# Check data

Smith et al., 2004 did not publish supplementary data.

However, the supplement of Bedford et al. 2014 contains data from Smith et al. 2014.

This notebook imports the data from the Bedford et al. supplement, and checks that it contains the observations used by Bedford et al. 2014 or Smith et al. 2004 to infer maps.

```
library(tidyverse)
```

## Import the data

```
download.file("https://raw.githubusercontent.com/cobeylab/data-emporium/main/Influenza-titer-panel-data/
    destfile = "Bedford_2014_data.txt", method = "curl")
Bedford_data <- read_delim("Bedford_2014_data.txt")
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 10059 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (6): virusIsolate, virusStrain, serumIsolate, serumStrain, titer, source
## dbl (2): virusYear, serumYear
##
## 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.
```

## Bedford et al. expected data

The methods of Bedford et al. 2014 state:

> We compiled an antigenic dataset of hemagglutination inhibition (HI) measurements of virus isolates against post-infection ferret sera **for influenza A/H3N2** by collecting data from previous publications (Hay and Gregory, 2001; Smith et al., 2004; Russell et al., 2008; Barr et al., 2010), NIMR vaccine strain selection reports for 2002 and 2008–2012 (Hay et al., 2002; 2008a, 2009a; McCauley et al., 2010a; 2010b, 2011b, 2012) and the February 2011 VRBPAC report (Cox, 2011).
>
> Because we are interested in longer-term antigenic evolution, **we subsampled the data to have at most 20 virus isolates per year, preferentially keeping those isolates with more antigenic comparisons.** We then kept only those serum isolates that are relatively informative to the antigenic placement of viruses, dropping serum isolates that are compared to four or fewer different virus isolates.
>
> **This censoring left 402 virus isolates, 519 serum** [for H3N2].

## 1. Check that these data contain 402 virus isolates of H3N2

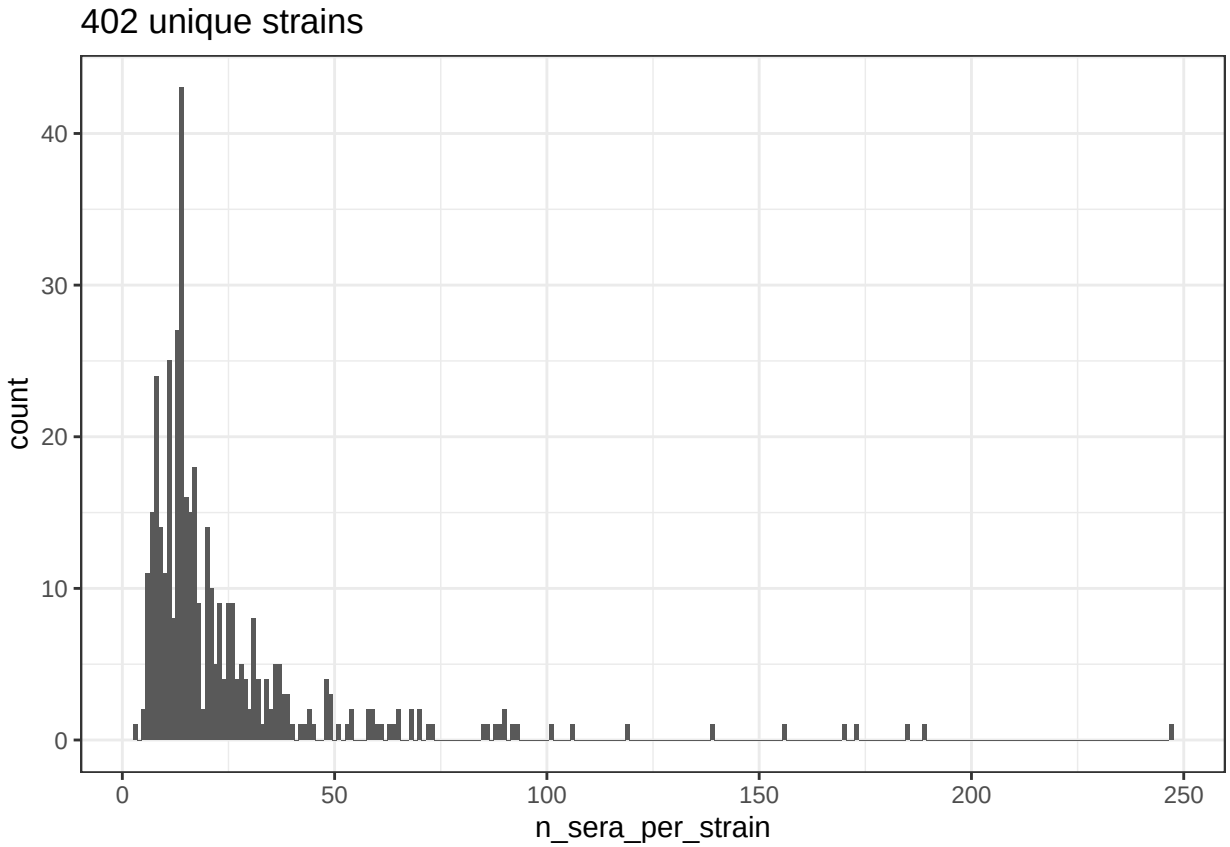
```
Bedford_data %>%
  group_by(virusIsolate) %>%
  summarise(n_strains_per_isolate = n()) %>%
  mutate(n_virus_isolates = nrow(.))
```

```
## # A tibble: 427 x 3
##    virusIsolate                       n_strains_per_isolate n_virus_isolates
##    <chr>                                              <int>            <int>
##  1 A_Human_ANHUI_1238_2005                               26              427
##  2 A_Human_ANHUI_1239_2005                               70              427
##  3 A_Human_AUCKLAND_6_2003                               31              427
##  4 A_Human_Baden-Wurttemberg_38_2005                     21              427
##  5 A_Human_Beijing_51_2002                               16              427
##  6 A_Human_Brazil_1742_2005                              72              427
##  7 A_Human_BRISBANE_3_2005                               31              427
##  8 A_Human_BRISBANE_5_2002                                9              427
##  9 A_Human_BRISBANE_6_2002                               14              427
## 10 A_Human_BRISBANE_7_2003                               24              427
## # ... with 417 more rows
## # i Use `print(n = ...)` to see more rows
```

