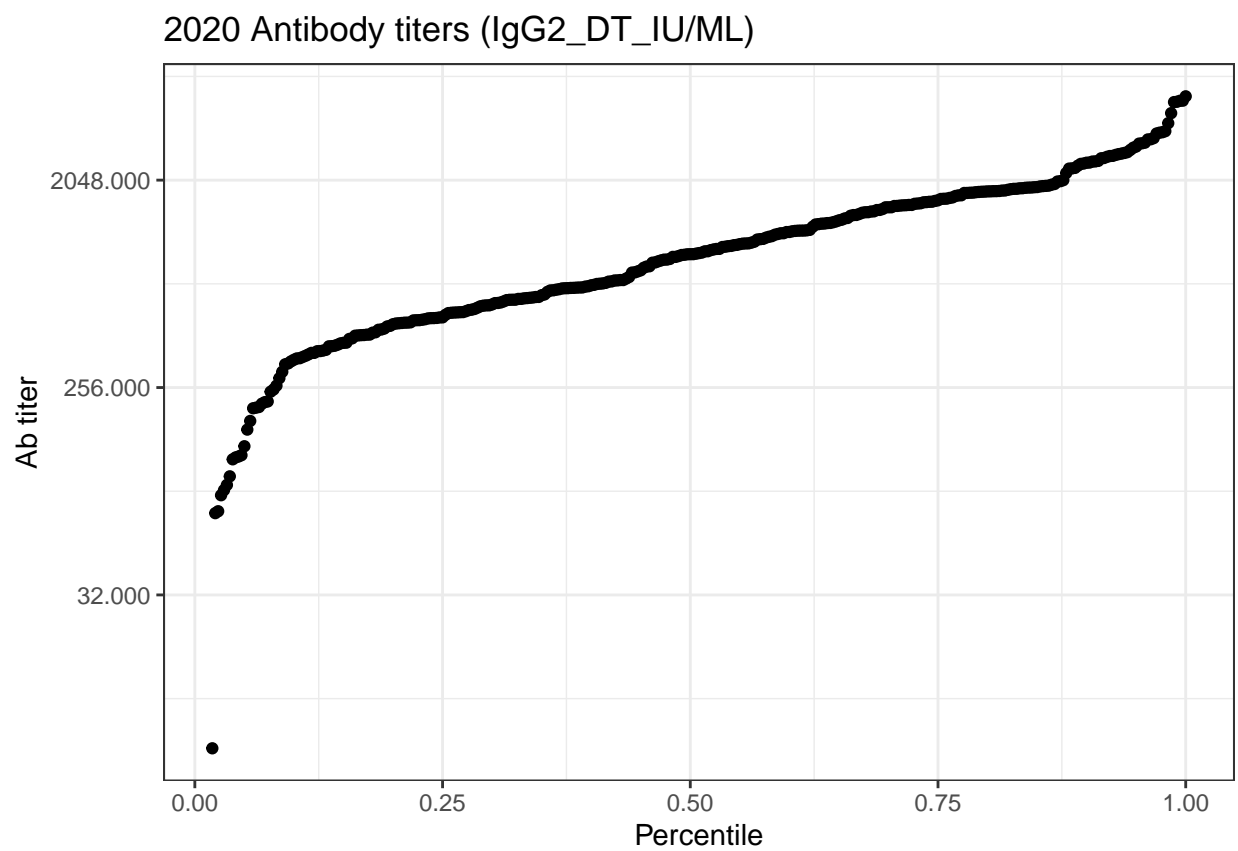
2020 Antibody titers (IgG2_PRN_IU/ML)

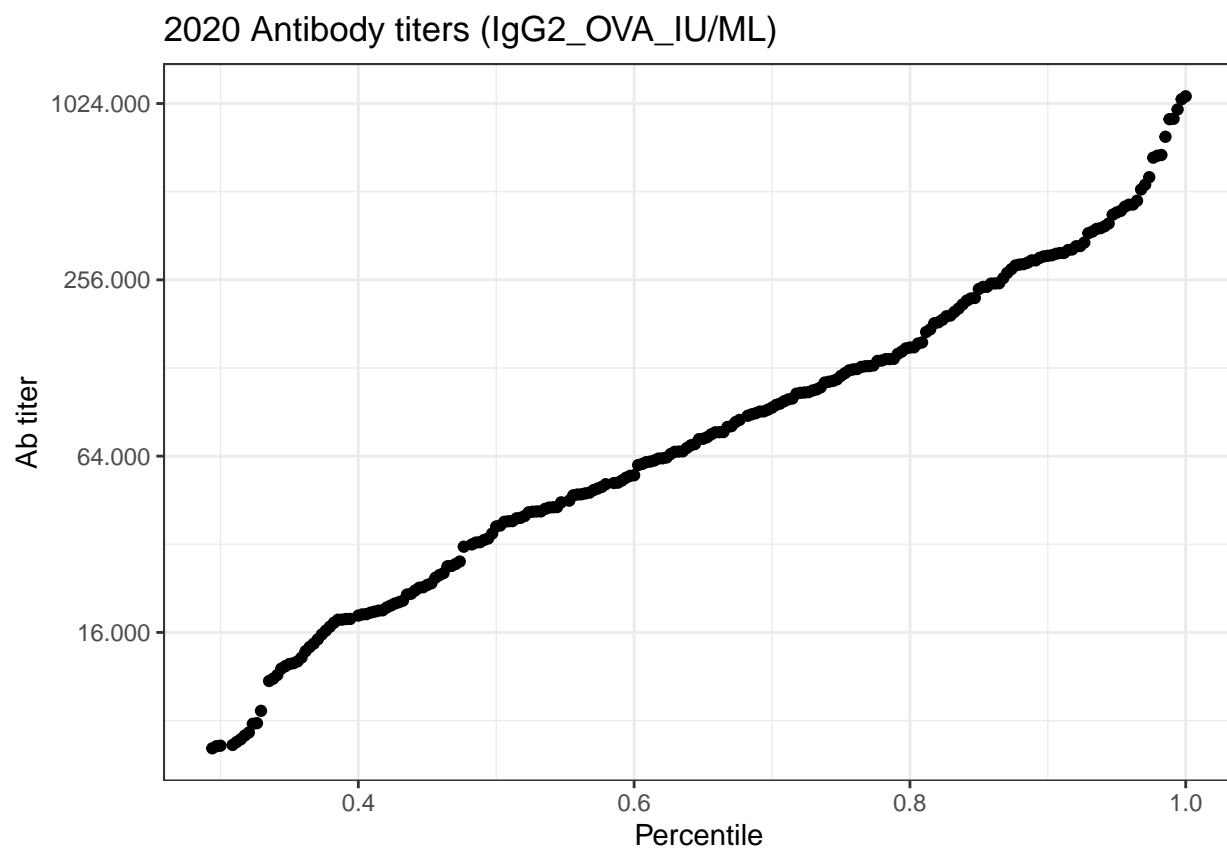


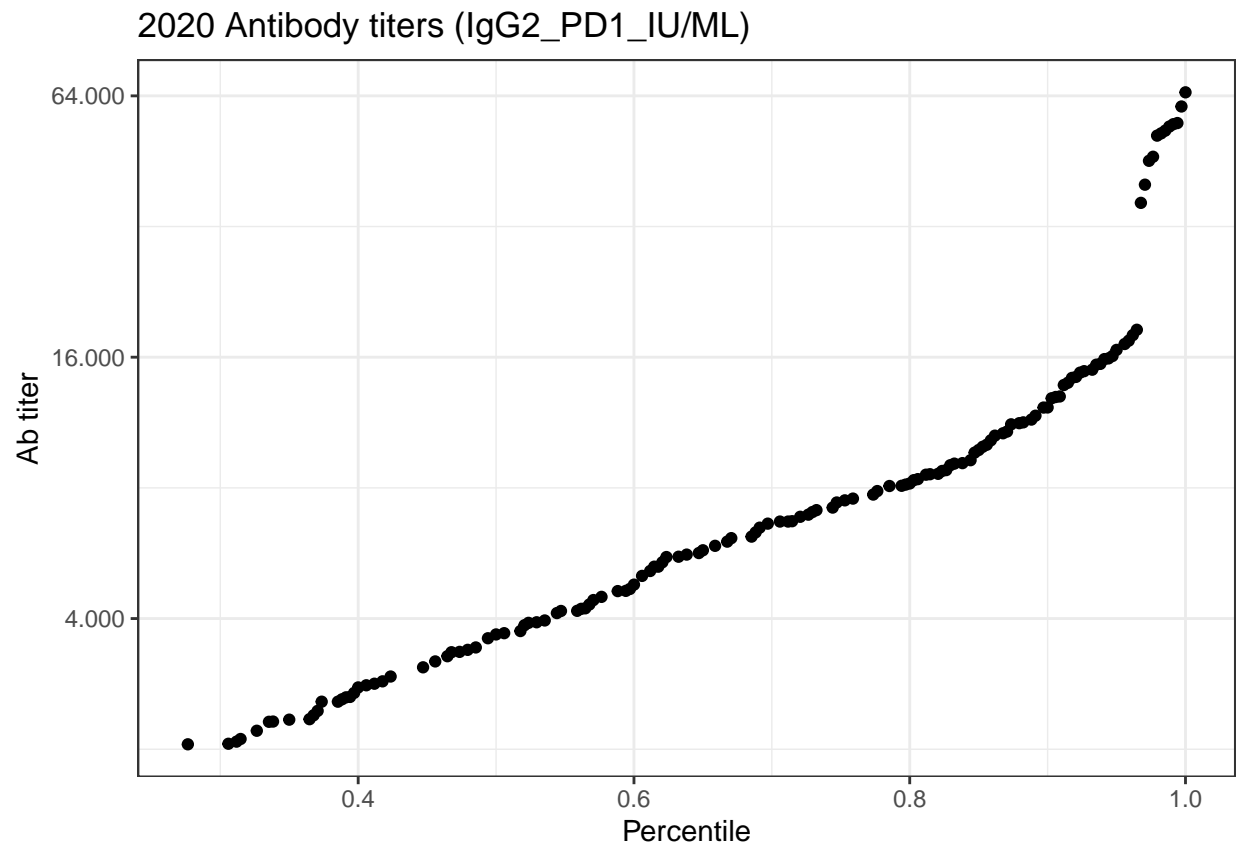




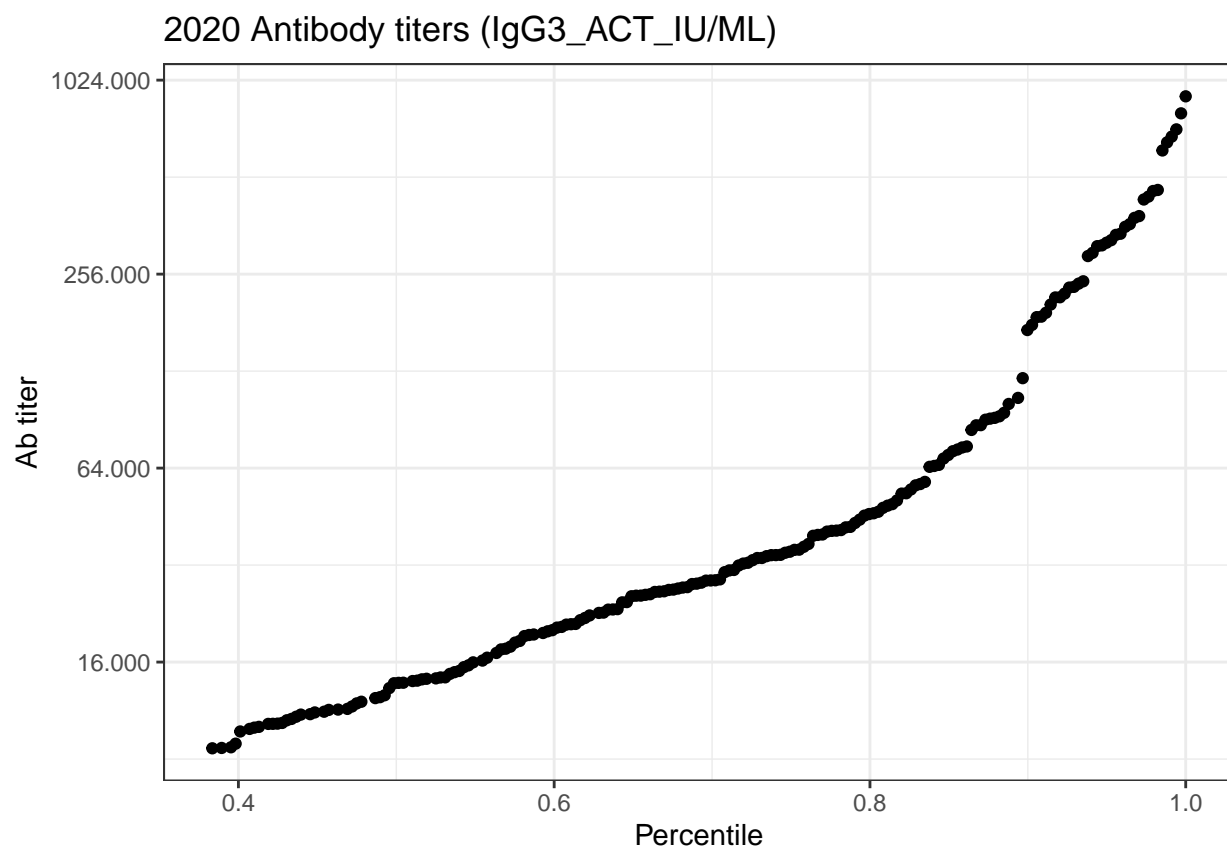


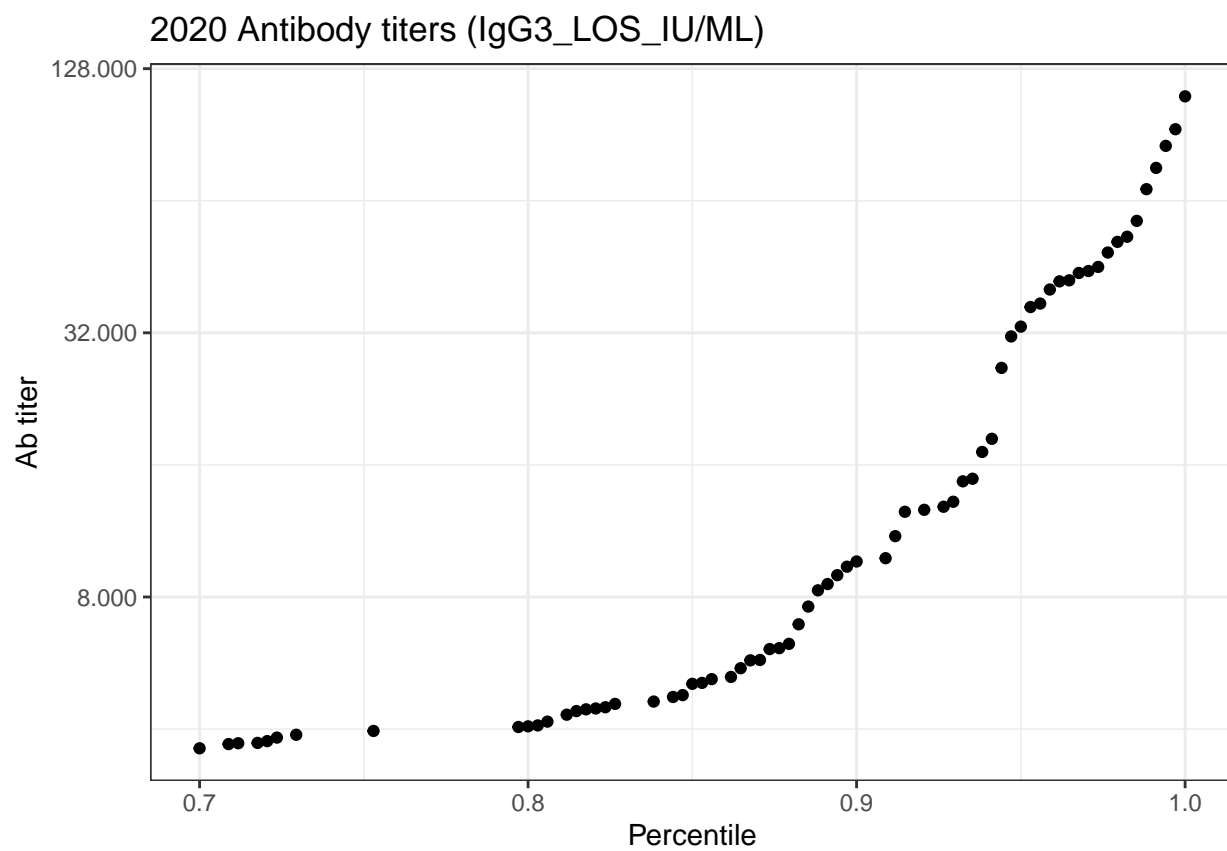




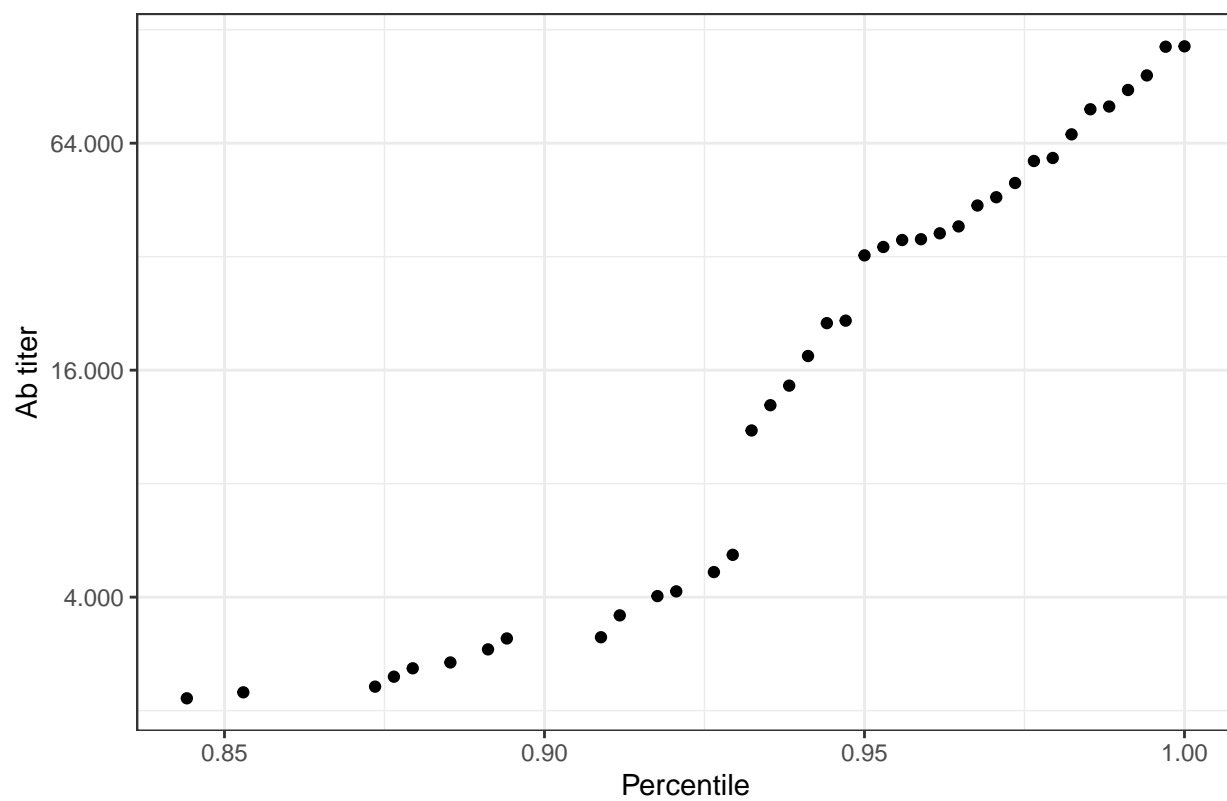


```
## Warning: Removed 1 rows containing non-finite values (stat_ecdf).
```

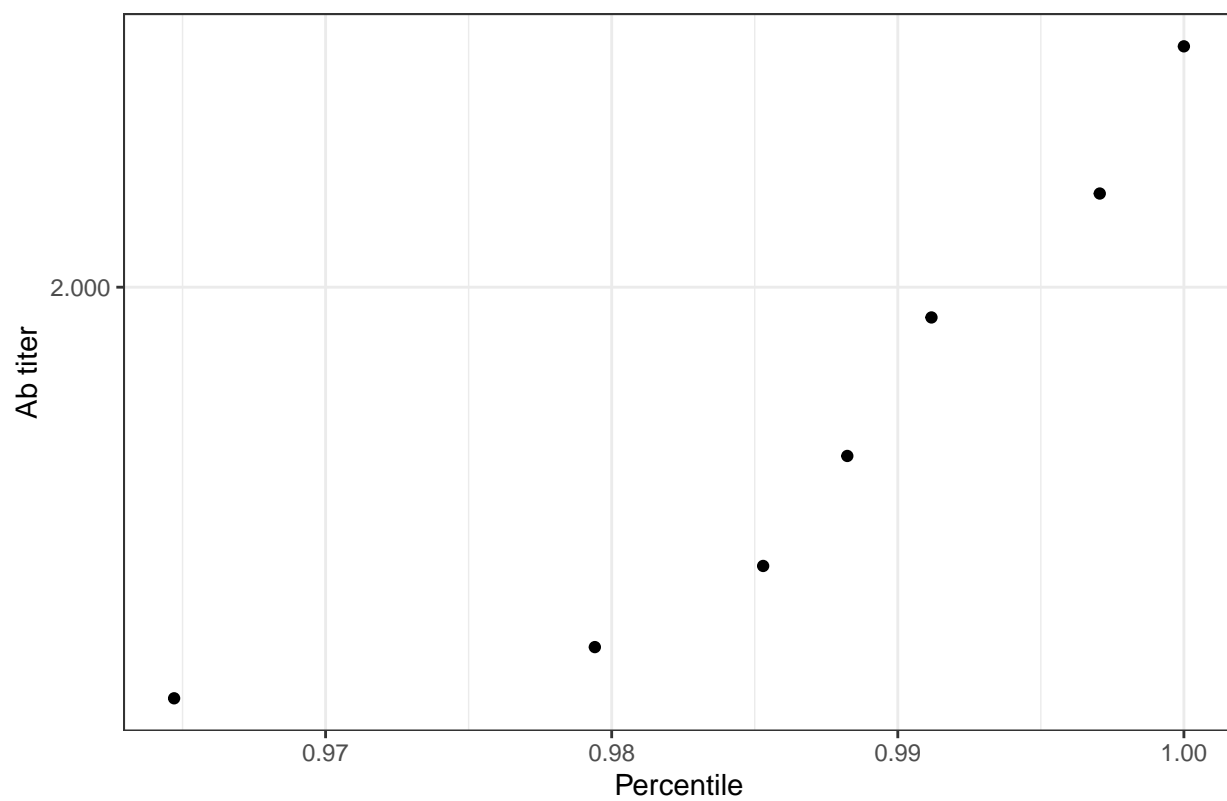


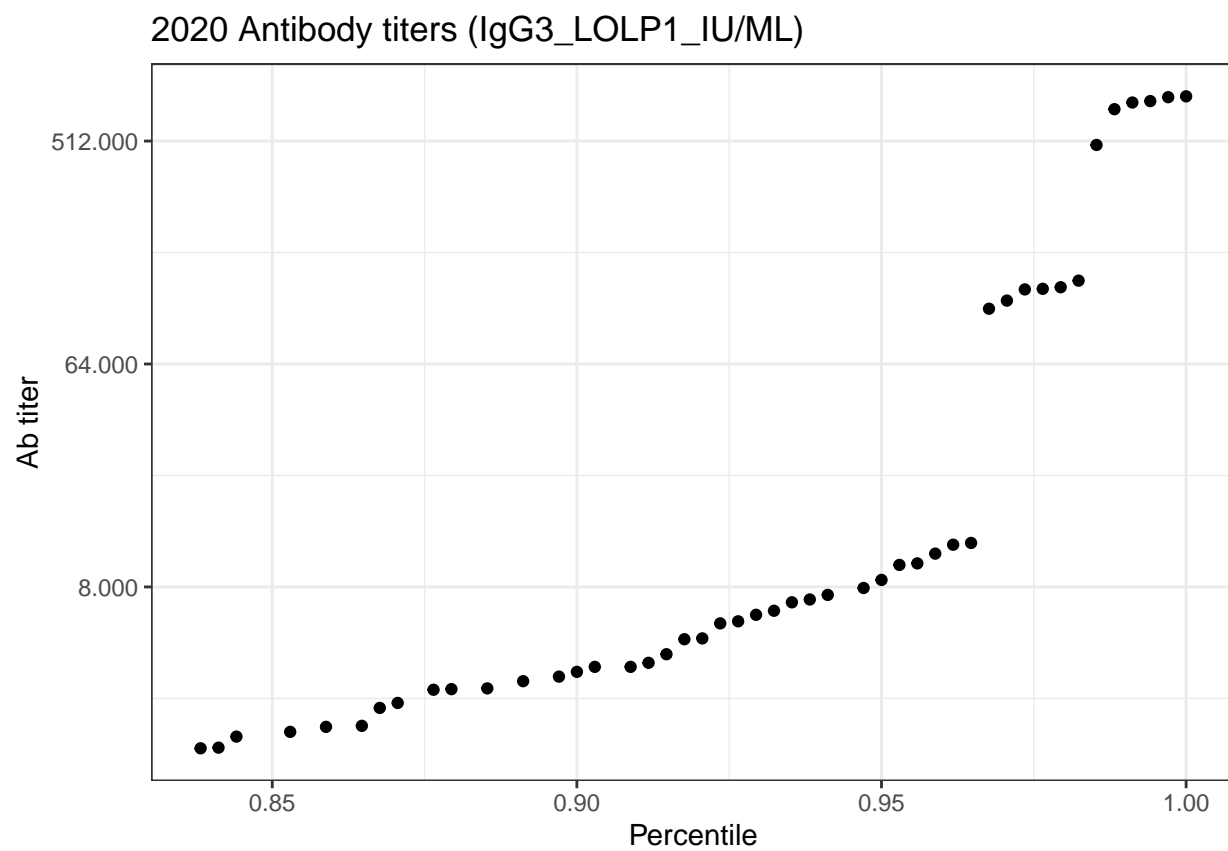


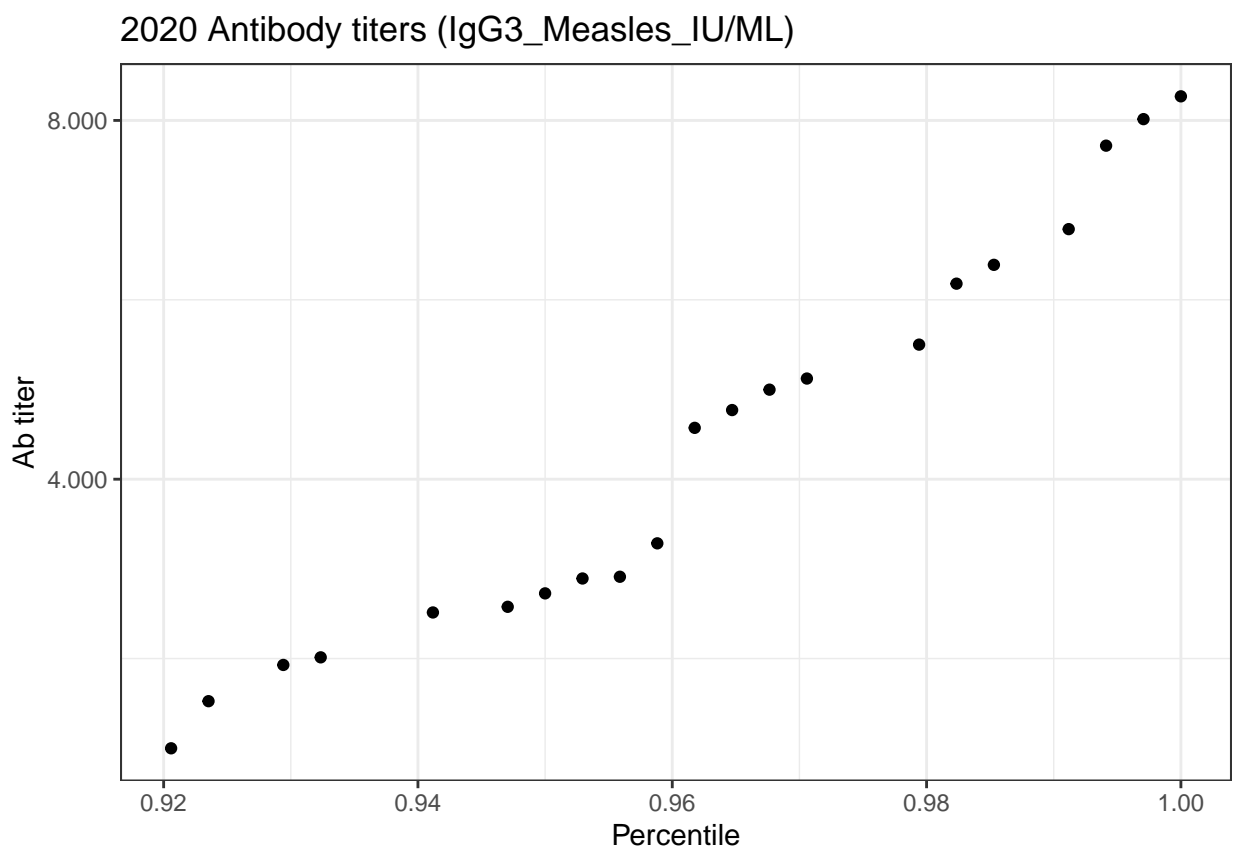
2020 Antibody titers (IgG3_FELD1_IU/ML)



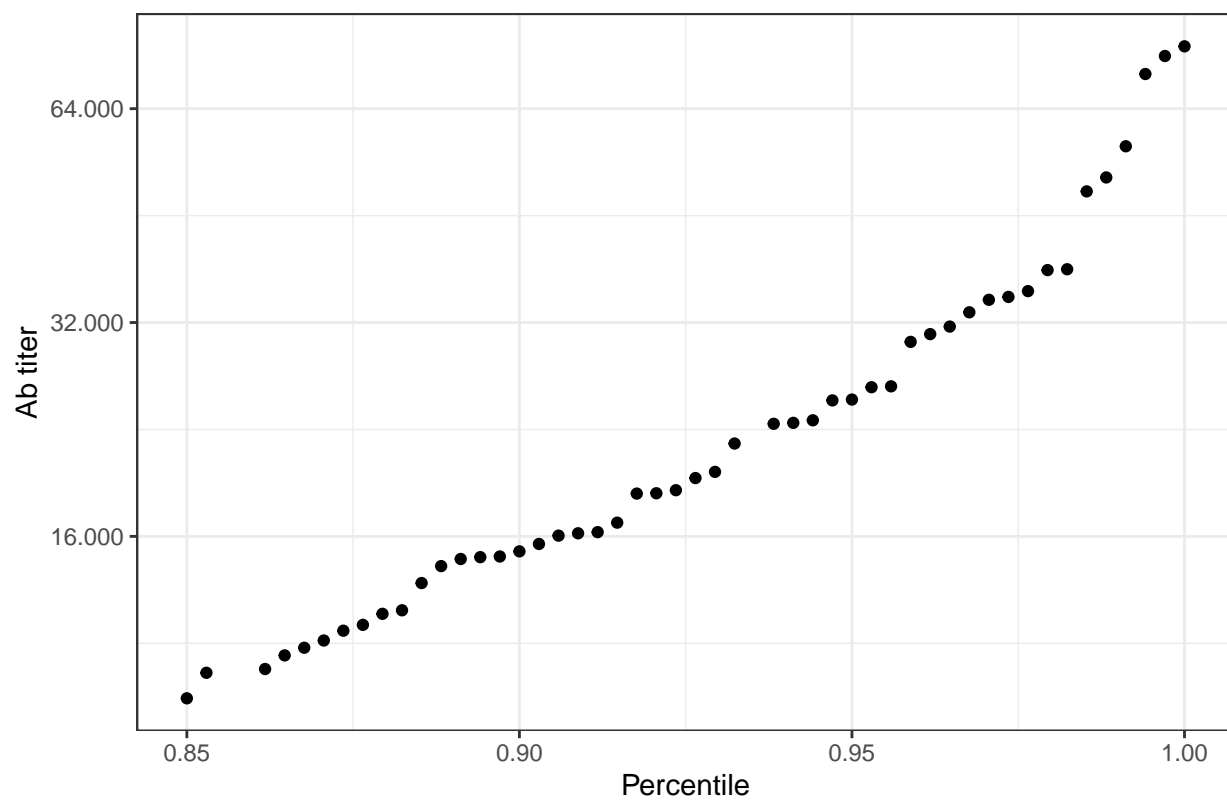
2020 Antibody titers (IgG3_BETV1_IU/ML)



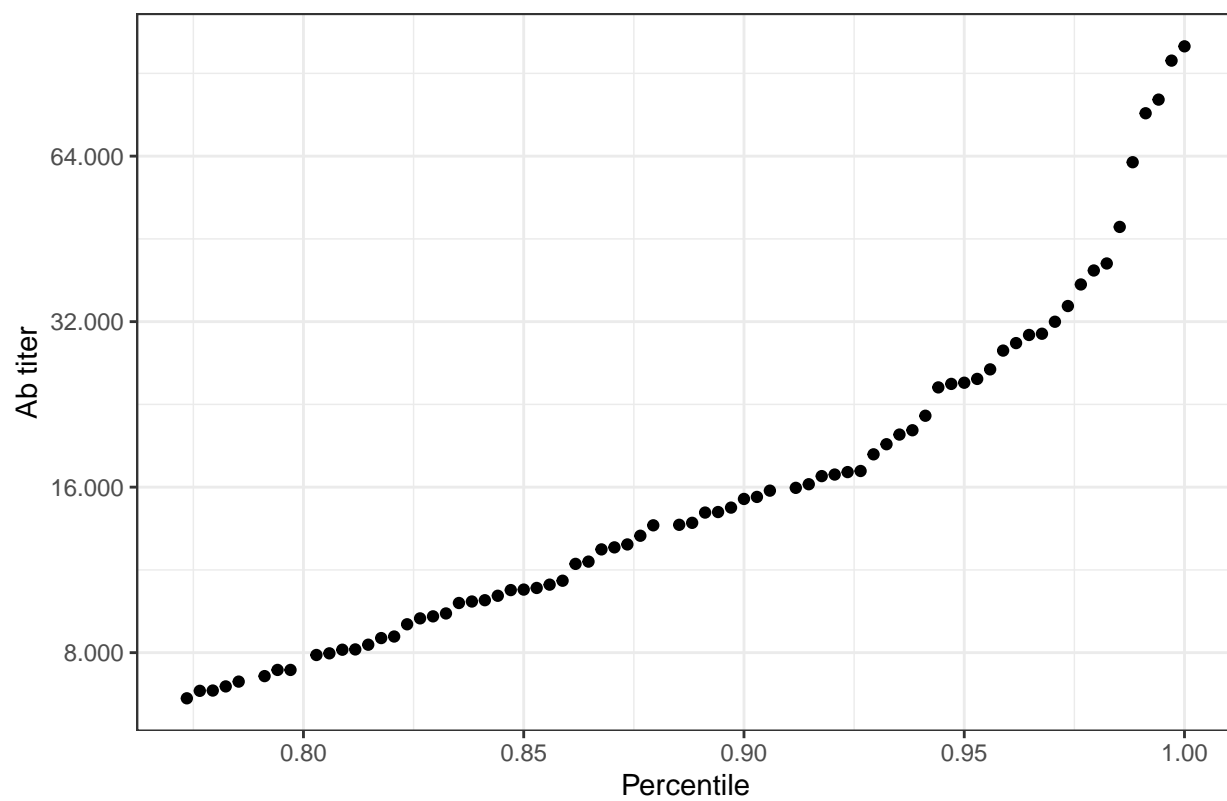




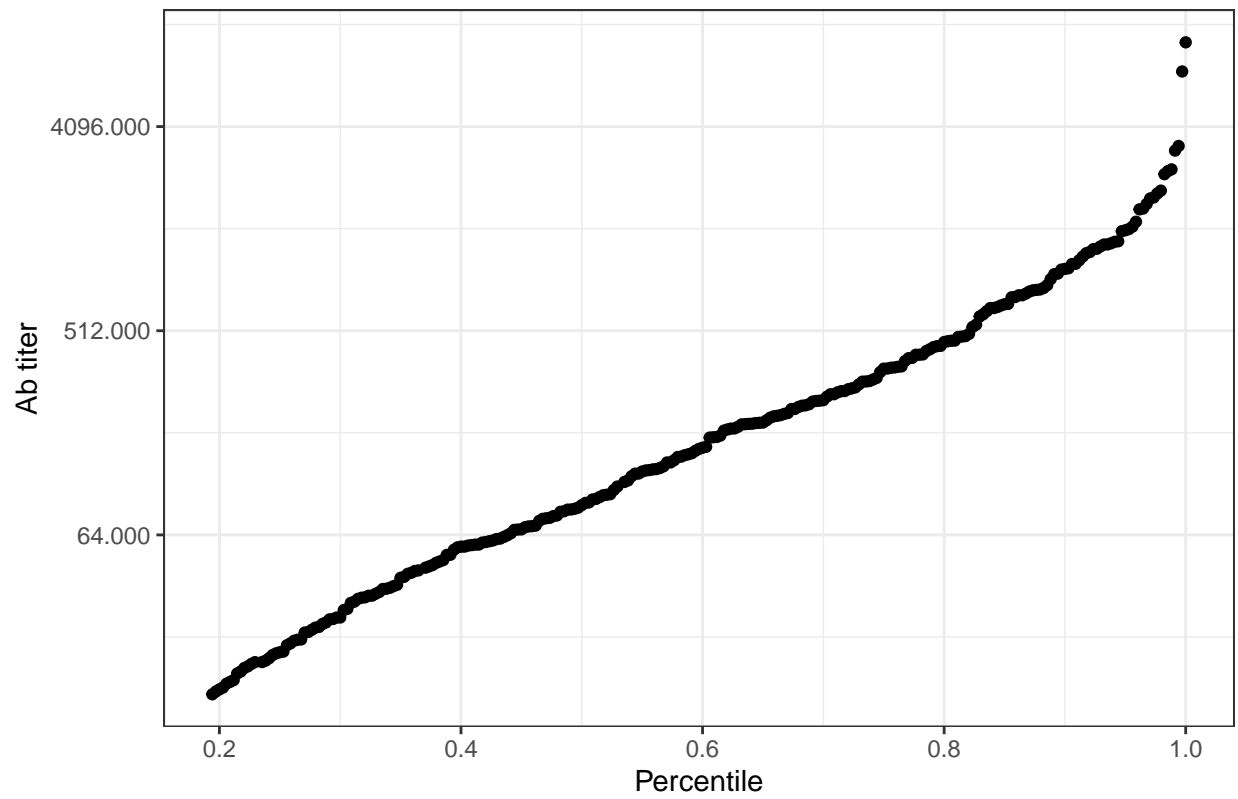
2020 Antibody titers (IgG3_PTM_IU/ML)

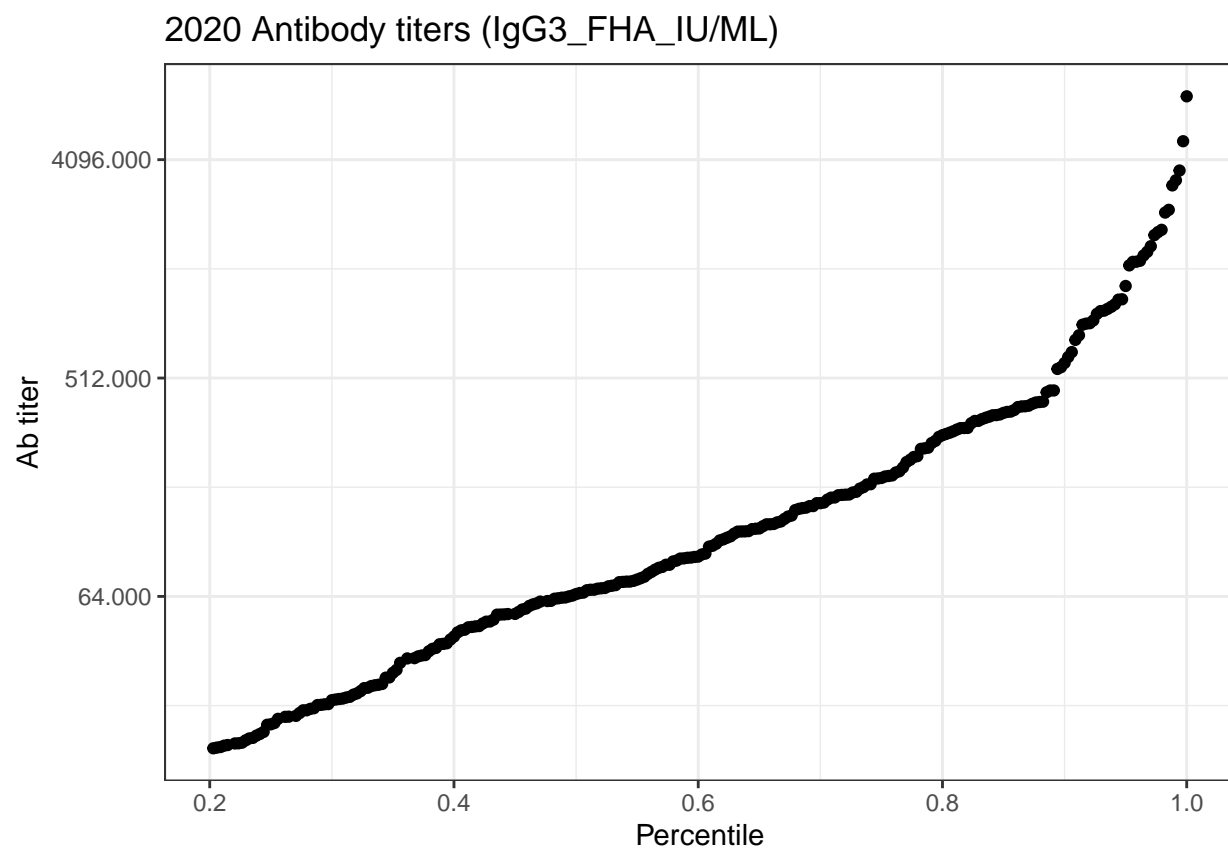


2020 Antibody titers (IgG3_PT_IU/ML)

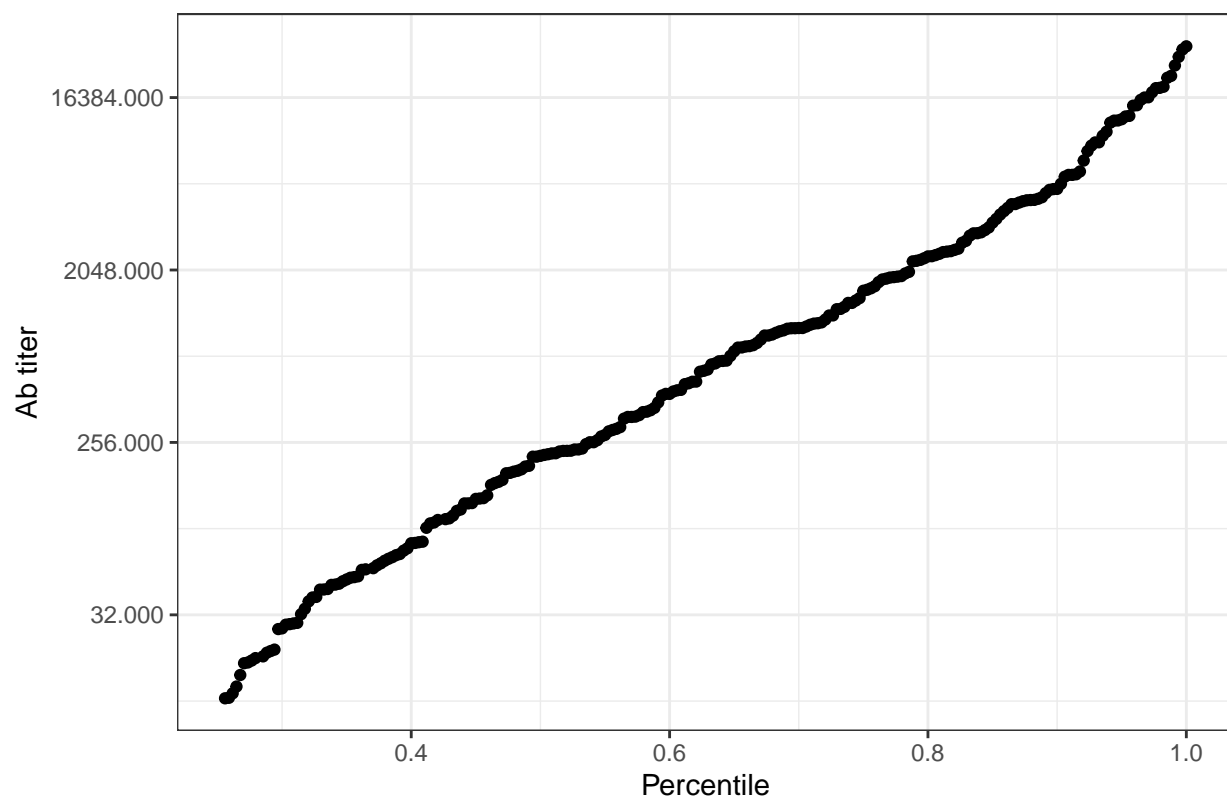


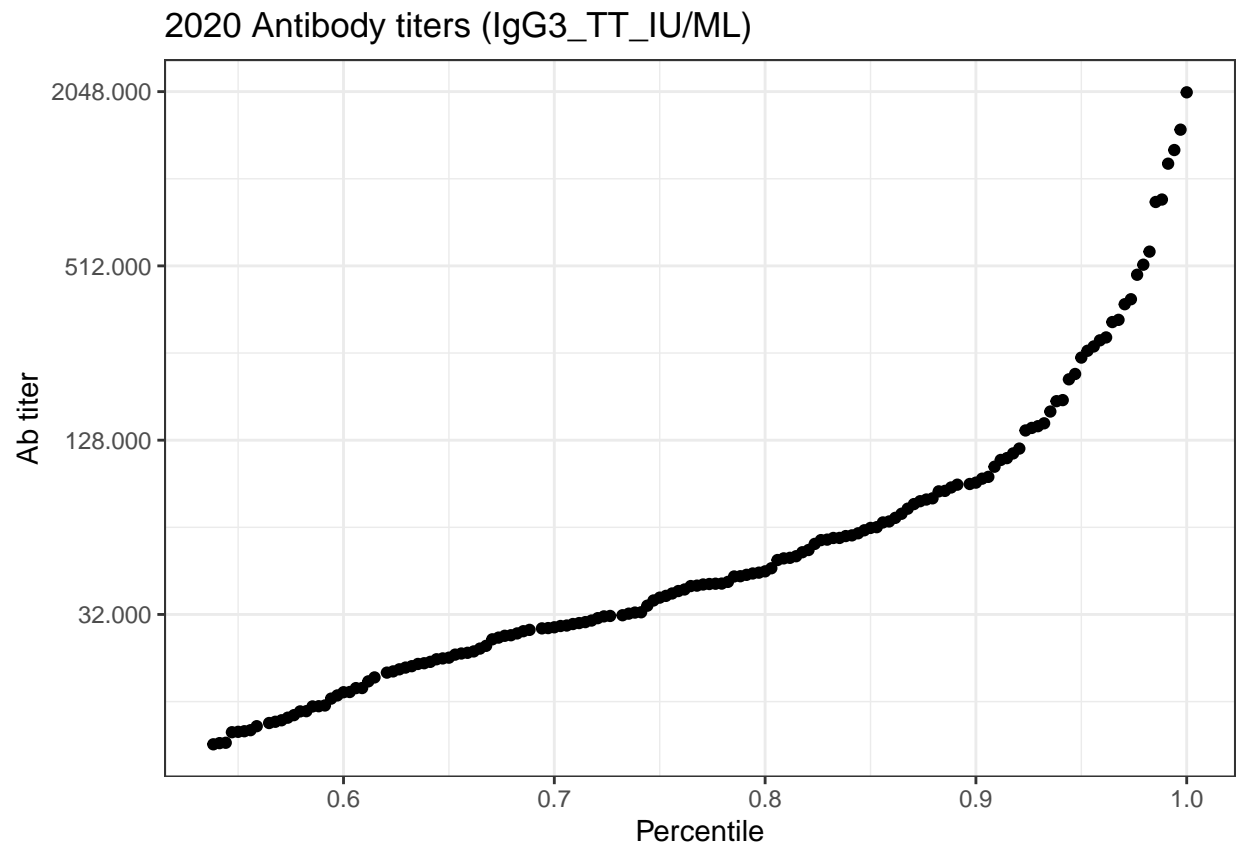
2020 Antibody titers (IgG3_PRN_IU/ML)