It looks like there are 427, not 402 isolates in the data. Maybe if we group by virusStrain there will be 402?

```
df <- Bedford_data %>%
  group_by(virusStrain) %>%
  summarise(n_sera_per_strain = n()) %>%
  mutate(n_strains = nrow(.))

df %>%
  ggplot() +
  geom_histogram(aes(x=n_sera_per_strain), binwidth = 1) +
  ggtitle(sprintf('%s unique strains', unique(df$n_strains)))
```




[yes] - There are 402 unique virusStrain values in the data frame.

Look at the rows in which a virusStrain contains more than one virusIsolate

```
Bedford_data %>%
  group_by(virusStrain, virusIsolate) %>%
  summarise(n = n()) %>%
  ungroup() %>%
  group_by(virusStrain) %>%
  mutate(n_strain_repeats = n()) %>%
  dplyr::filter(n_strain_repeats > 1)
```

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

```
## # A tibble: 49 x 4
## # Groups:   virusStrain [24]
##    virusStrain        virusIsolate                n n_strain_repeats
##    <chr>              <chr>                   <int>            <int>
##  1 A/Bangkok/1/1979   A/Bangkok/1/79             13                2
##  2 A/Bangkok/1/1979   BA/1/79                    25                2
##  3 A/Beijing/32/1992  A/Beijing/32/92            13                2
##  4 A/Beijing/32/1992  BE/32/92                   35                2
##  5 A/Beijing/352/1989 A/Beijing/352/89           13                2
##  6 A/Beijing/352/1989 BE/352/89                  19                2
##  7 A/England/42/1972  A/England/42/72            13                2
##  8 A/England/42/1972  EN/42/72                   15                2
##  9 A/Finland/170/2003 A_Human_FINLAND_170_2003   13                2
## 10 A/Finland/170/2003 FI/170/03                  12                2
```