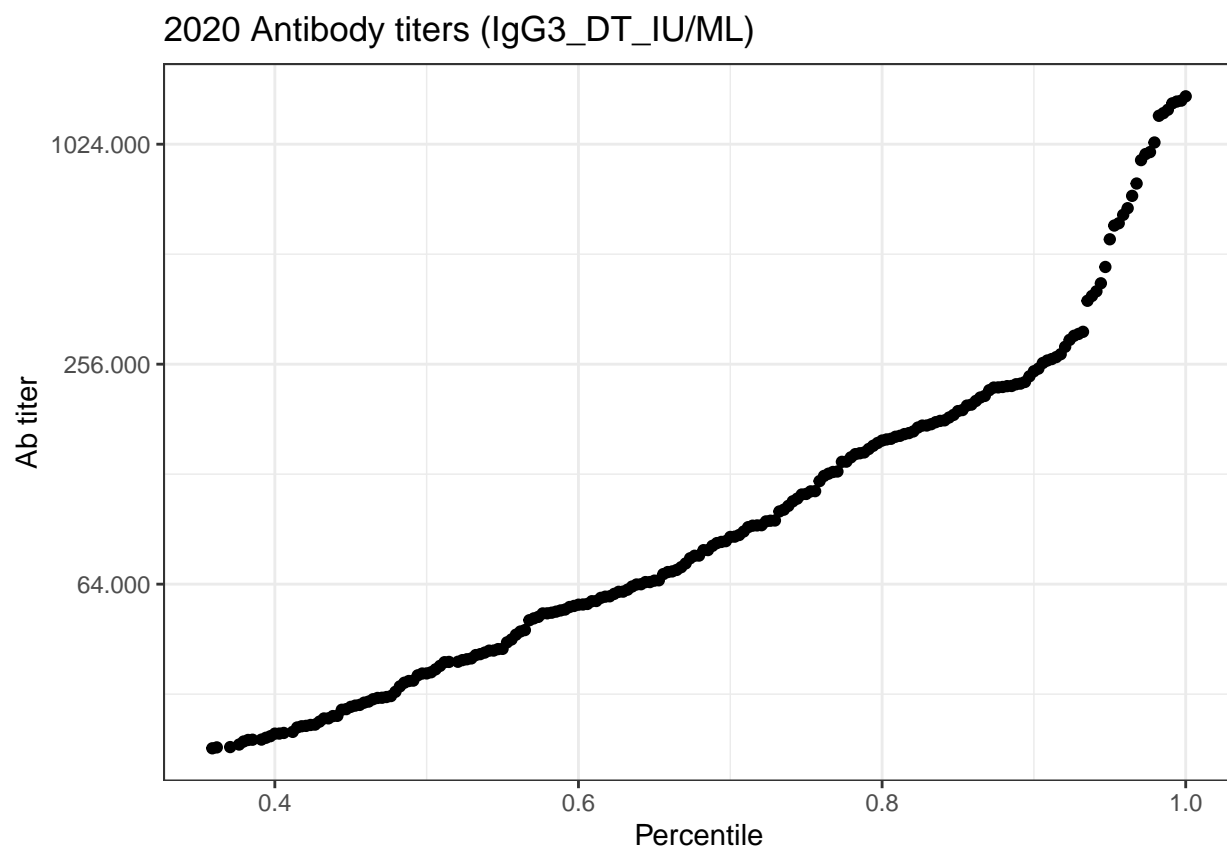




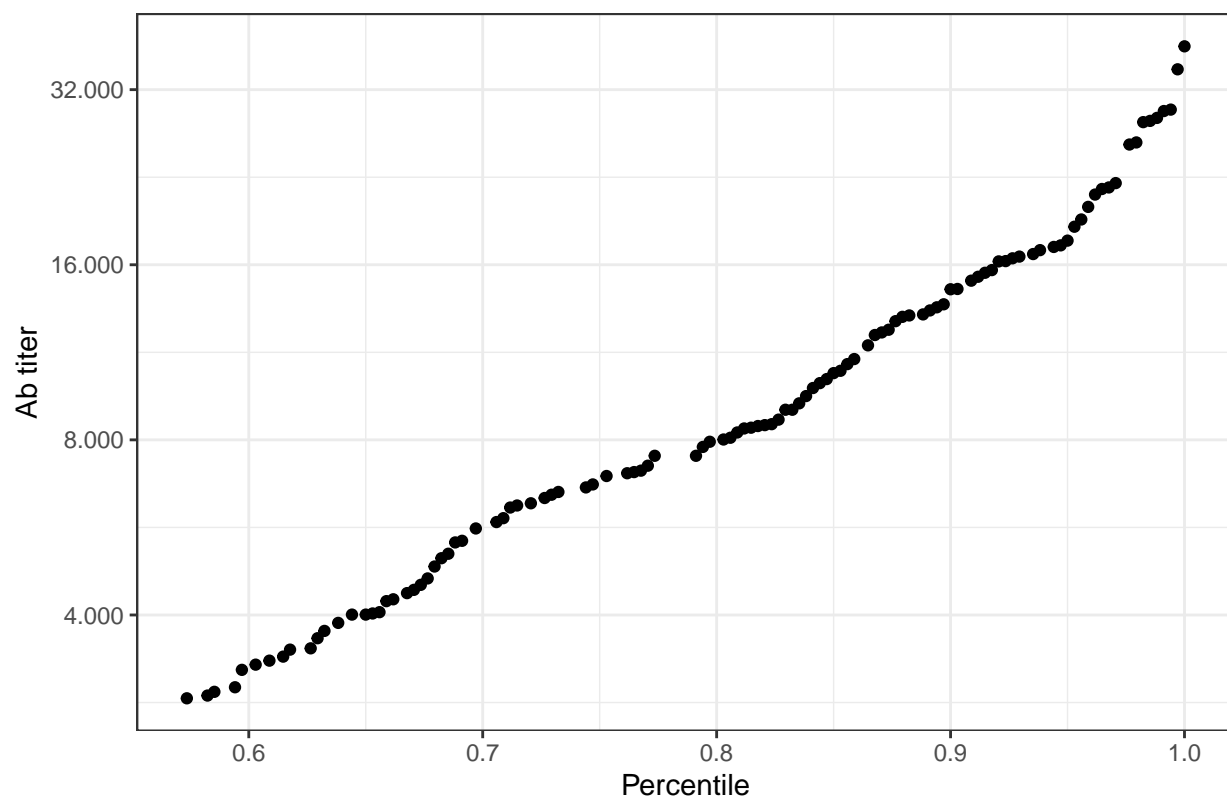
2020 Antibody titers (IgG3_FIM2/3_IU/ML)



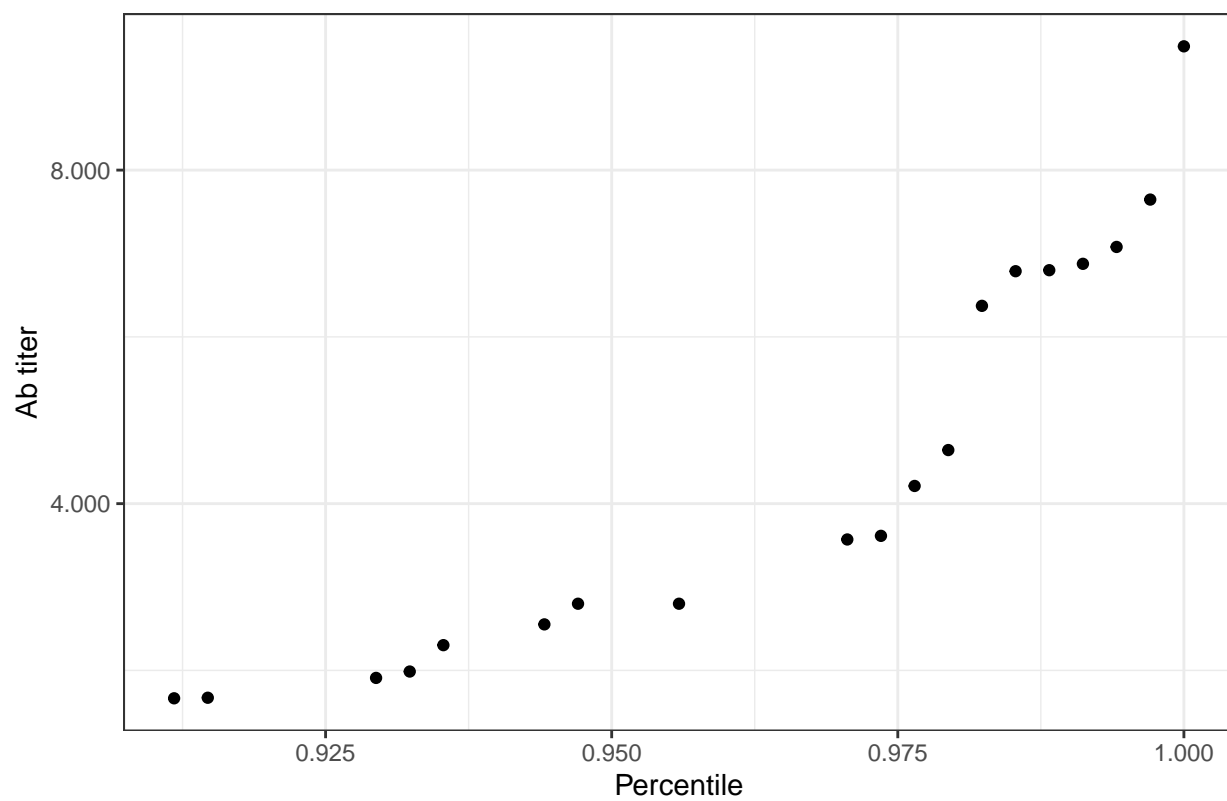


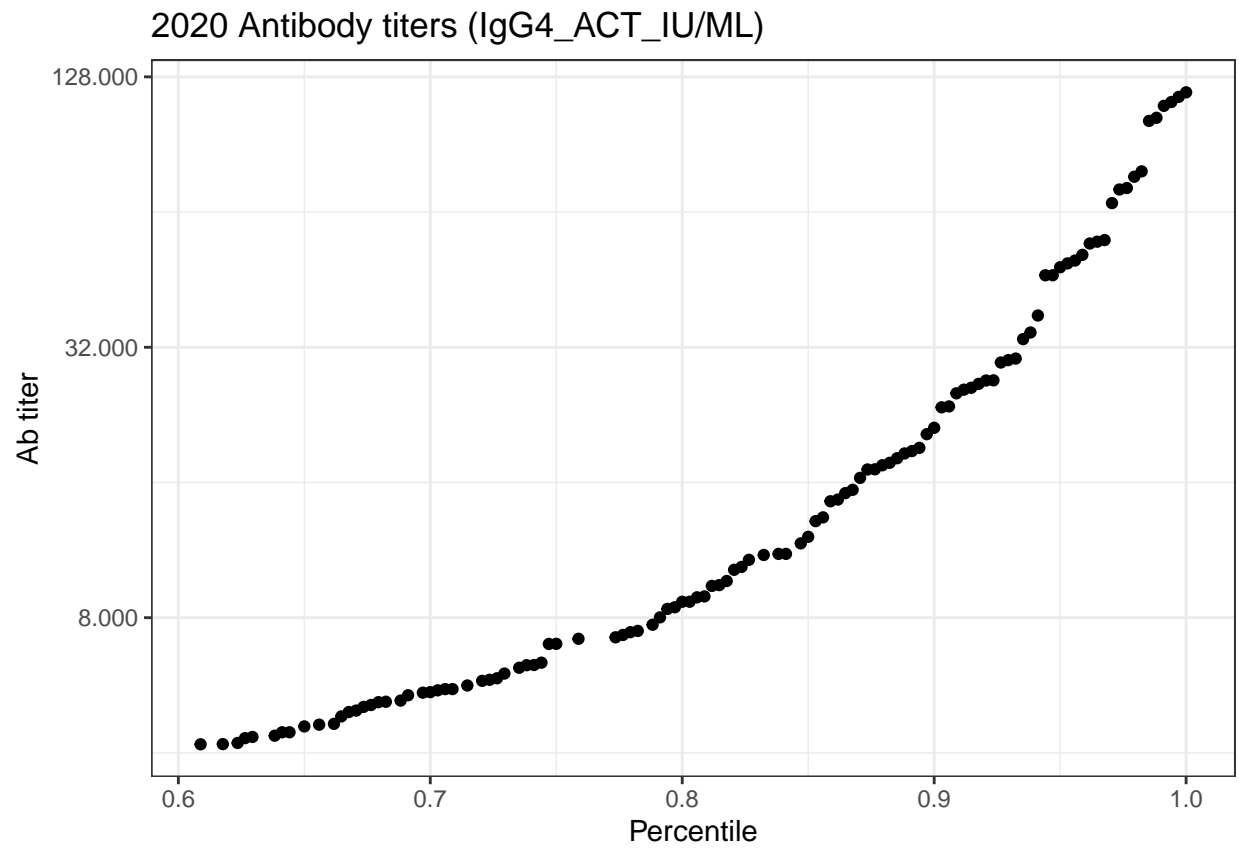


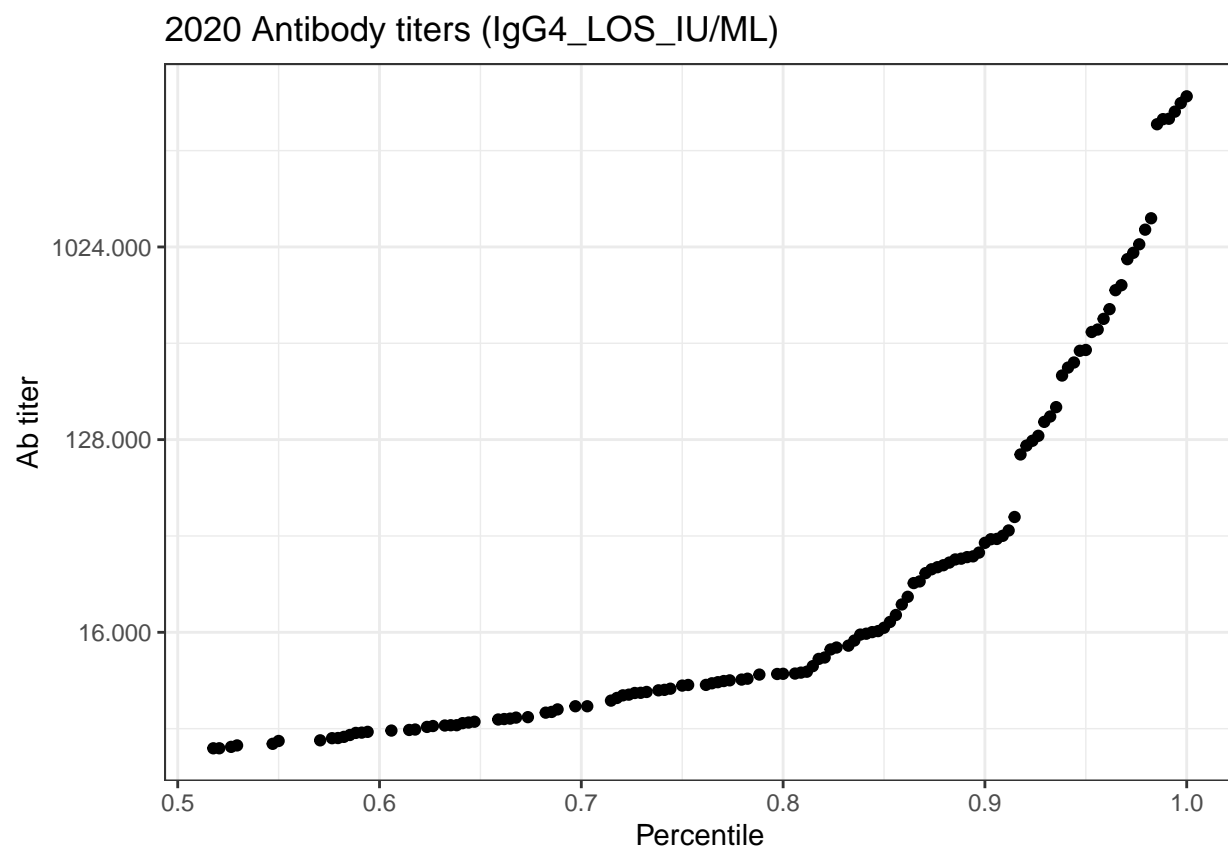
2020 Antibody titers (IgG3_OVA_IU/ML)

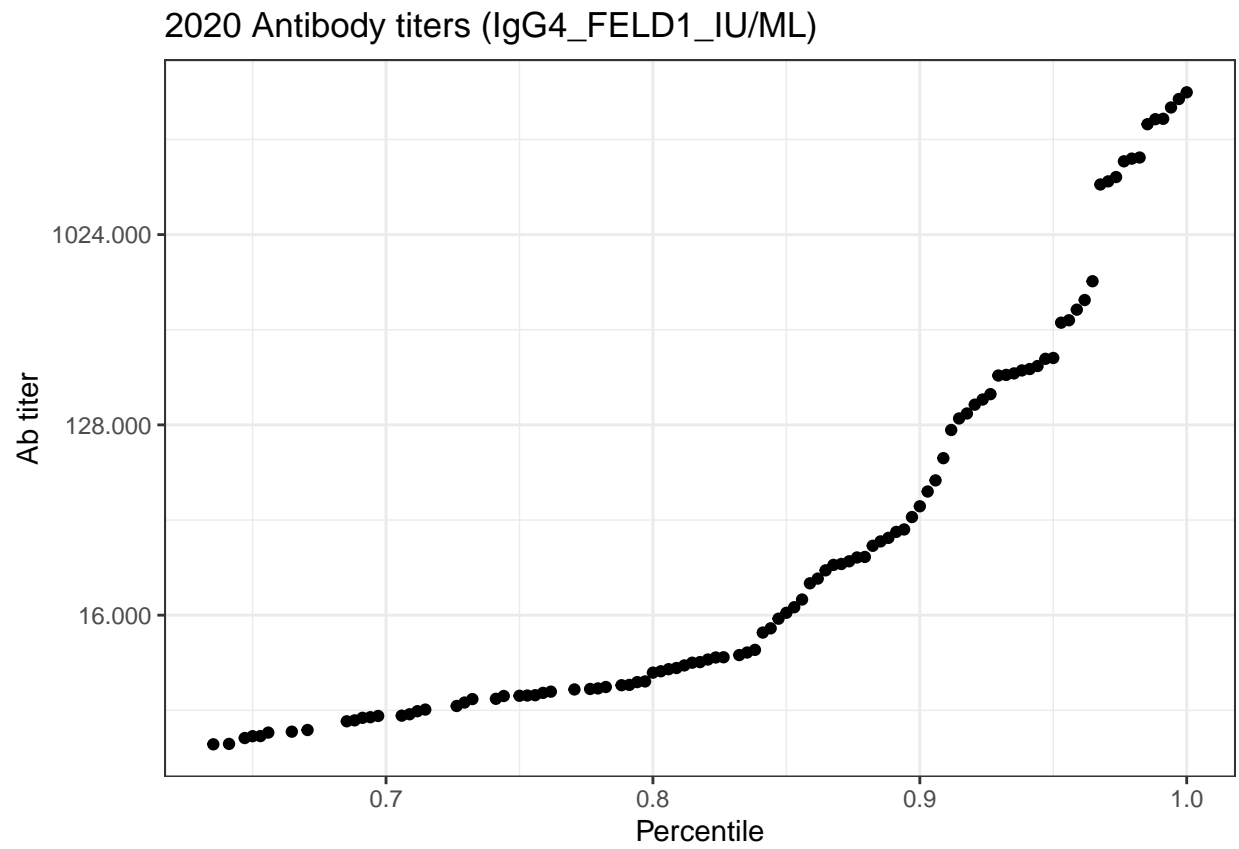


2020 Antibody titers (IgG3_PD1_IU/ML)

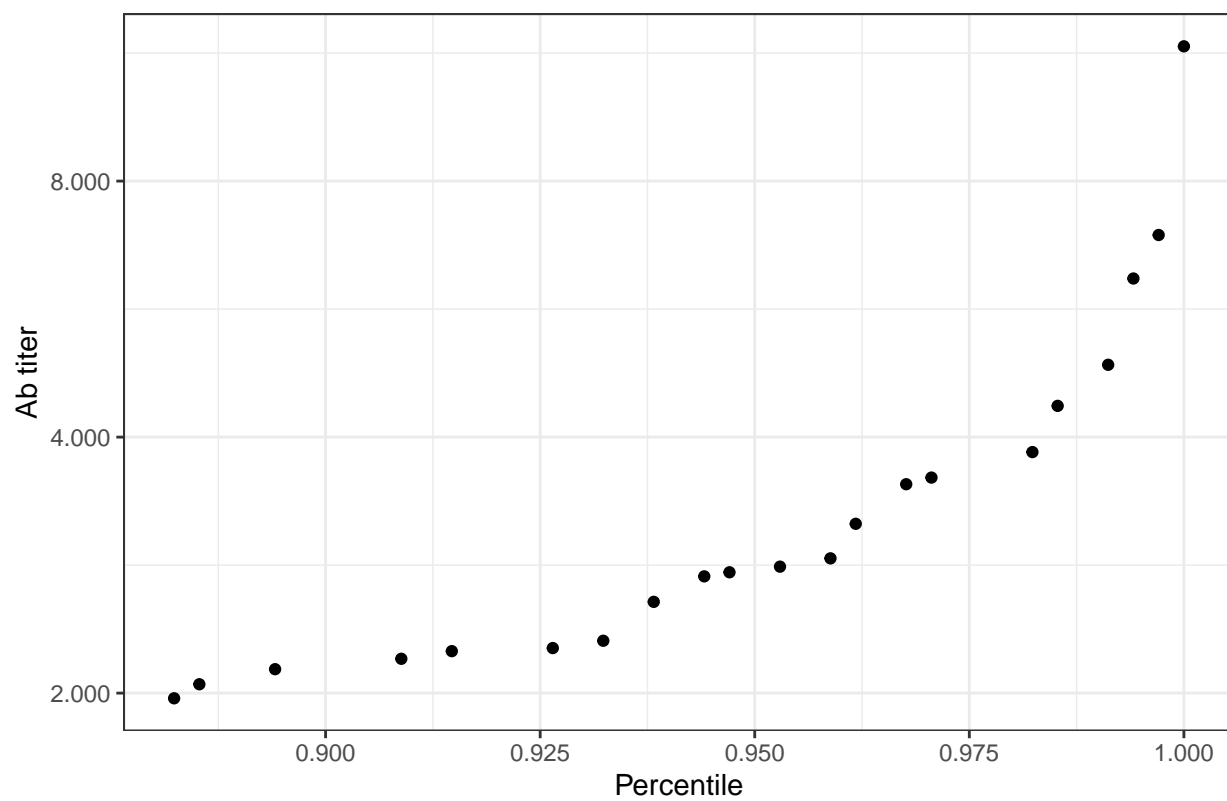




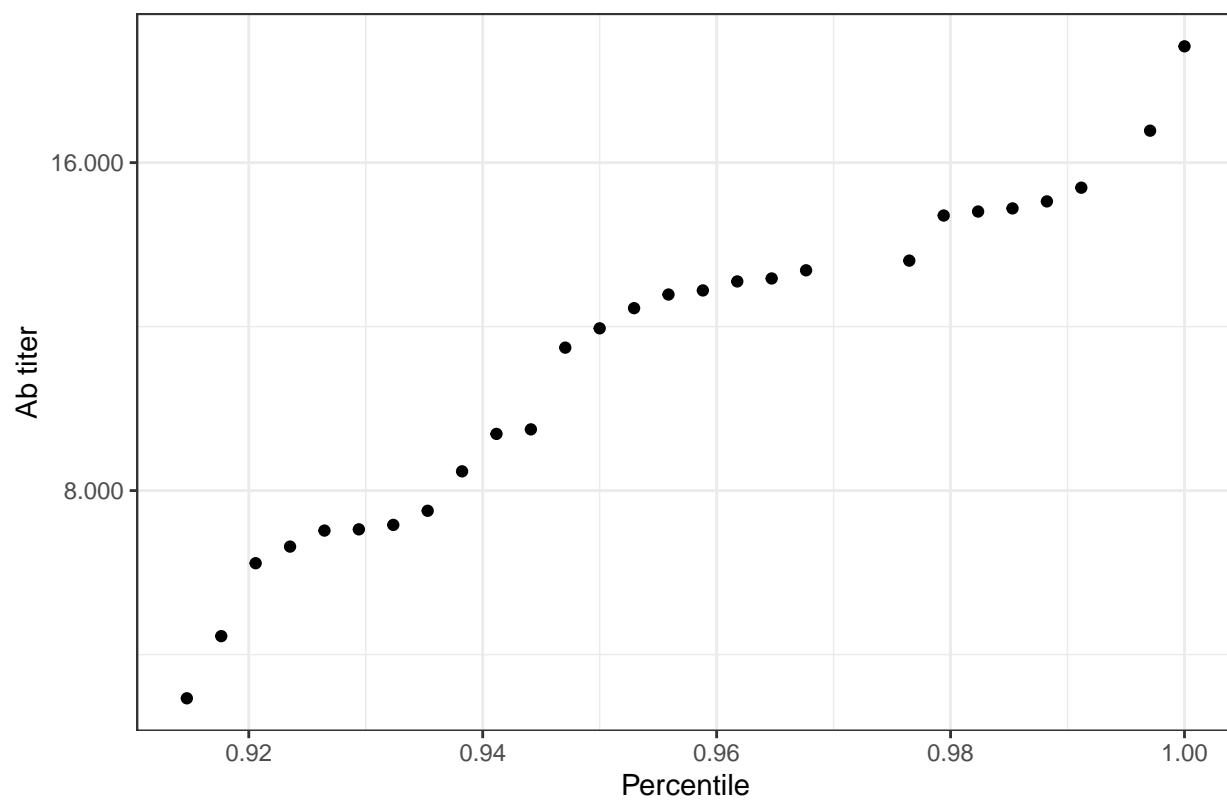


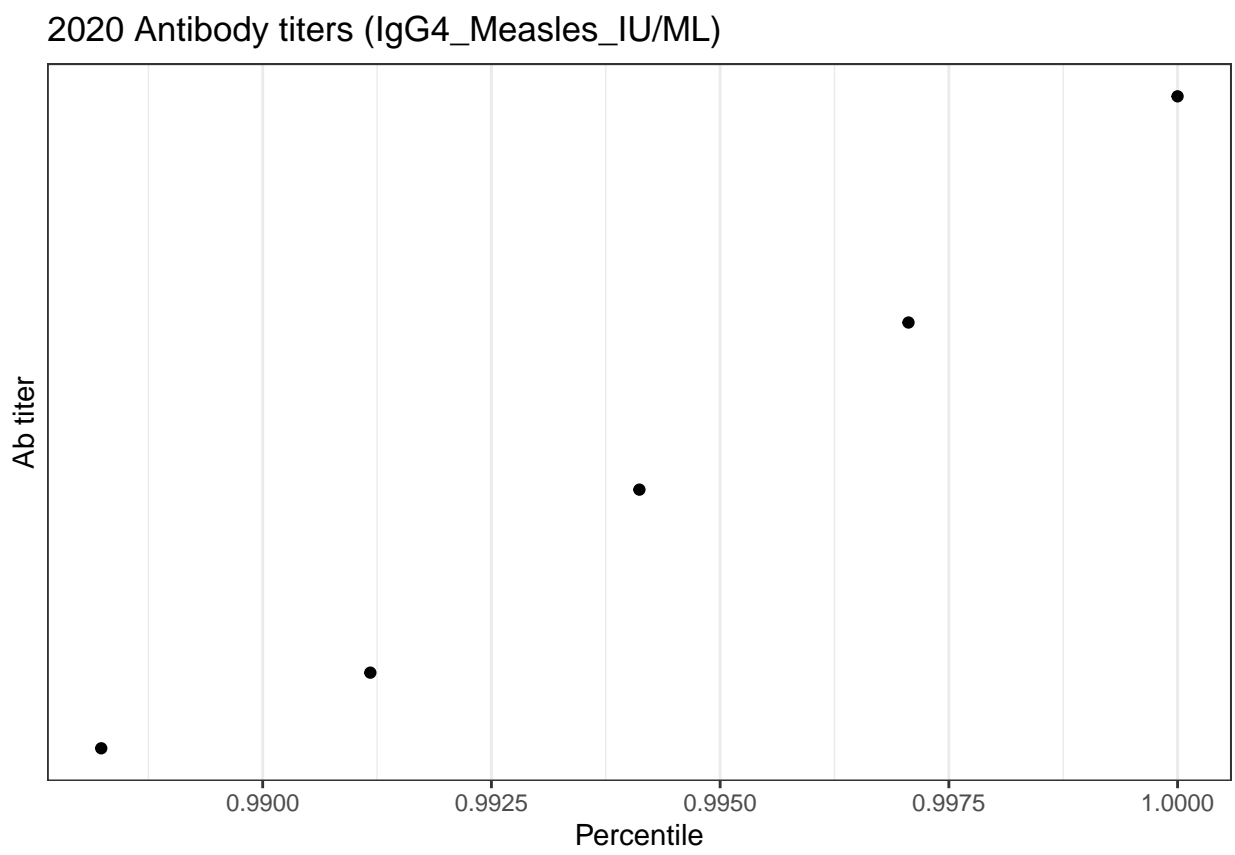


2020 Antibody titers (IgG4_BETV1_IU/ML)

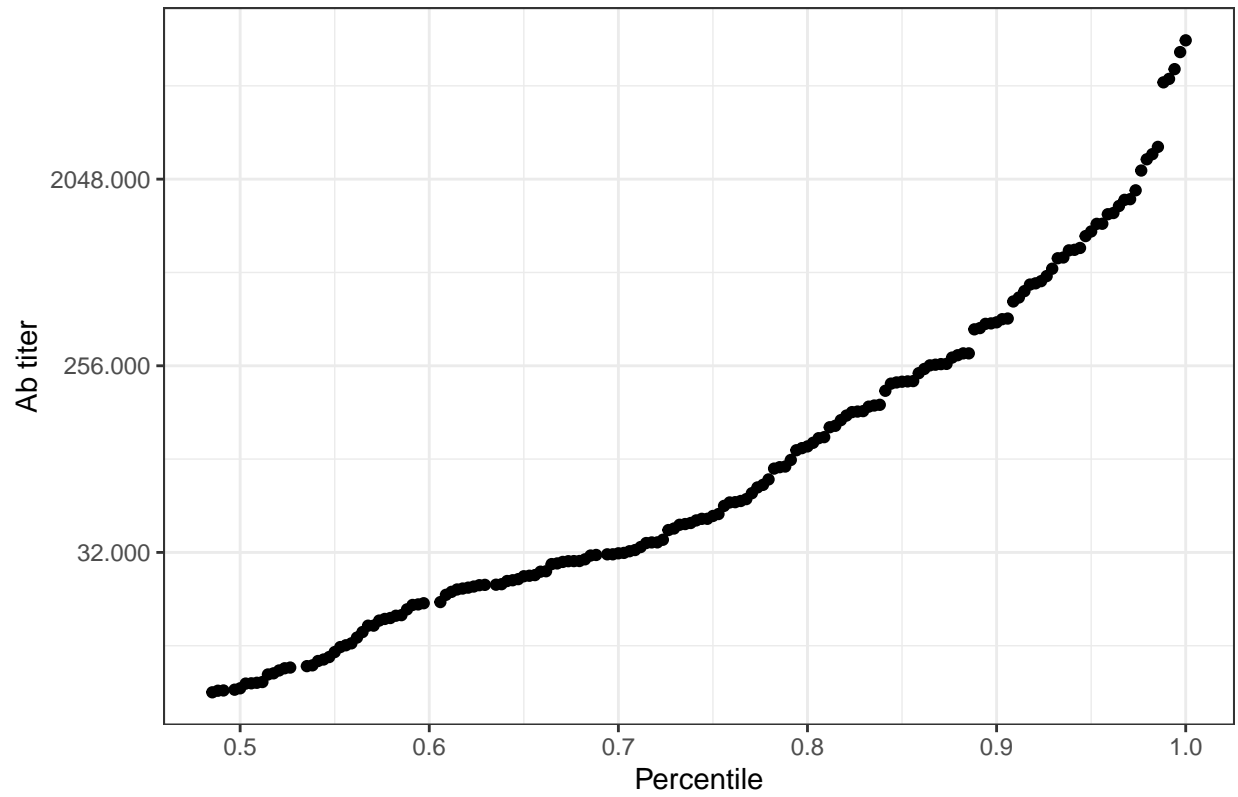


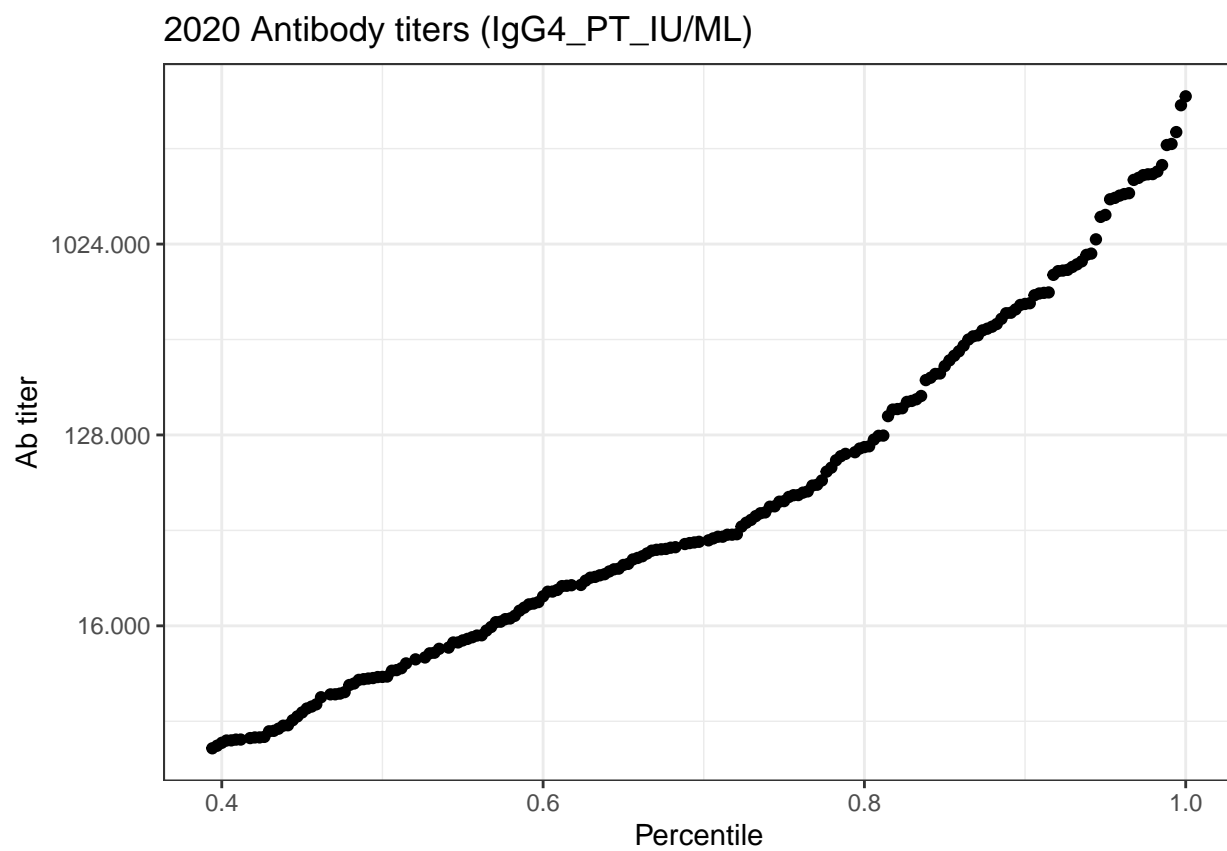
2020 Antibody titers (IgG4_LOLP1_IU/ML)

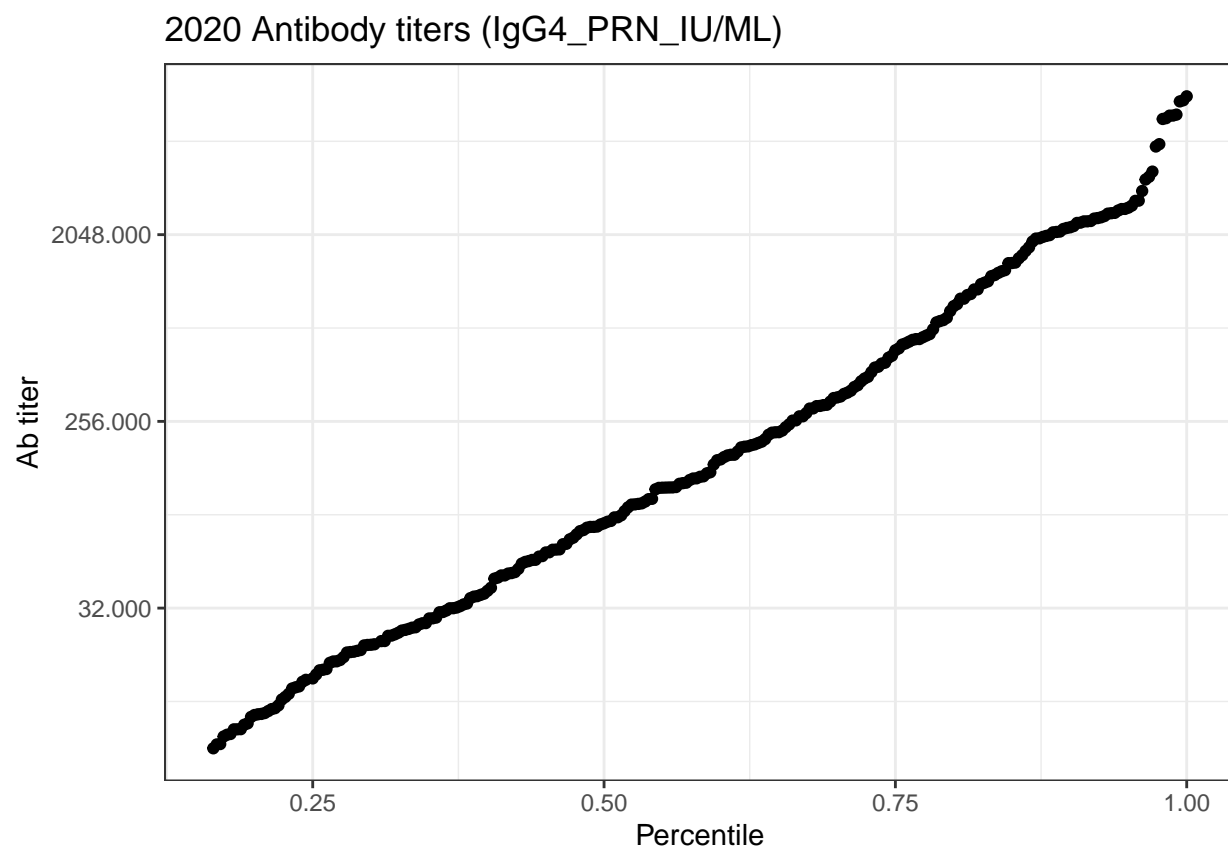


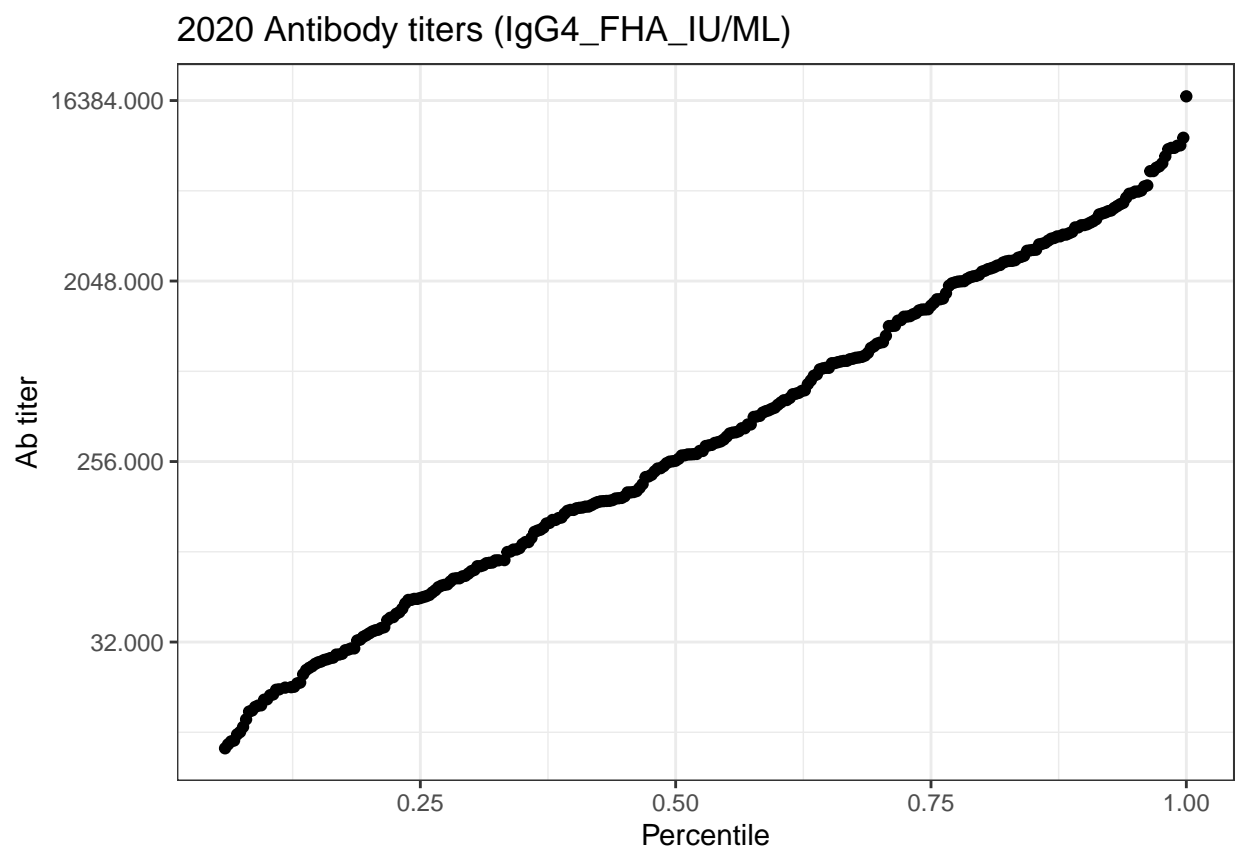


2020 Antibody titers (IgG4_PTM_IU/ML)

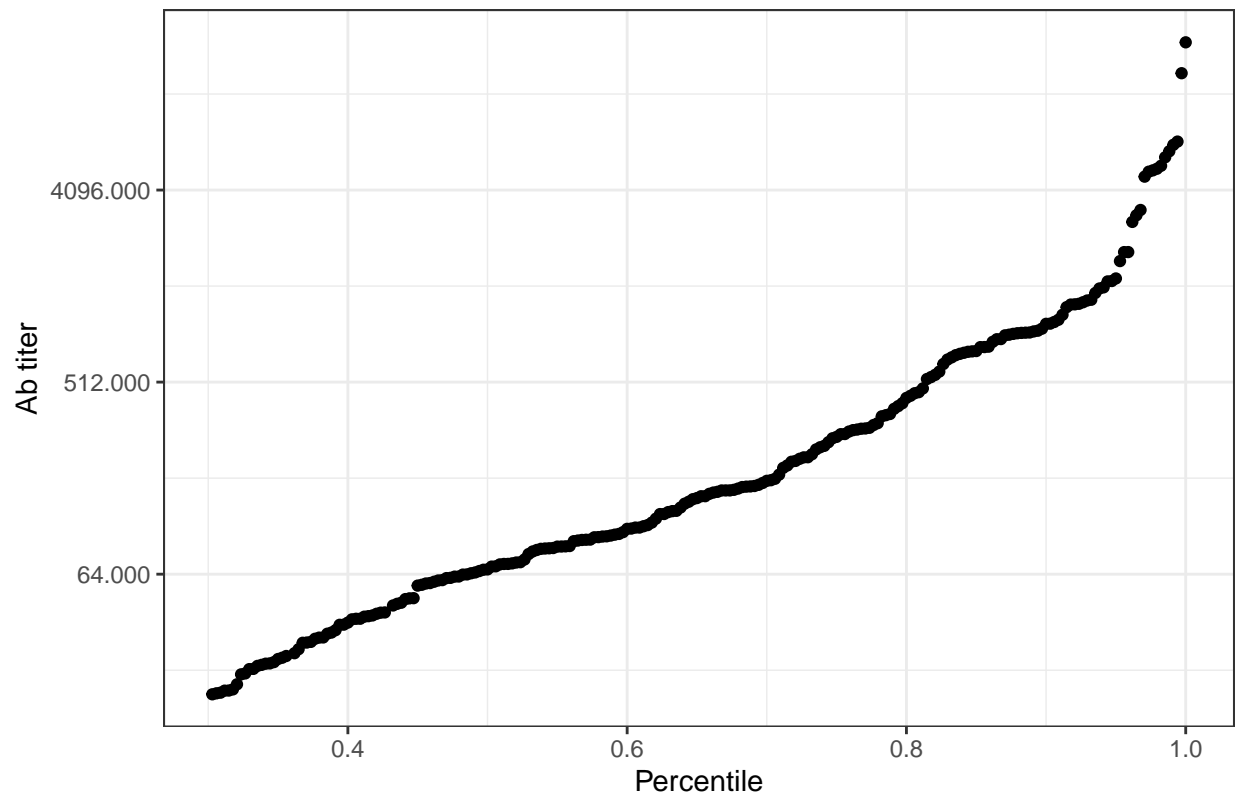




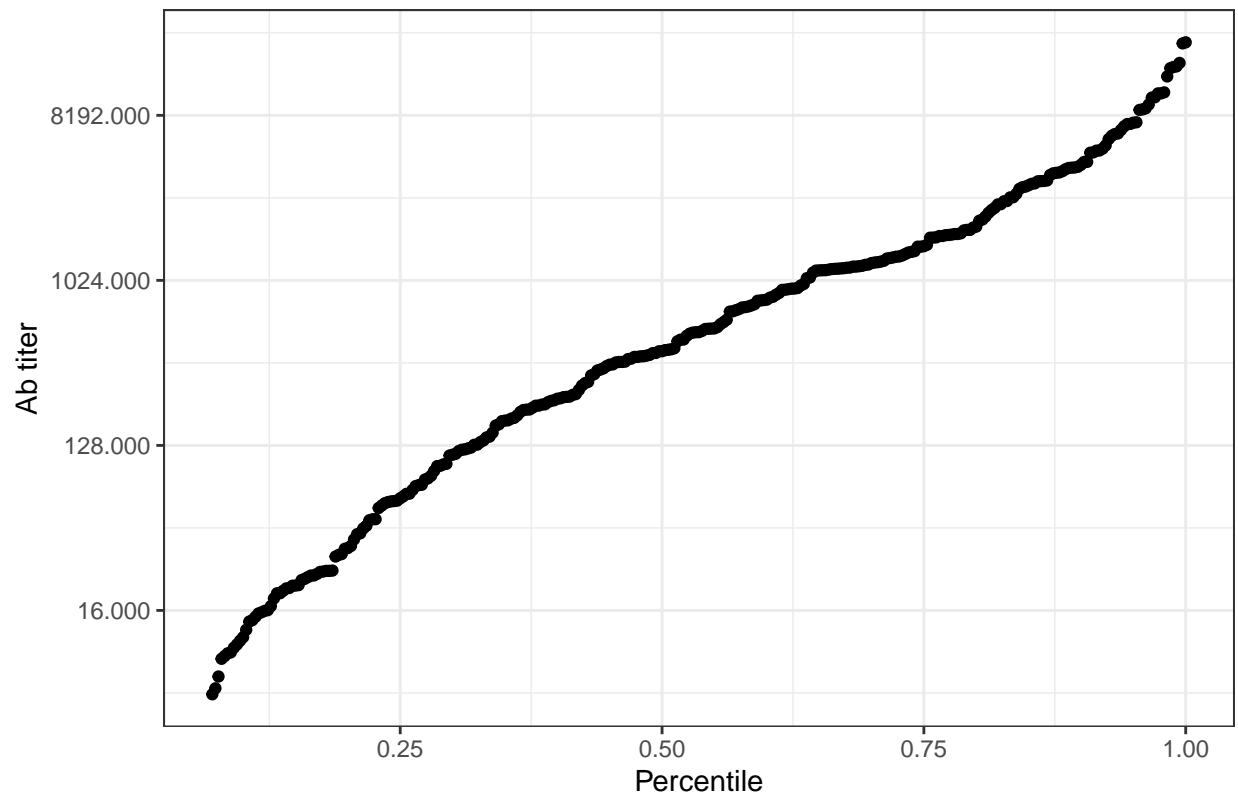


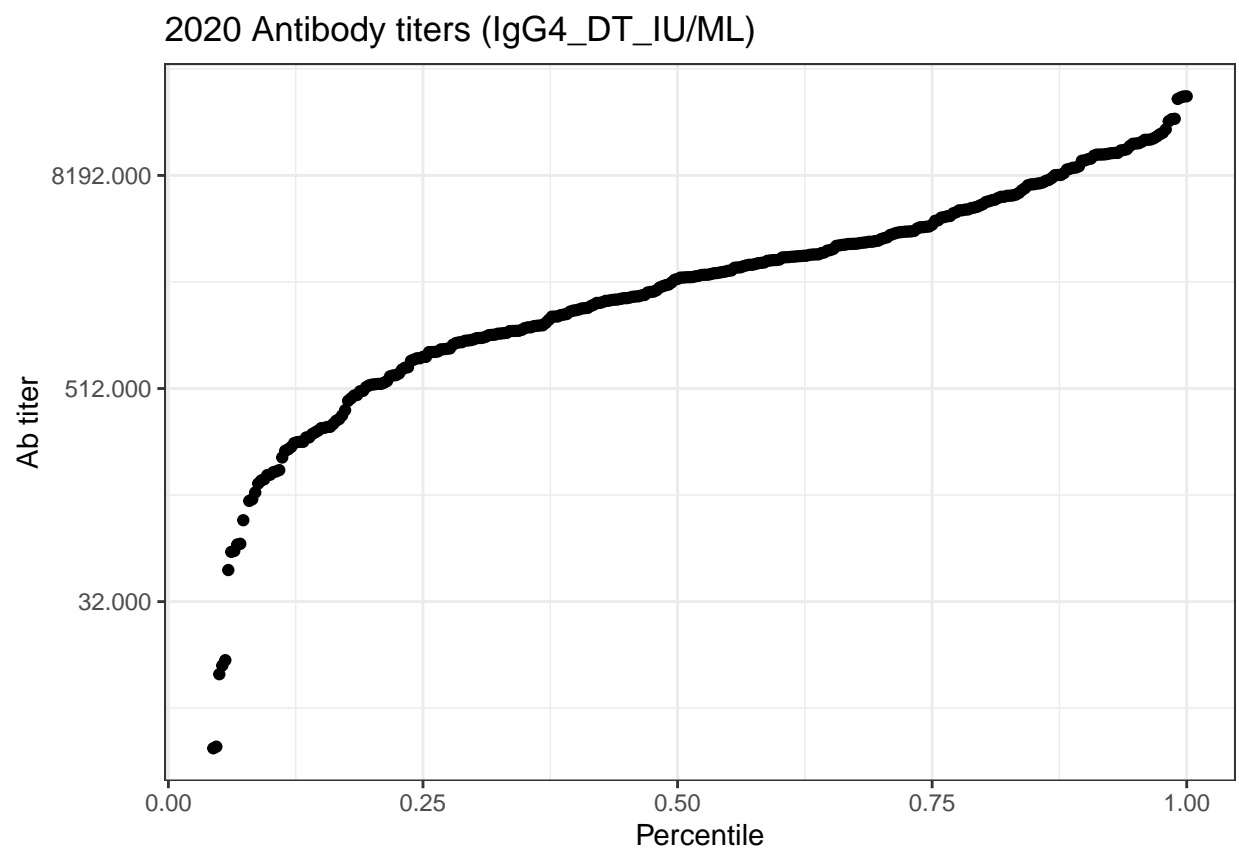


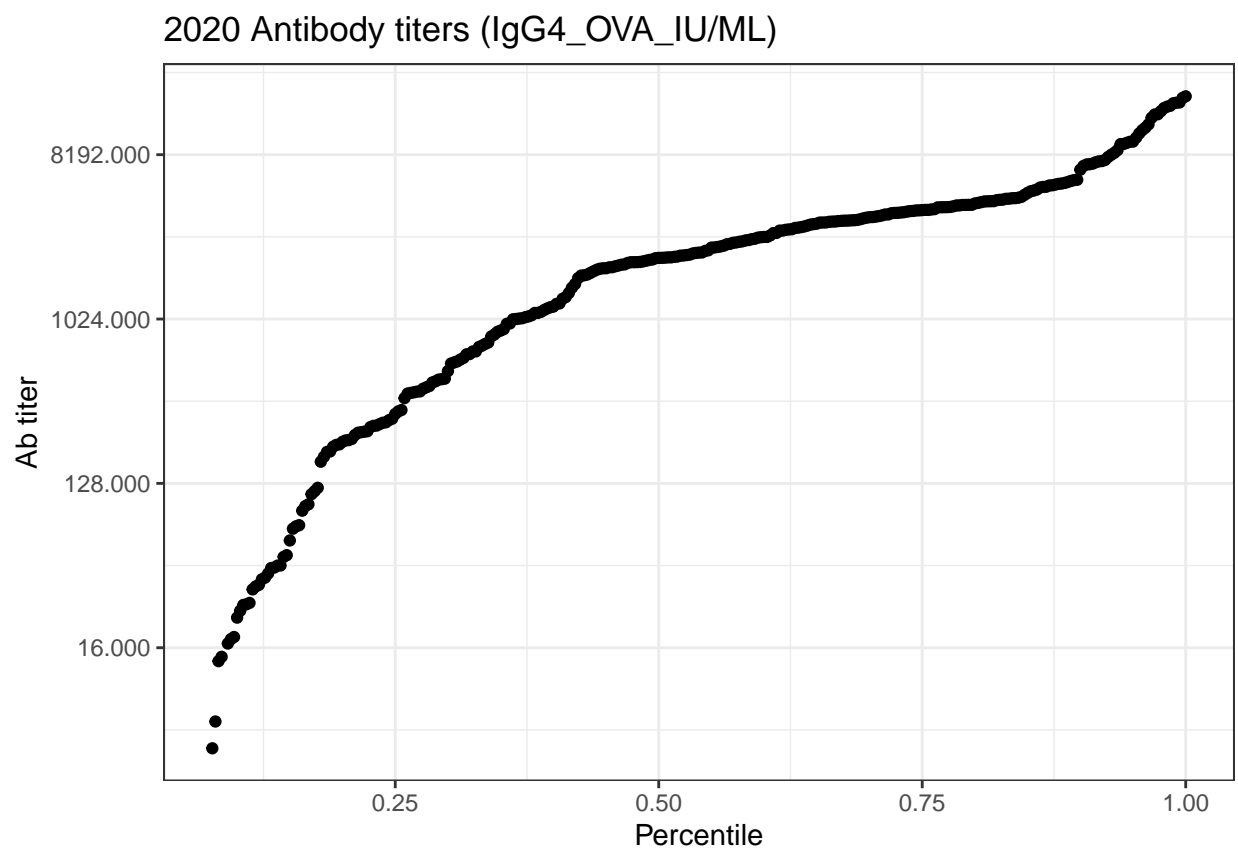
2020 Antibody titers (IgG4_FIM2/3_IU/ML)



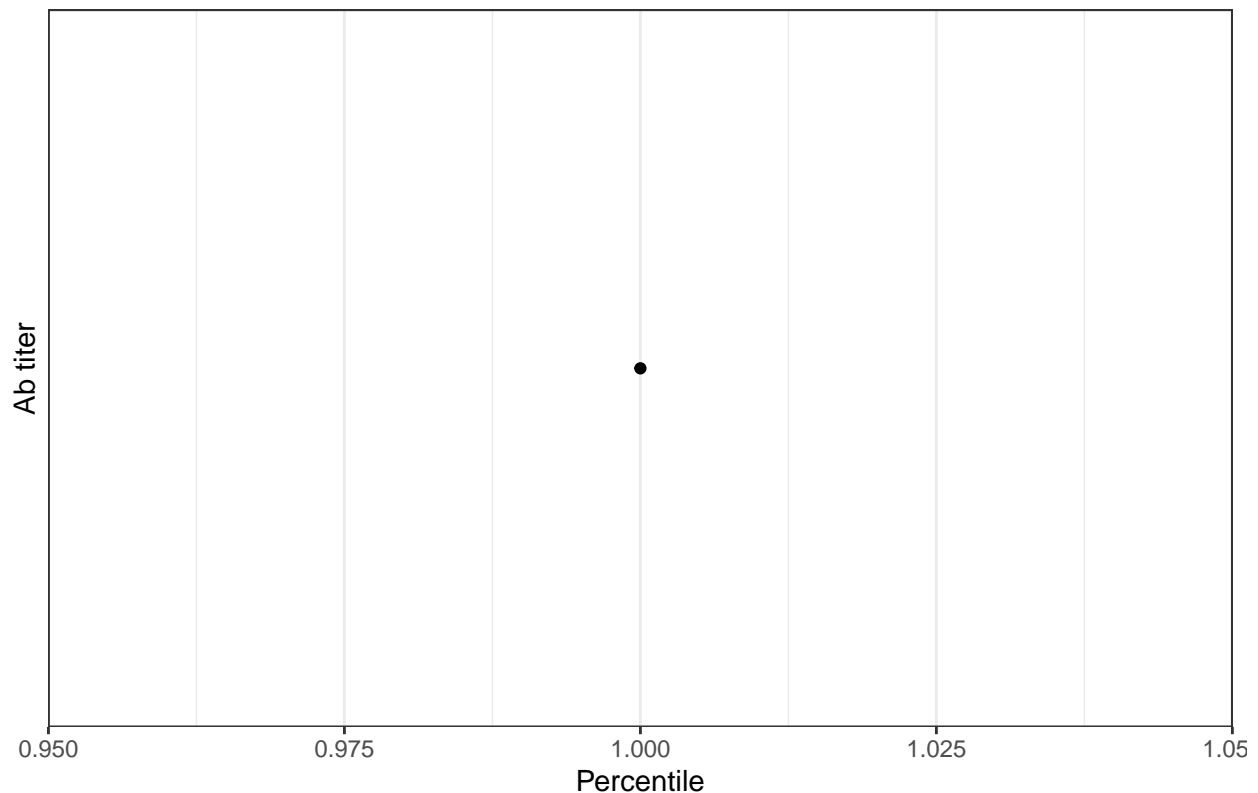
2020 Antibody titers (IgG4_TT_IU/ML)







2020 Antibody titers (IgG4_PD1_IU/ML)



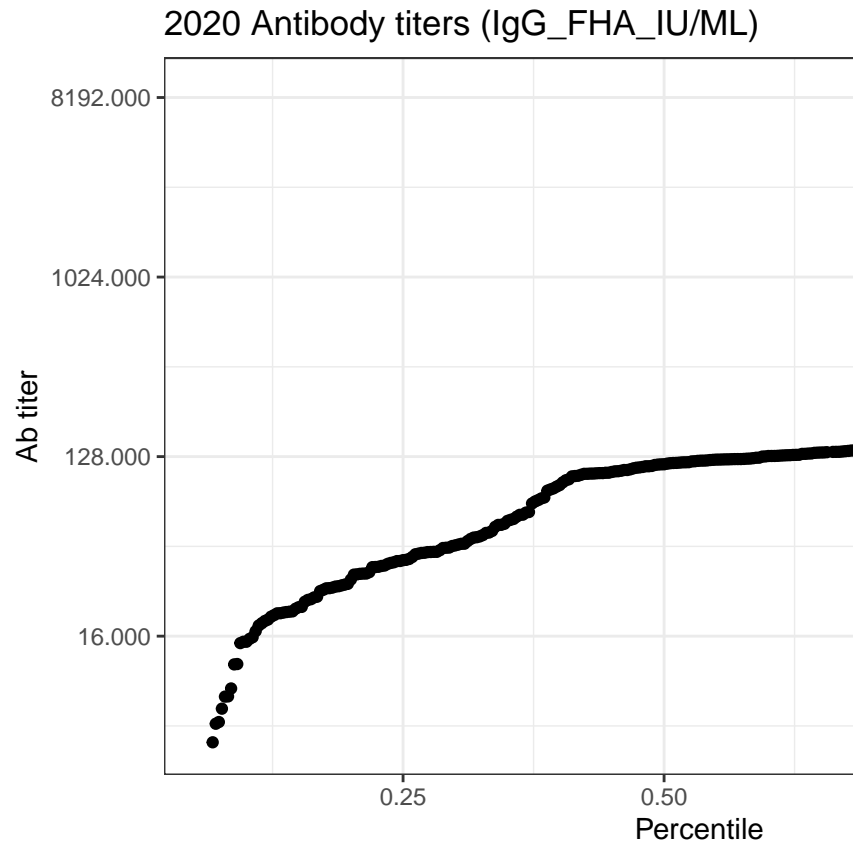
LOD Corrections

1. Top outliers: IgG_PT

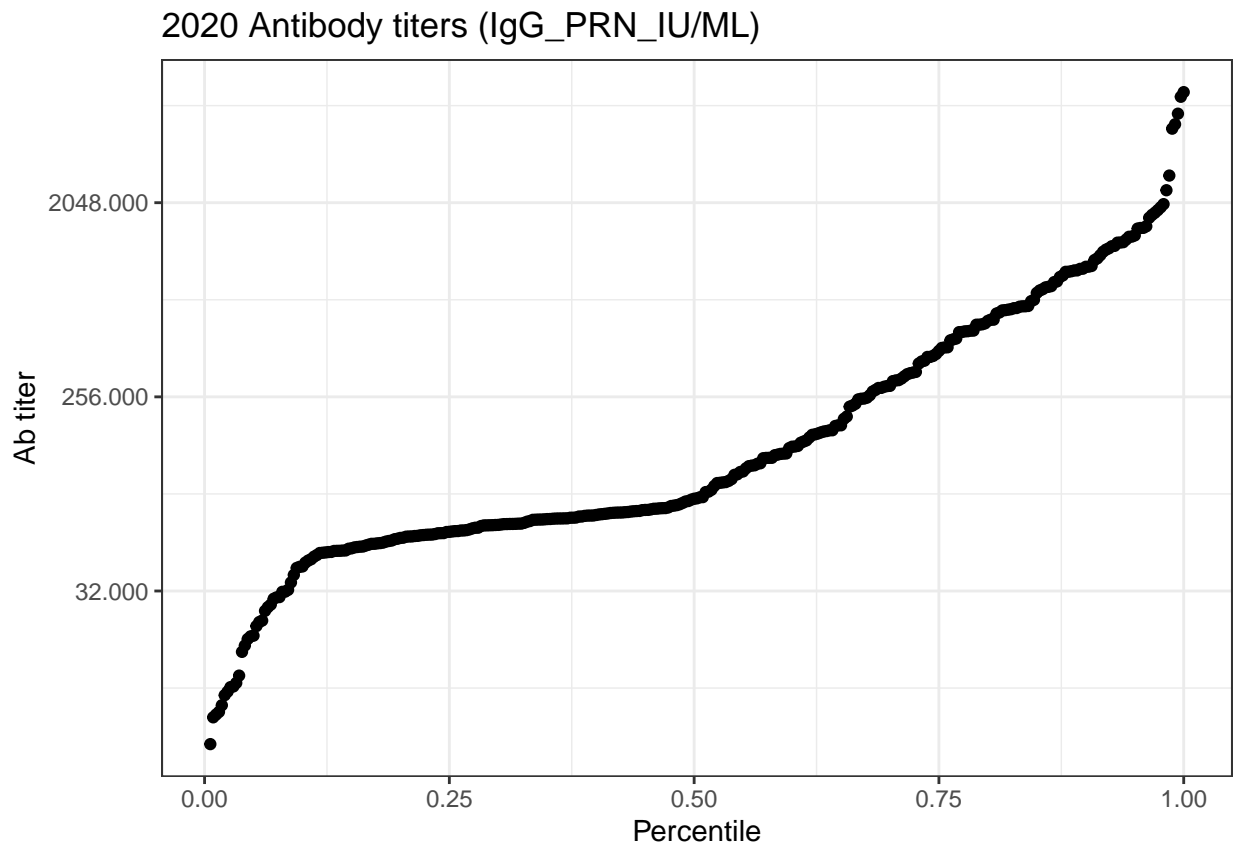
```
titters_2020_feature_removed_lod_corrected <- titters_2020_feature_removed_lod %>%
  mutate(
    lower_limit_of_detection = if_else(isotype_antigen == "IgG_FHA_IU/ML", 4.67953450834645, lower_limit_of_detection),
    lower_limit_of_detection = if_else(isotype_antigen == "IgG_PRN_IU/ML", 6.20594906363301, lower_limit_of_detection),
    ab_titer = if_else(isotype_antigen == "IgG_PT_IU/ML" & ab_titer > 15000, lower_limit_of_detection, ab_titer)
  ) %>%
  mutate(ab_titer = if_else(ab_titer < lower_limit_of_detection | is.na(ab_titer) == TRUE, lower_limit_of_detection, ab_titer))

## Validation plots
for(select_ia in c("IgG_FHA_IU/ML", "IgG_PRN_IU/ML", "IgG_PT_IU/ML"))
{
  plot1 <- titters_2020_feature_removed_lod_corrected %>%
    mutate(subject_id = as.factor(subject_id)) %>%
    filter(isotype_antigen == select_ia, planned_day_relative_to_boost < 50) %>%
    arrange(desc(ab_titer)) %>%
    ggplot(aes(y=ab_titer)) + stat_ecdf(geom = "point", pad = FALSE) +
    labs(y = "Ab titer", x = "Percentile")+ ggtitle(paste0("2020 Antibody titers (", select_ia, ")")) +
    scale_y_continuous(trans = 'log2', labels=scaleFUN)
```

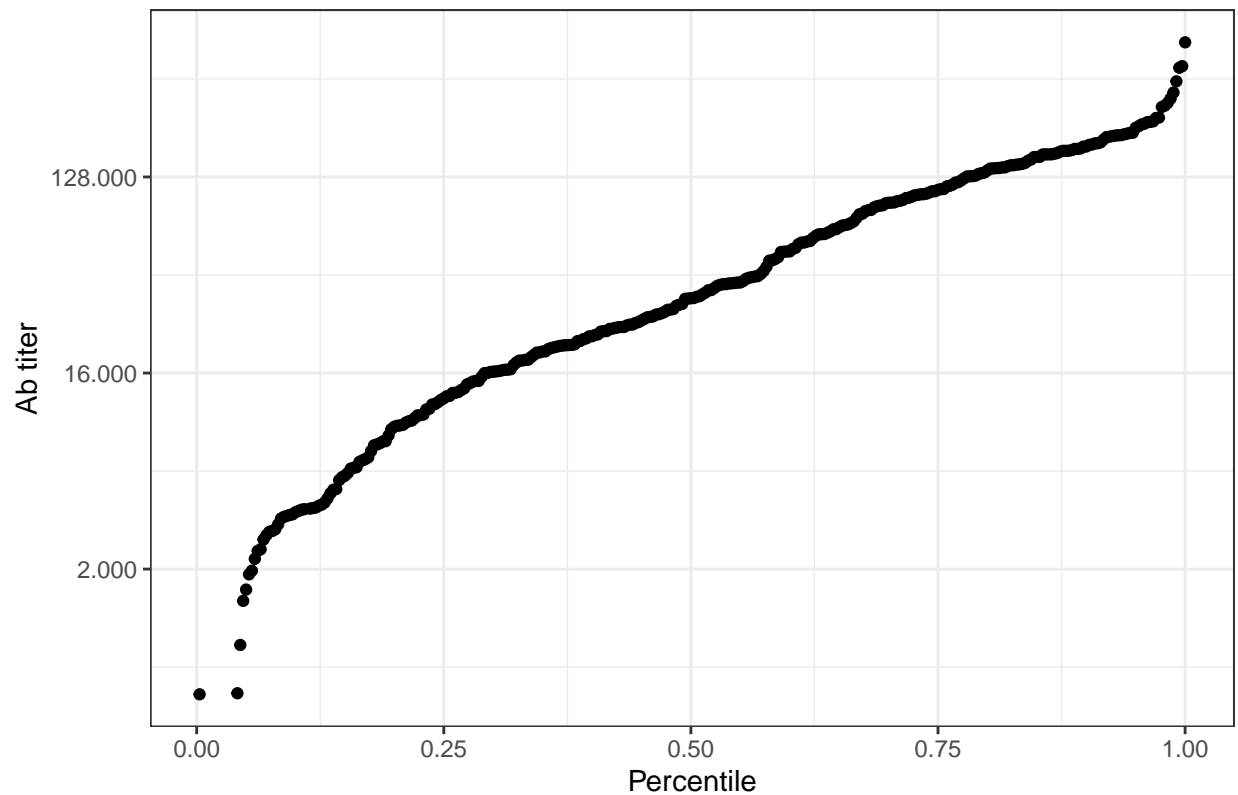
```
plot(plot1)
}
```



2. Bottom outliers: IgG_FHA, IgG_PRN



2020 Antibody titers (IgG_PT_IU/ML)



Performing median based normalisation

```
## Setting MFI zero values to lower limit of detection
titters2020_calculate_lod <- titters_2020_feature_removed_lod_corrected %>%
  #filter(isotype_antigen == 'IgE_Total') %>%
  group_by(isotype_antigen) %>%
  mutate(
    lod_new = min(ab_titer[ab_titer > 0], na.rm = TRUE),
    MFI_na_removed = if_else(ab_titer == 0 | is.na(ab_titer) == TRUE, lod_new, ab_titer)
  ) %>%
  ungroup()

## Calculate Overall median using MFI at day post boost 0
df_d0_median_2020 <- titters2020_calculate_lod %>%
  filter(planned_day_relative_to_boost == 0) %>%
  group_by(isotype_antigen) %>%
  summarise(
    MFI_median_d0 = median(MFI_na_removed),
  ) %>%
  ungroup()

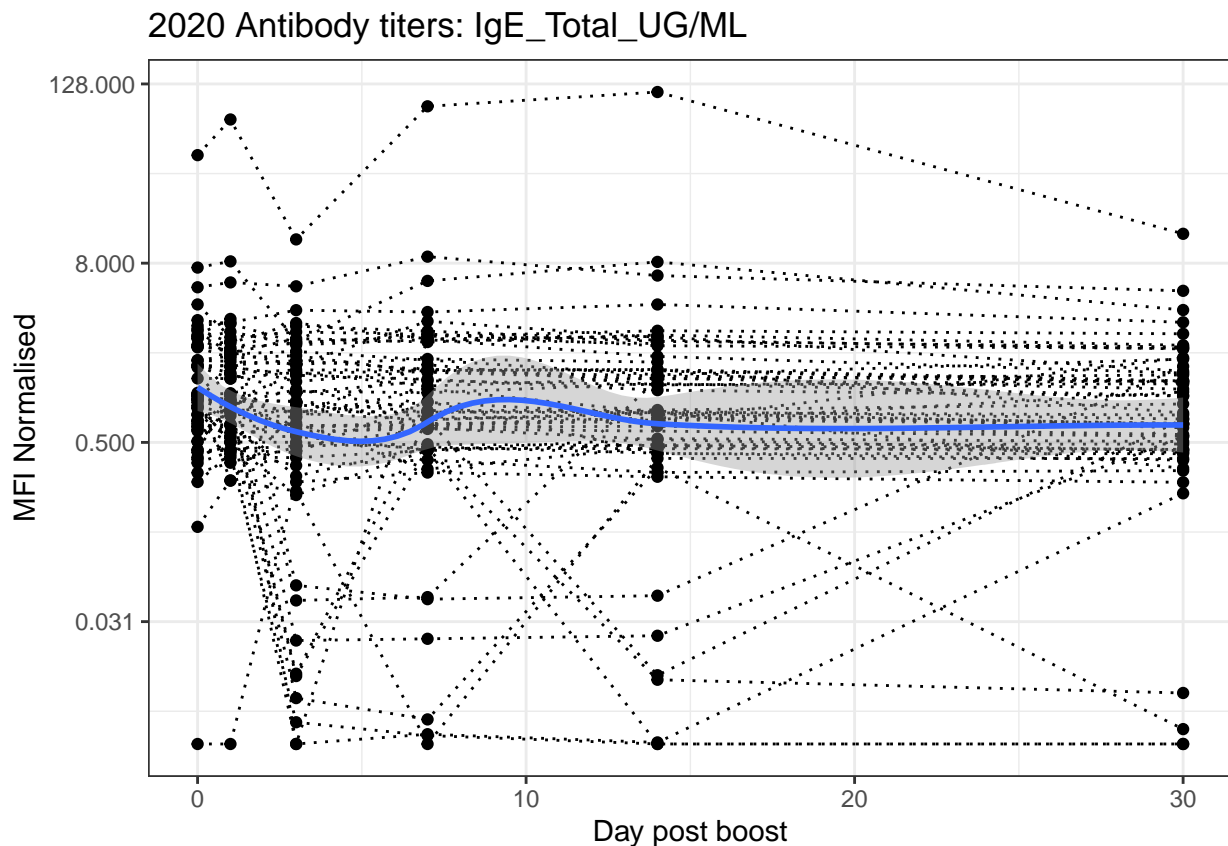
titters_2020_new_raw <- left_join(titters2020_calculate_lod , df_d0_median_2020 , by = "isotype_antigen")
mutate(
  MFI_normalized = MFI_na_removed / MFI_median_d0,
)
```

Plot longitudinal data after median based normalisation

```
#for(select_ia in c('IgE_Total_IU/ML'))
for(select_ia in unique(titers_2020_new_raw$isotype_antigen))
{
plot1 <- titers_2020_new_raw %>%
  mutate(subject_id = as.character(subject_id)) %>%
  filter(isotype_antigen == select_ia, planned_day_relative_to_boost < 50) %>%
  ggplot(aes(x=planned_day_relative_to_boost, y=MFI_normalized, )) +
    geom_line(aes(group=subject_id), linetype = "dotted") +
    geom_point() +
    labs(x = "Day post boost", y = "MFI Normalised") +
    geom_smooth(size = 1) +
    theme_bw() +
    theme(strip.background = element_blank(), strip.placement = "outside") +
    ggtitle(paste0("2020 Antibody titers: ", select_ia)) +
    scale_y_continuous(trans = 'log2', labels=scaleFUN)

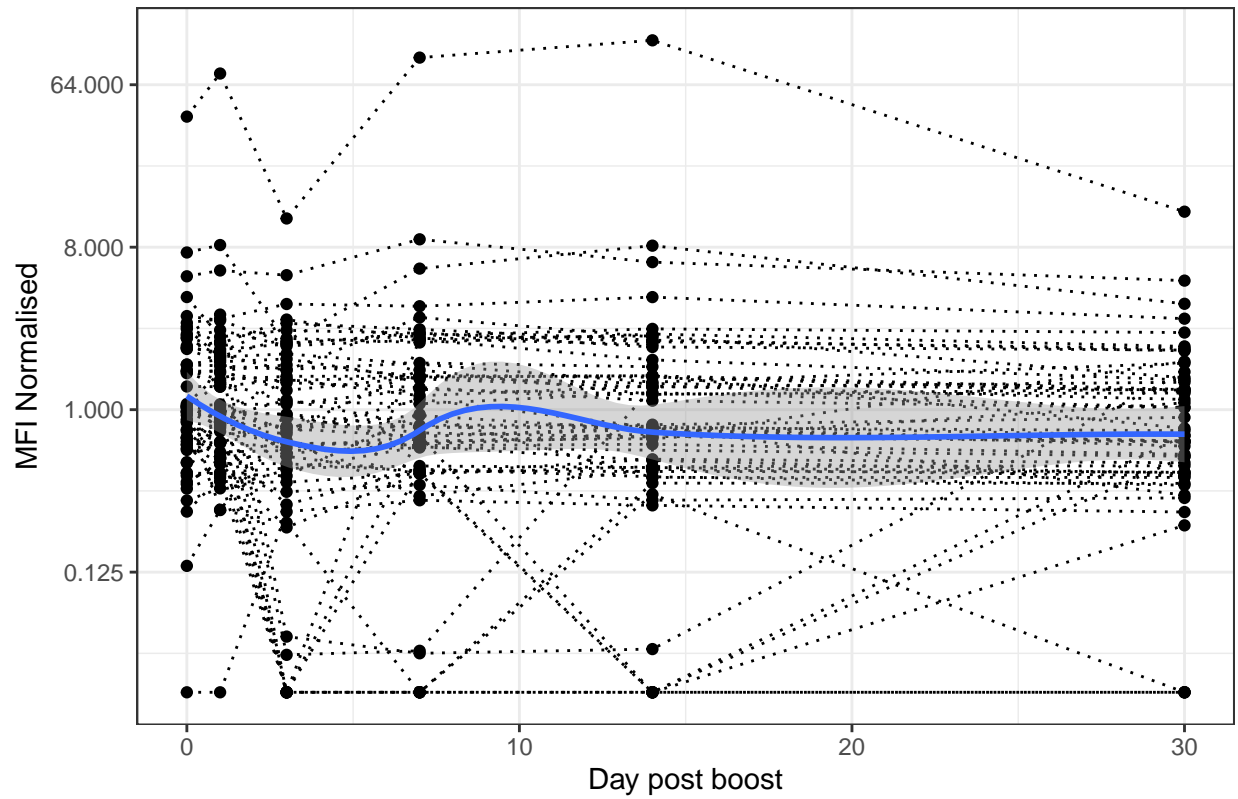
plot(plot1)
}
```

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



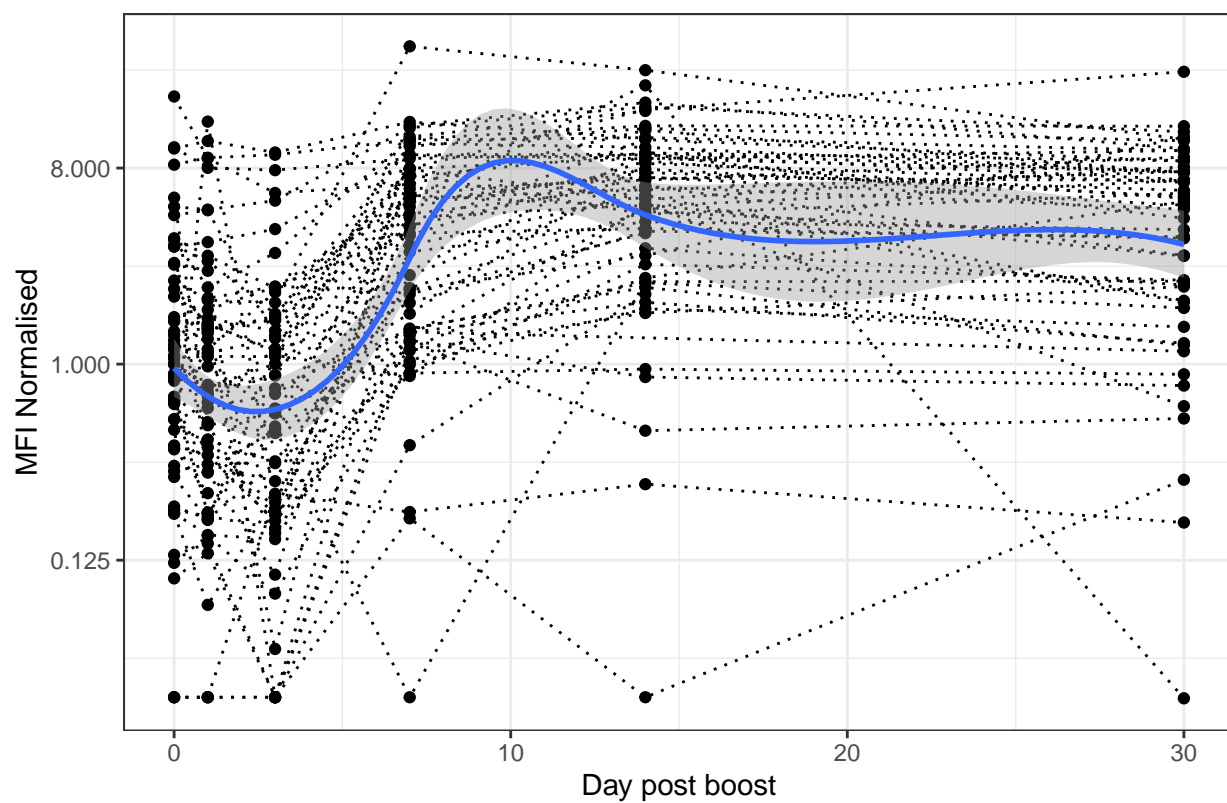
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_Total_IU/ML



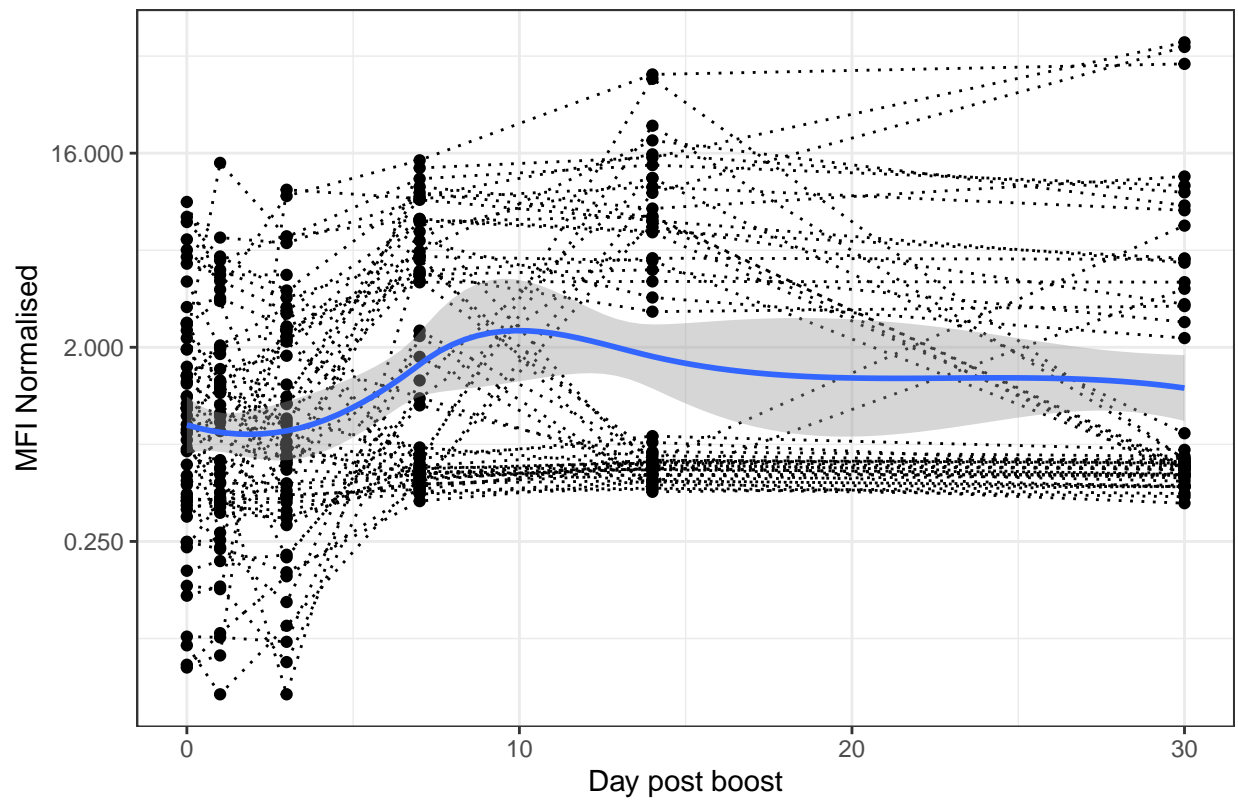
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgG_PT_IU/ML



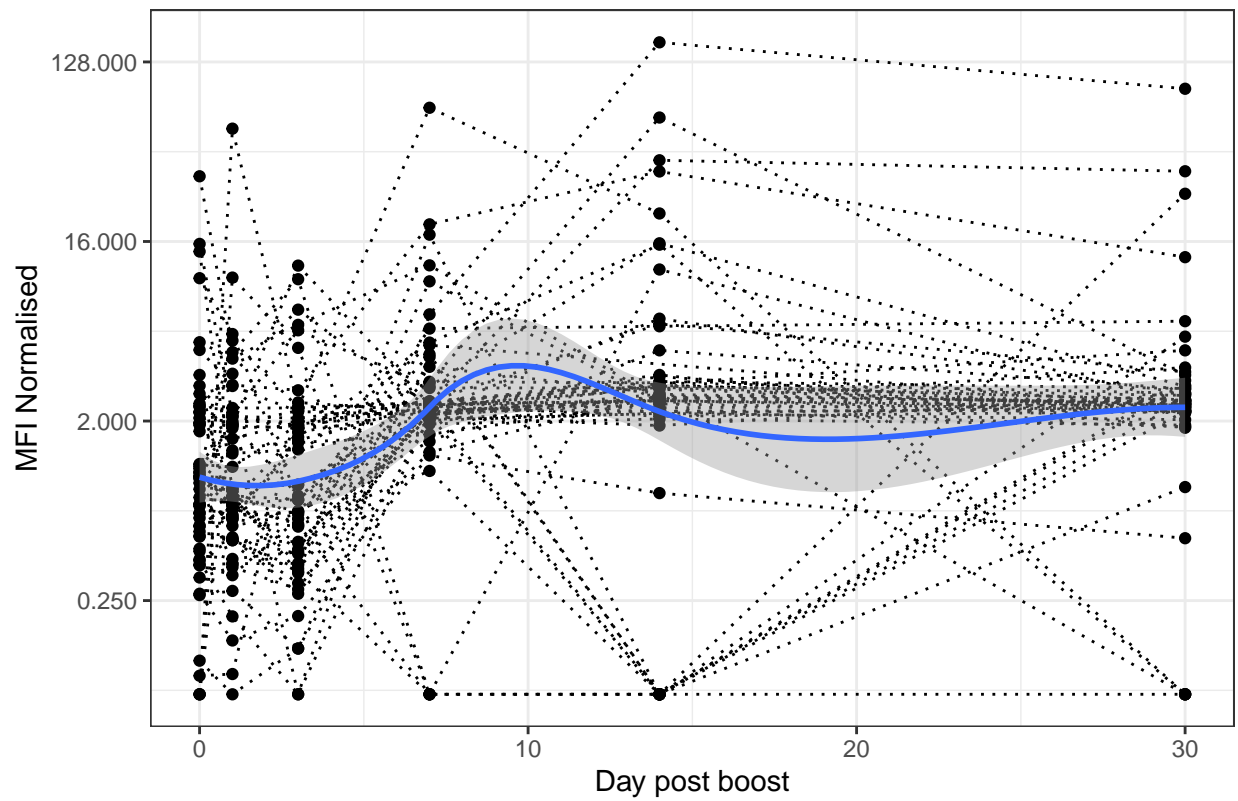
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG_PRN_IU/ML

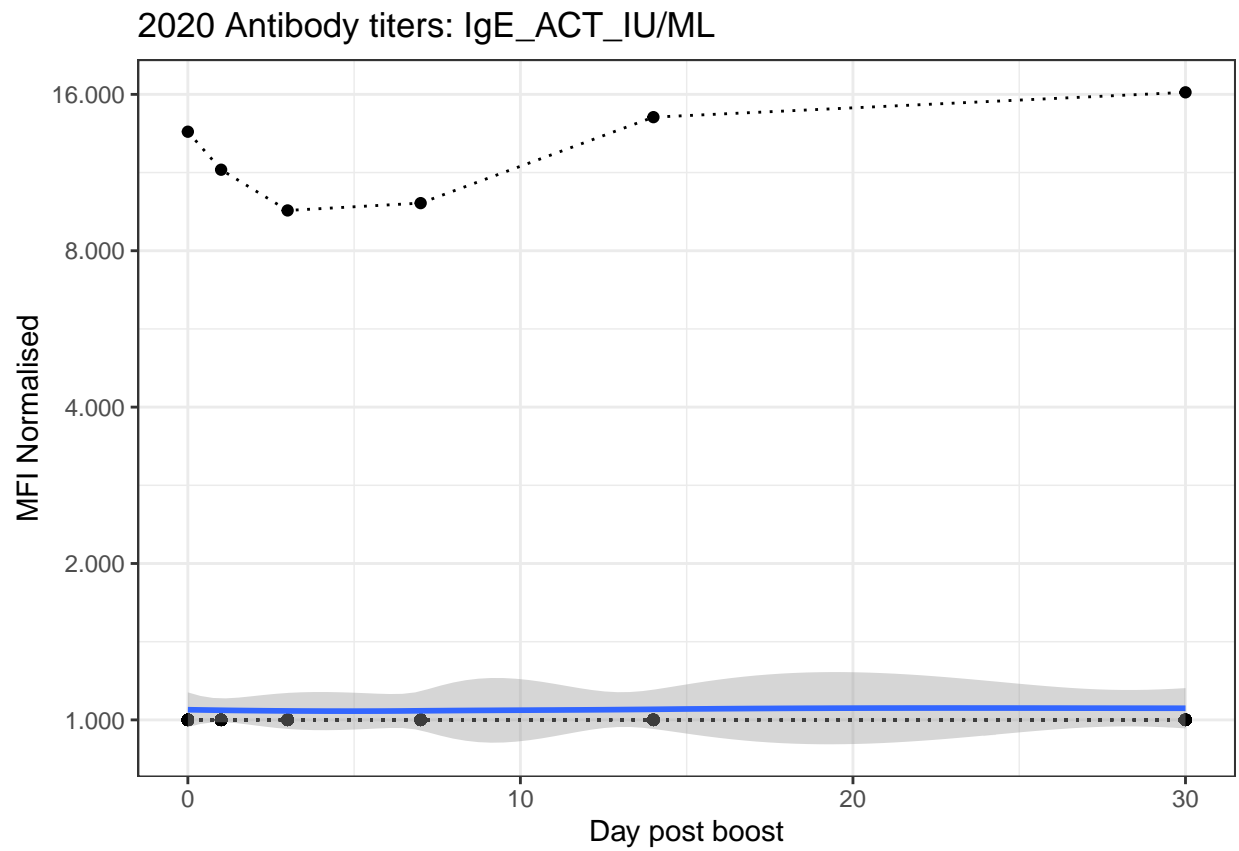


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

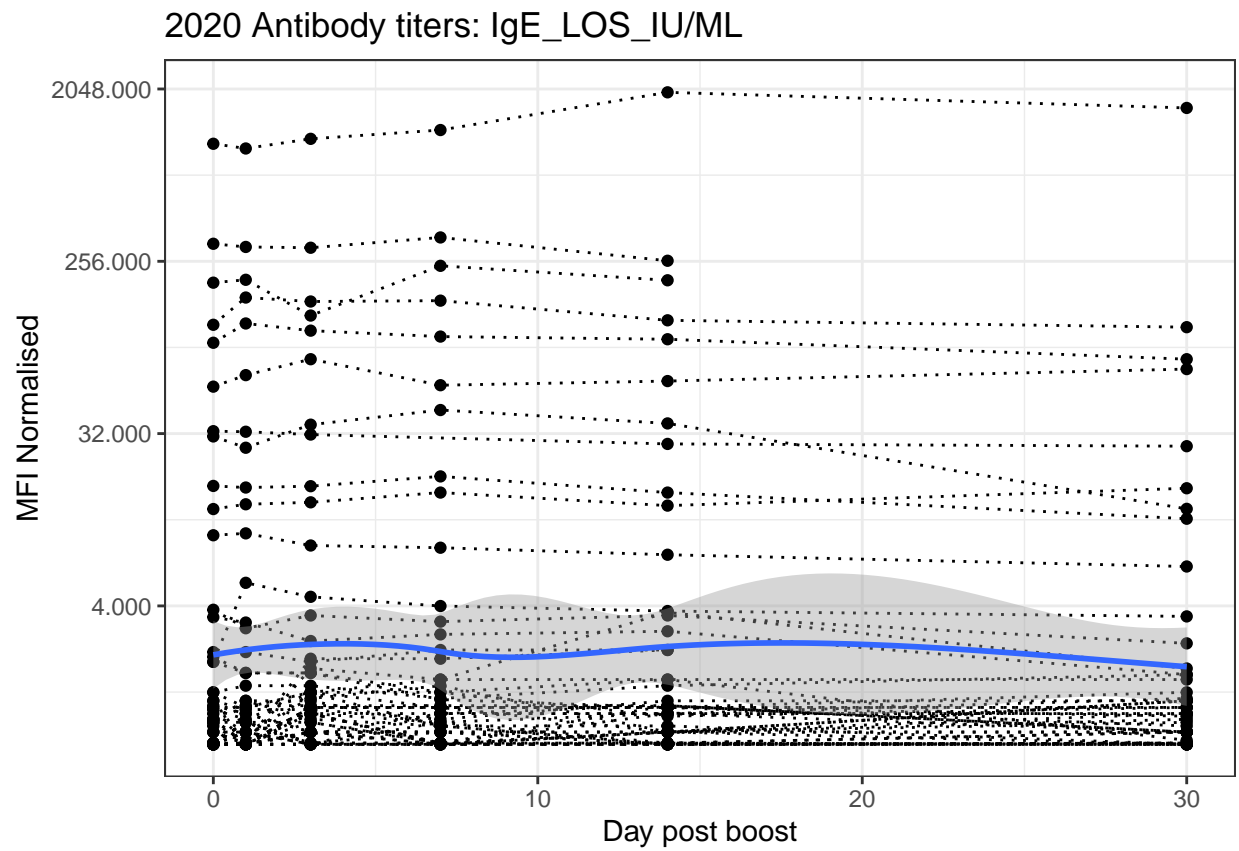
2020 Antibody titers: IgG_FHA_IU/ML



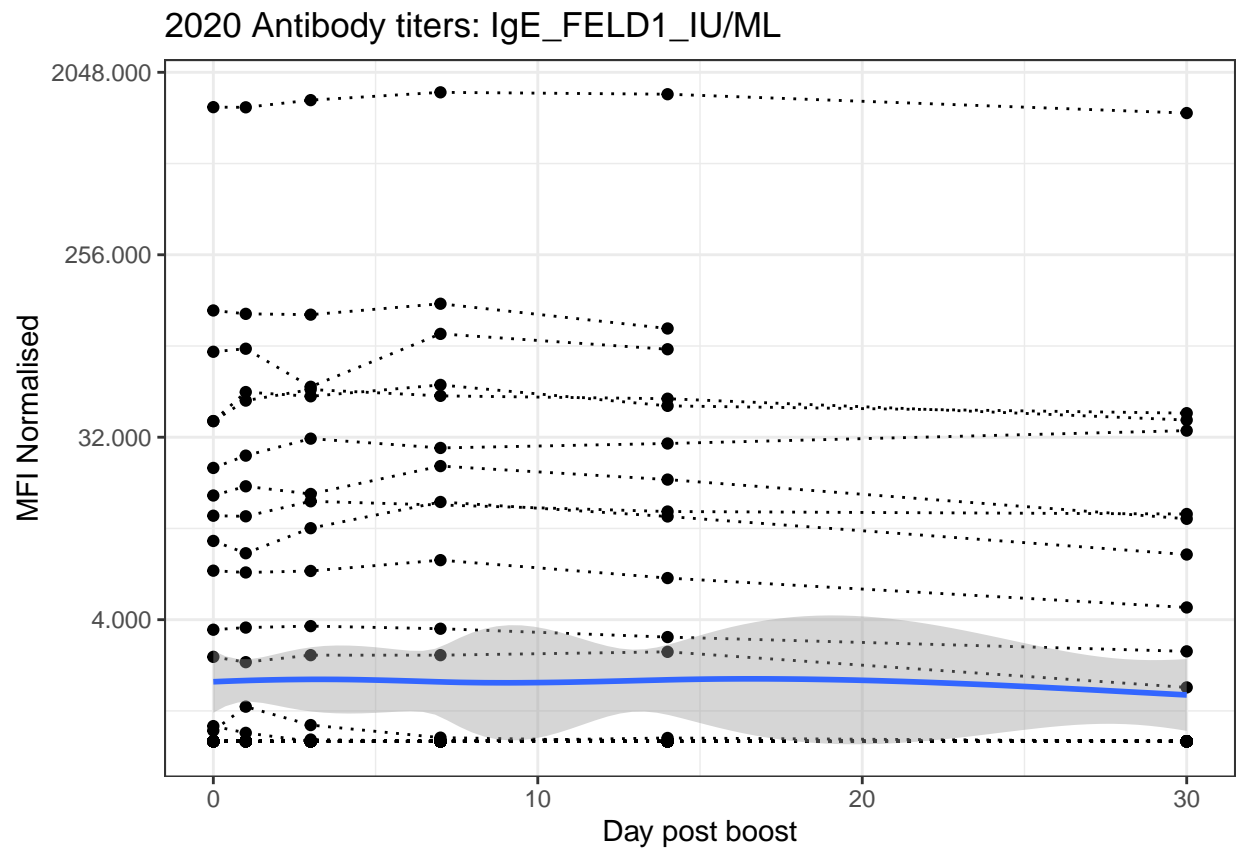
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