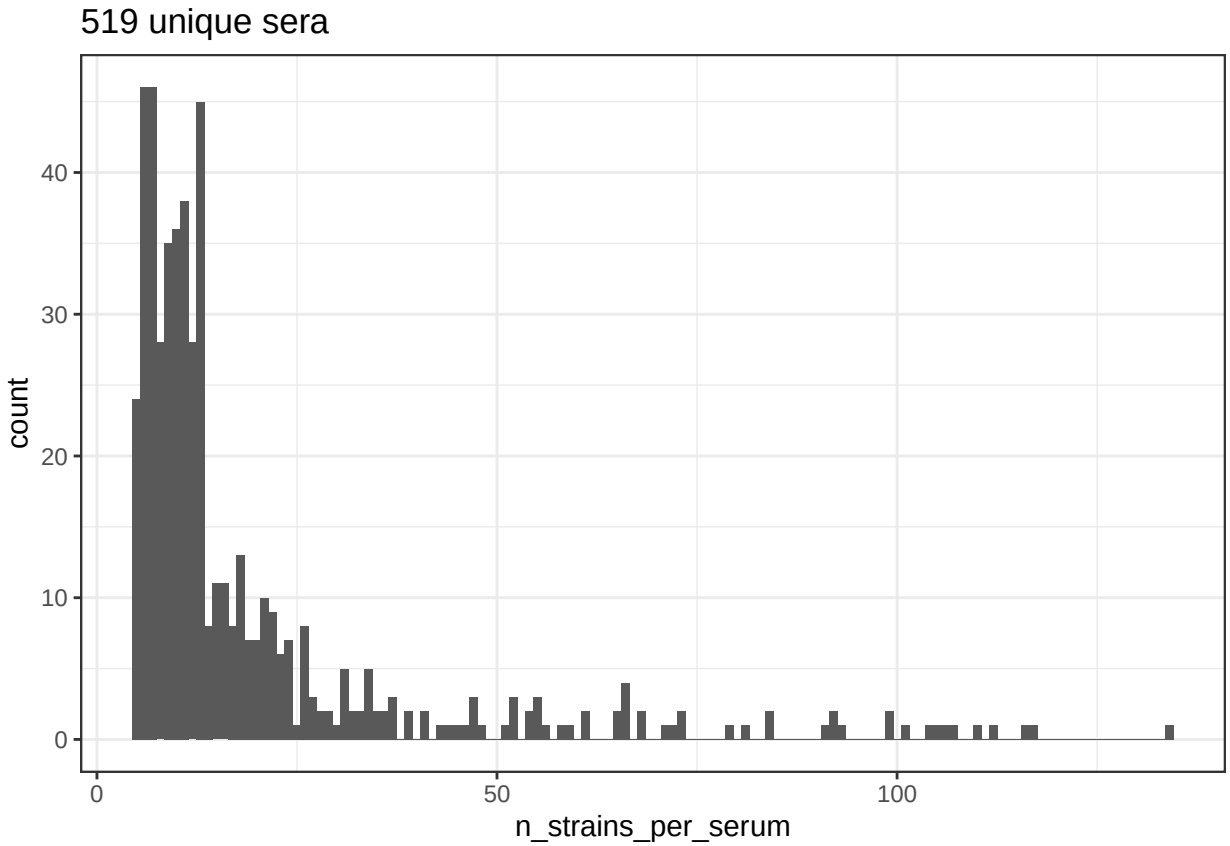
```
## # ... with 39 more rows
## # i Use `print(n = ...)` to see more rows
```

It looks like the virusStrain column is just a standardized version of the virusIsolate column. Nomenclature is more variable in the virus Isolate column.

## 2. Check that these data contain 519 sera for H3N2

```
df <- Bedford_data %>%
  group_by(serumIsolate) %>%
  summarise(n_strains_per_serum = n()) %>%
  mutate(n_sera = nrow(.))

df %>%
  ggplot() +
  geom_histogram(aes(x=n_strains_per_serum), binwidth = 1) +
  ggtitle(sprintf('%s unique sera', unique(df$n_sera)))
```




Confusingly, it looks like we're supposed to use the `serumIsolate` column and the `virusStrain` column, not `serumStrain` and `virusStrain`.

I guess this is because:

- The batch matters for sera.
- The antigenic structure (not the batch) matters for the virus?

---

# Check the Smith data

The methods of Smith et al. 2004 state:

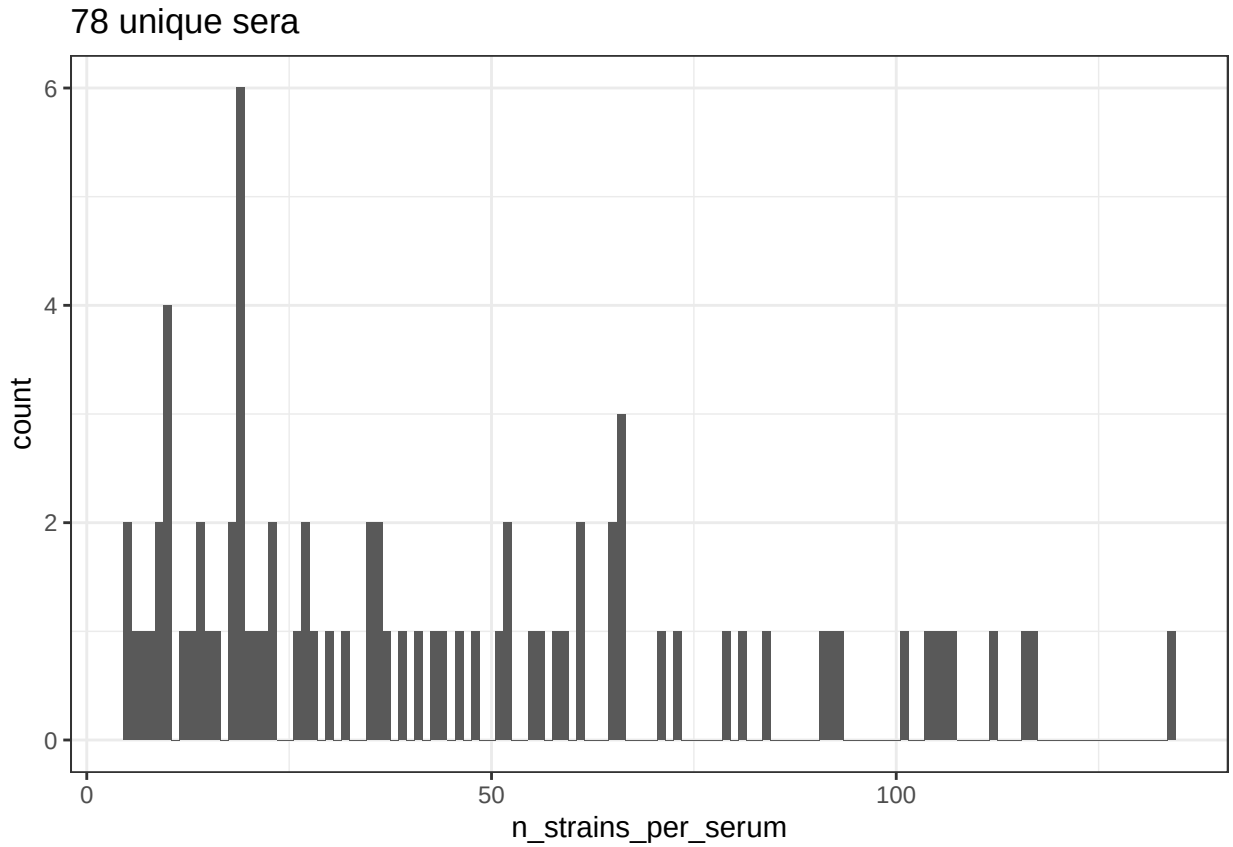
> Antigenic data from 35 years of [H3N2] influenza surveillance between 1968 and 2003 were combined into a single dataset. We sequenced the HA1 domain of a subset of thesevirus isolates (26,27) and restricted the an-tigenic analysis to these sequenced isolates tofacilitate a direct comparison of antigenic and genetic evolution.
>
> The resulting antigenic dataset consisted of a table of **79 post infection ferret antisera by 273 viral isolates**, with **4215 individual HI measurements** as entries in the table.
>
> Ninety-four of the isolates werefrom epidemics in the Netherlands, and 179were from elsewhere in the world.

```
Smith_data <- Bedford_data %>%
  dplyr::filter(source %in% c('Smith2004'))
write_csv(Smith_data, file = 'Smith_2004_subset_of_Bedford_2014.csv')
```

## 1. Check that there are 79 unique antisera

```
df <- Smith_data %>%
  group_by(serumIsolate) %>%
  summarise(n_strains_per_serum = n()) %>%
  mutate(n_sera = nrow(.))

df %>%
  ggplot() +
  geom_histogram(aes(x=n_strains_per_serum), binwidth = 1) +
  ggtitle(sprintf('%s unique sera', unique(df$n_sera)))
```