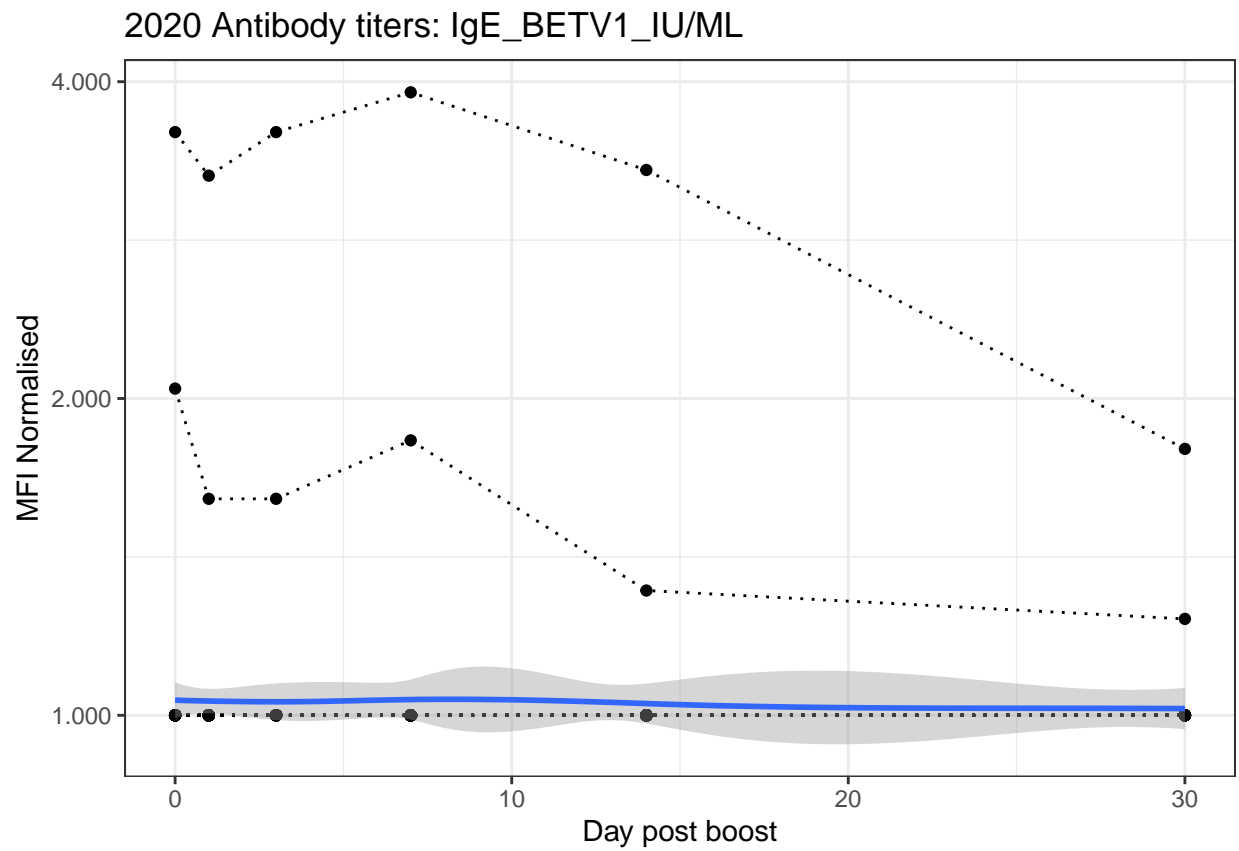
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



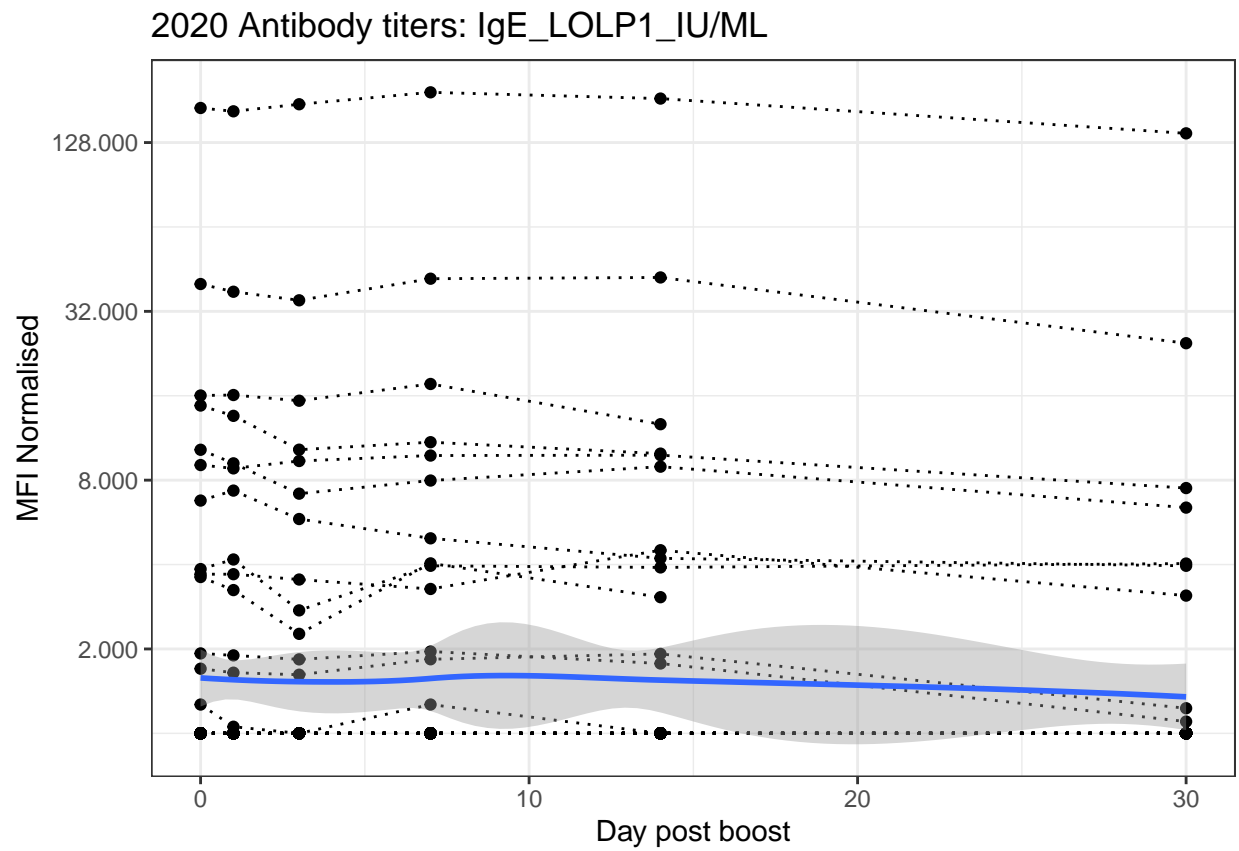
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



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

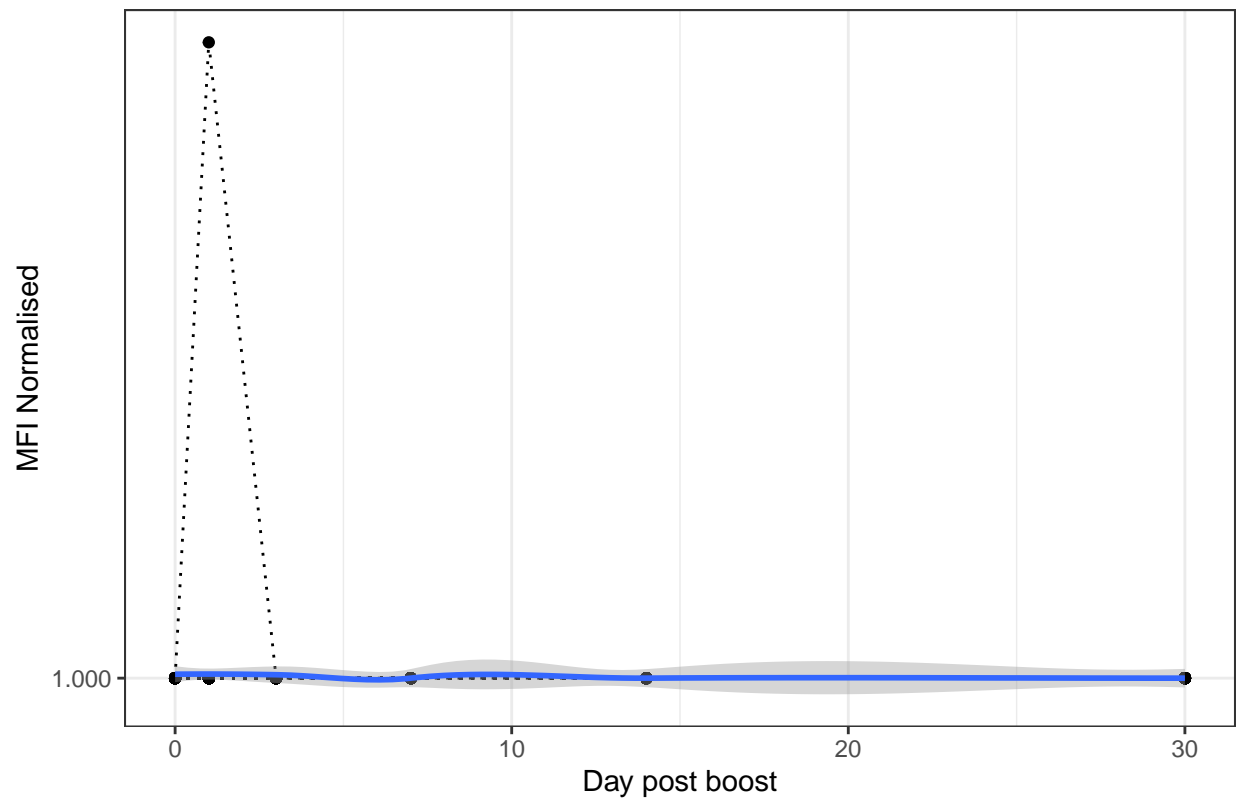


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



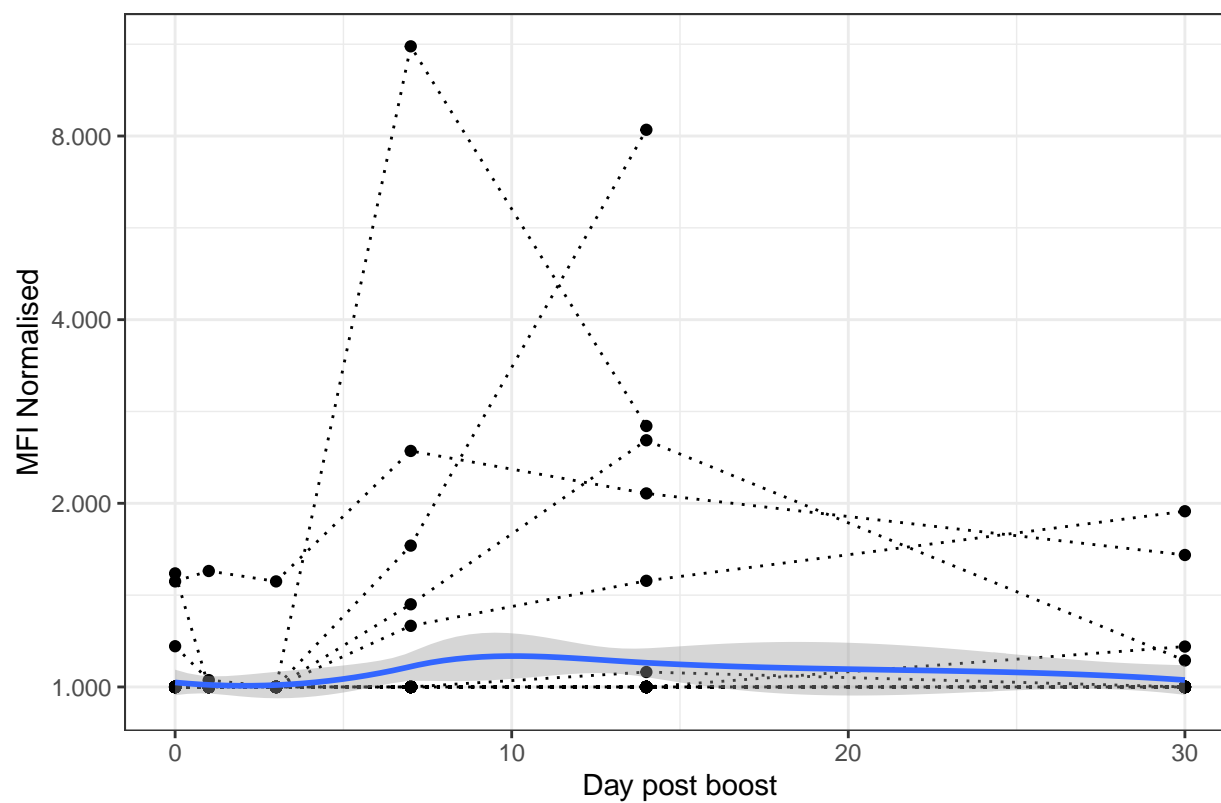
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgE_Measles_IU/ML



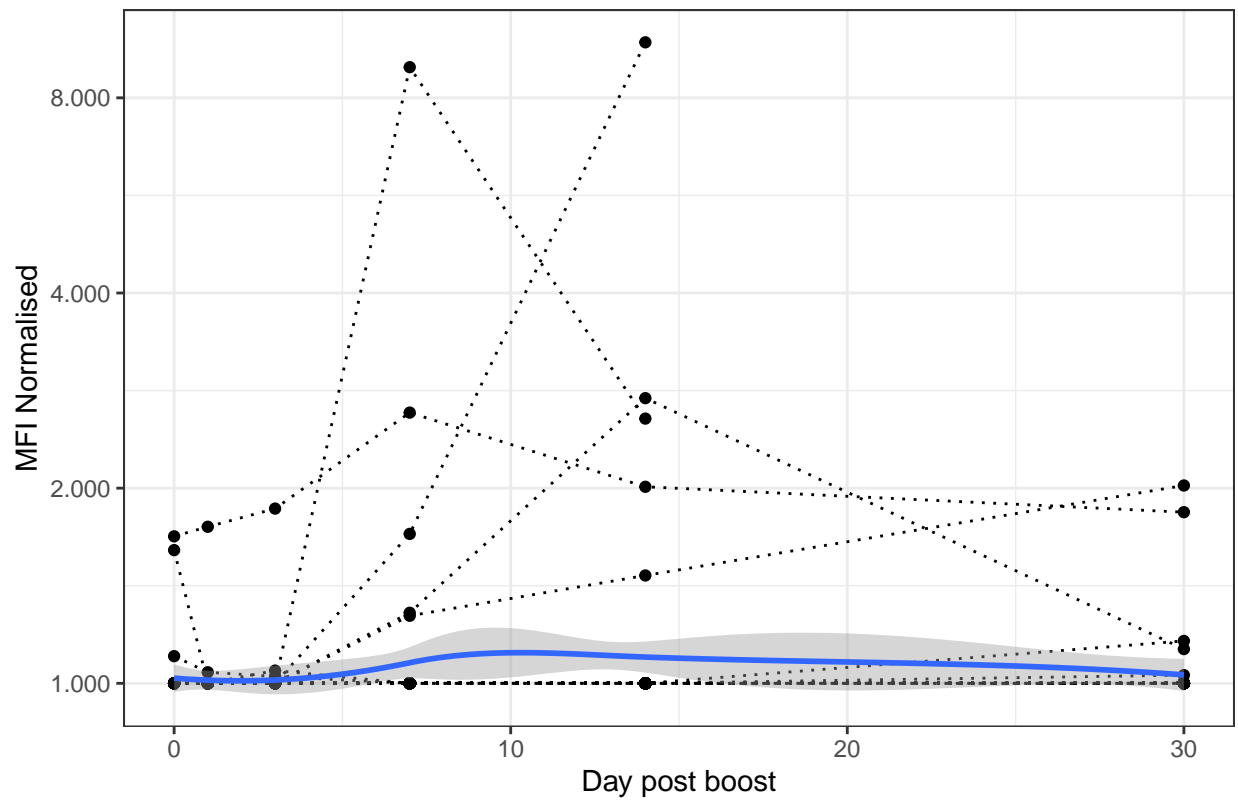
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_PTM_IU/ML

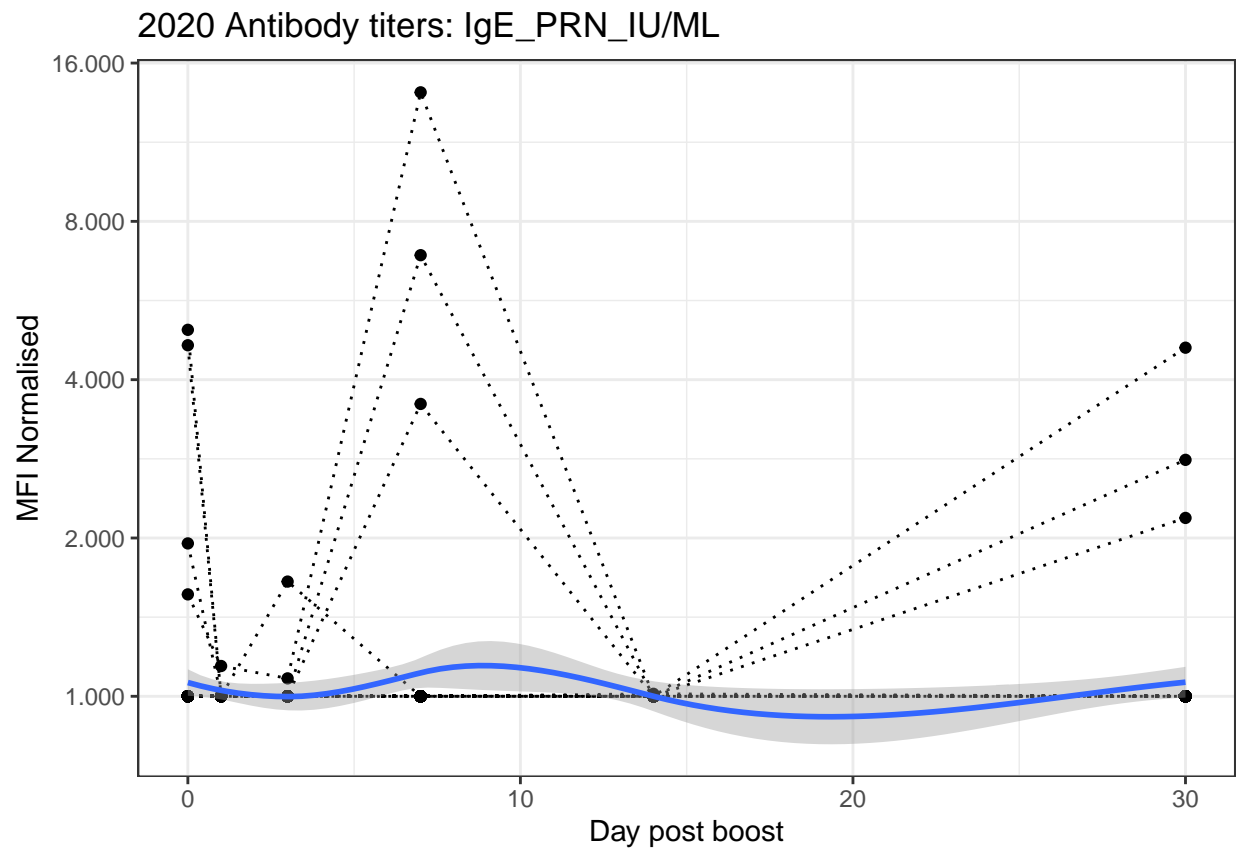


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_PT_IU/ML

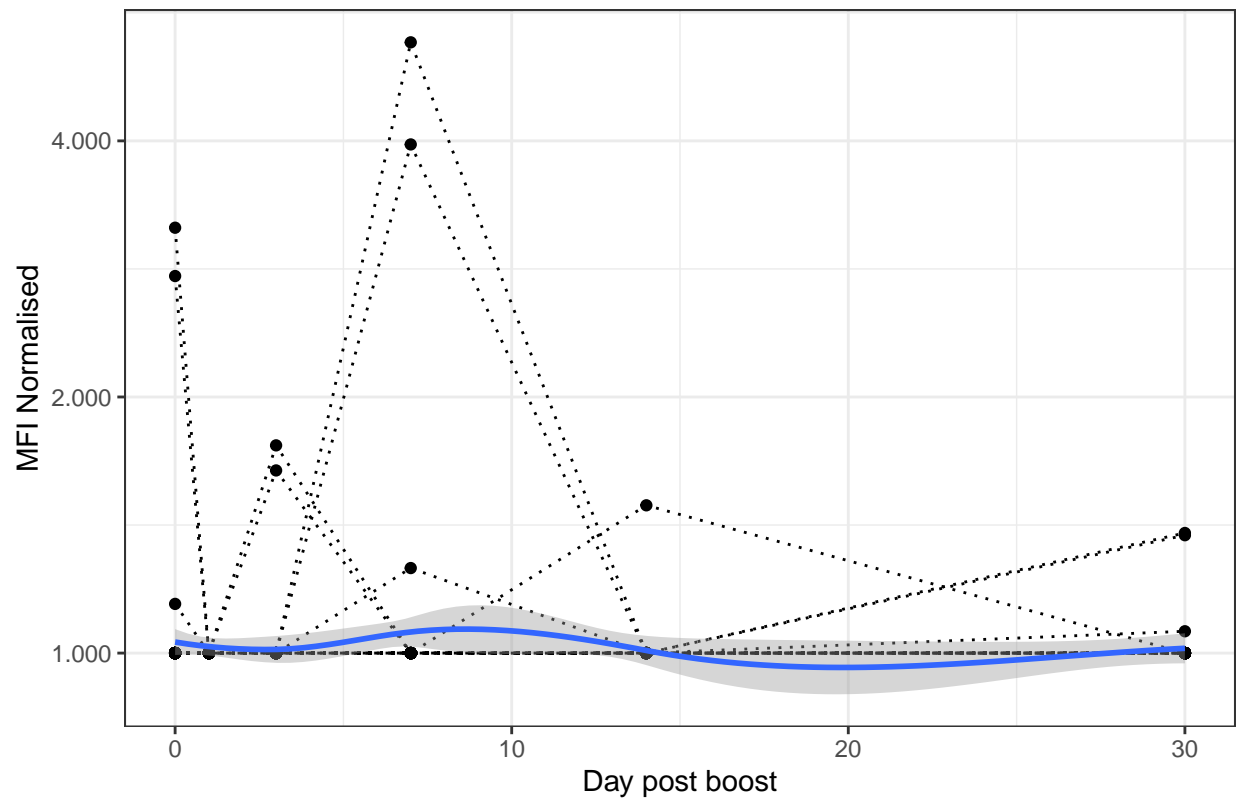


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



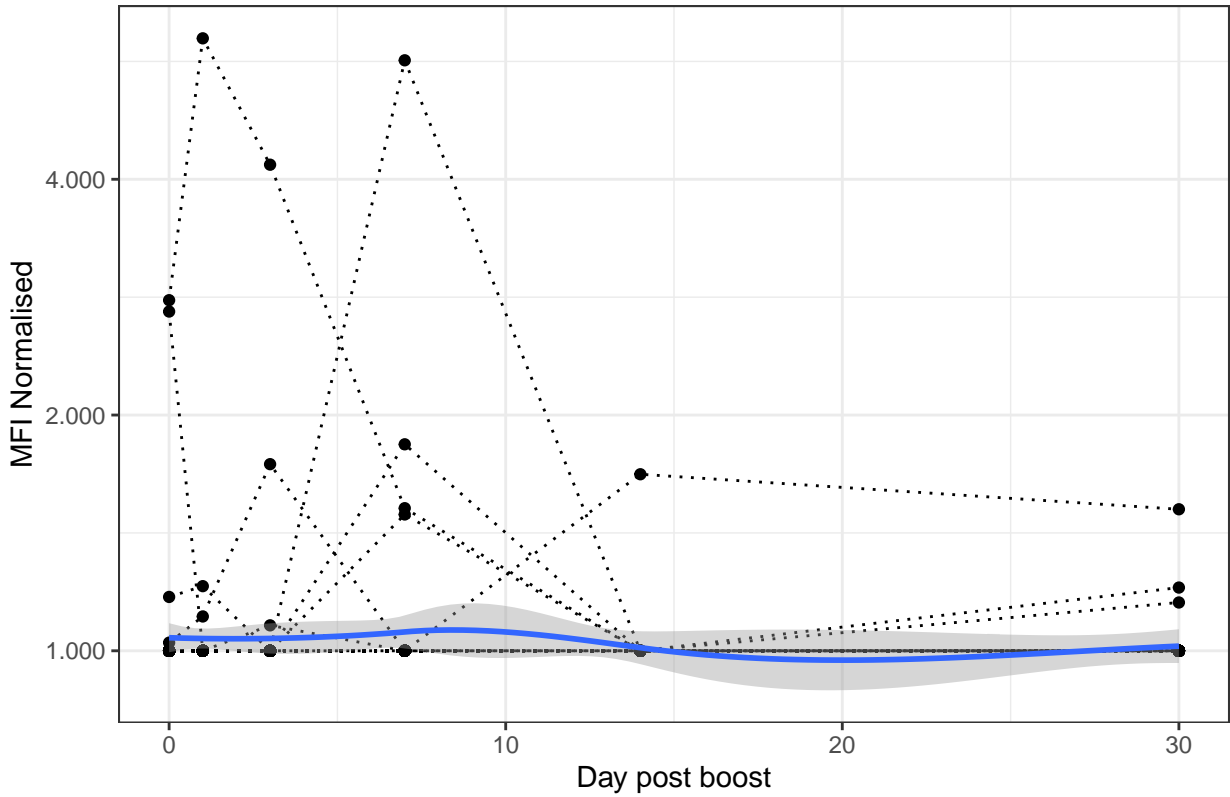
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_FHA_IU/ML



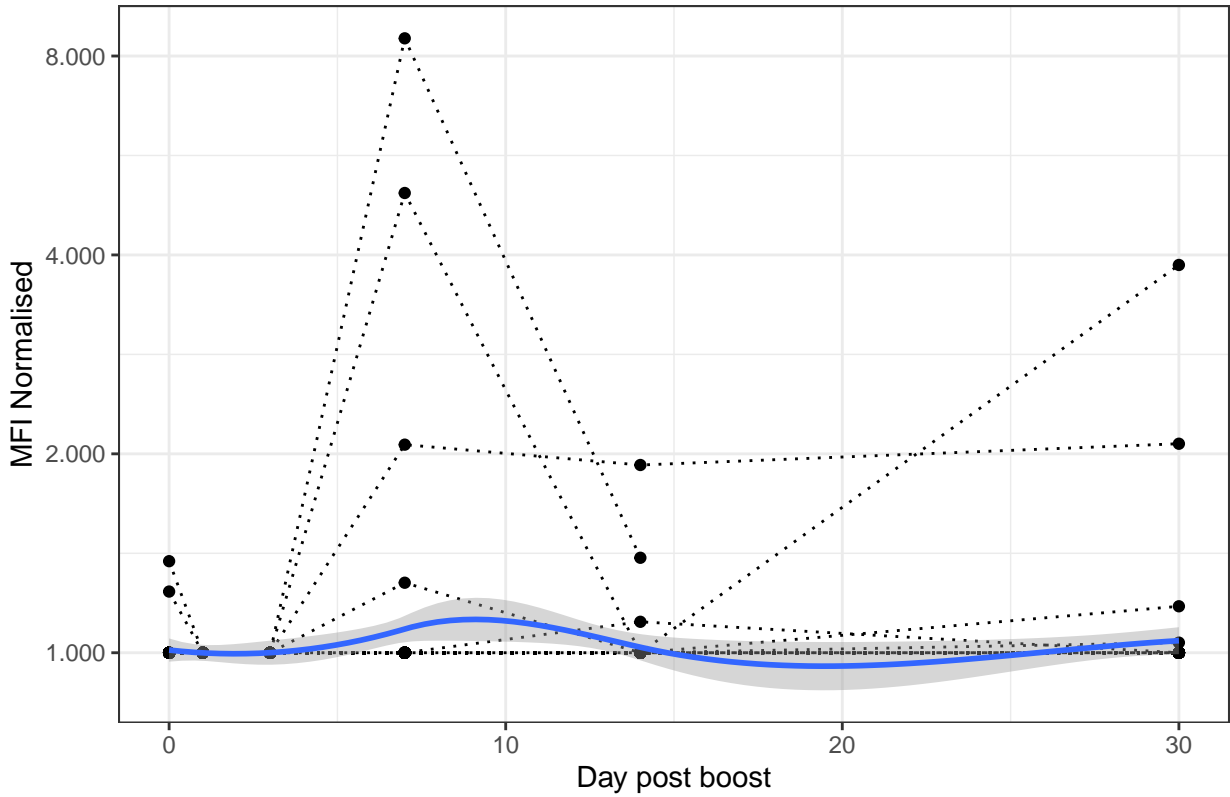
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_FIM2/3_IU/ML



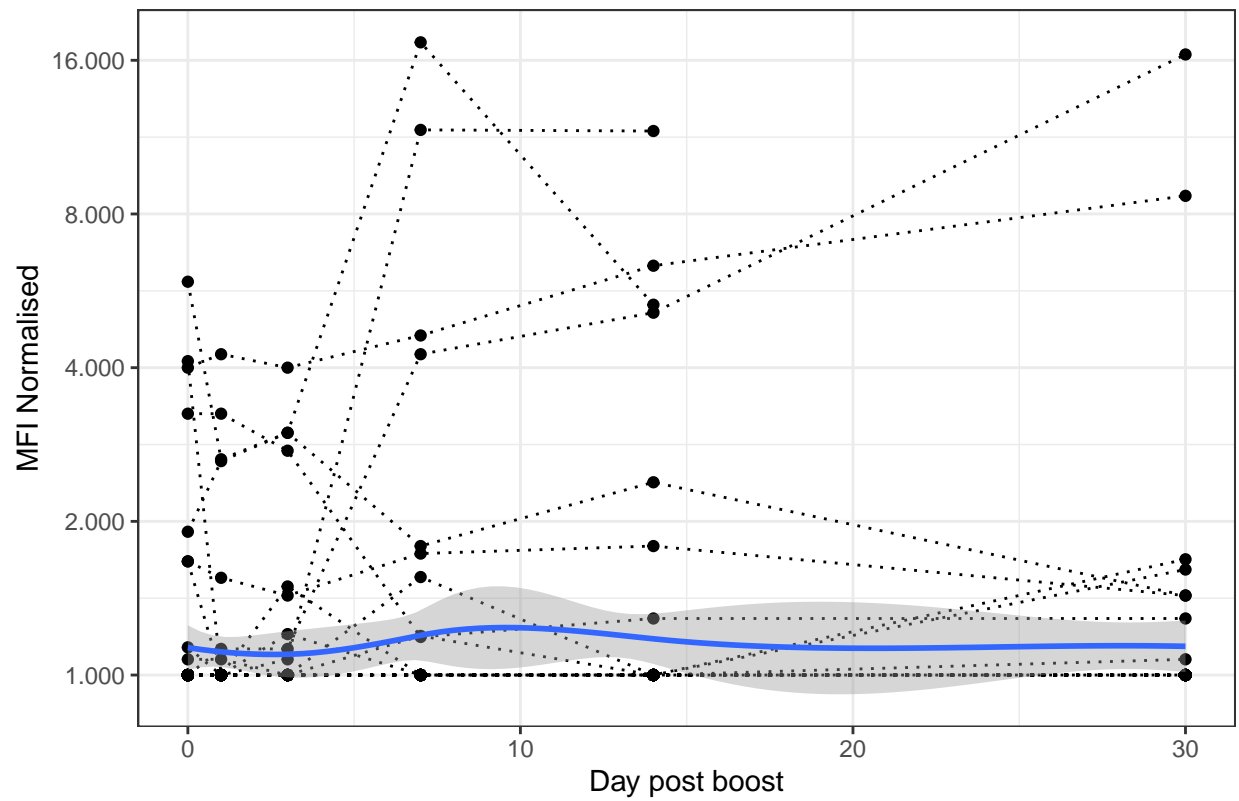
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_TT_IU/ML



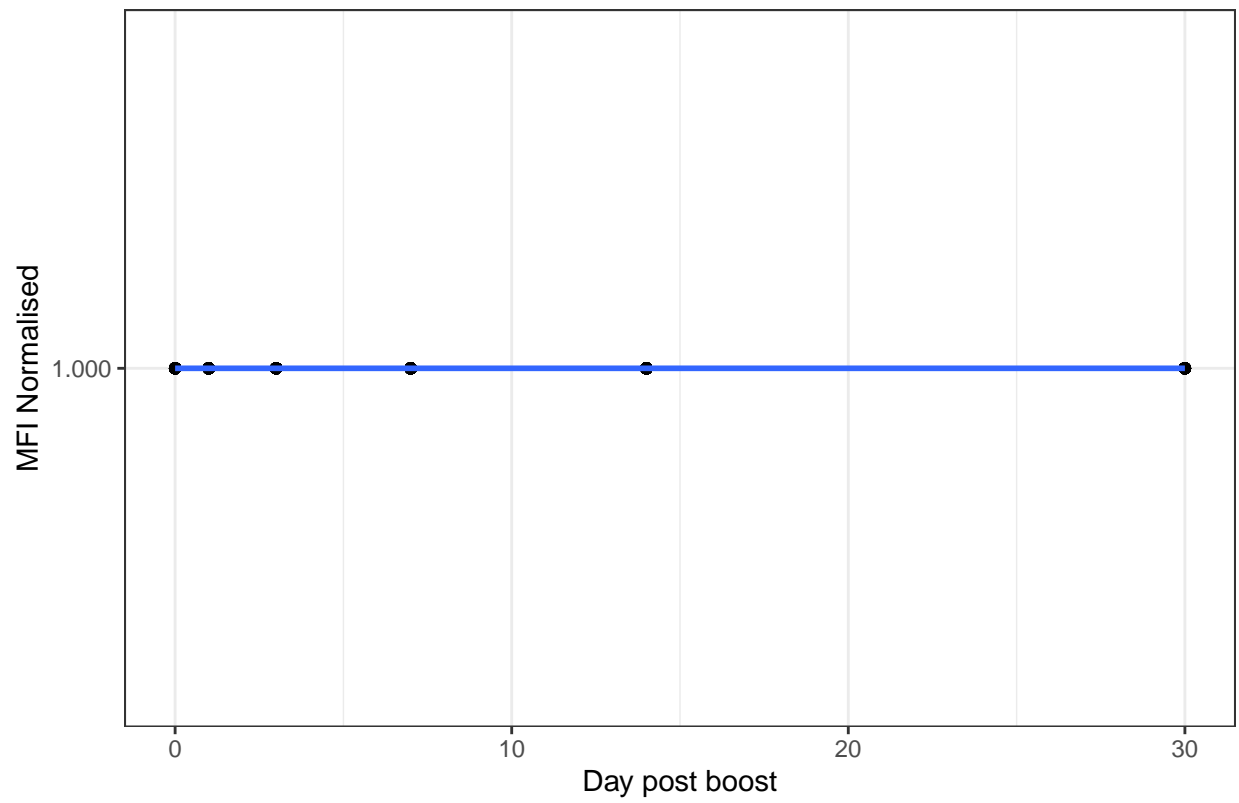
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_DT_IU/ML



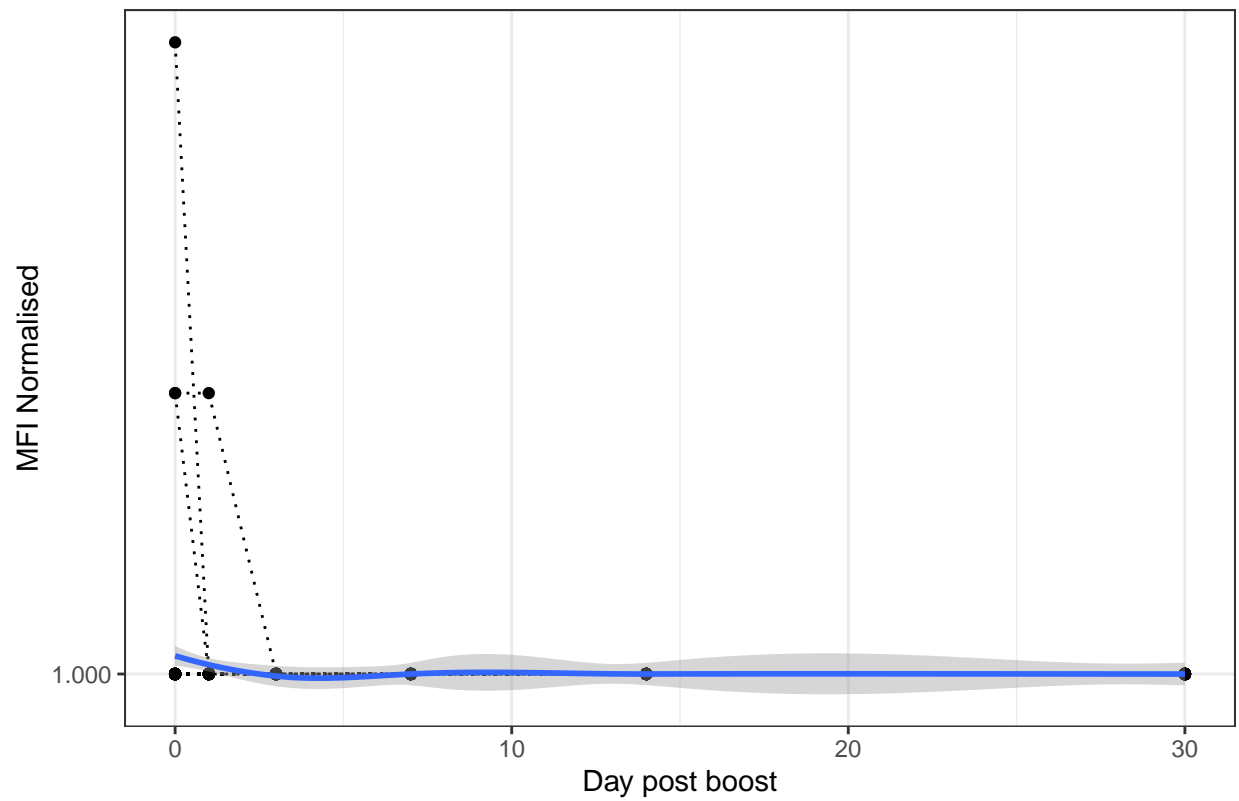
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgE_OVA_IU/ML



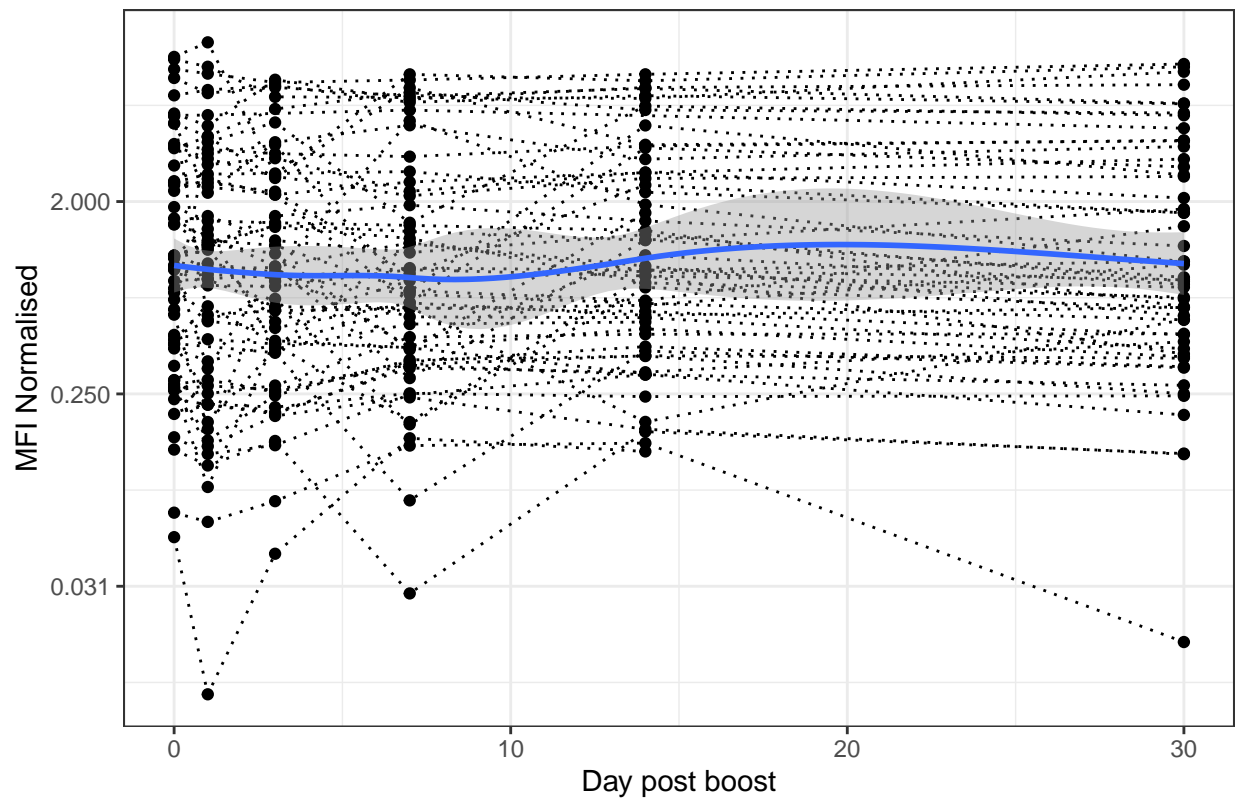
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgE_PD1_IU/ML

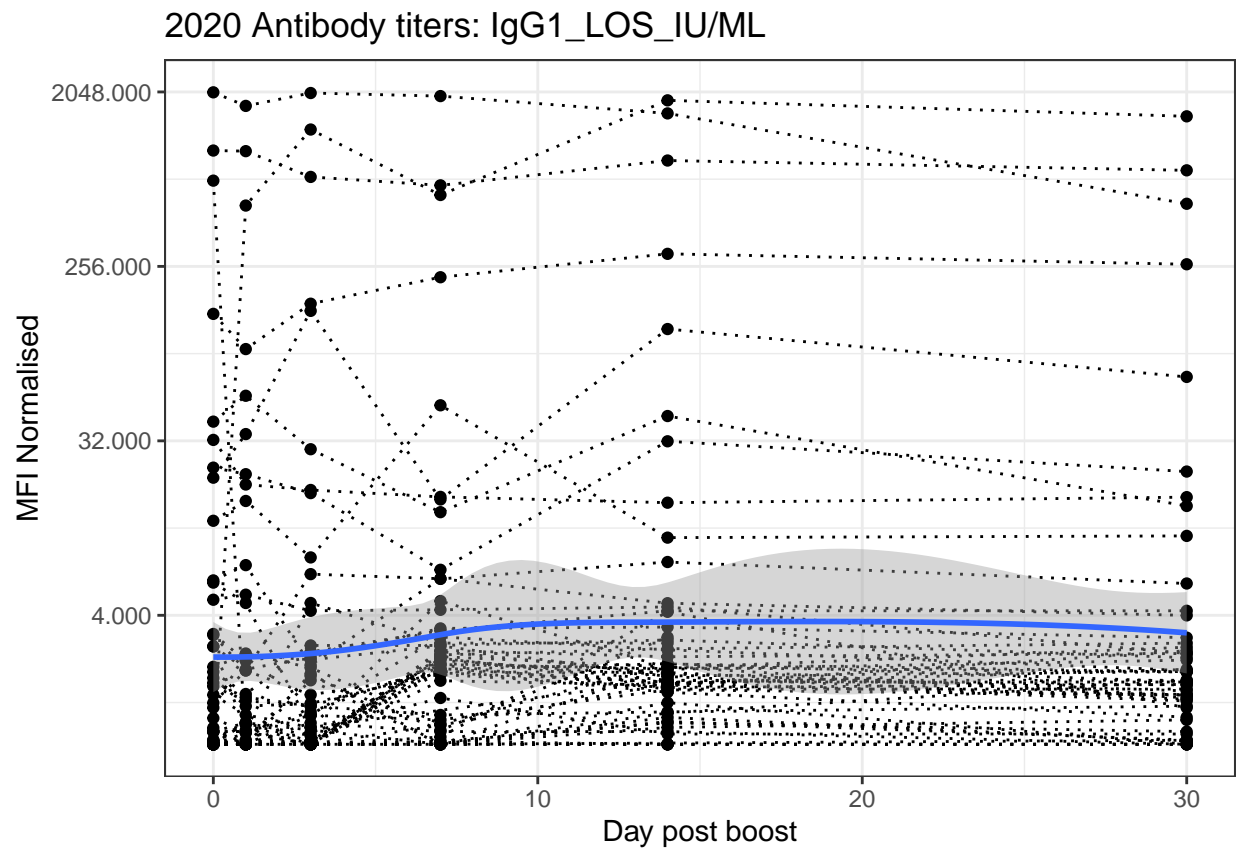


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_ACT_IU/ML

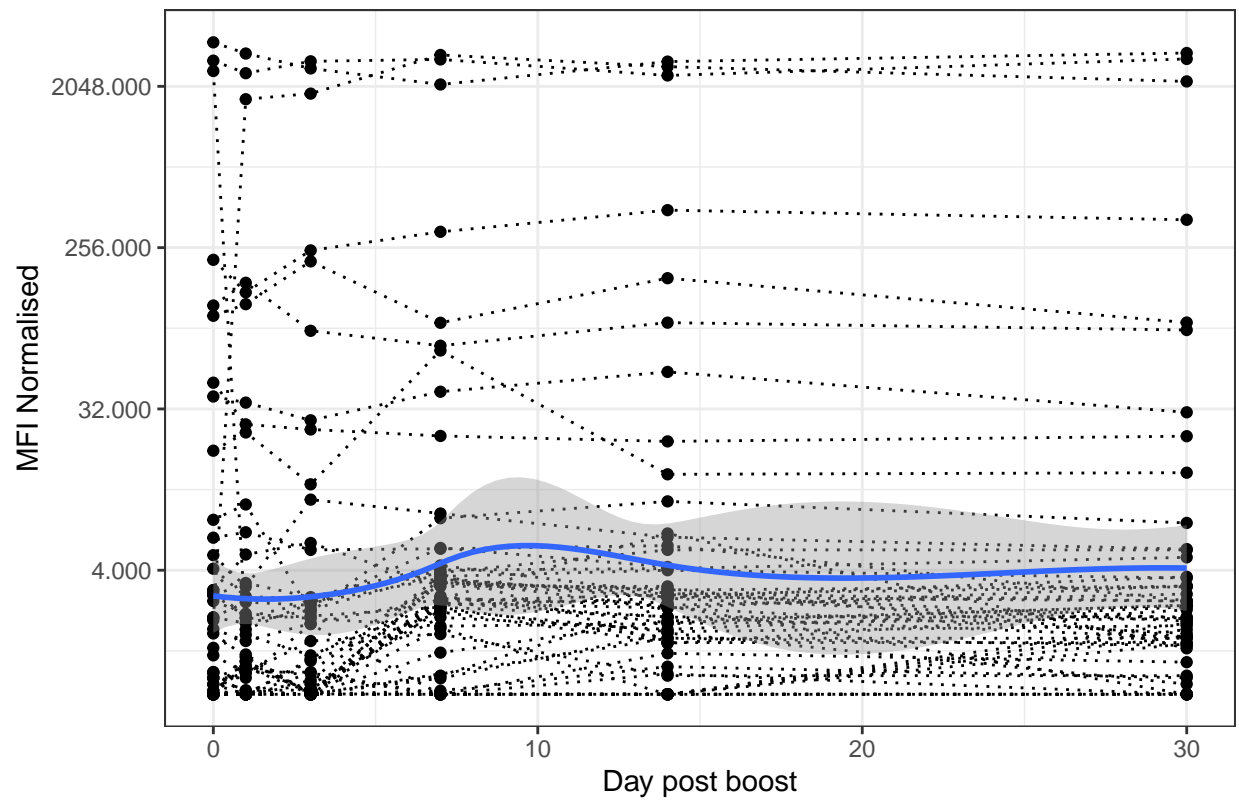


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



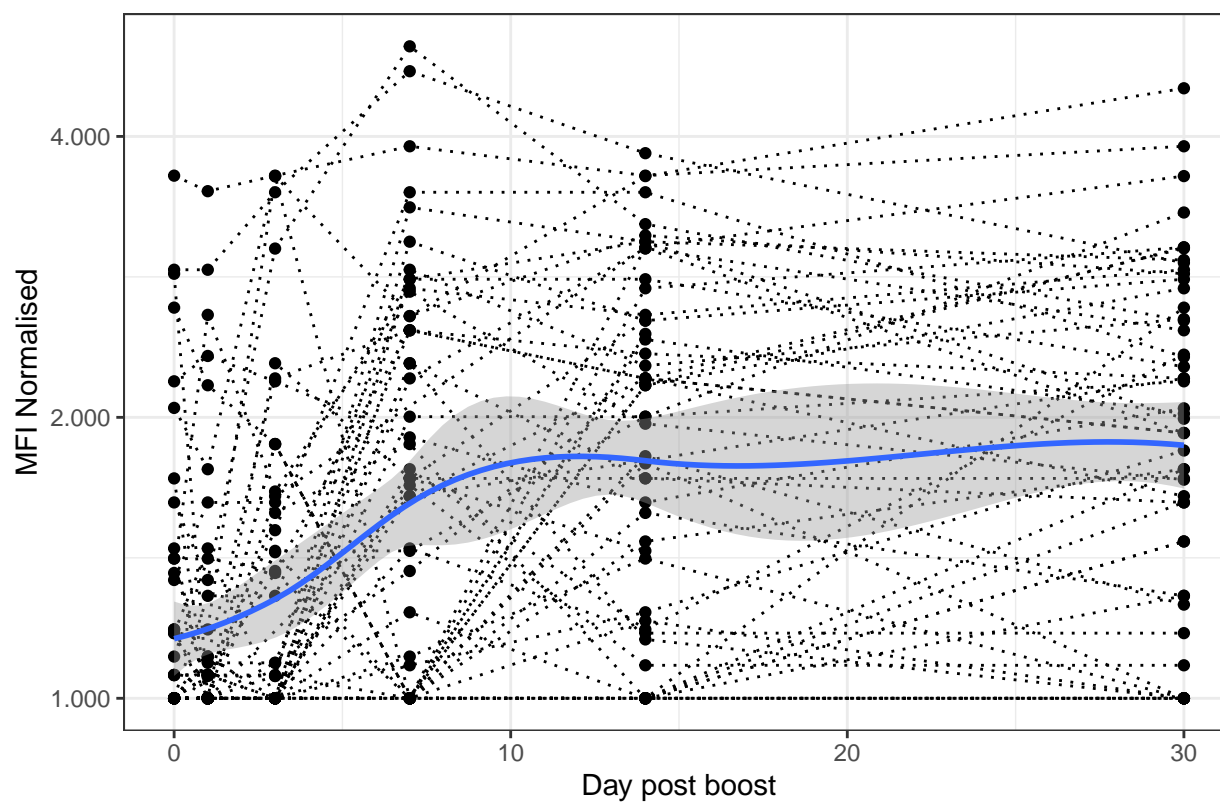
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_FELD1_IU/ML

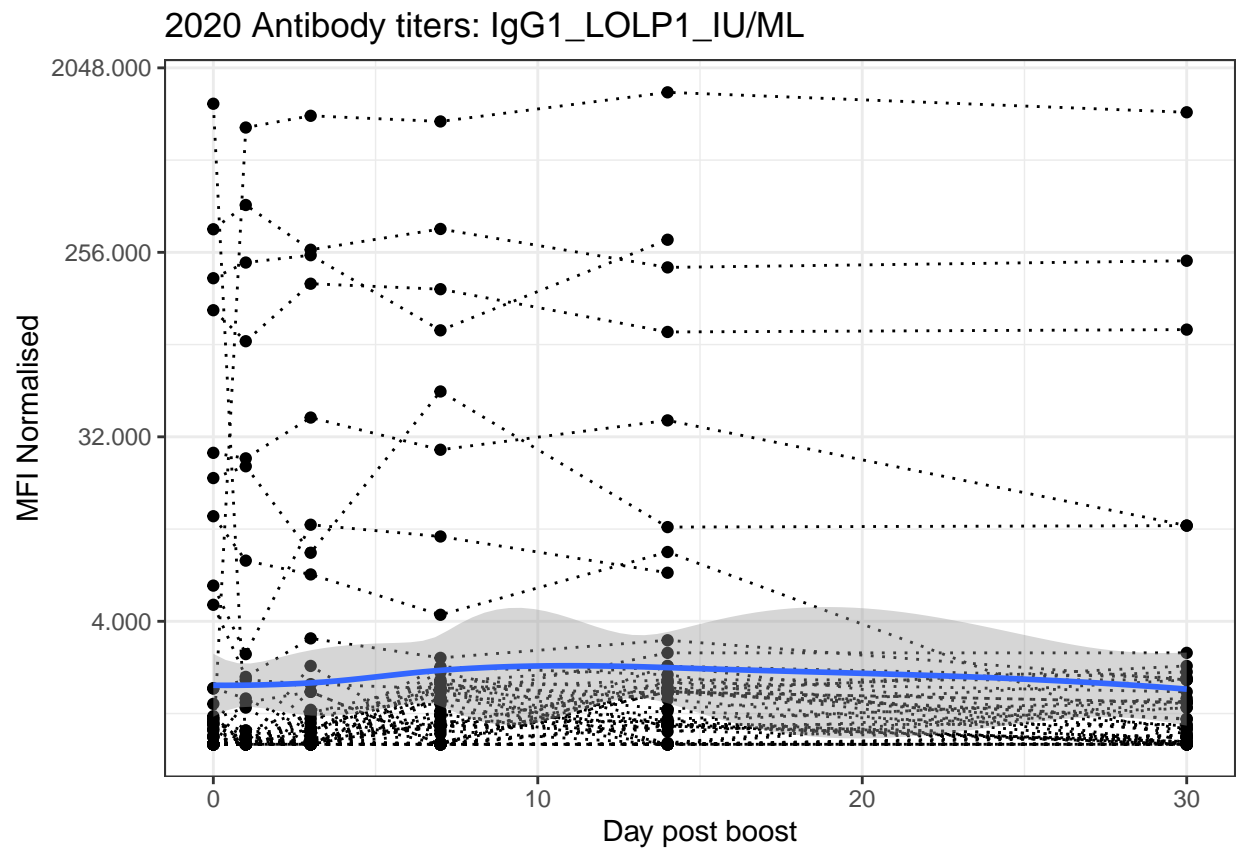


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_BETV1_IU/ML

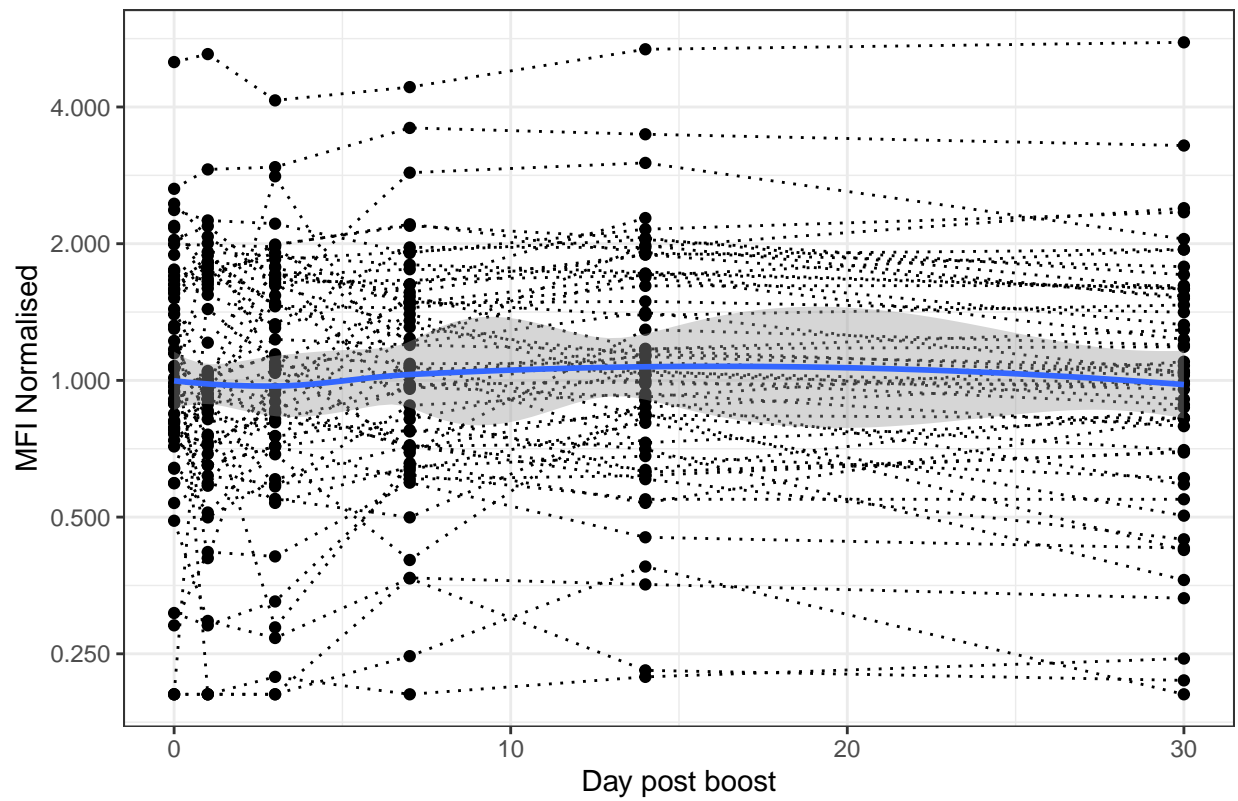


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



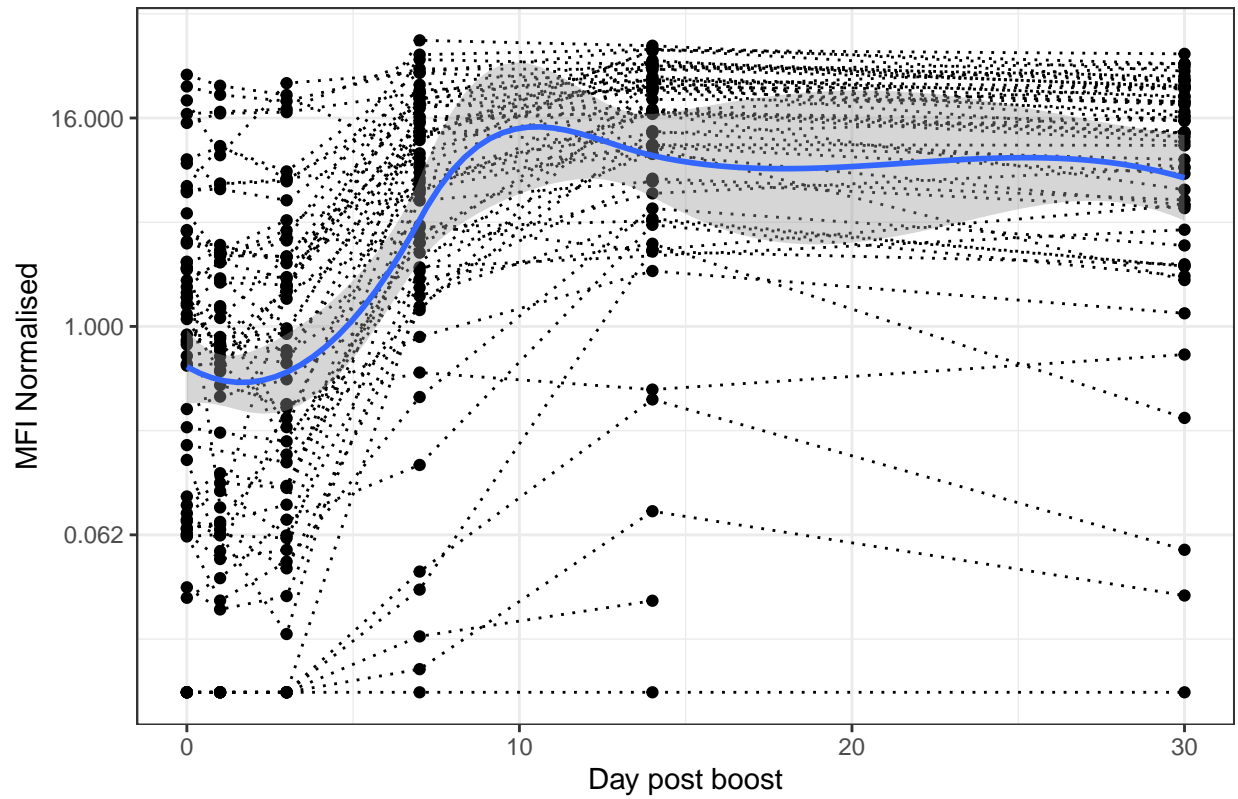
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_Measles_IU/ML



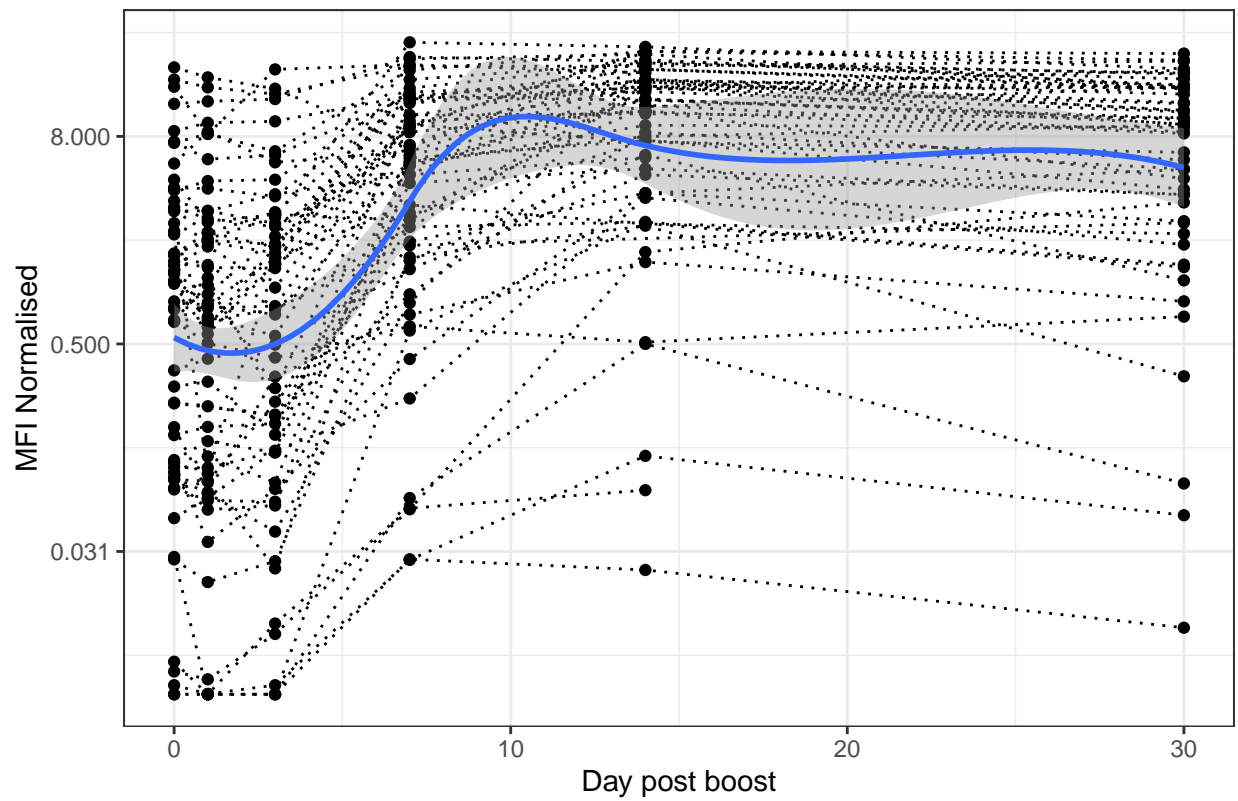
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgG1_PTM_IU/ML



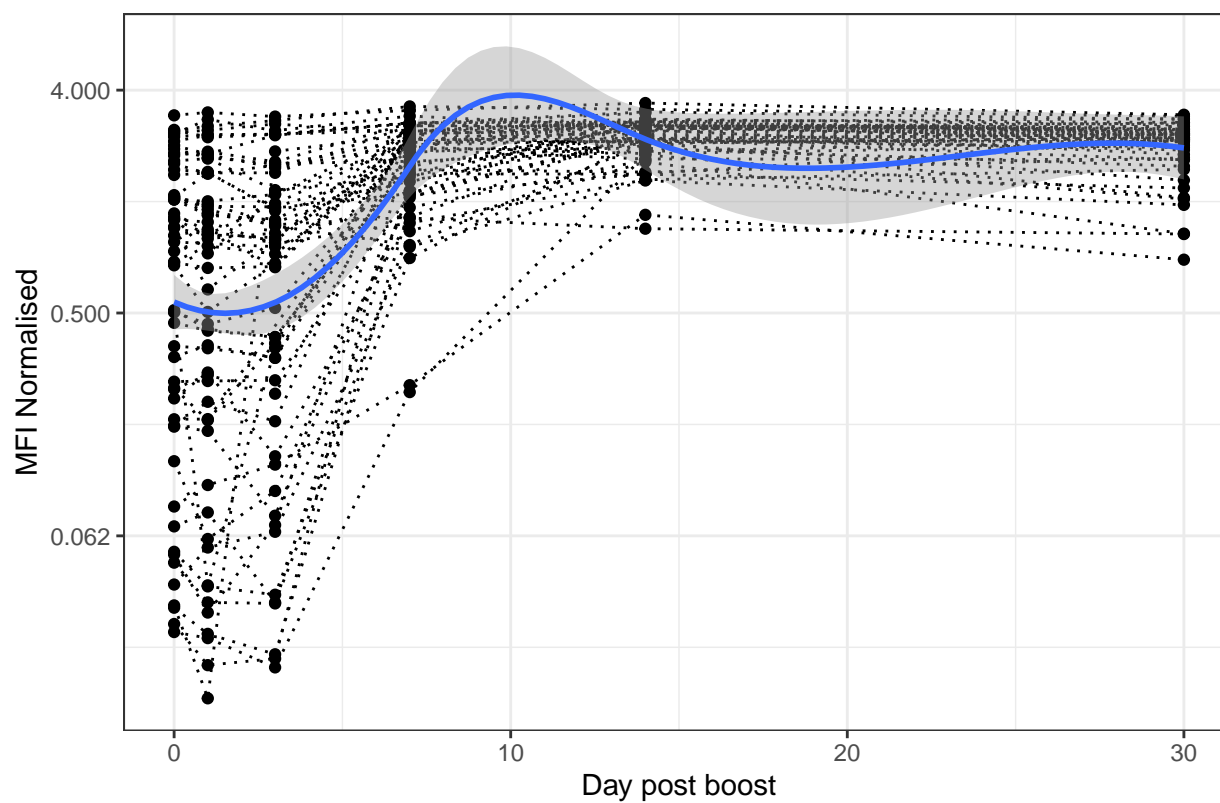
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_PT_IU/ML



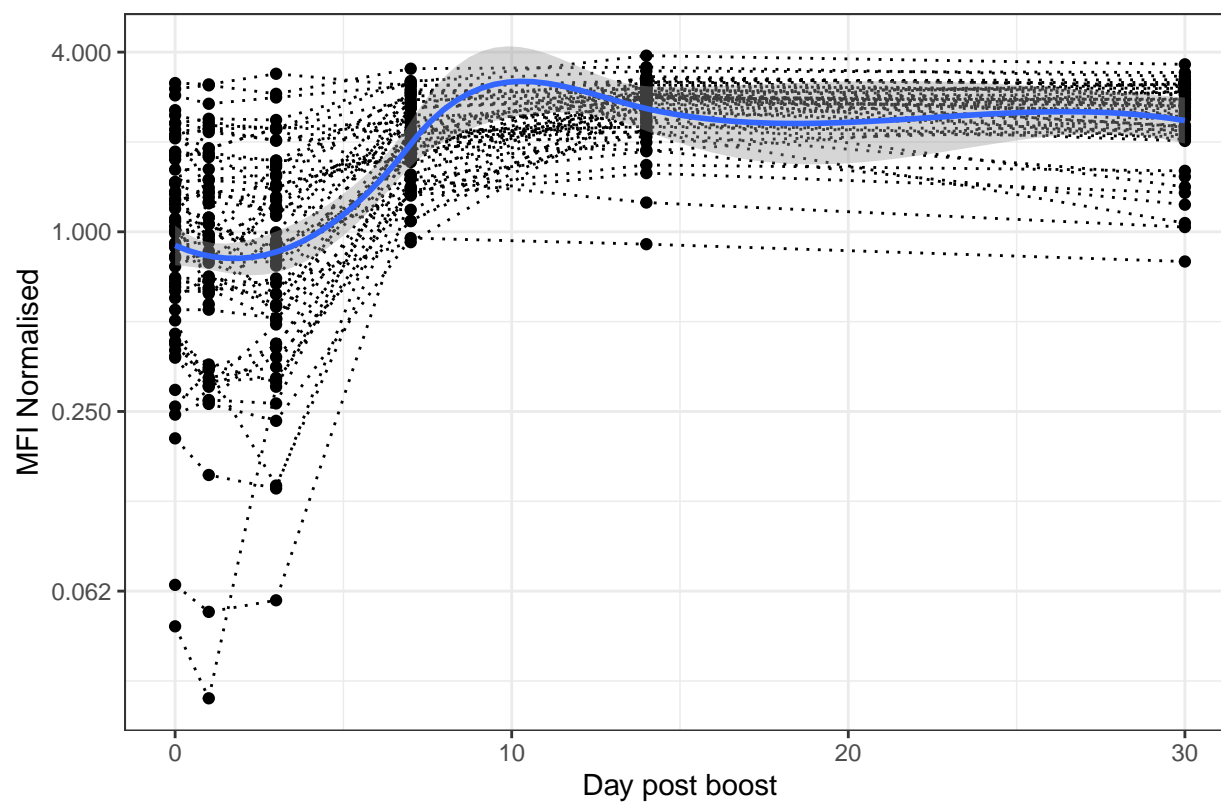
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_PRN_IU/ML



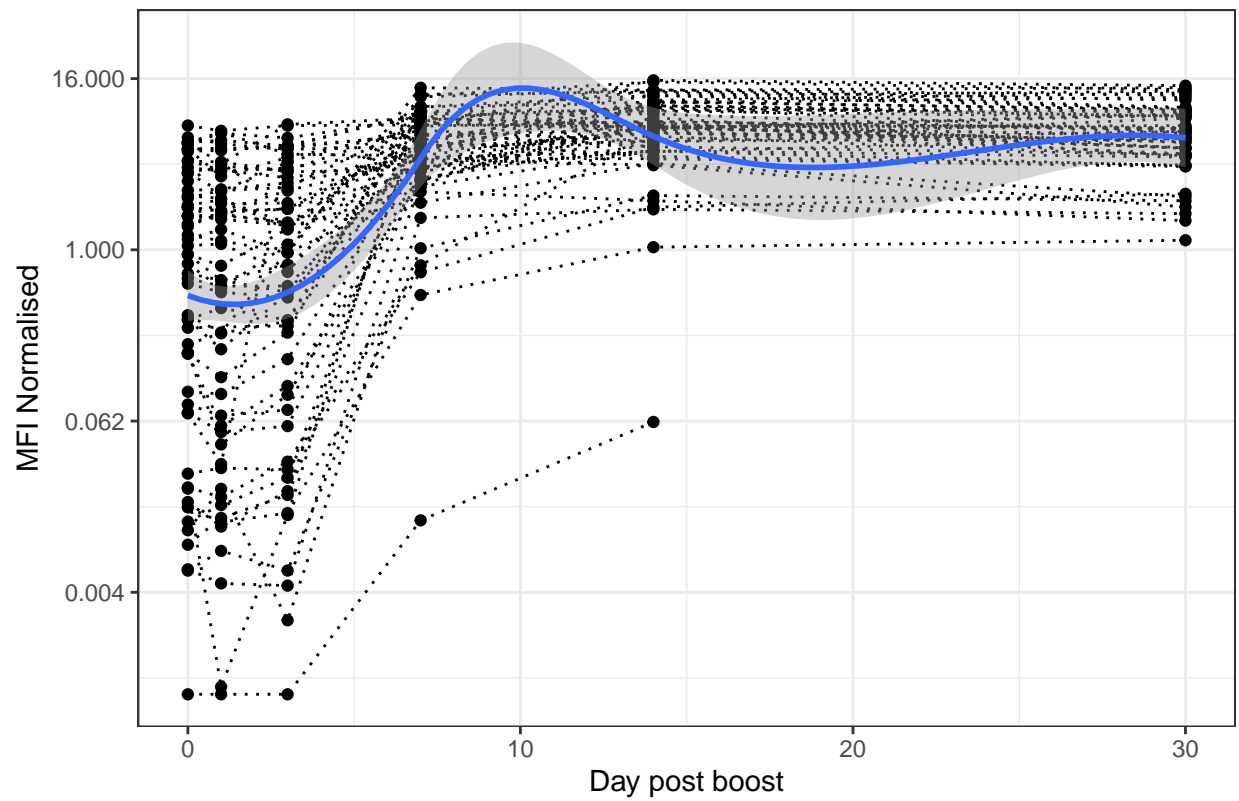
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_FHA_IU/ML



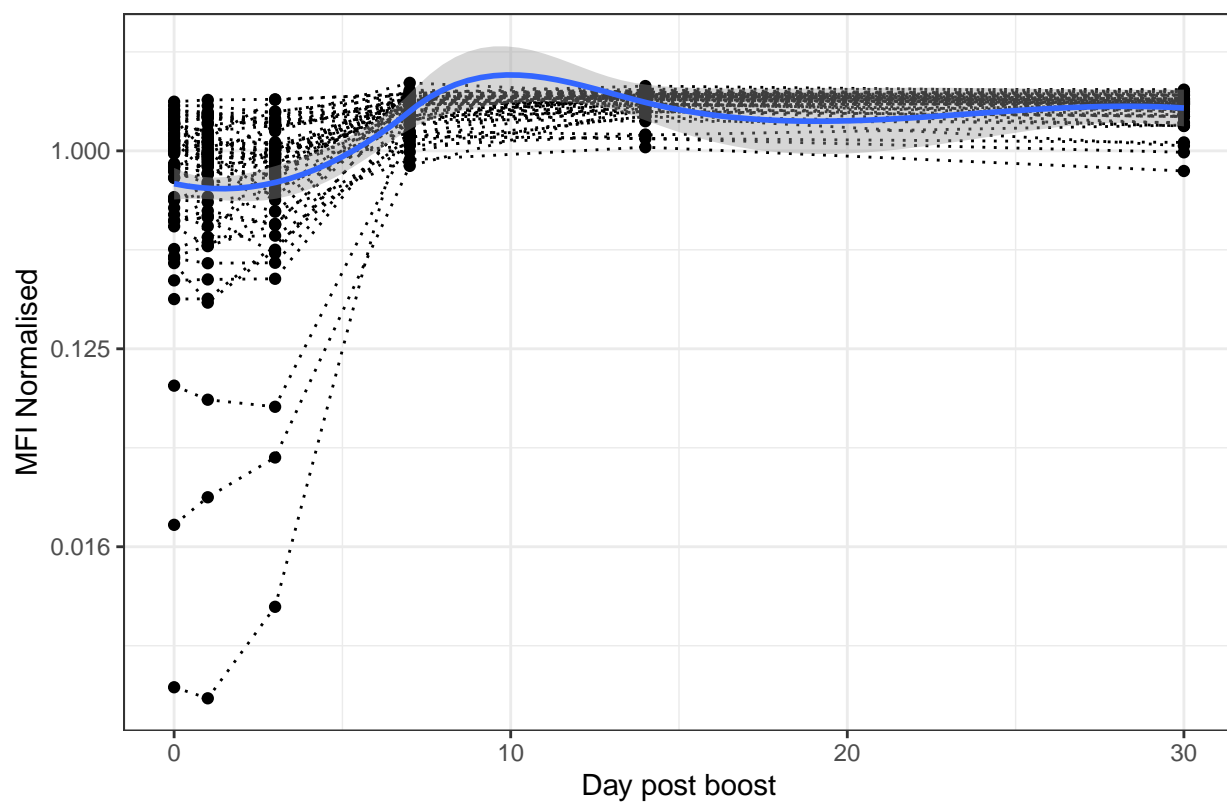
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_FIM2/3_IU/ML



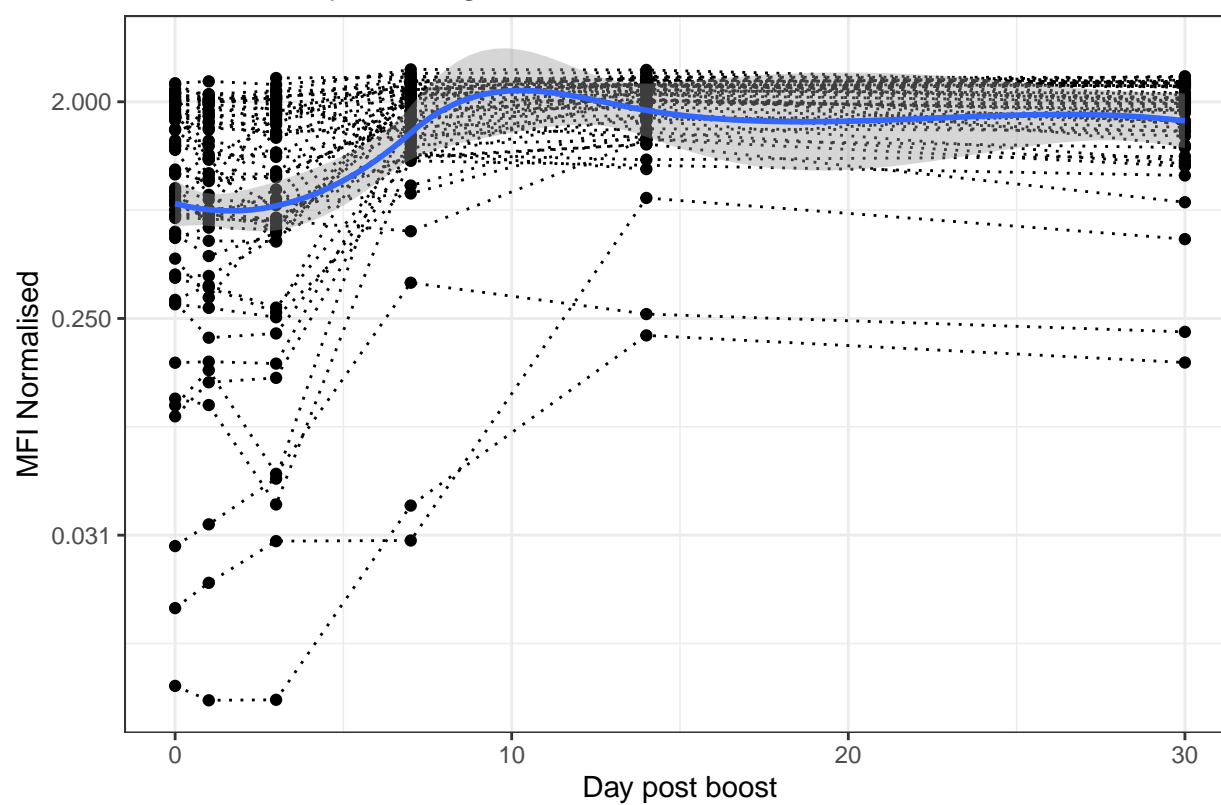
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_TT_IU/ML



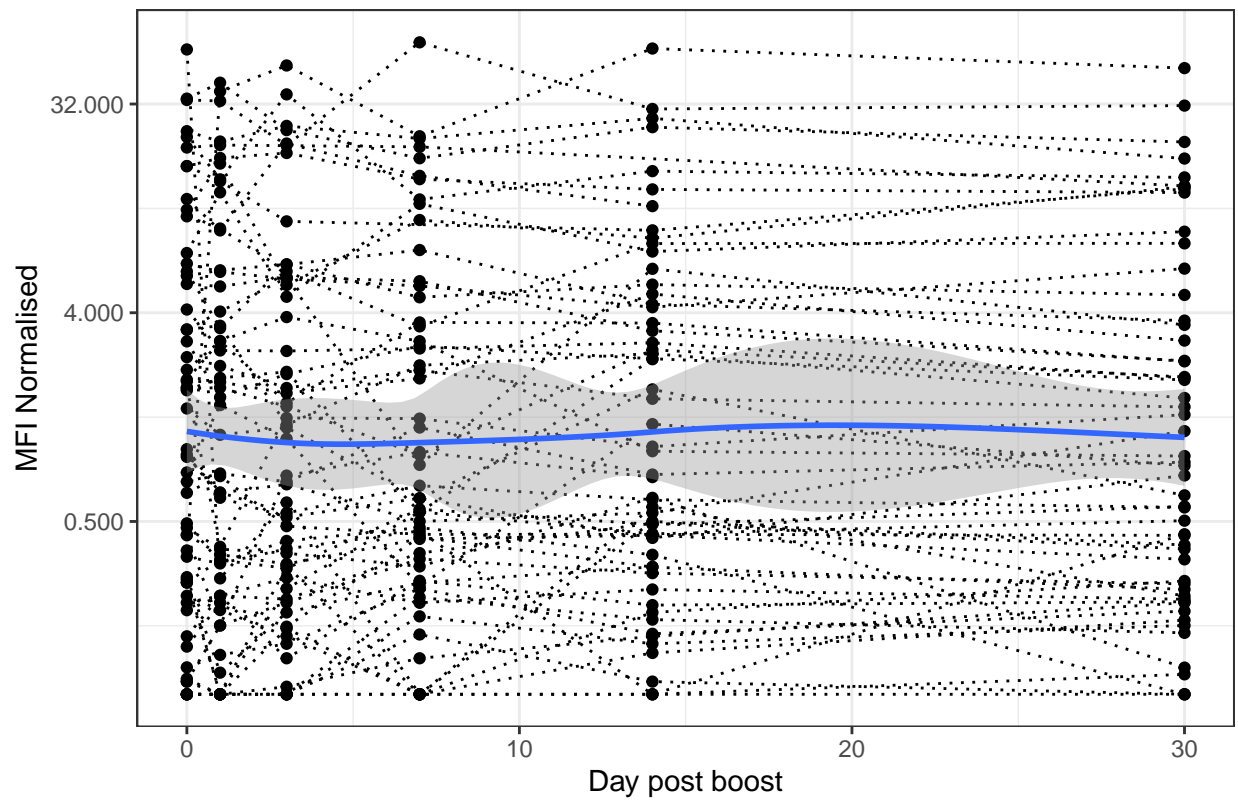
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_DT_IU/ML