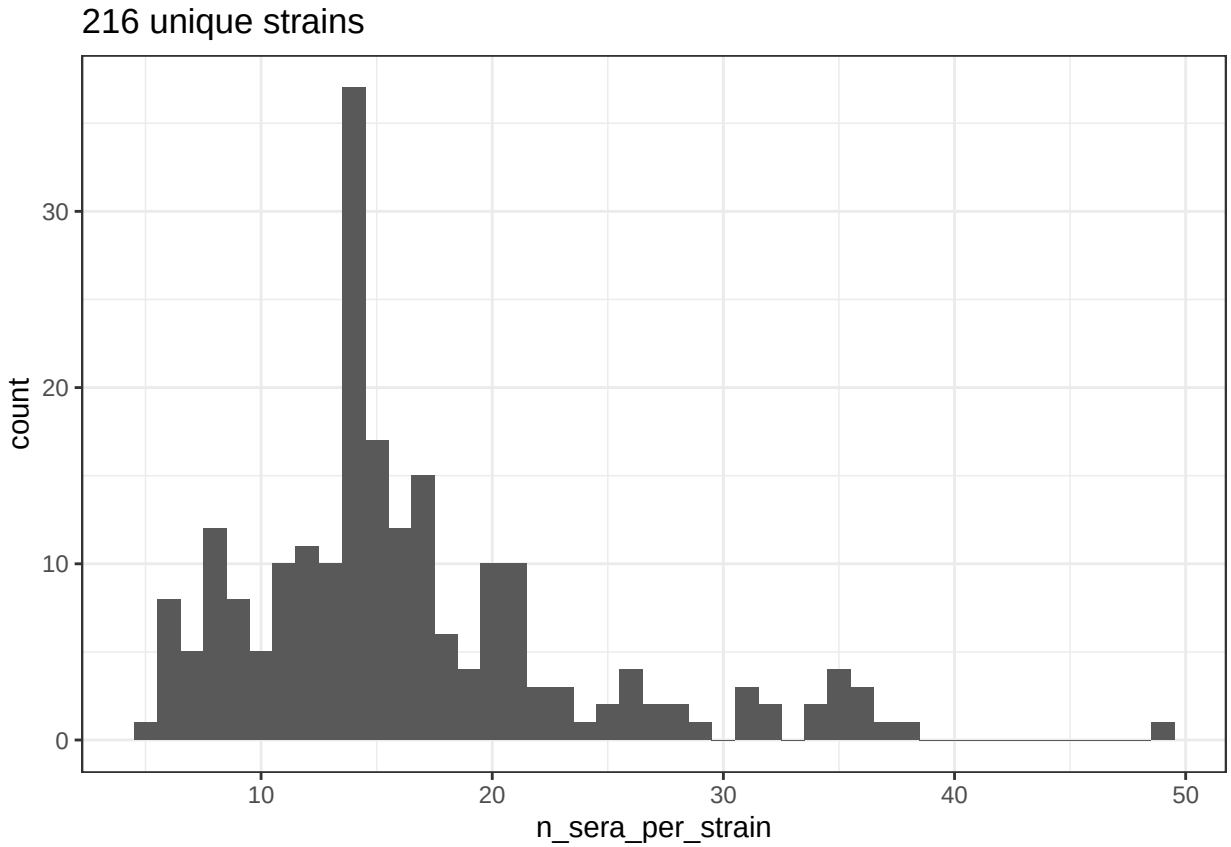



-> We have one less serum than the original Smith 2004 paper

## 2. Check that there are 273 unique viruses

```
df <- Smith_data %>%
  group_by(virusStrain) %>%
  summarise(n_sera_per_strain = n()) %>%
  mutate(n_strains = nrow(.))

df %>%
  ggplot() +
  geom_histogram(aes(x=n_sera_per_strain), binwidth = 1) +
  ggtitle(sprintf('%s unique strains', unique(df$n_strains)))
```




–> There are about 60 fewer strains in the panel than in the original Smith 2004 paper

### 3. Check that there are 4215 HI measurements

```
Smith_data
```

```
## # A tibble: 3,541 x 8
##    virusIsolate virusStrain         virus~1 serum~2 serum~3 serum~4 titer source
##    <chr>        <chr>                 <dbl> <chr>   <chr>     <dbl> <chr> <chr>
##  1 BI/15793/68  A/Bilthoven/15793/~    1968 HK/1/68 A/Hong~    1968 354   Smith~
##  2 BI/15793/68  A/Bilthoven/15793/~    1968 EN/42/~ A/Engl~    1972 501   Smith~
##  3 BI/15793/68  A/Bilthoven/15793/~    1968 PC/1/73 A/Port~    1973 109   Smith~
##  4 BI/15793/68  A/Bilthoven/15793/~    1968 VI/3A/~ A/Vict~    1975 63    Smith~
##  5 BI/15793/68  A/Bilthoven/15793/~    1968 LE/360~ A/Leni~    1986 <10   Smith~
##  6 BI/15793/68  A/Bilthoven/15793/~    1968 TE/1A/~ A/Texa~    1977 <20   Smith~
##  7 BI/15793/68  A/Bilthoven/15793/~    1968 NL/330~ A/Neth~    1985 <10   Smith~
##  8 BI/15793/68  A/Bilthoven/15793/~    1968 BA/1/79 A/Bang~    1979 <10   Smith~
##  9 BI/15793/68  A/Bilthoven/15793/~    1968 PH/2/82 A/Phil~    1982 <10   Smith~
## 10 BI/15793/68  A/Bilthoven/15793/~    1968 WE/4/85 A/Well~    1985 <10   Smith~
## # ... with 3,531 more rows, and abbreviated variable names 1: virusYear,
## #   2: serumIsolate, 3: serumStrain, 4: serumYear
## # i Use `print(n = ...)` to see more rows
```