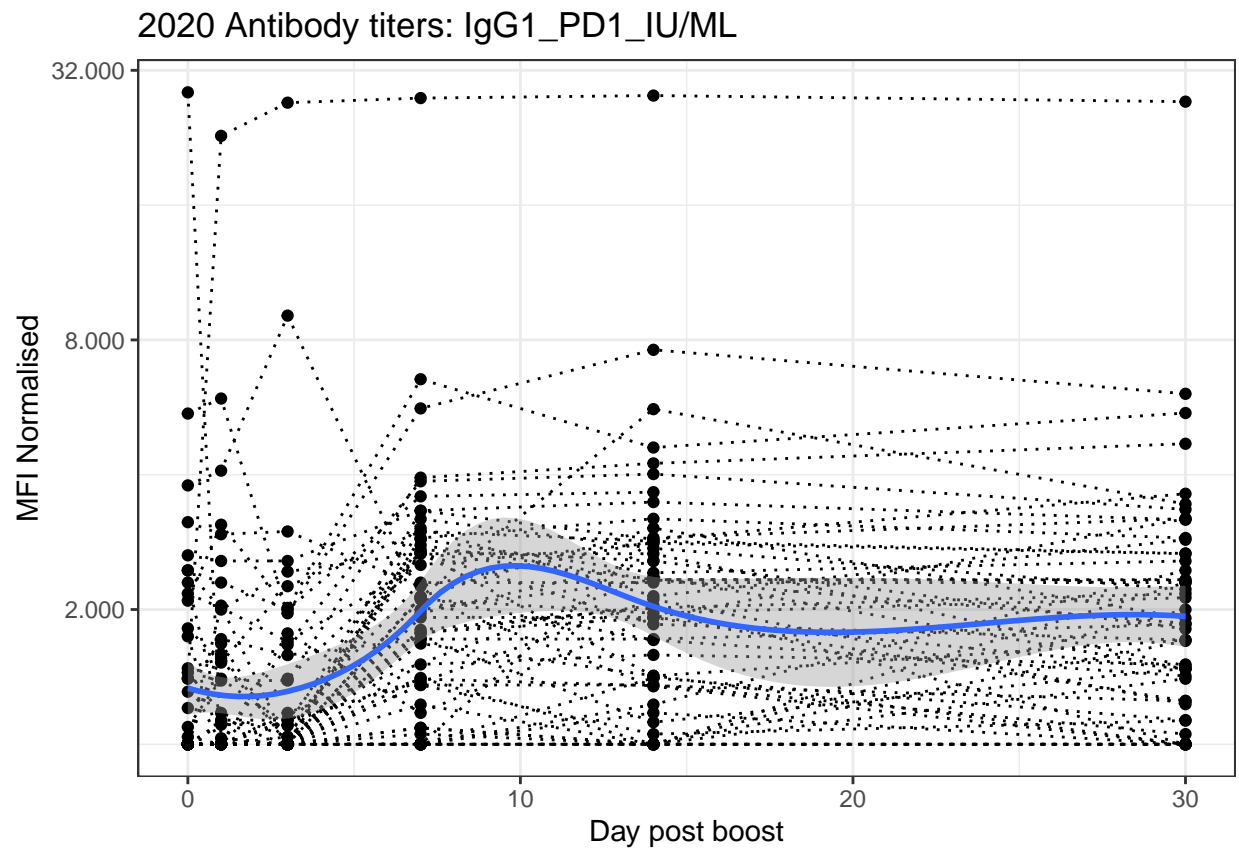


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG1_OVA_IU/ML

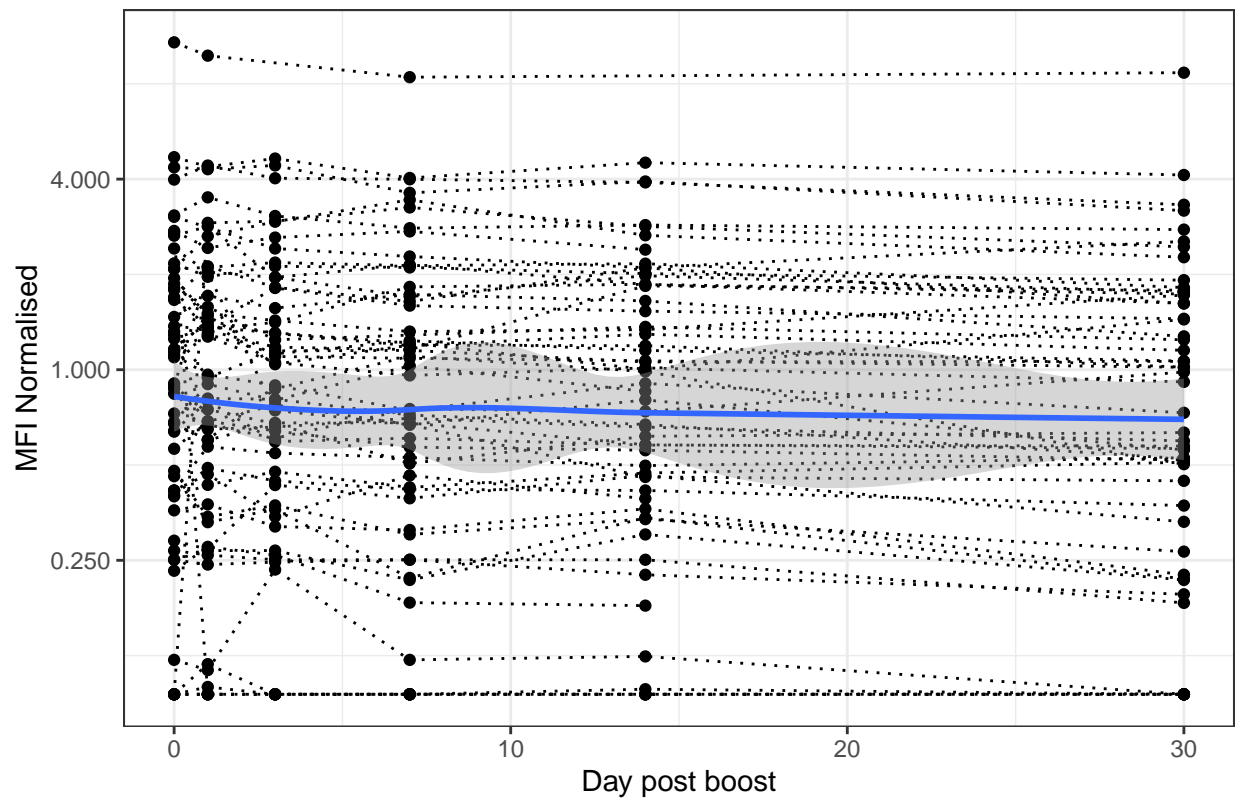


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

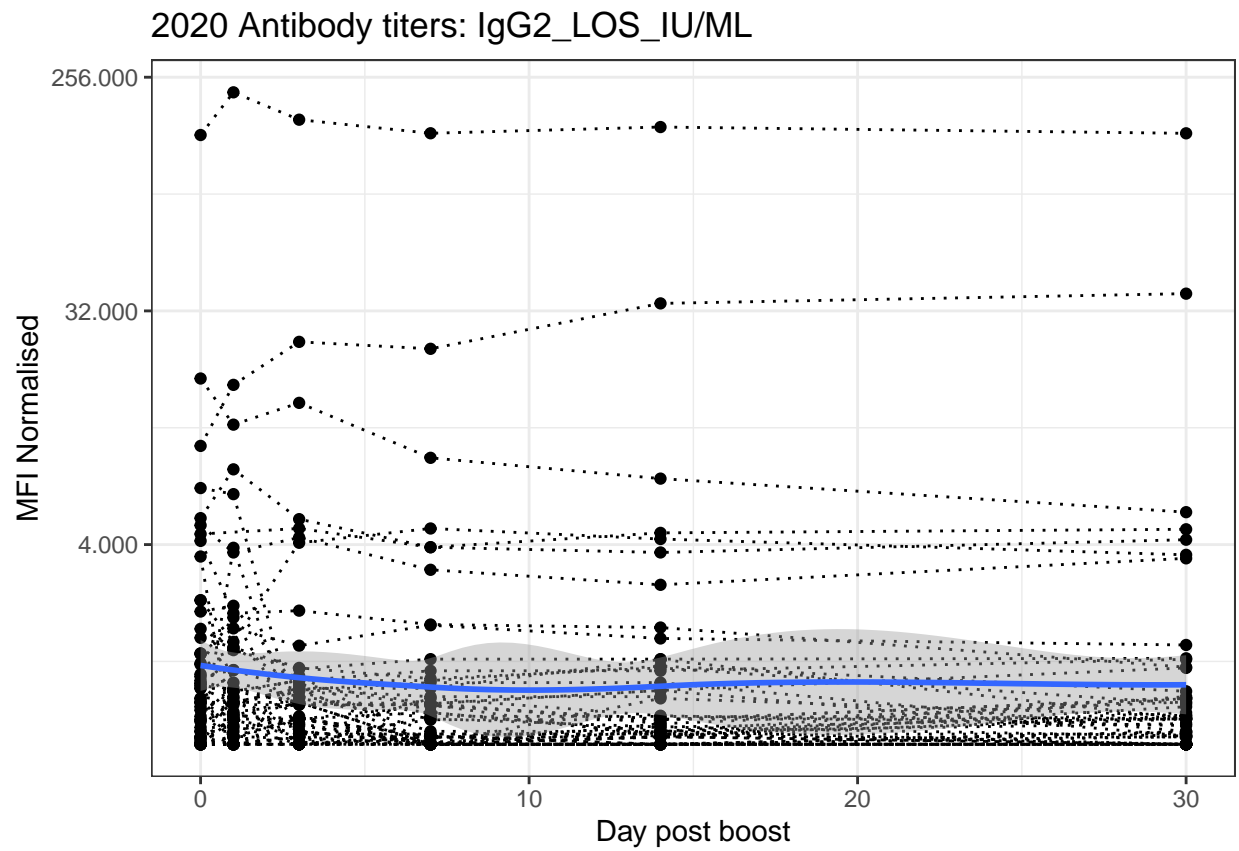



```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

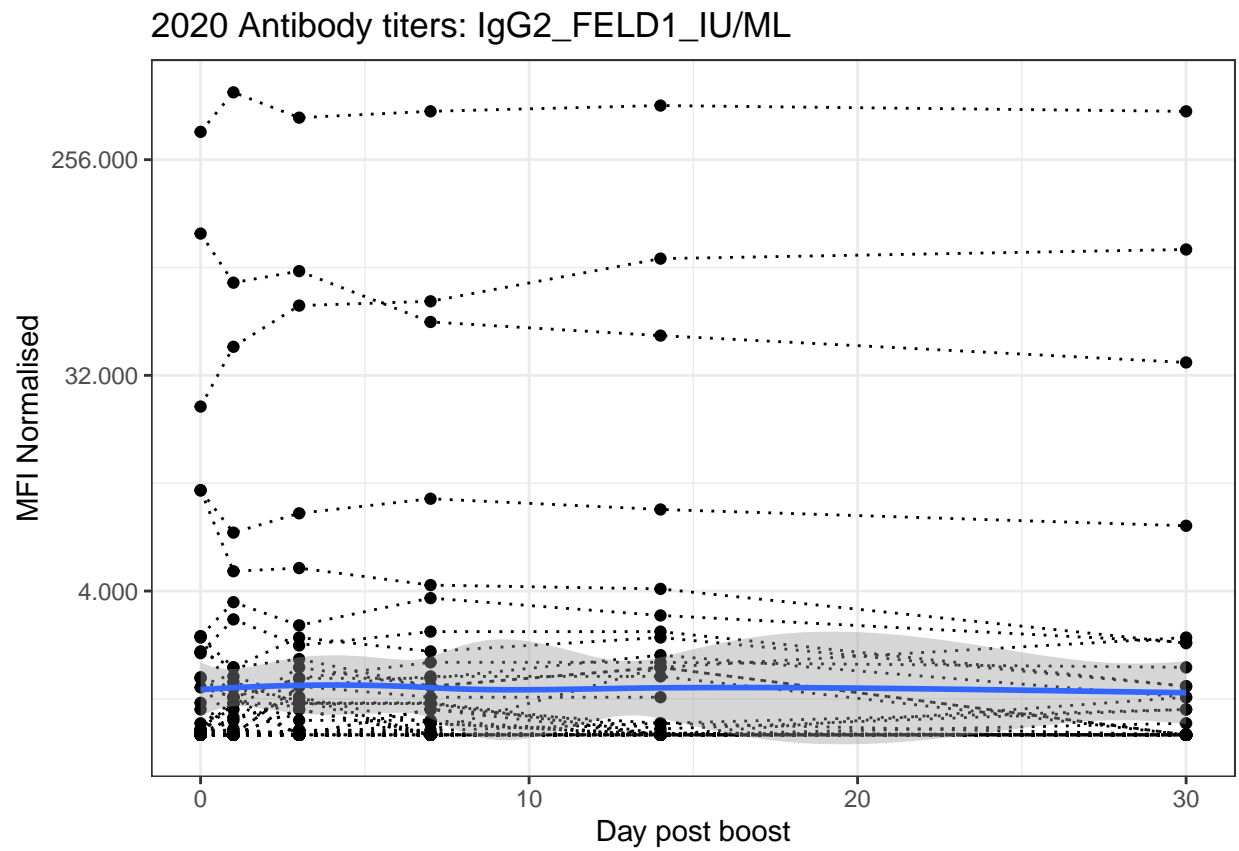
2020 Antibody titers: IgG2_ACT_IU/ML



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

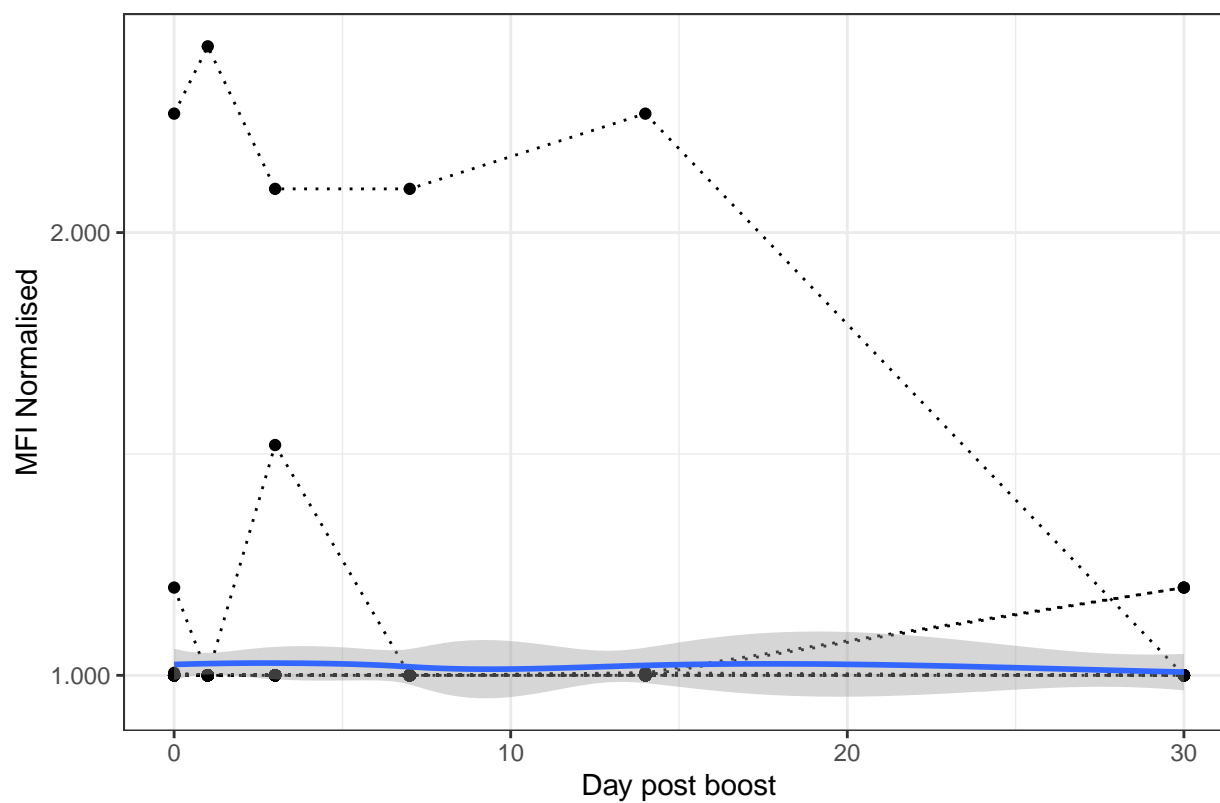


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

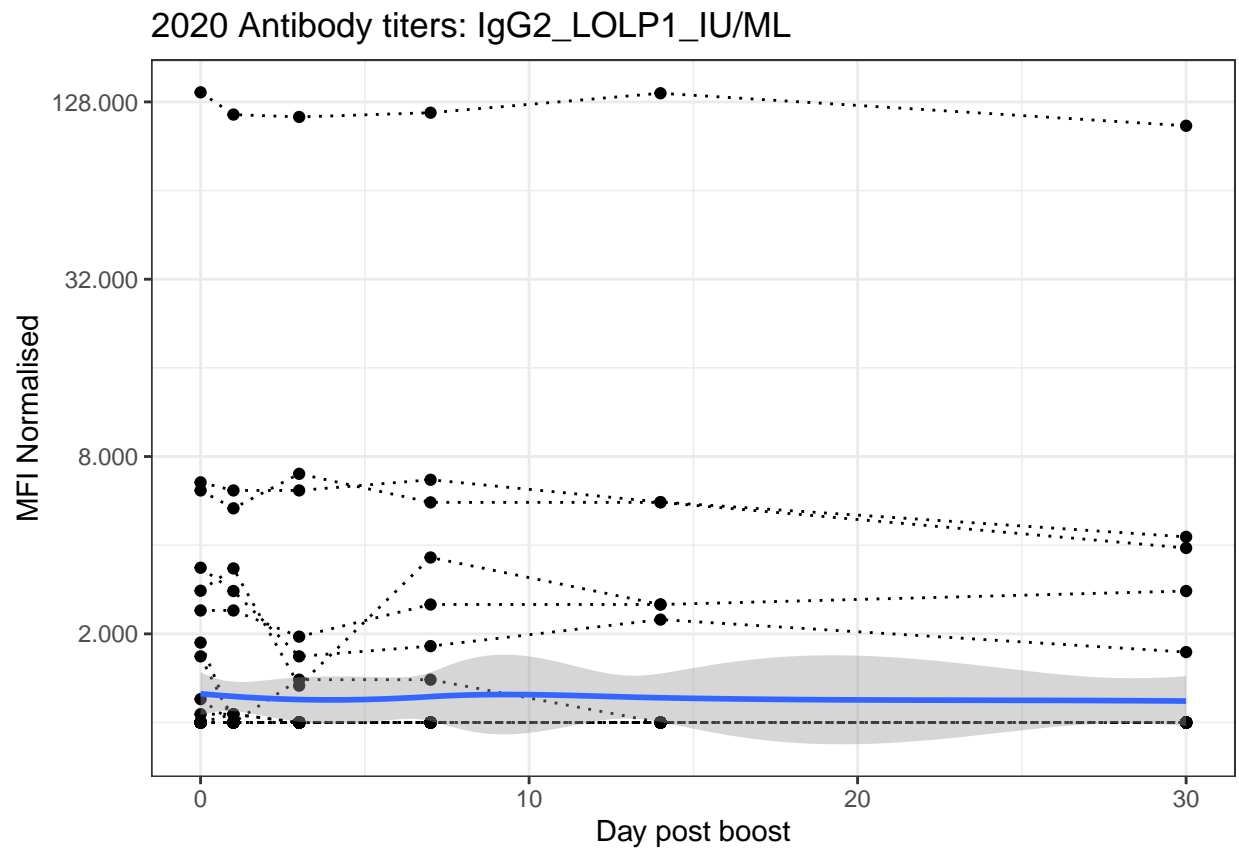


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_BETV1_IU/ML

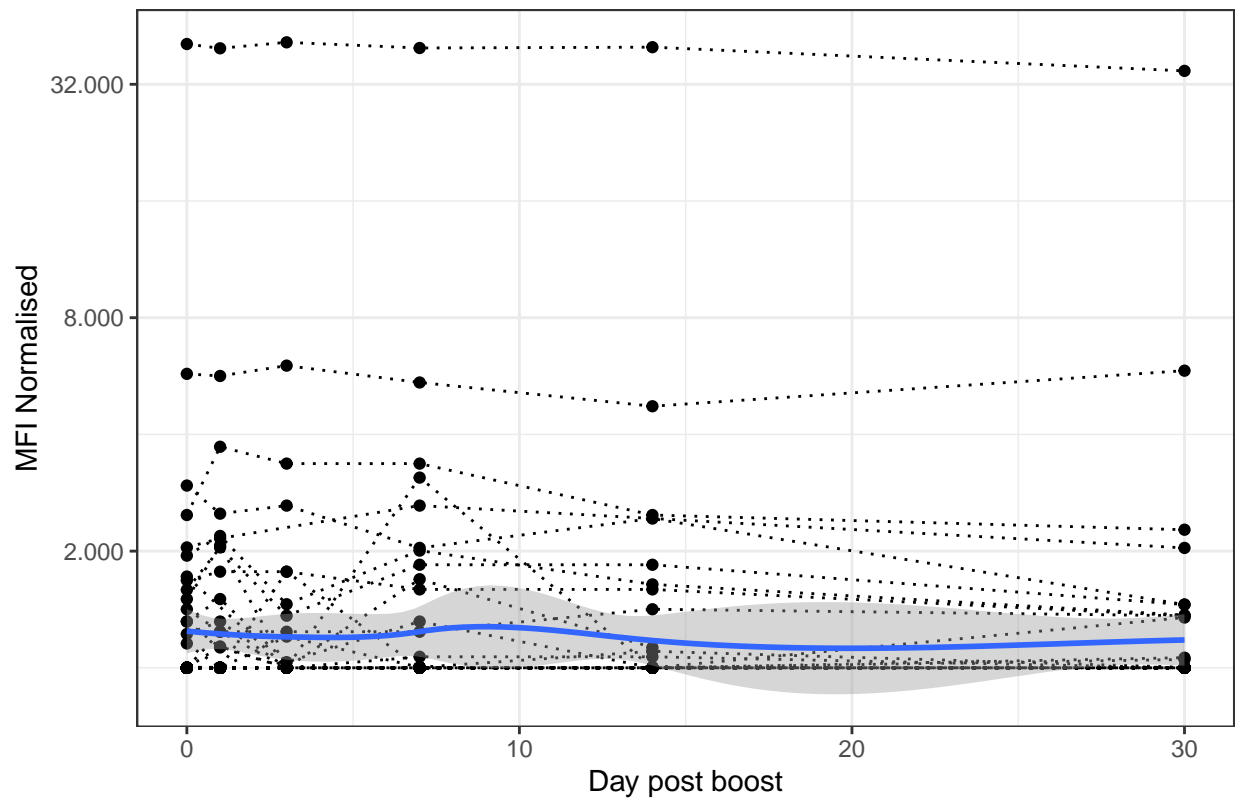


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

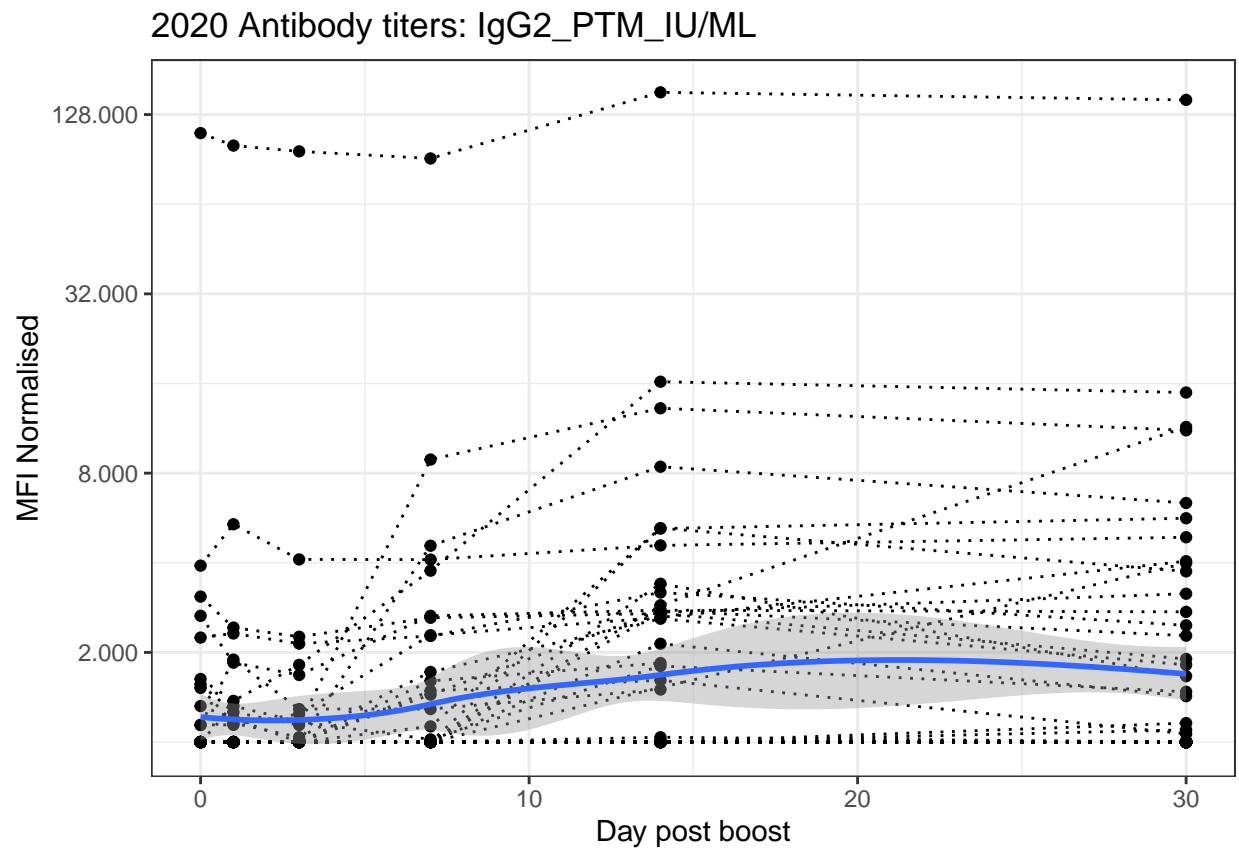


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

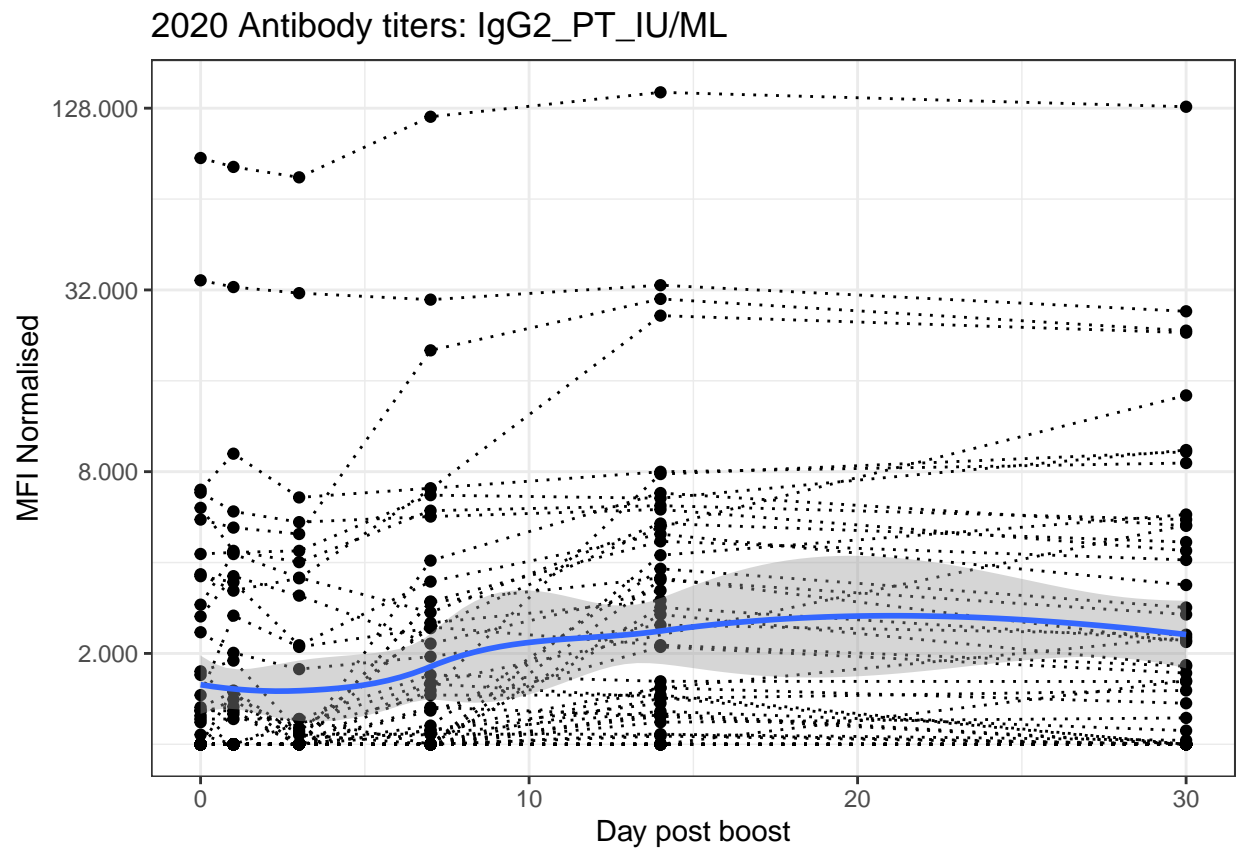
2020 Antibody titers: IgG2_Measles_IU/ML



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

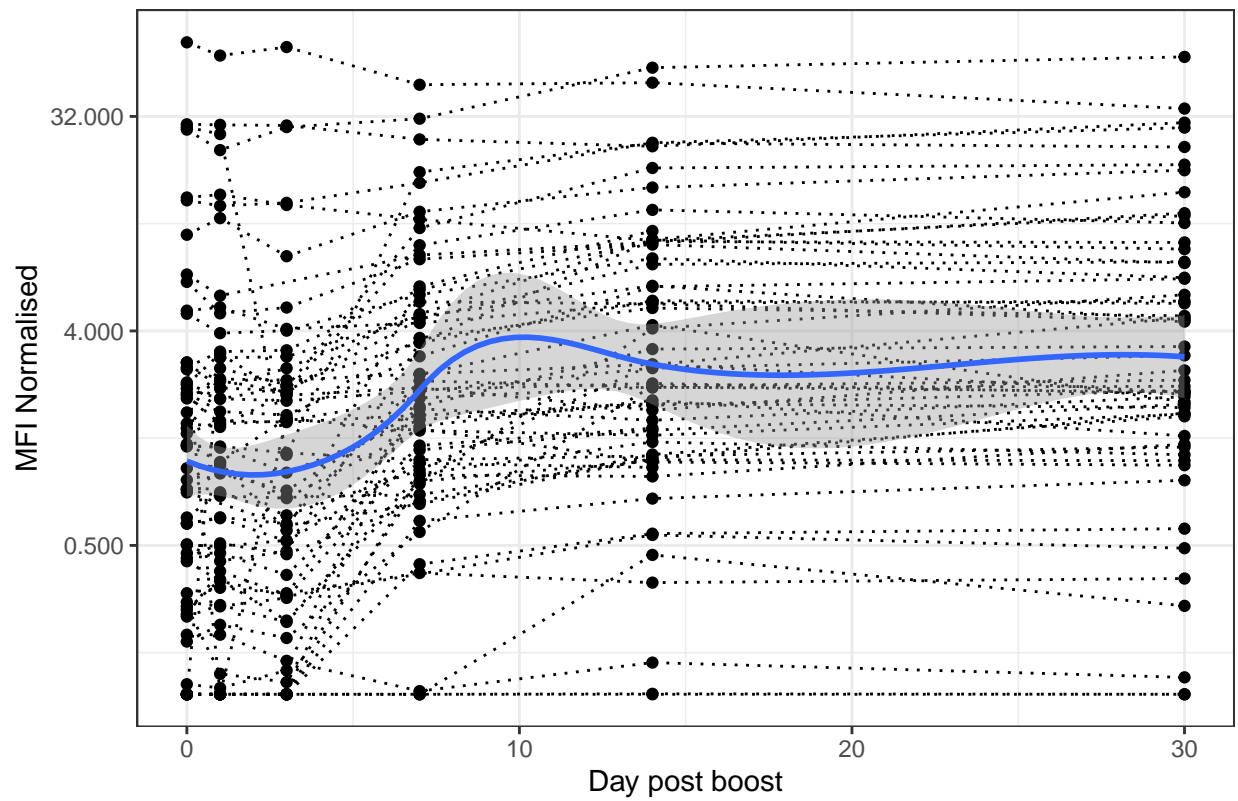


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

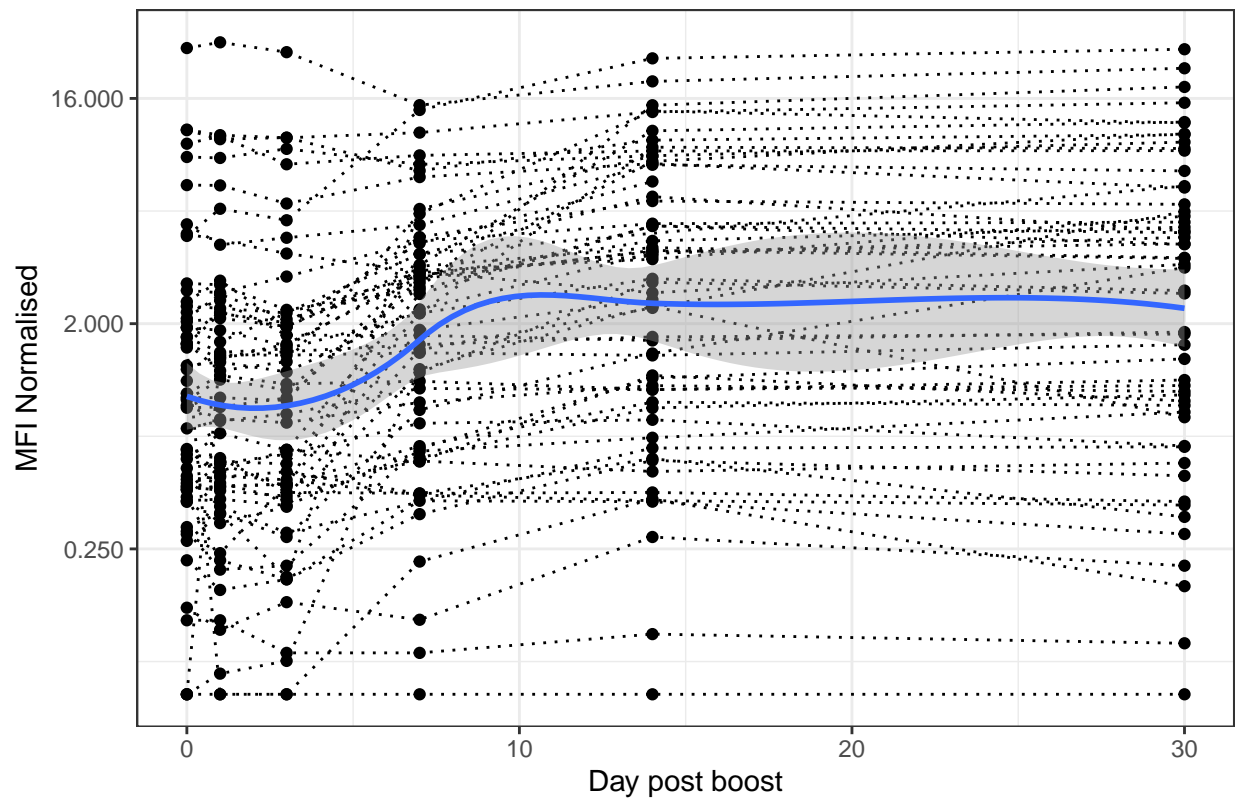
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_PRN_IU/ML

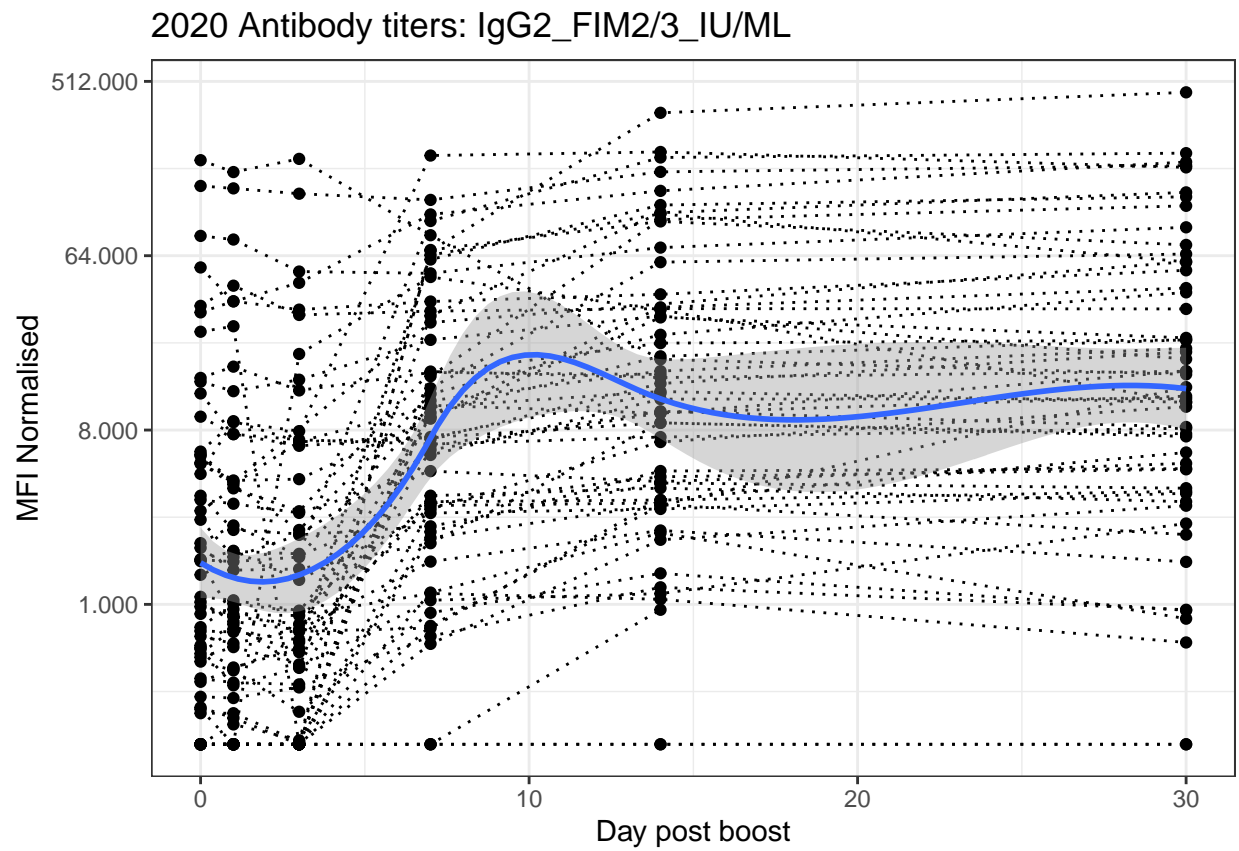


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_FHA_IU/ML

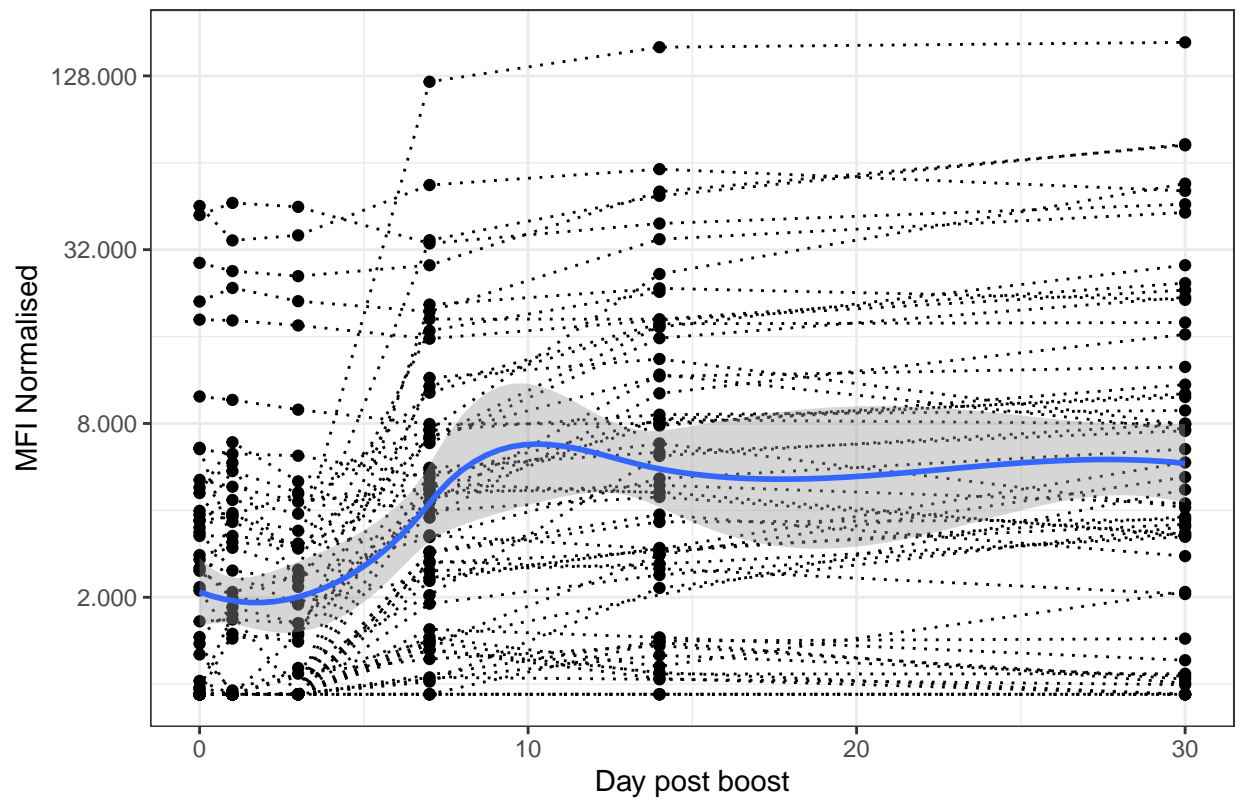


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



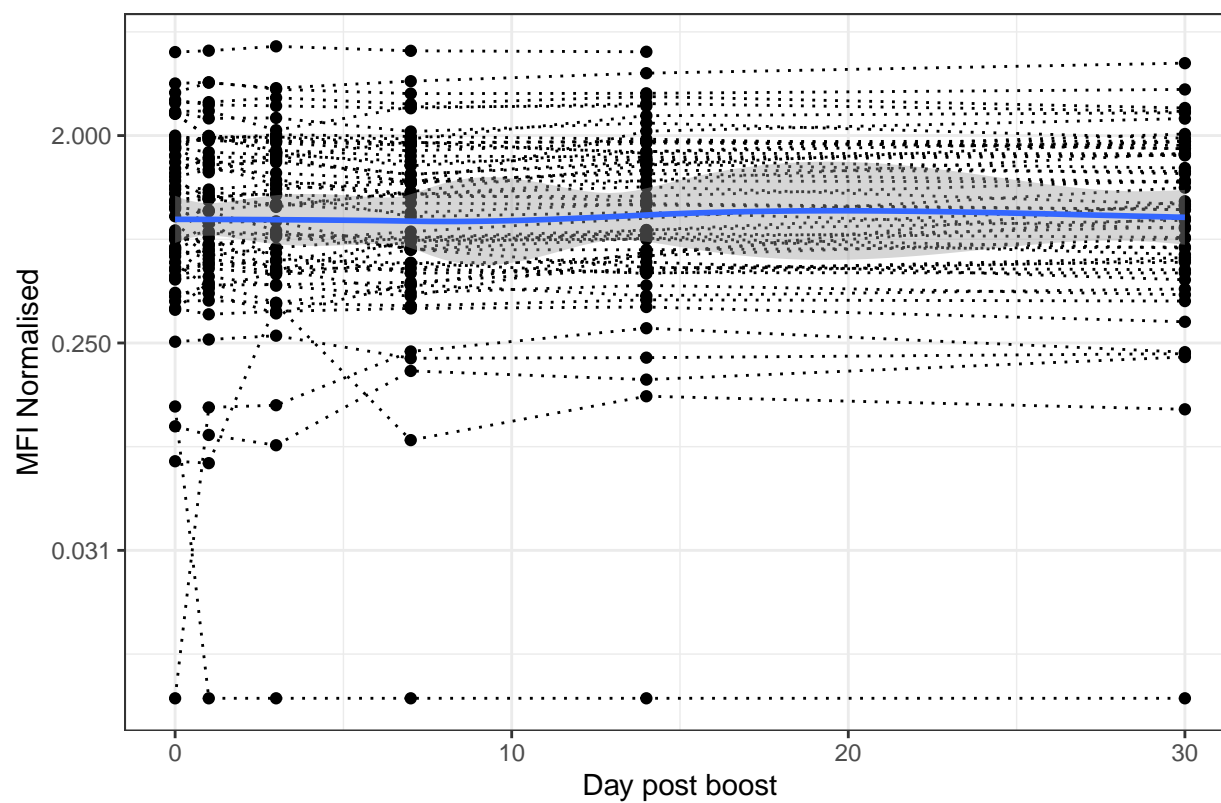
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_TT_IU/ML



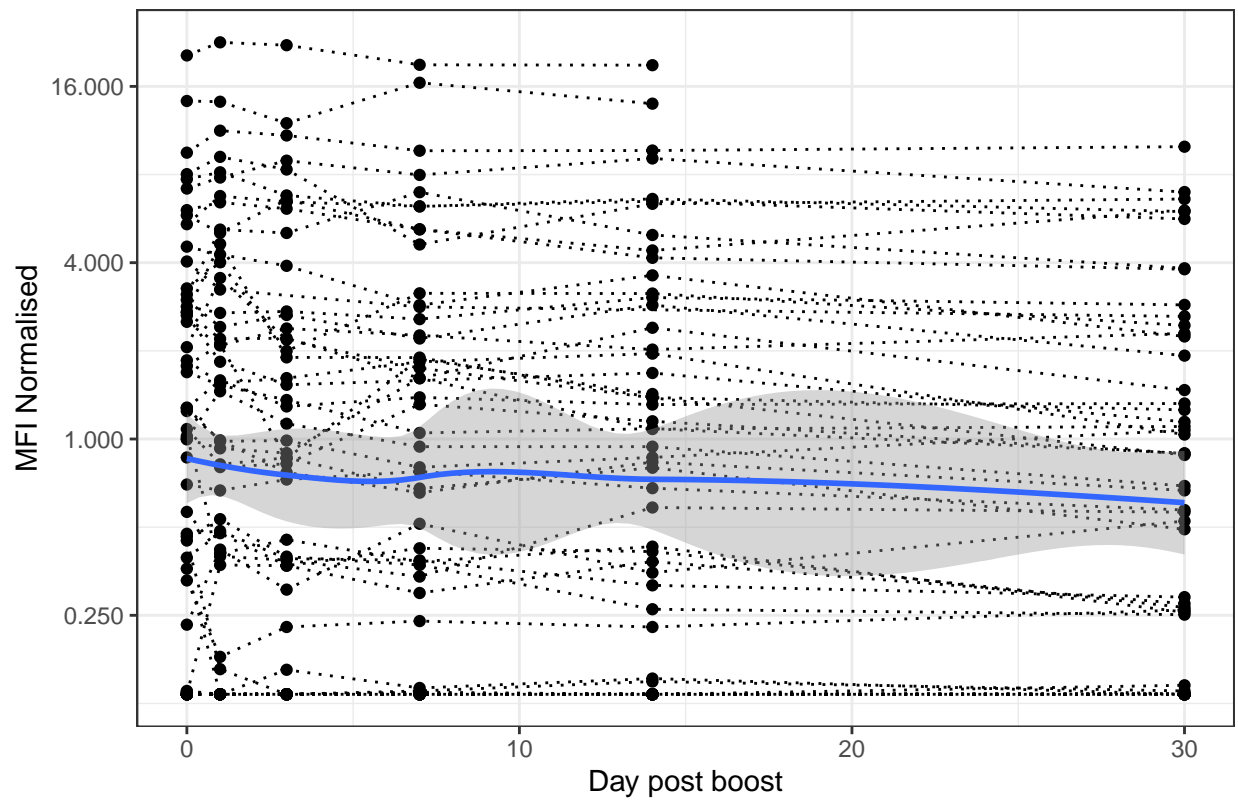
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_DT_IU/ML



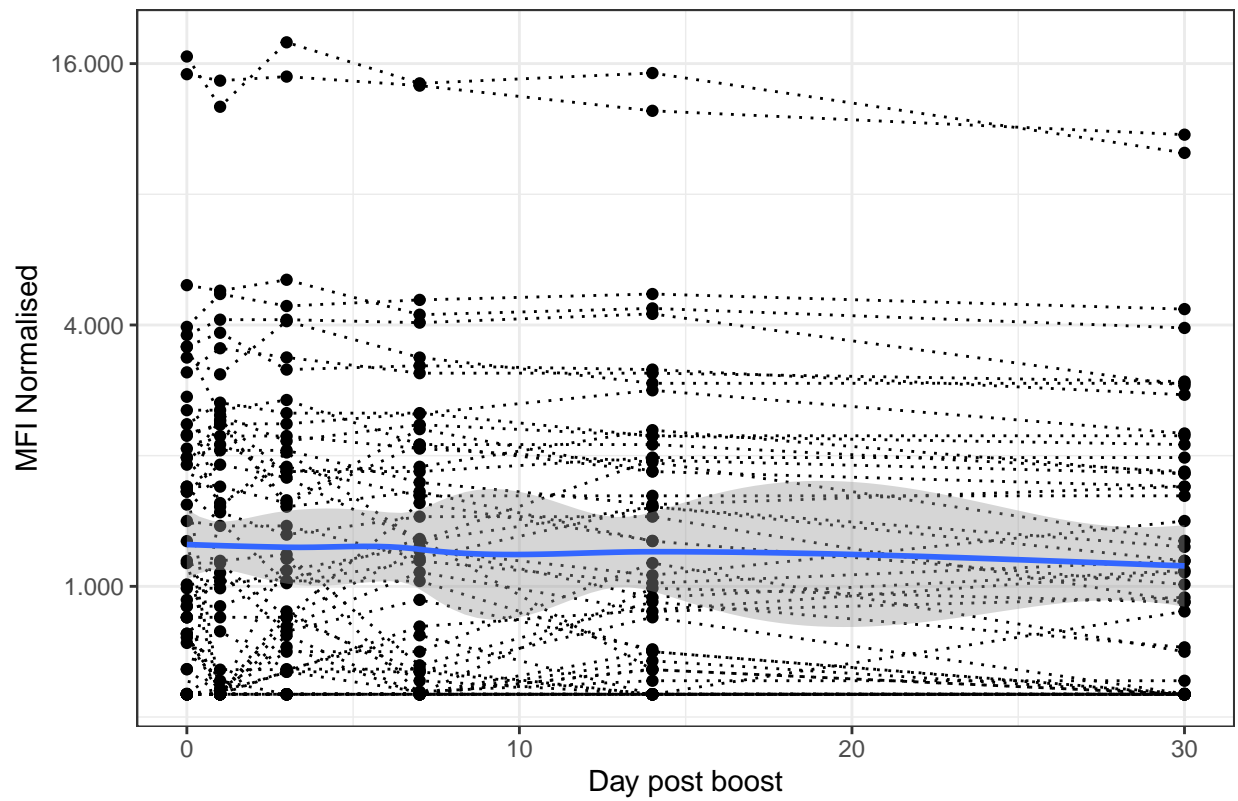
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_OVA_IU/ML



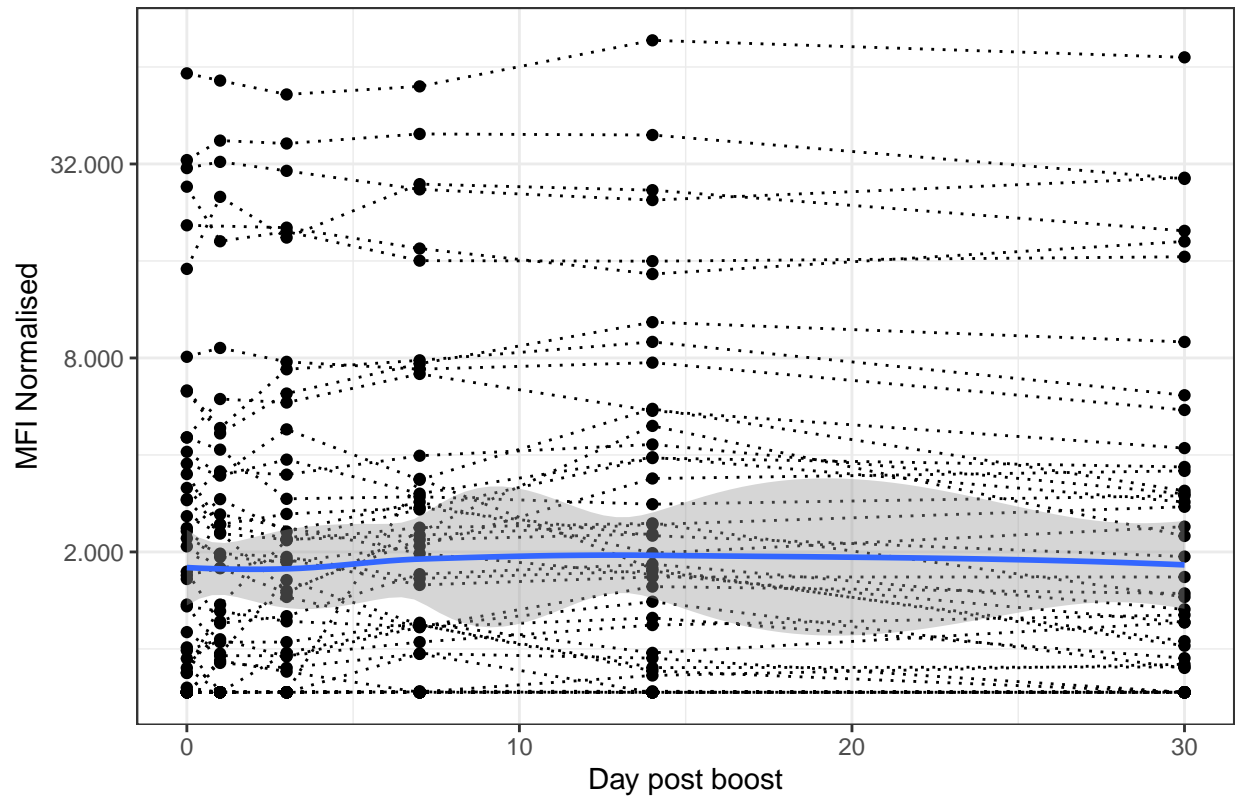
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG2_PD1_IU/ML

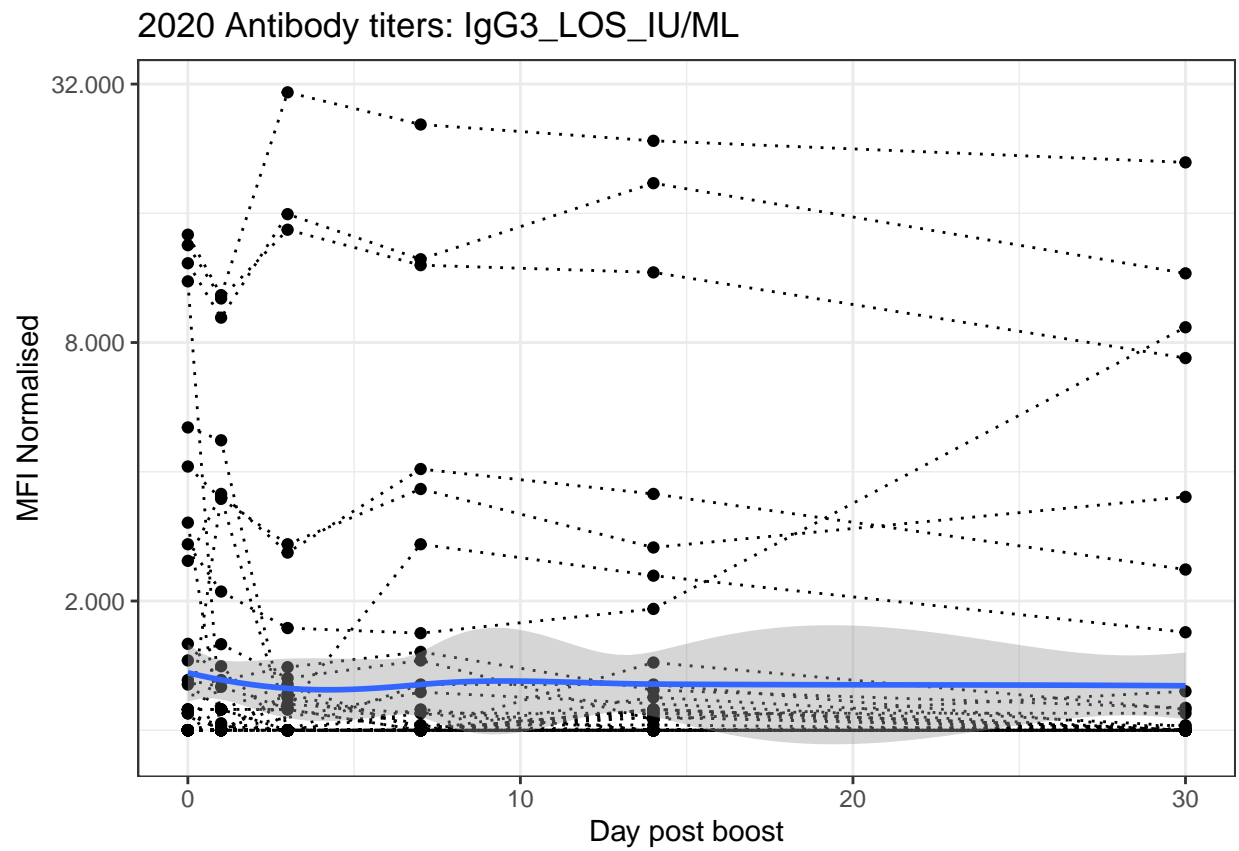


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgG3_ACT_IU/ML

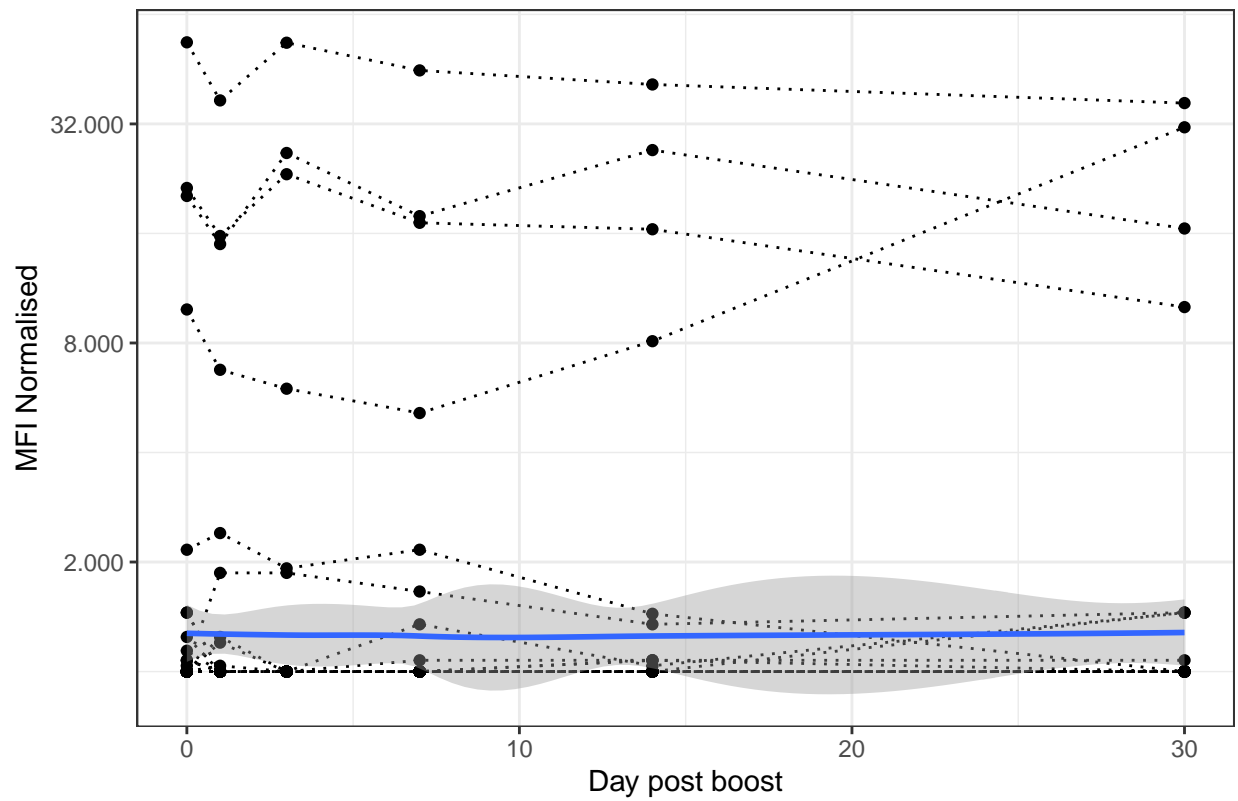


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



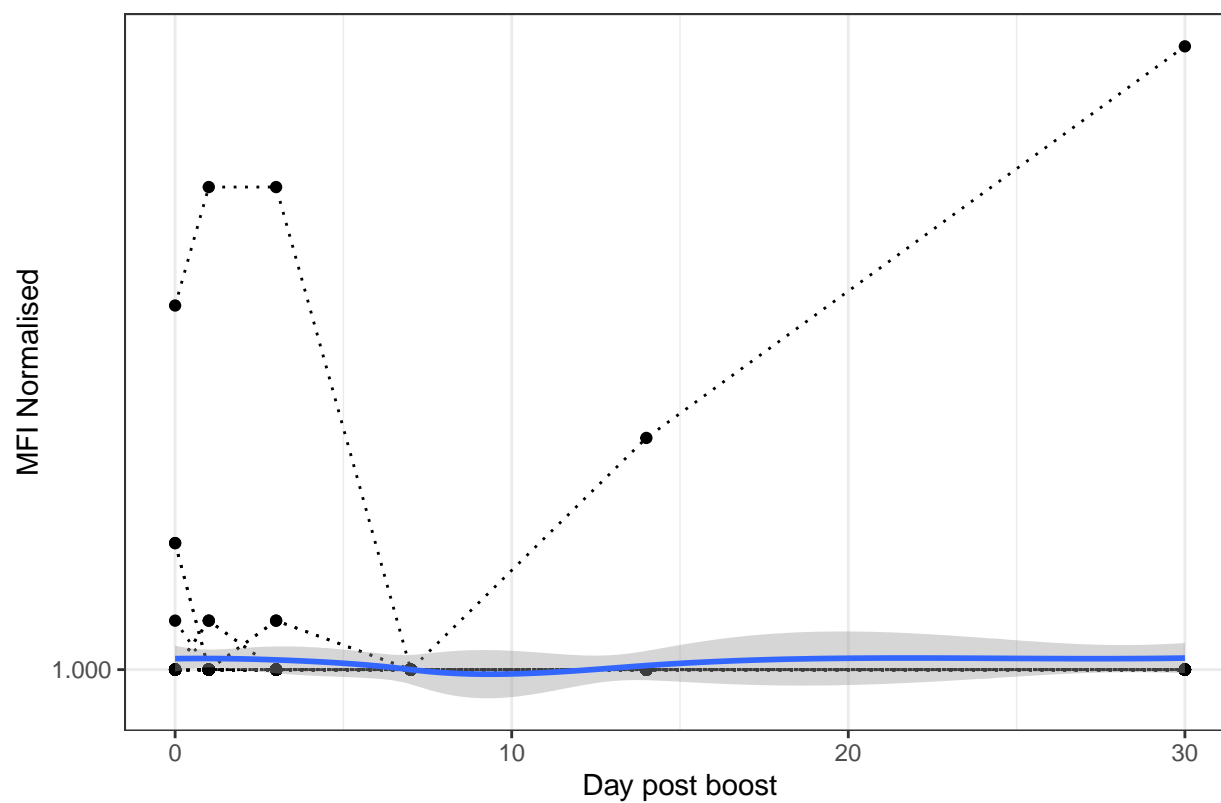
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_FELD1_IU/ML

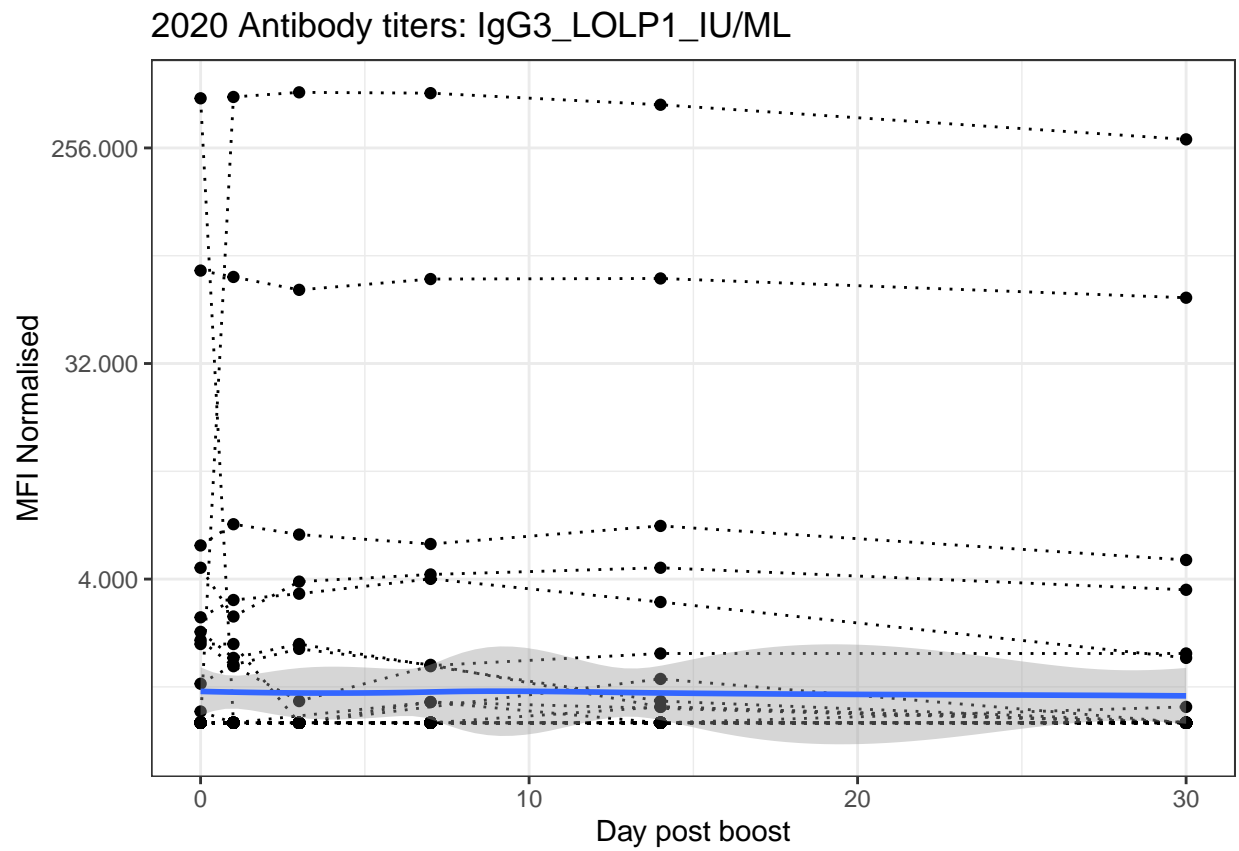


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_BETV1_IU/ML