### There are only 3541 rows in this dataset, and the number of strains and antisera is lower than expected.

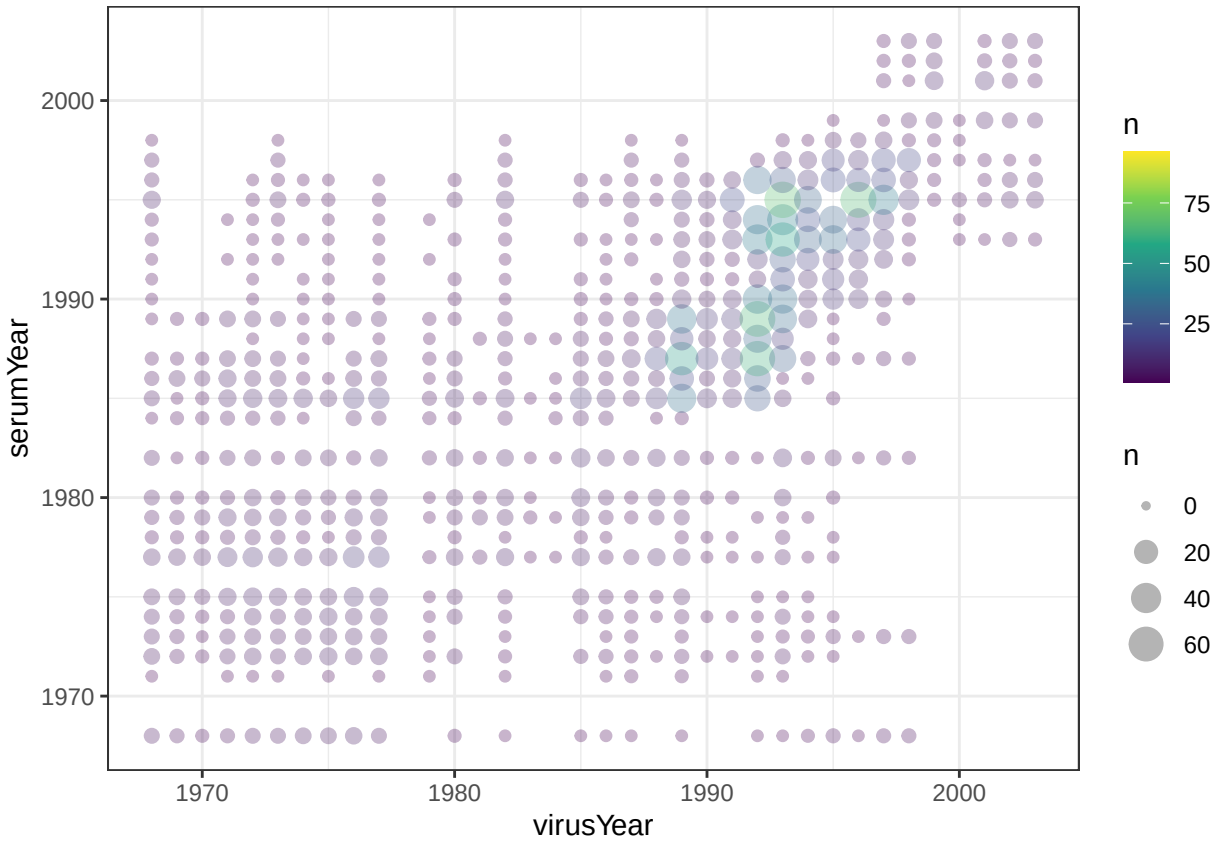
It looks like about 84% (3541 of 4215) of the observations from Smith 2004 are present here. The available rows account for almost all sera (78/79), and most strains (216/273) It's possible that some observations were filtered out because there were fewer than 4 strains per serum.

### Visualize the distribution of strain and titer years

```
Smith_data %>%
  group_by(virusYear, serumYear) %>%
  summarise(n = n()) %>%
  ggplot() +
  geom_point(aes(x = virusYear, y = serumYear, size = n, color = n), alpha = .3) +
  scale_color_viridis_c() +
  scale_size(limits = c(0, 75))
```

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

```
## Warning: Removed 2 rows containing missing values (geom_point).
```

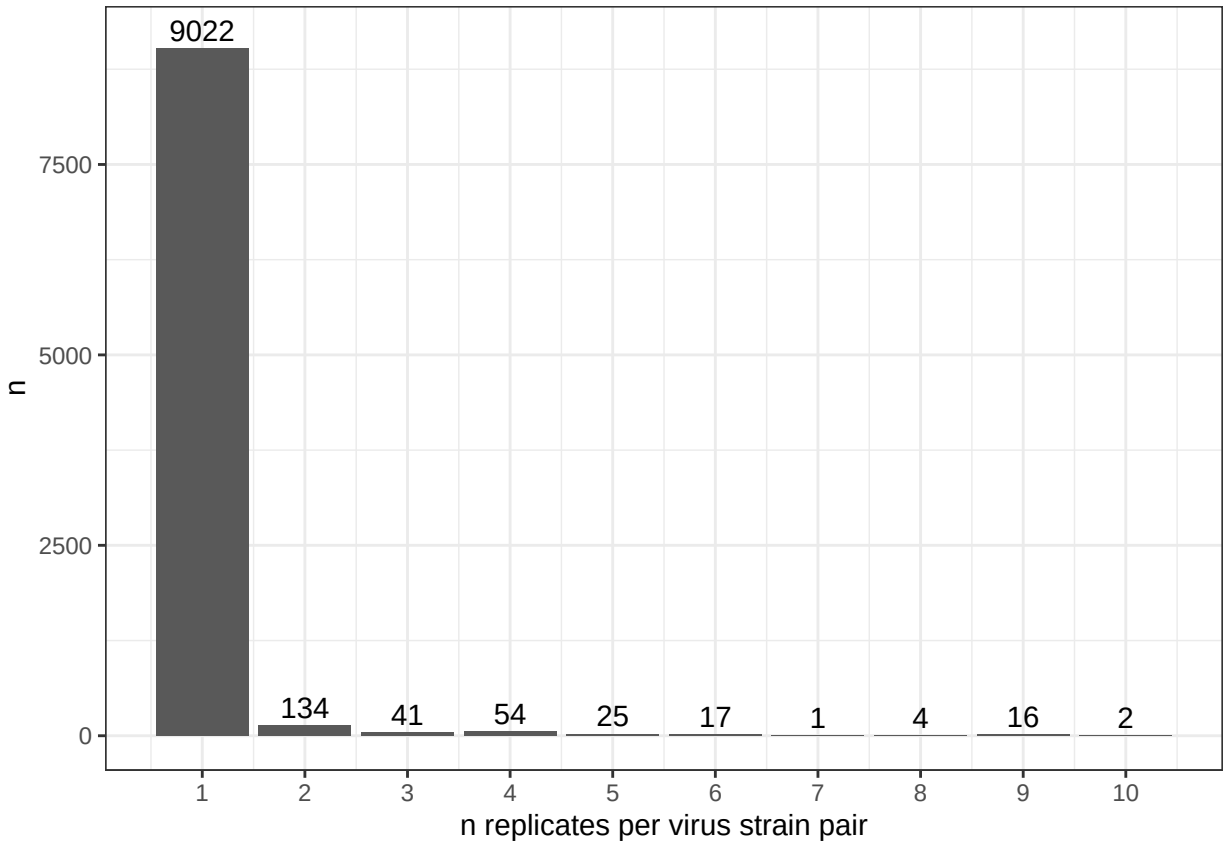


## Assess variance in individual titers observed among ferrets infected by the same virus and tested against the same strain.

### In the Bedford dataset, assess the number of replicates (individual titer observations) for each virus-strain pair:

```
Bedford_data %>%
  select(virusStrain, serumIsolate, titer, source) %>%
  group_by(virusStrain, serumIsolate) %>%
  summarise(n_replicates = n()) %>%
  group_by(n_replicates) %>%
  summarise(n = n()) %>%
  ggplot() +
  geom_bar(aes(x = n_replicates, y = n), stat = "identity") +
  geom_text(aes(x = n_replicates, y = n+100, label = n), vjust = 0)+
  xlab('n replicates per virus strain pair') +
  scale_x_continuous(breaks = 1:10)
```

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




```
ggsave('n_replicates.png')
```

```
## Saving 6.5 x 4.5 in image
```

N.B. - we need to dig deeper into the methods used to produce these repeated observations. Are they technical replicates (sera drawn from the same ferret repeatedly tested in the same lab), or individual replicates (sera

from different ferrets)?

### Look at the individual distances for each strain-virus pair with n>=5:

```
## Get a list of virus-strain pairs with repeat observations
repeat_observations = Bedford_data %>%
  select(virusStrain, serumIsolate, titer, source) %>%
  group_by(virusStrain, serumIsolate) %>%
  summarise(n_replicates = n()) %>%
  ungroup() %>%
  dplyr::filter(n_replicates >= 5)
```