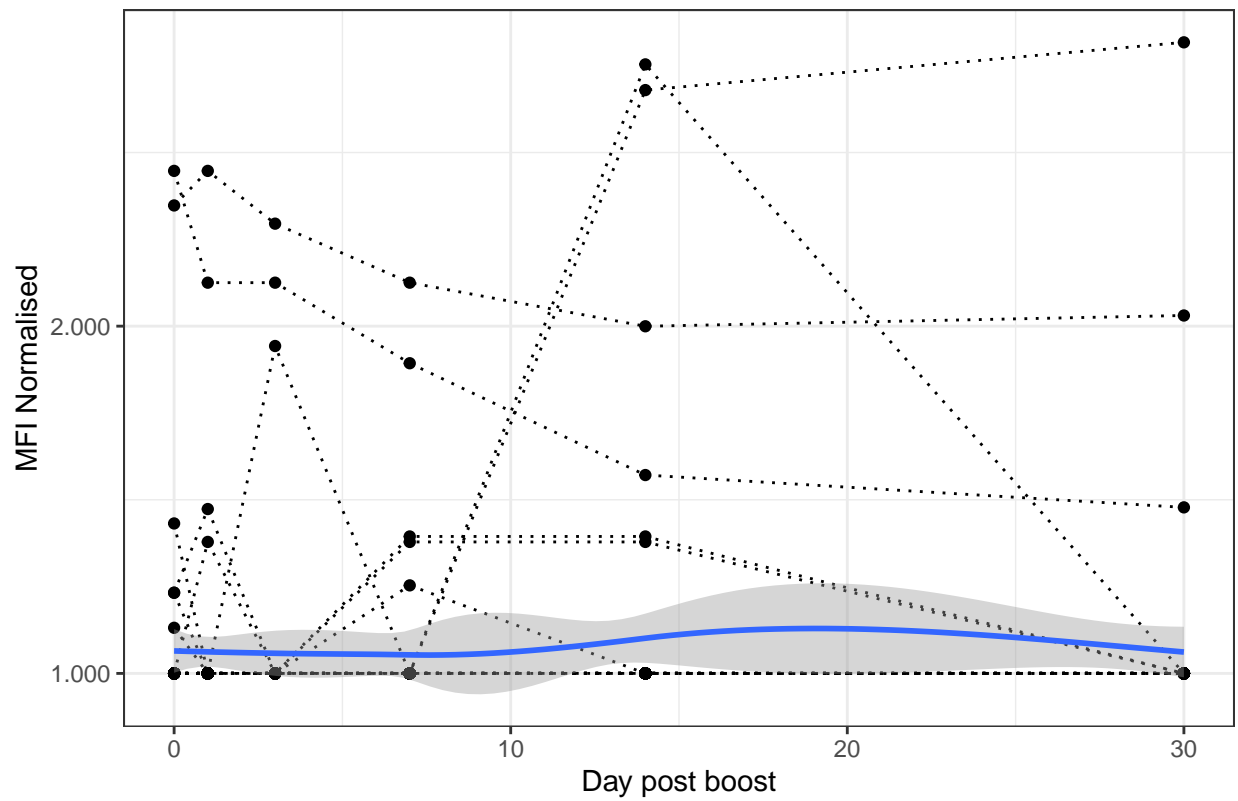


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



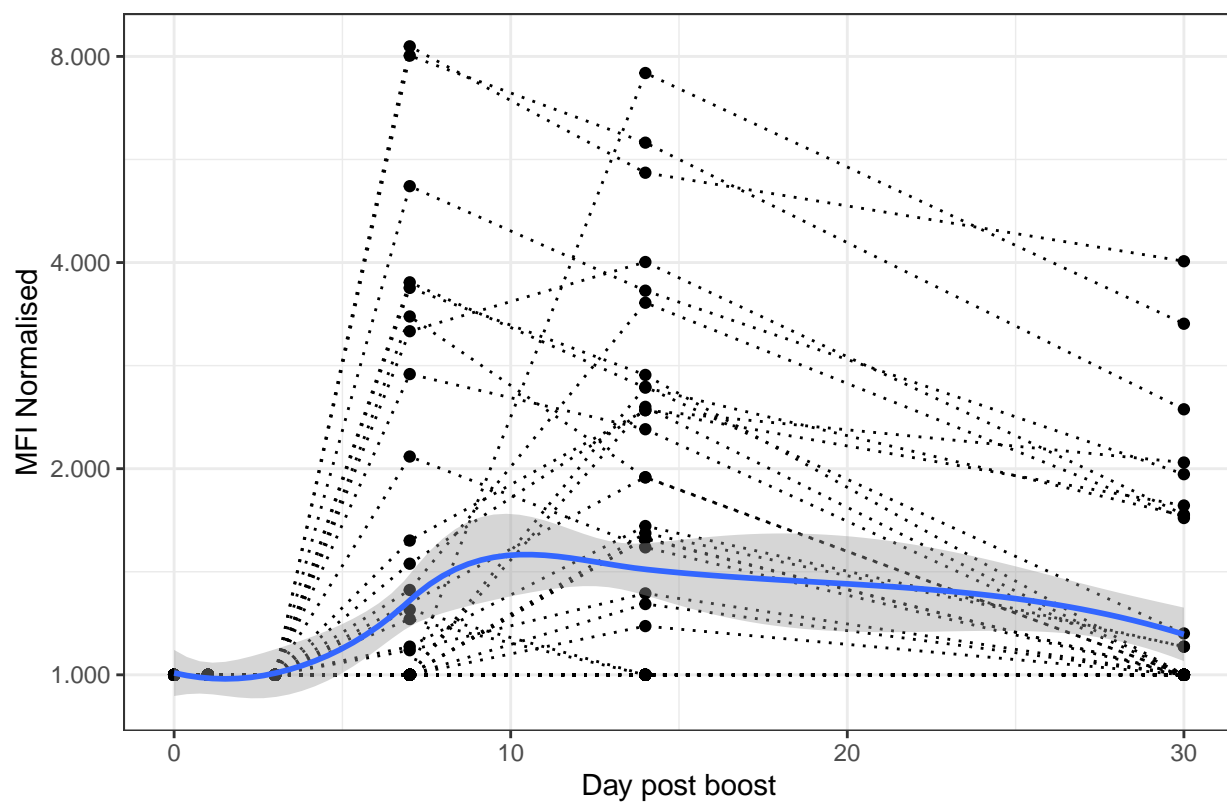
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_Measles_IU/ML

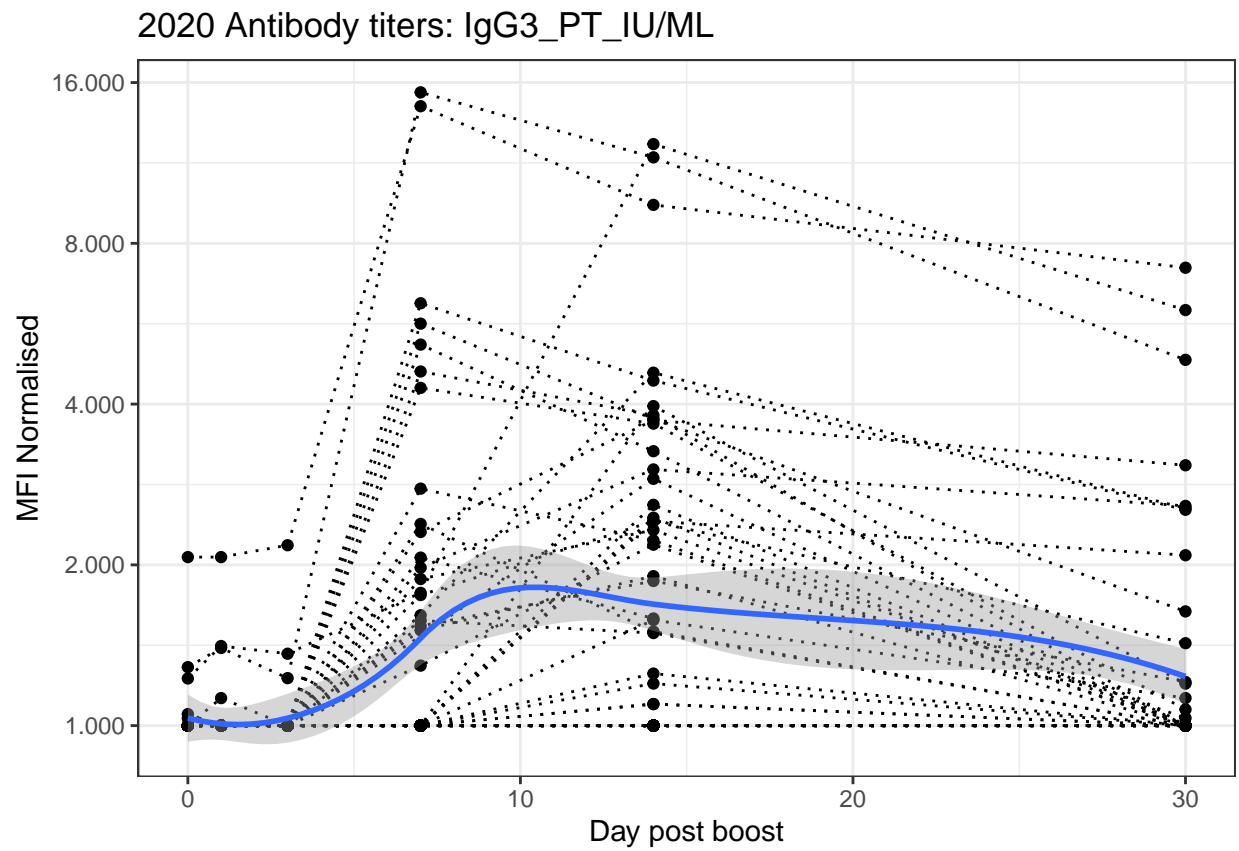


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_PTM_IU/ML

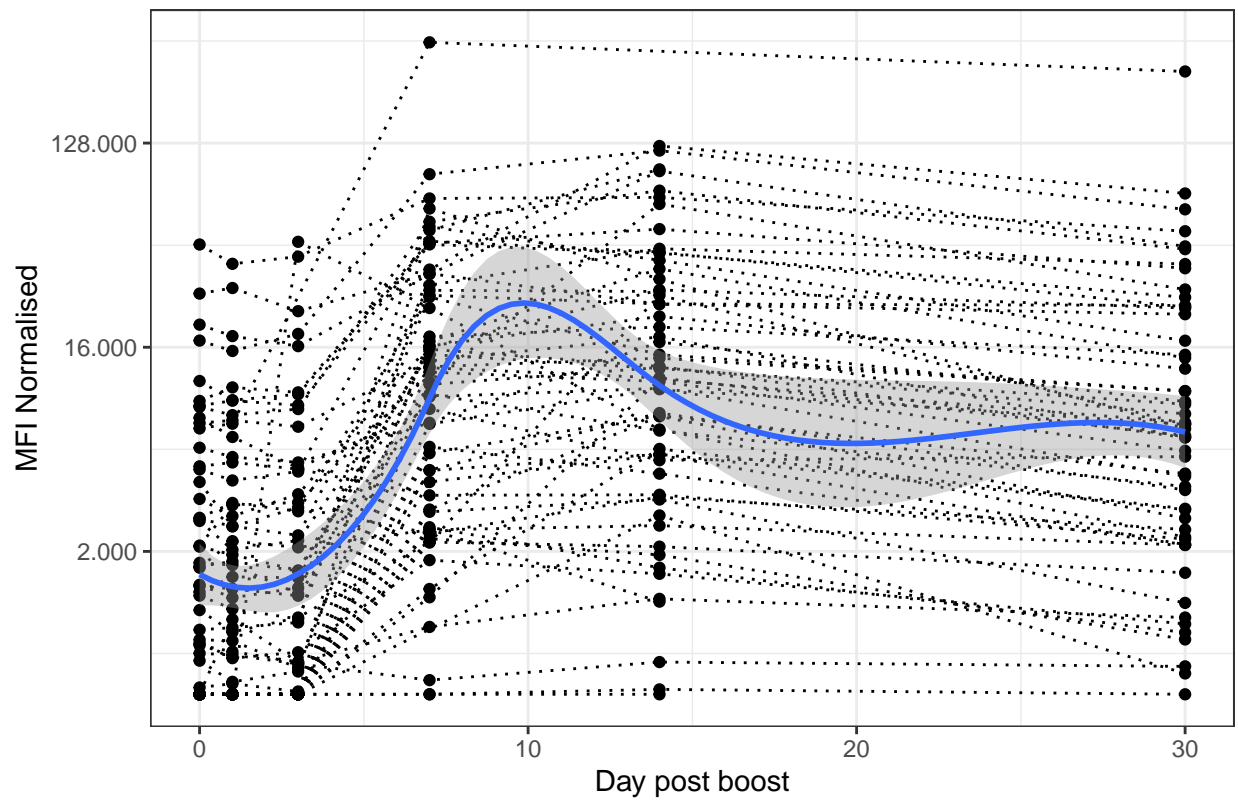


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



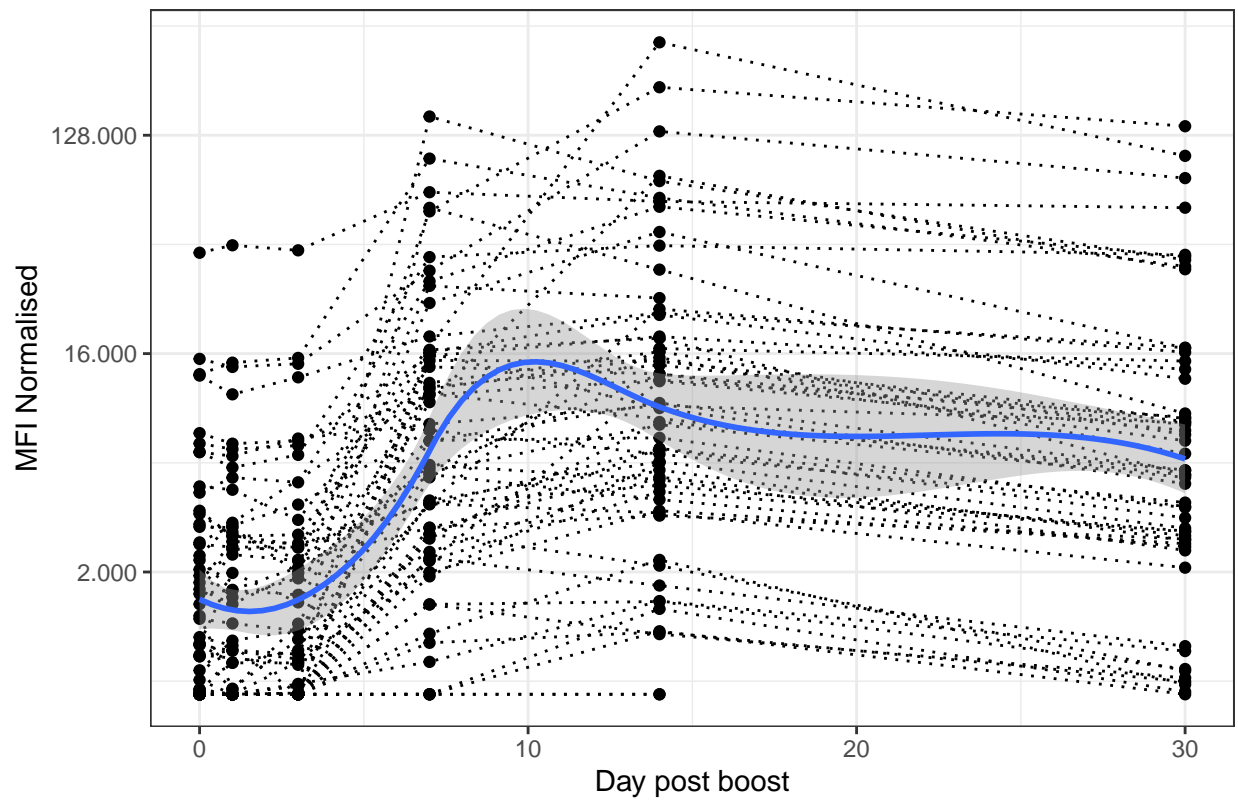
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgG3_PRN_IU/ML



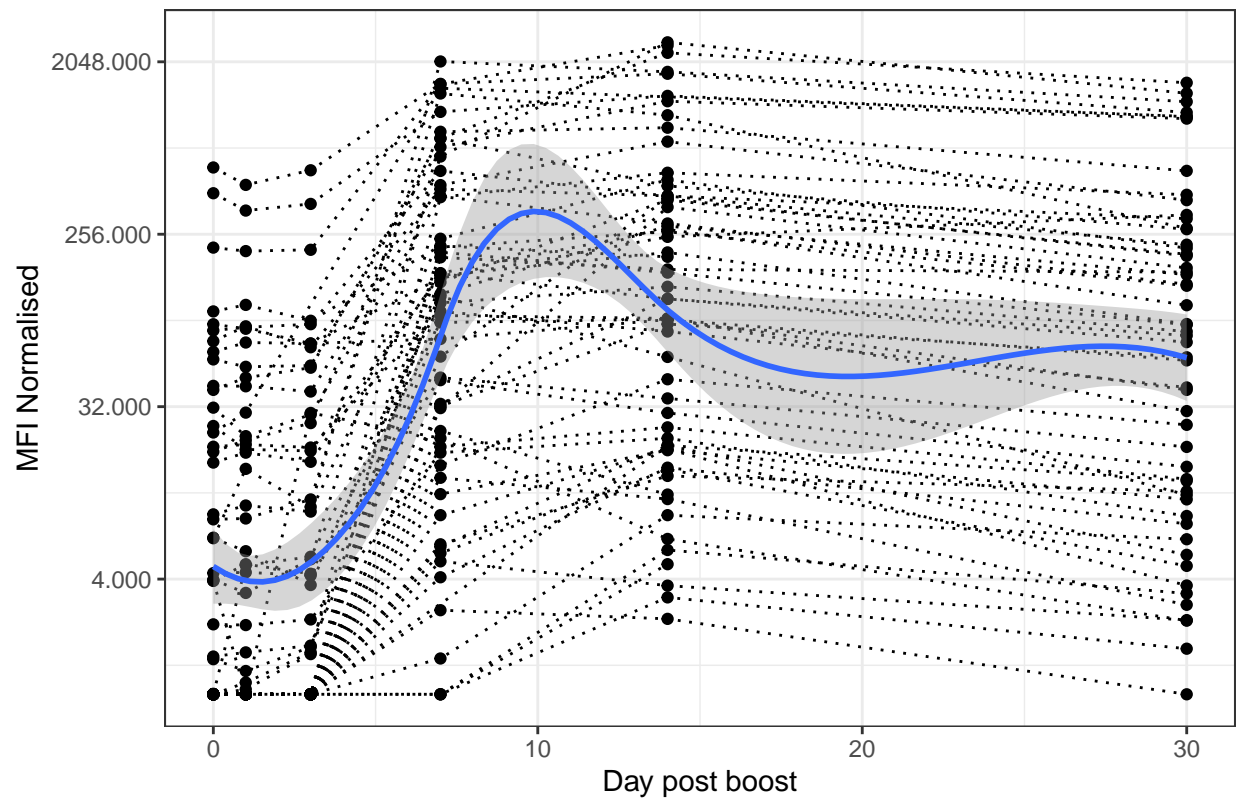
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_FHA_IU/ML



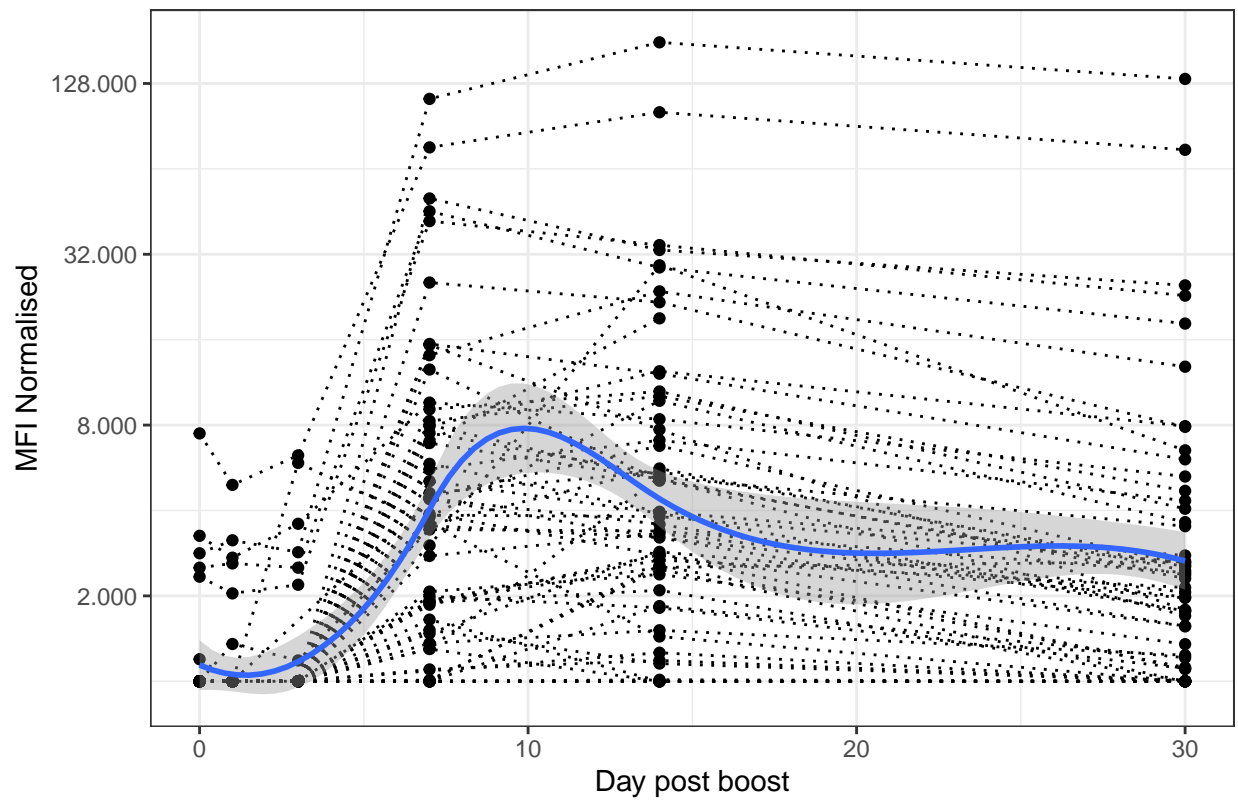
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_FIM2/3_IU/ML



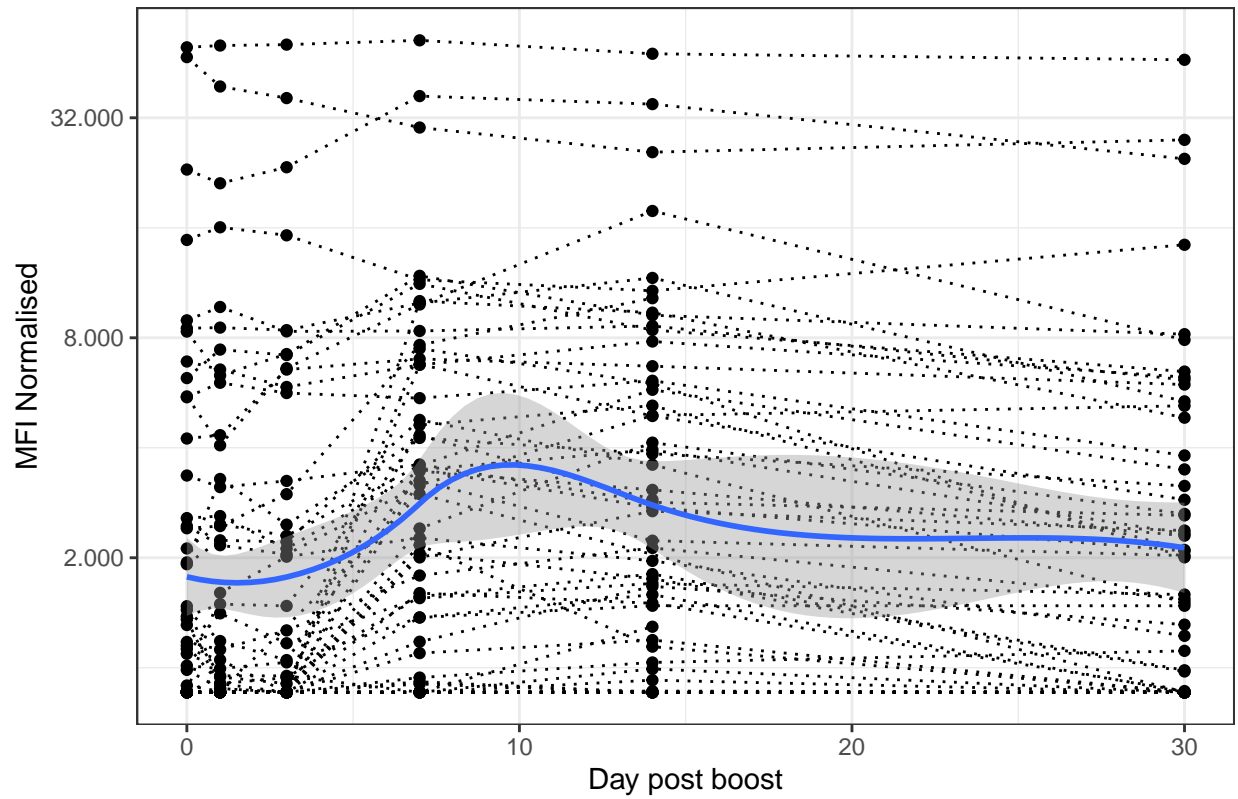
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_TT_IU/ML



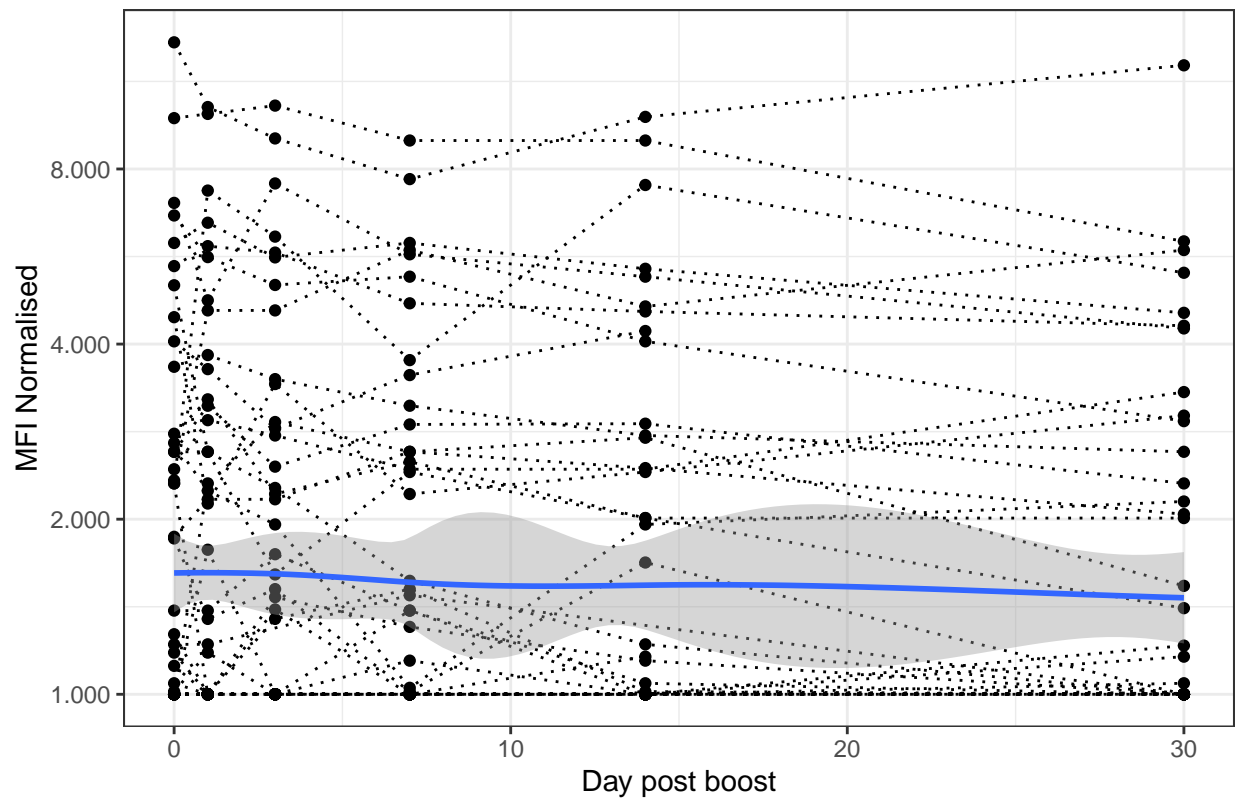
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG3_DT_IU/ML

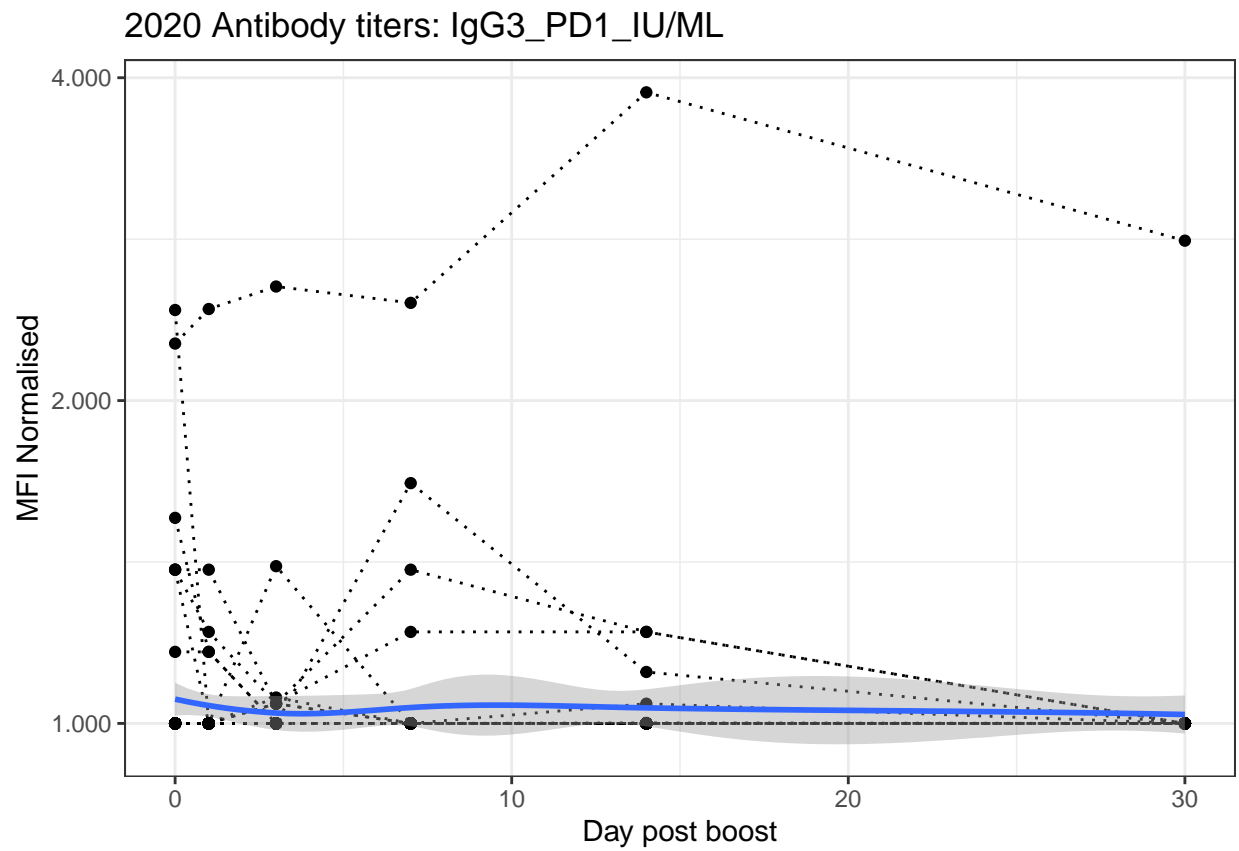


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

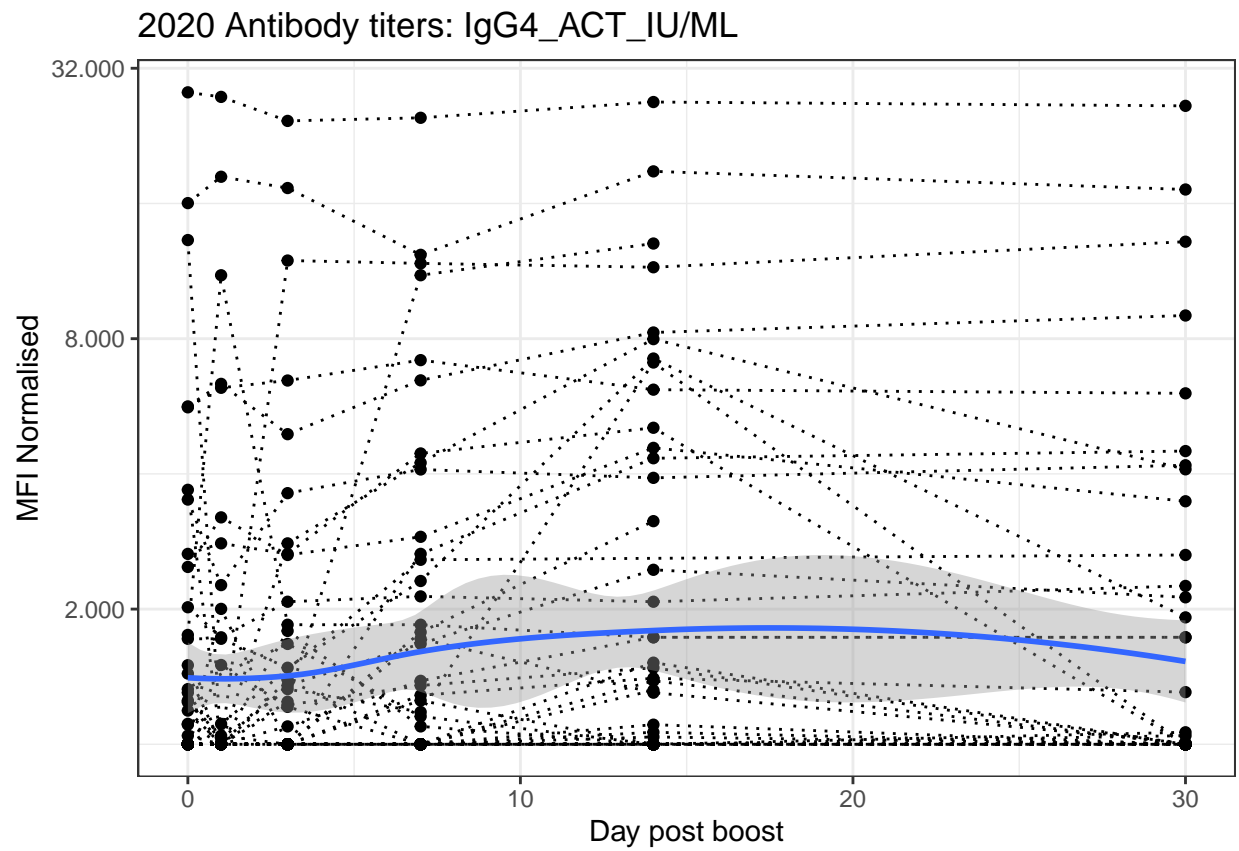
2020 Antibody titers: IgG3_OVA_IU/ML



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

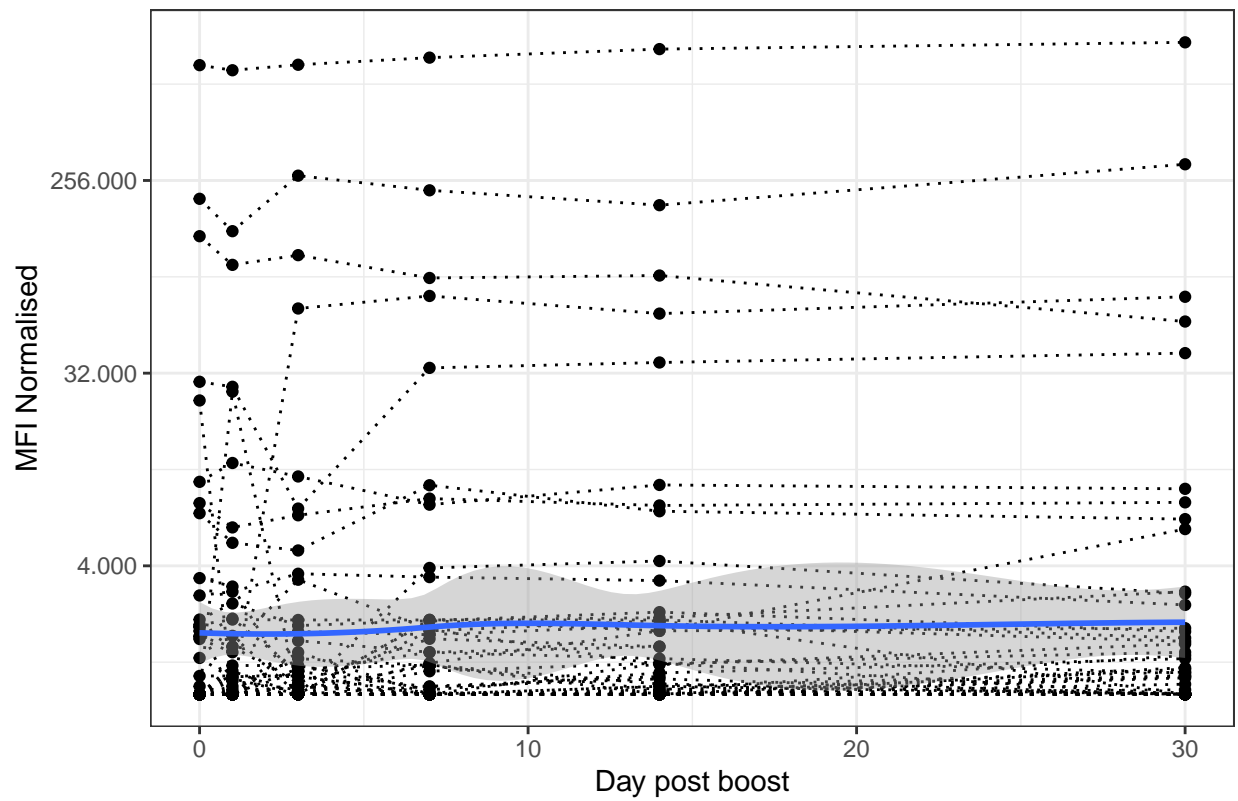


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



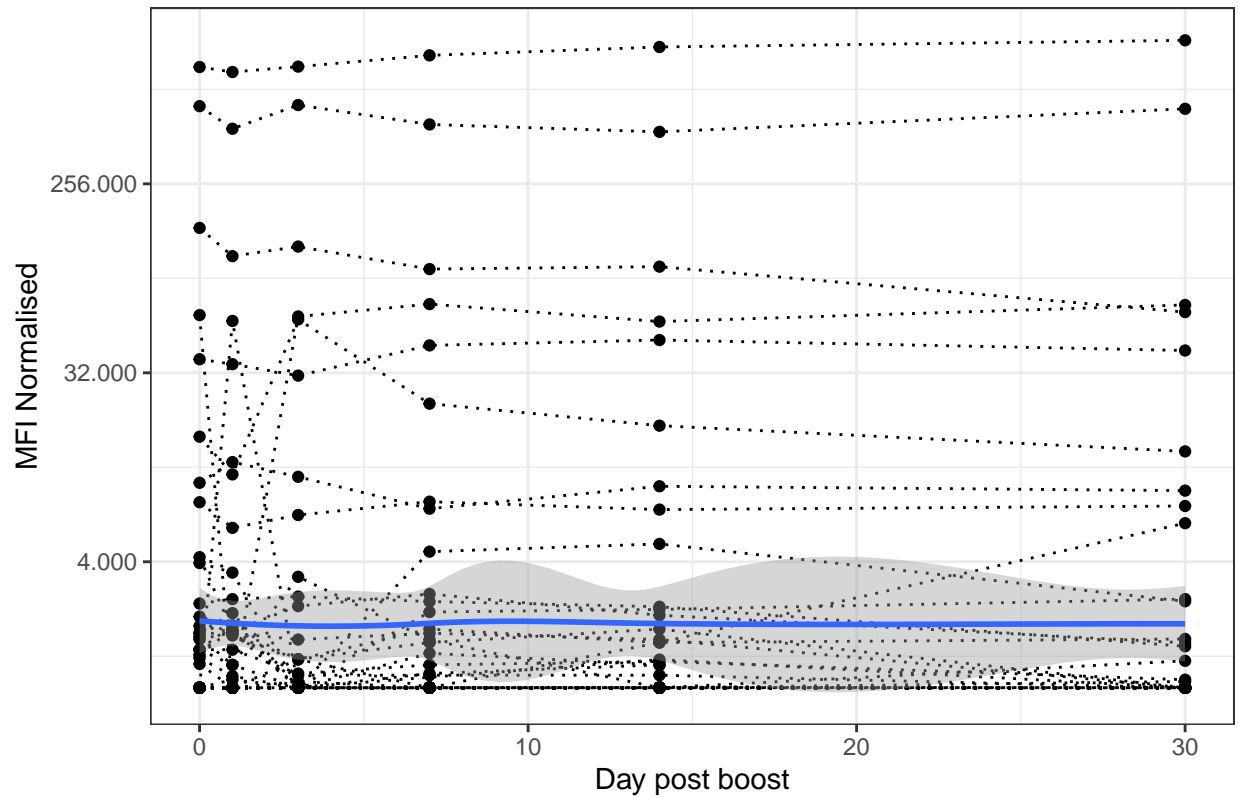
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```


2020 Antibody titers: IgG4_LOS_IU/ML



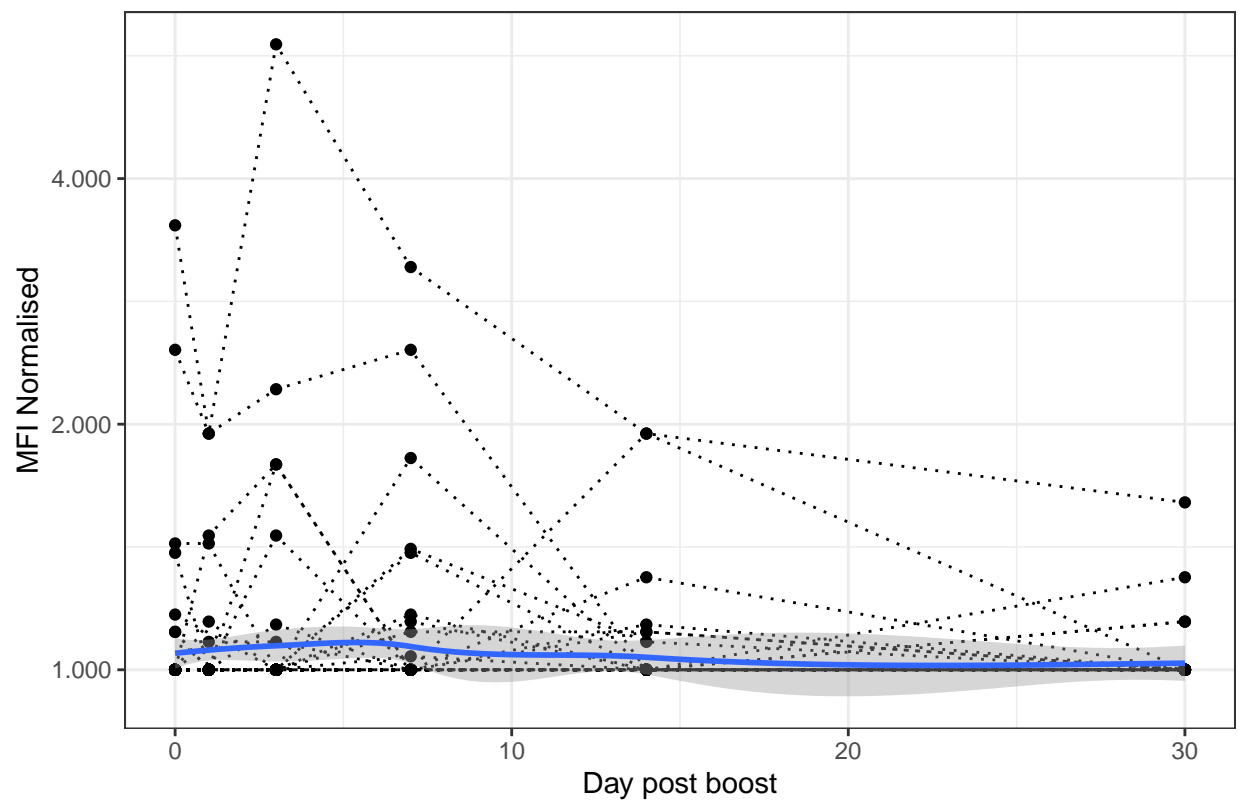
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_FELD1_IU/ML



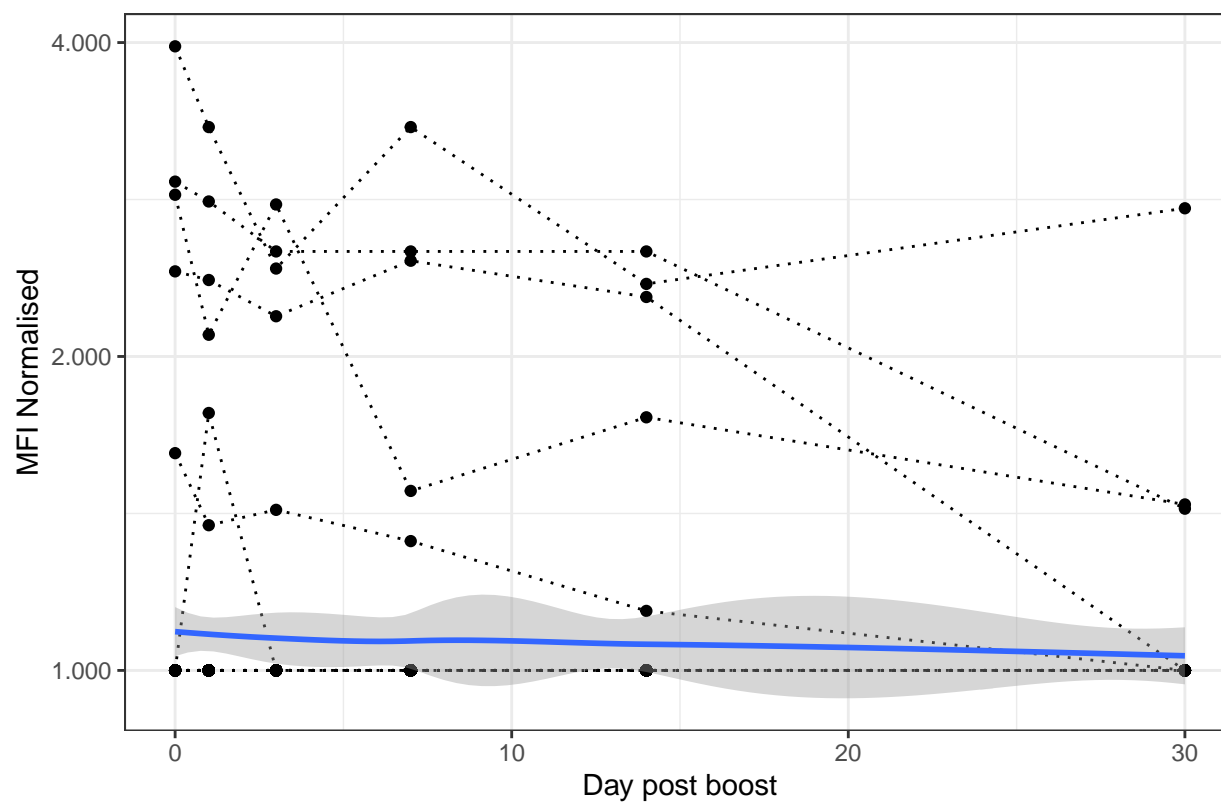
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_BETV1_IU/ML



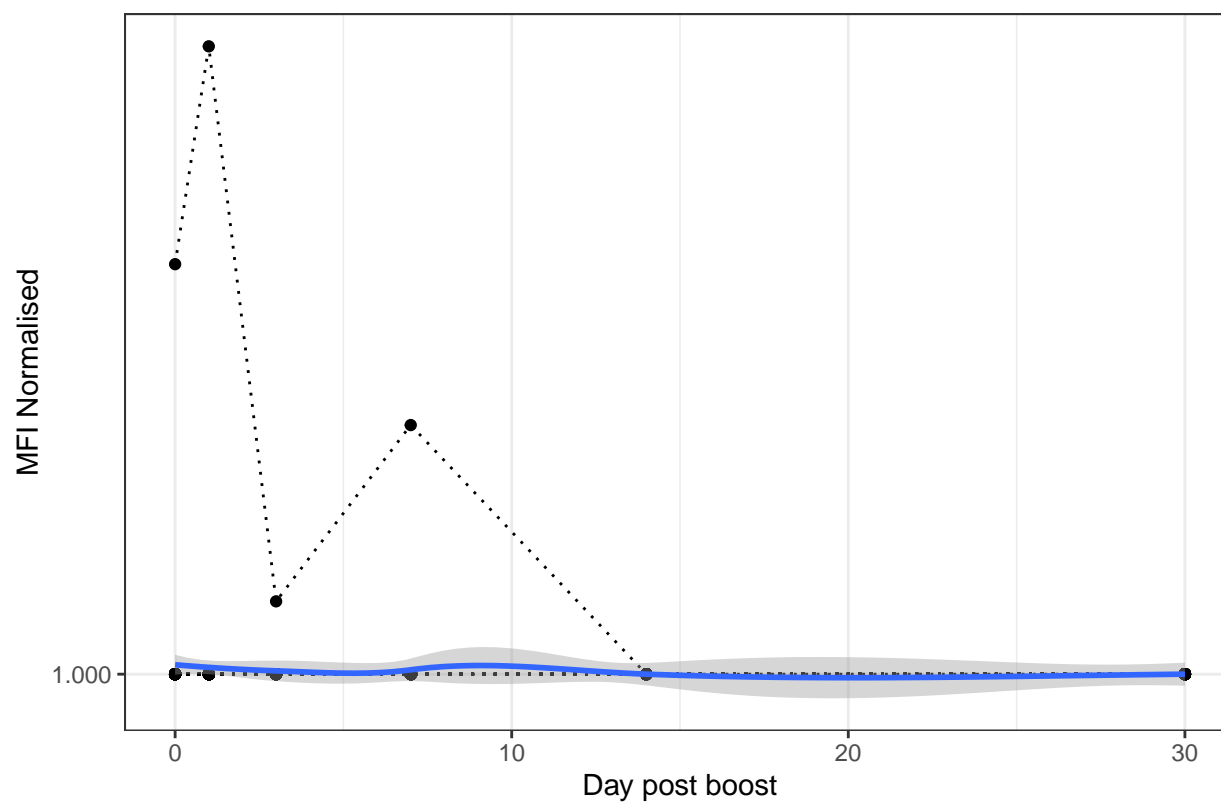
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_LOLP1_IU/ML

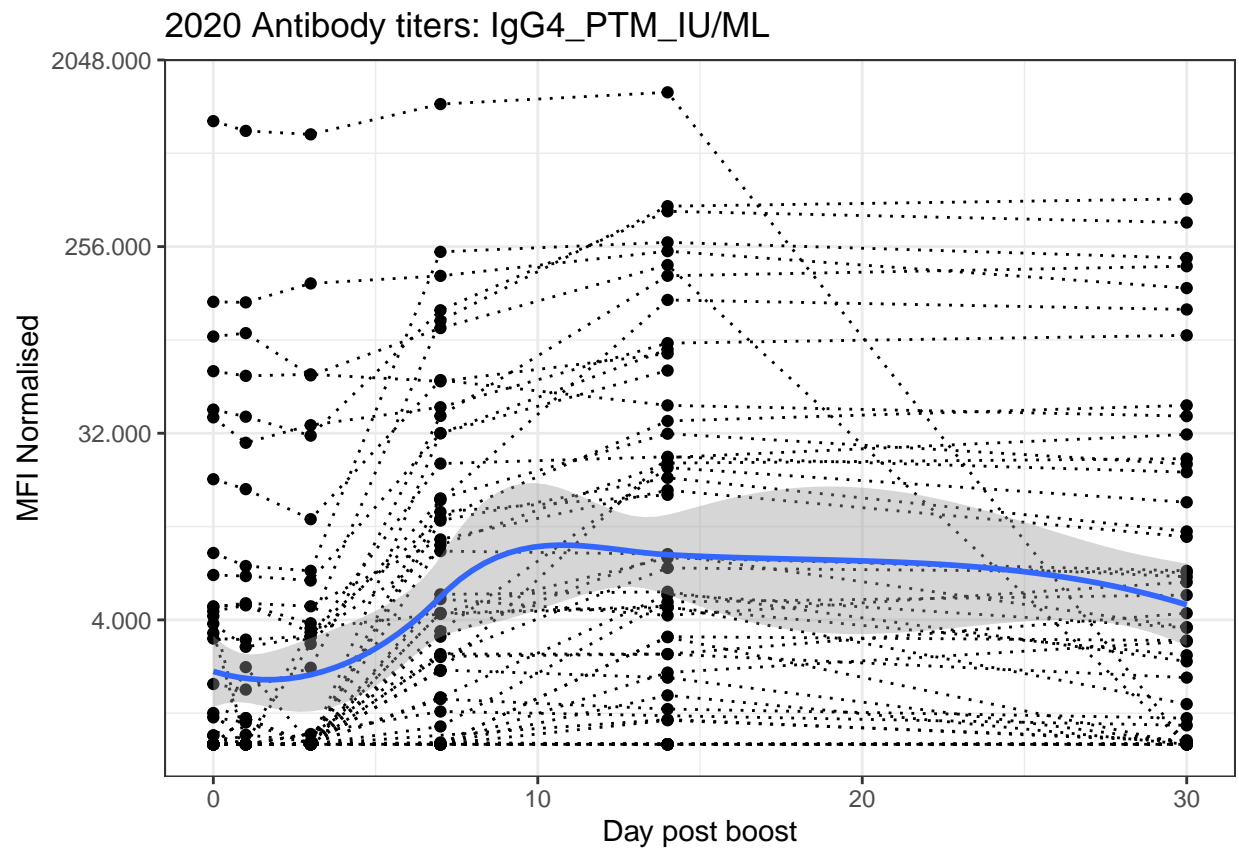


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_Measles_IU/ML

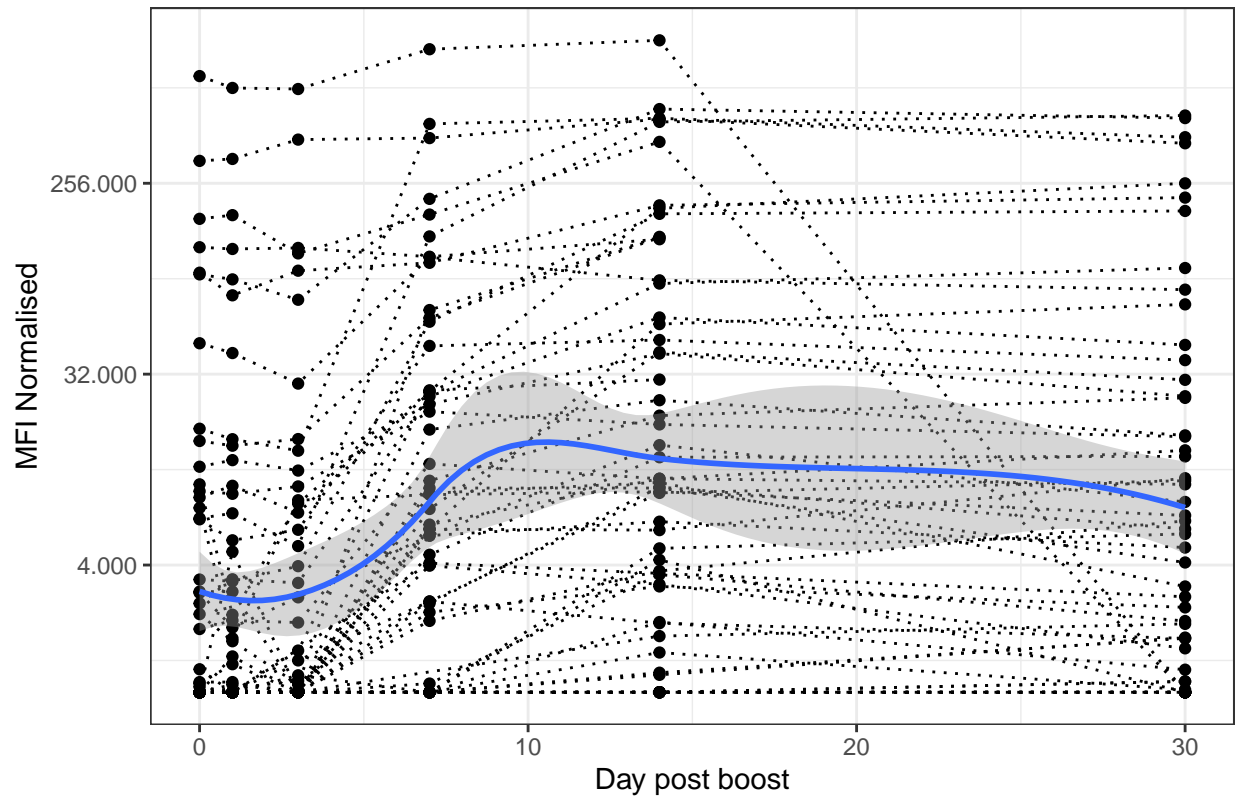


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



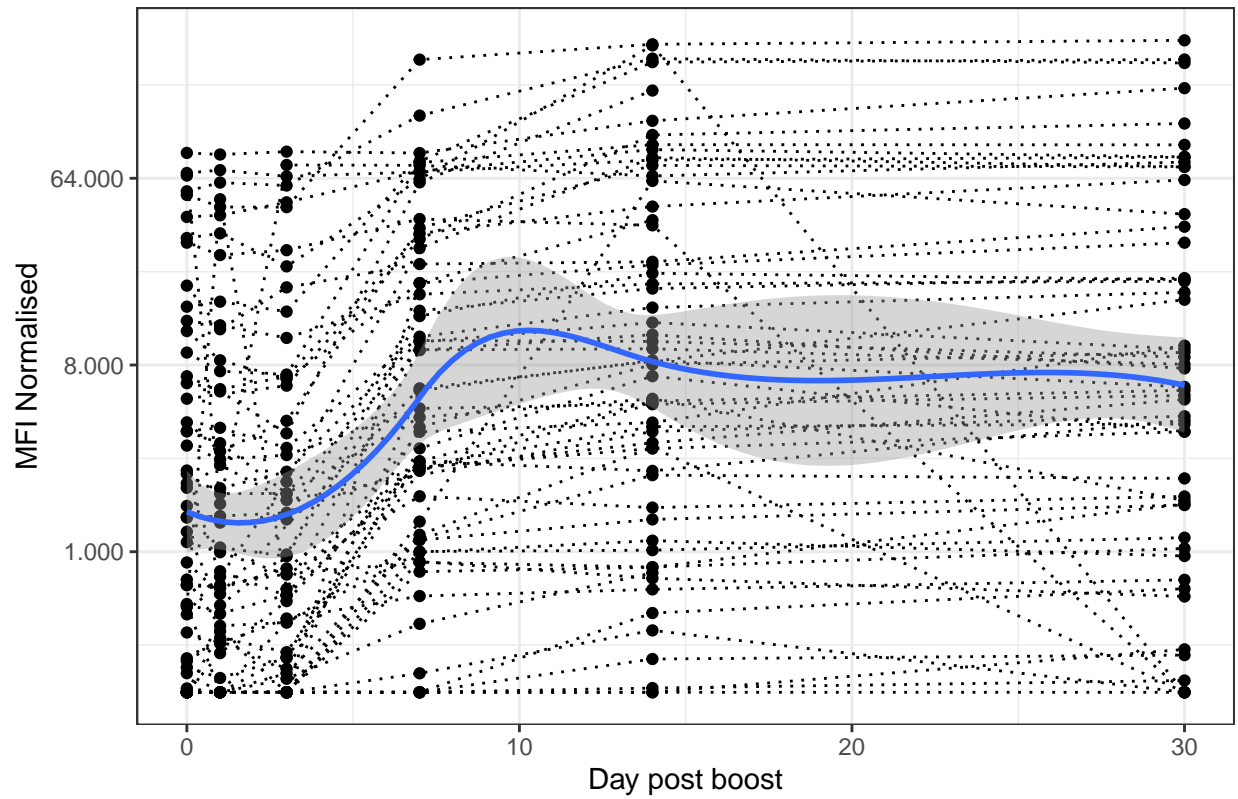
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_PT_IU/ML

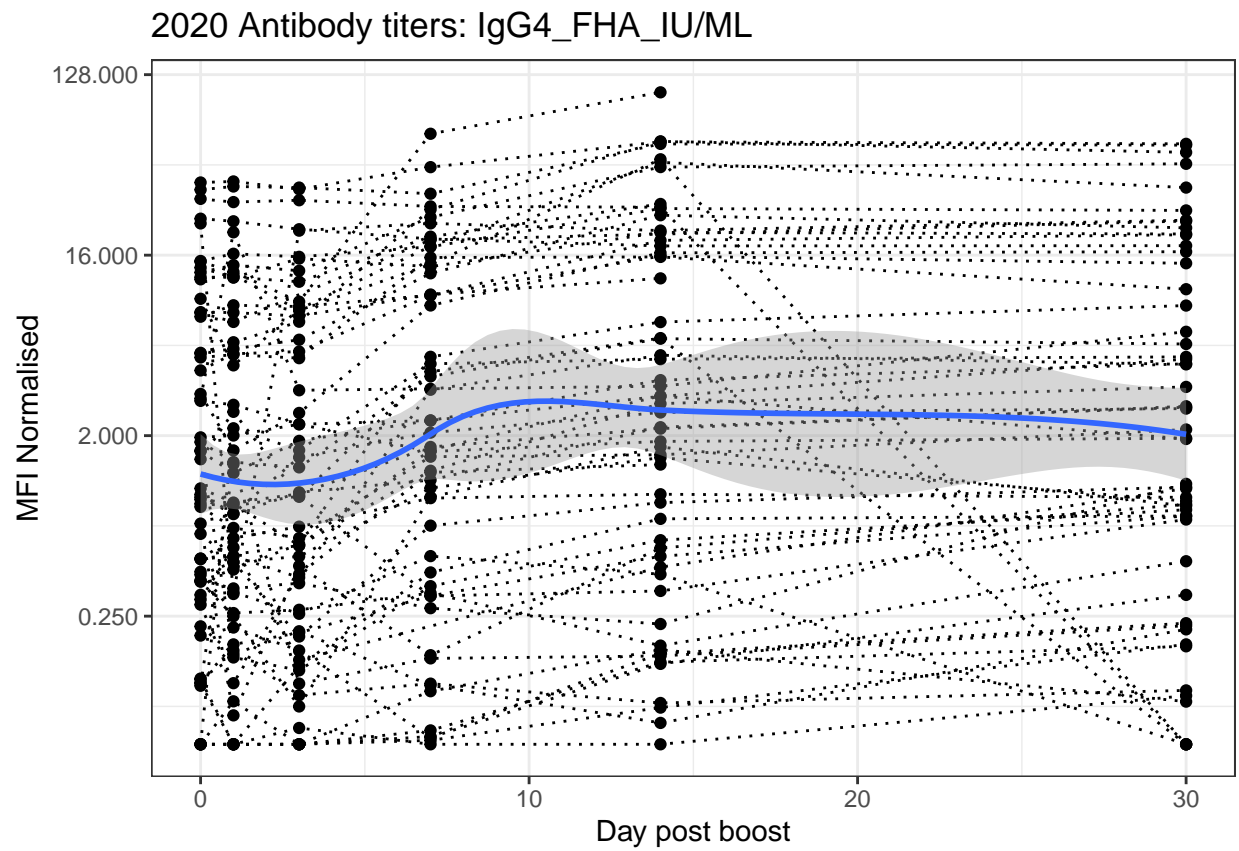


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_PRN_IU/ML

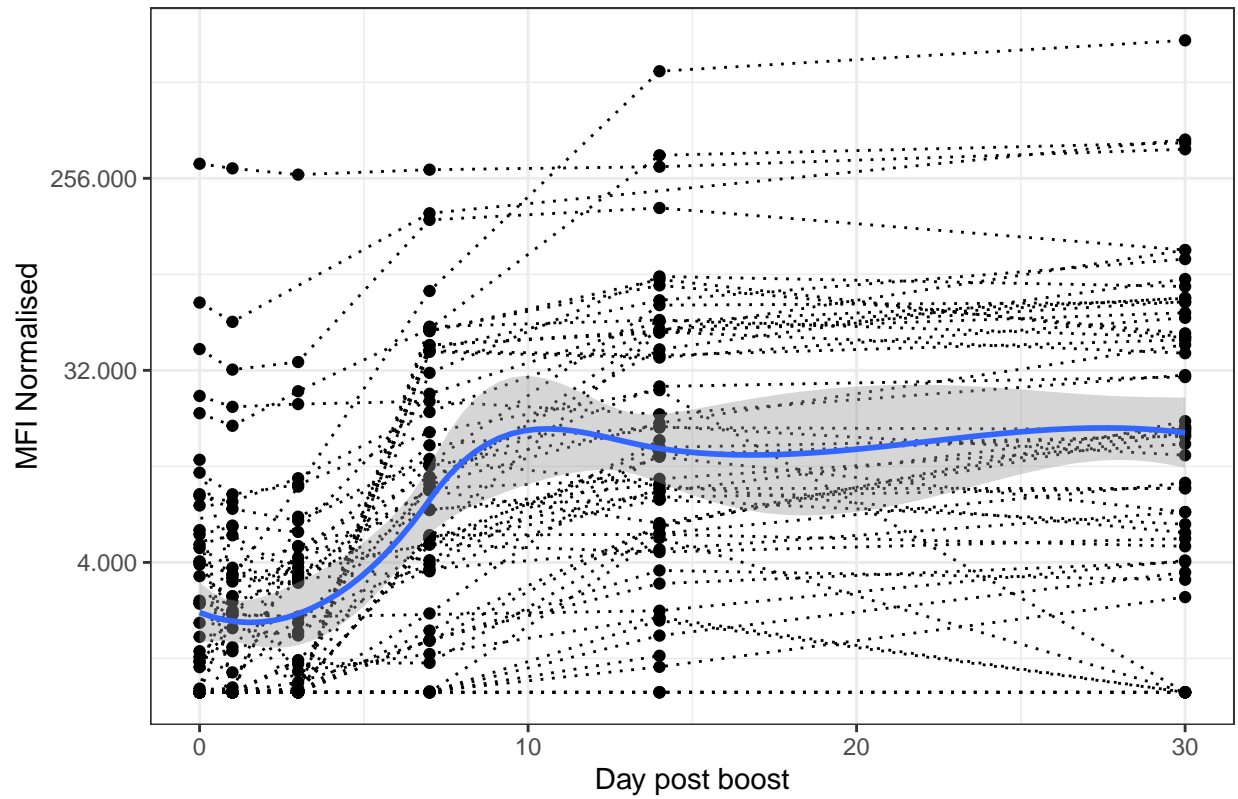


```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

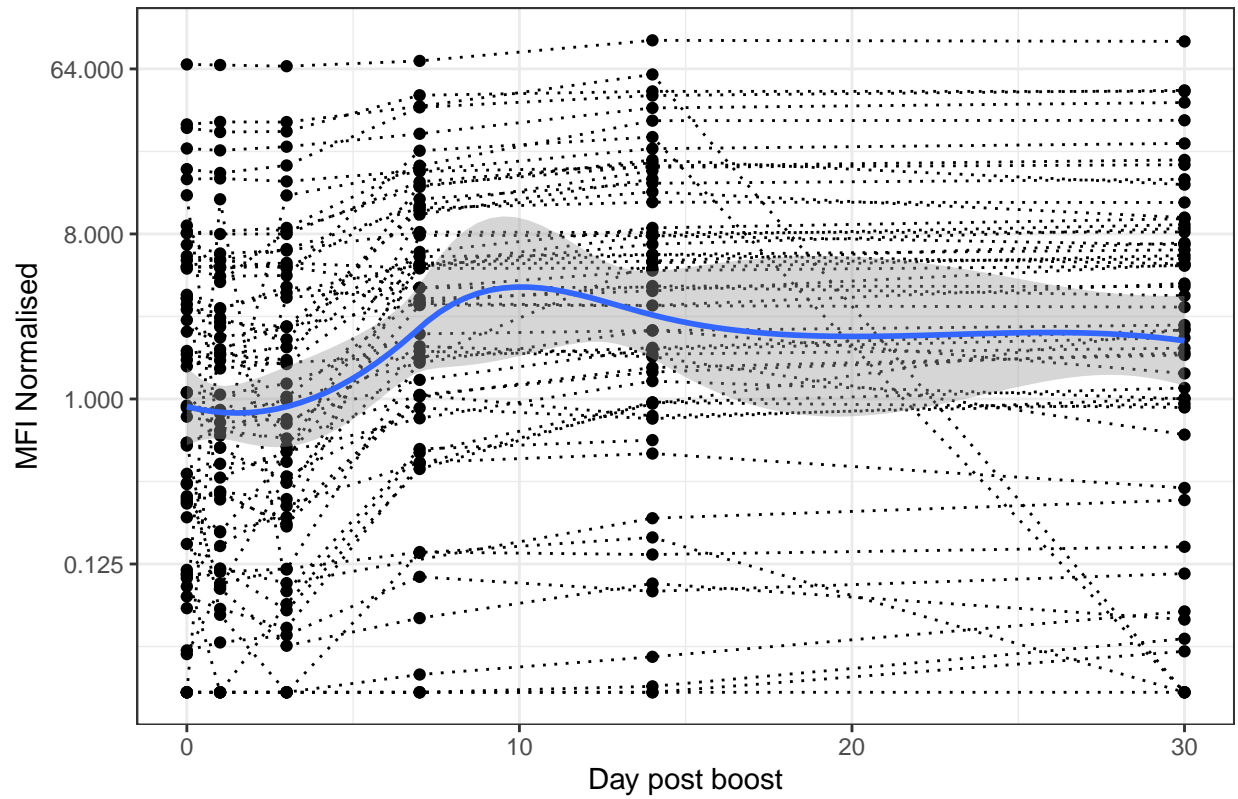
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_FIM2/3_IU/ML



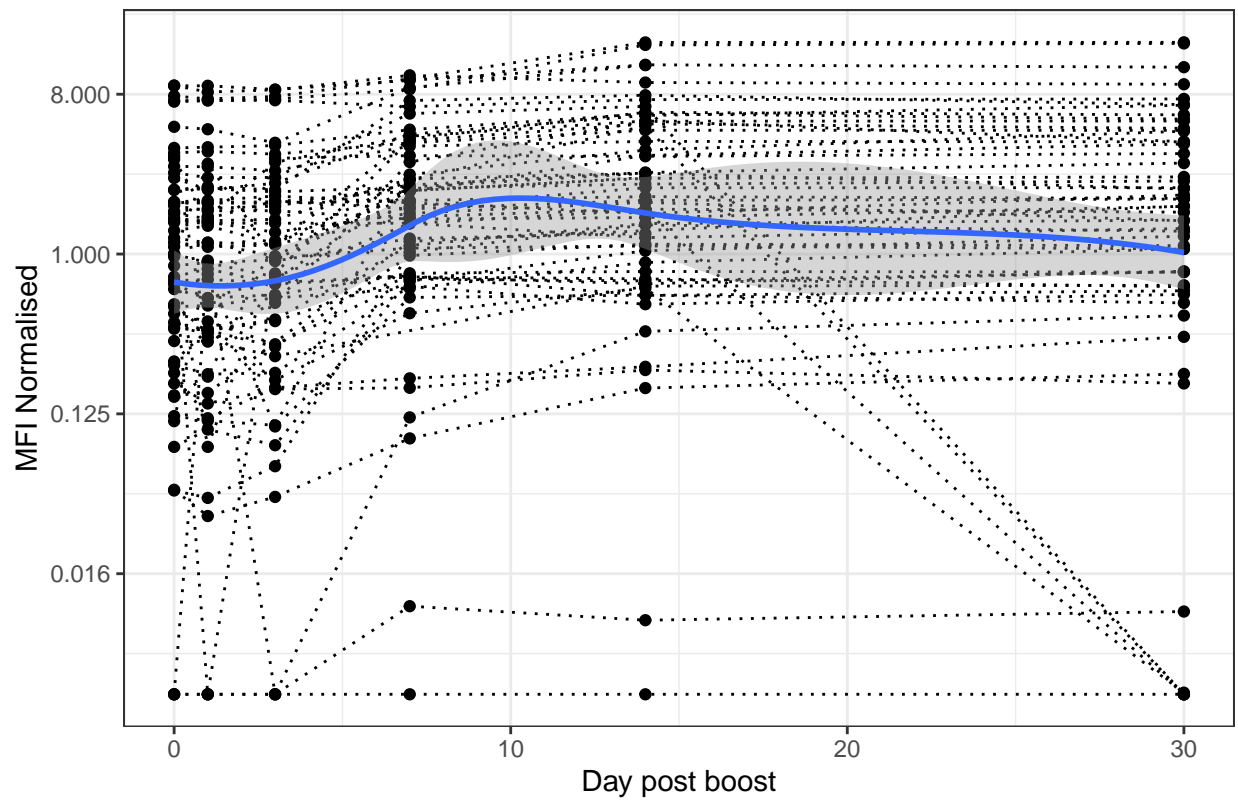
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_TT_IU/ML



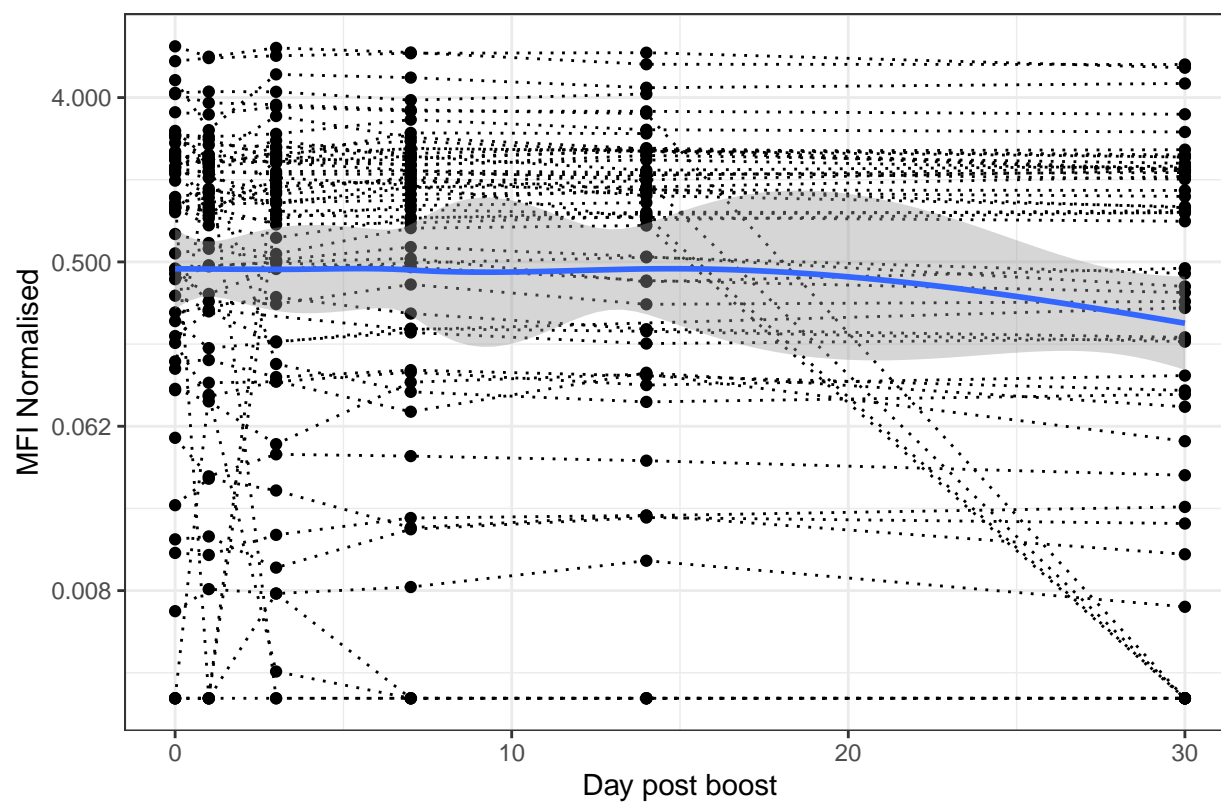
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_DT_IU/ML



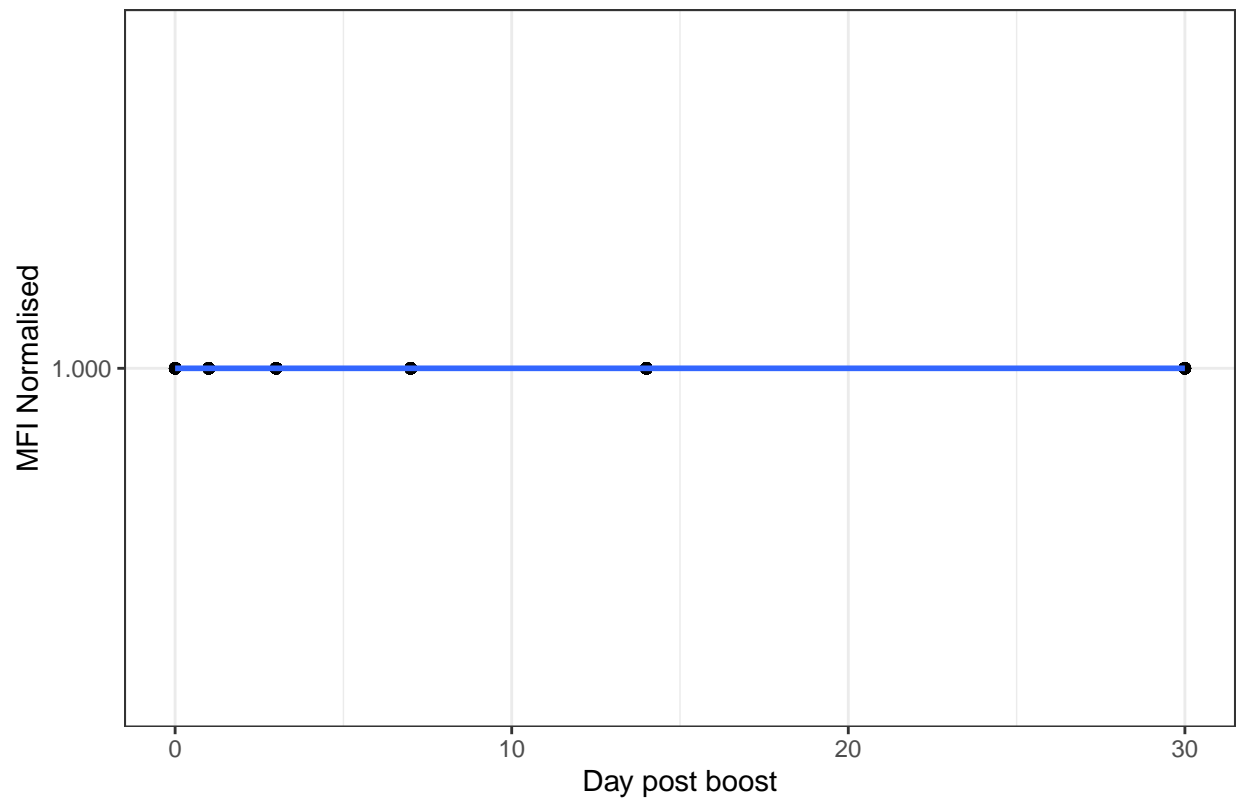
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

2020 Antibody titers: IgG4_OVA_IU/ML



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

2020 Antibody titers: IgG4_PD1_IU/ML



Output/final dataframe for database porpose

```
titers_2020_new <- titers_2020_new_raw %>%
  mutate(ab_titer_normalised = MFI_normalized,
         ab_titer = ab_titer_original) %>%
  select(specimen_id, isotype, antigen, unit, is_antigen_specific, ab_titer, ab_titer_normalised, lower)
write.csv(titers_2020_new, "titers_2020_new.csv")
```

““