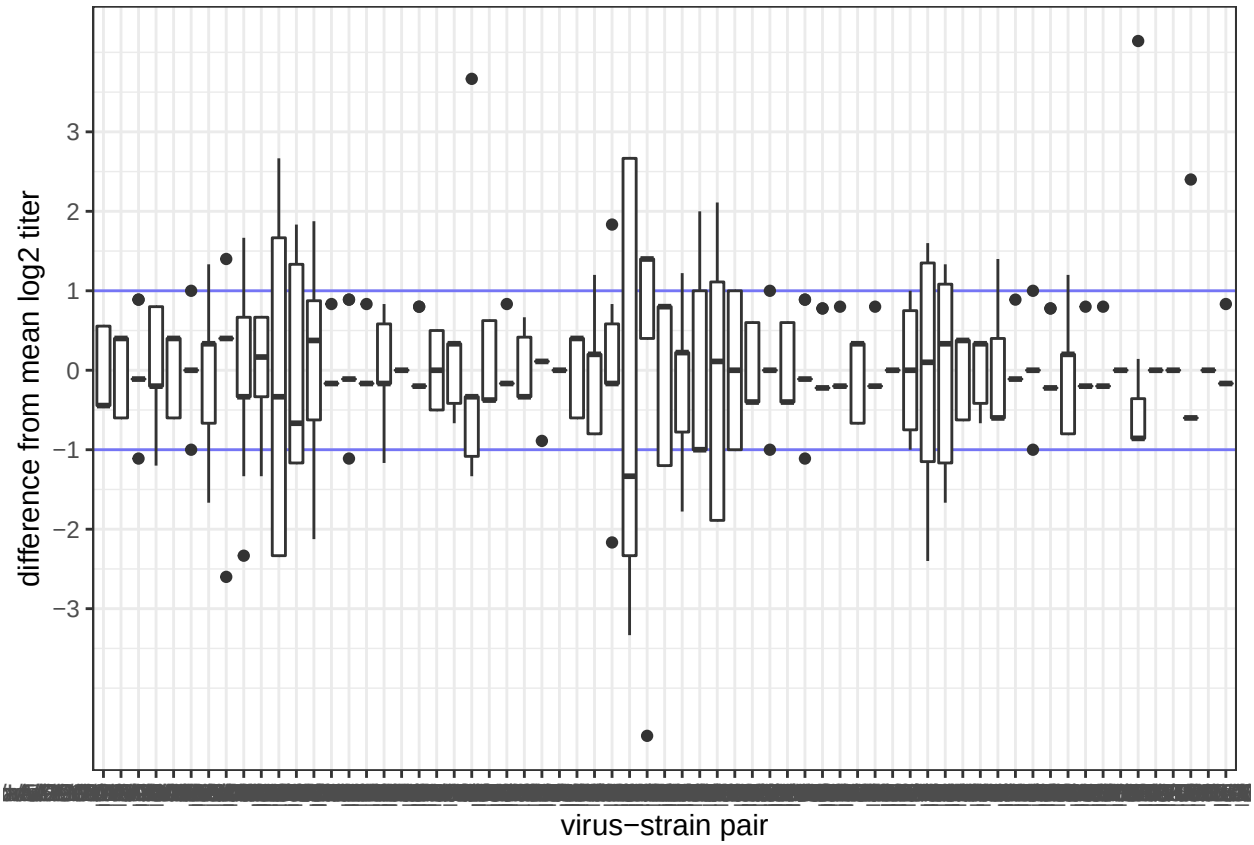
```
## `summarise()` has grouped output by 'virusStrain'. You can override using the
## `.groups` argument.
```

```
## Write a function to clean titers to numeric
clean_titers = function(titer_char){
  ## If the entry contains <, report half the lower limit of detection ("undetectable")
  ifelse(grepl("<", titer_char), as.numeric(gsub("<(\\d+)", "\\1", titer_char))/2, as.numeric(titer_cha
}

## Extract the repeat observations from the overall dataset
repeat_observations = merge(Bedford_data, repeat_observations, all.x = FALSE, all.y = TRUE) %>%
  select(virusStrain, serumIsolate, titer, n_replicates, source, virusYear, serumYear)
## Sort virus-strain pairs by temporal distance, dy
ordered_levels = repeat_observations %>%
  mutate(dy = abs(virusYear-serumYear)) %>%
  arrange(dy) %>%
  select(virusStrain, serumIsolate, dy) %>%
  distinct()

## Create virus-strain ids, ordered by temporal distance
repeat_observations %>%
mutate(id = factor(paste0(virusStrain, '_', serumIsolate), levels = paste0(ordered_levels$virusStrain,
mutate(is_undetectable = grepl("<", titer),
         titer = clean_titers(titer),
         logtiter = log2(titer/10)) %>%
  group_by(id) %>%
  mutate(sd = sd(logtiter),
         nlogtiter = logtiter-mean(logtiter)) %>%
  ungroup() %>%
  ## Plot observed log2 titers within each group
  ggplot() +
  geom_hline(aes(yintercept = -1), color = 'blue', alpha = .5)+
  geom_hline(aes(yintercept = 1), color = 'blue', alpha = .5)+
  geom_boxplot(aes(x = id, y = nlogtiter)) +
  xlab('virus-strain pair')+
  ylab('difference from mean log2 titer') +
  scale_y_continuous(breaks = -3:3)
```

```
## Warning in ifelse(grepl("<", titer_char), as.numeric(gsub("<(\\d+)", "\\1", :
## NAs introduced by coercion
```

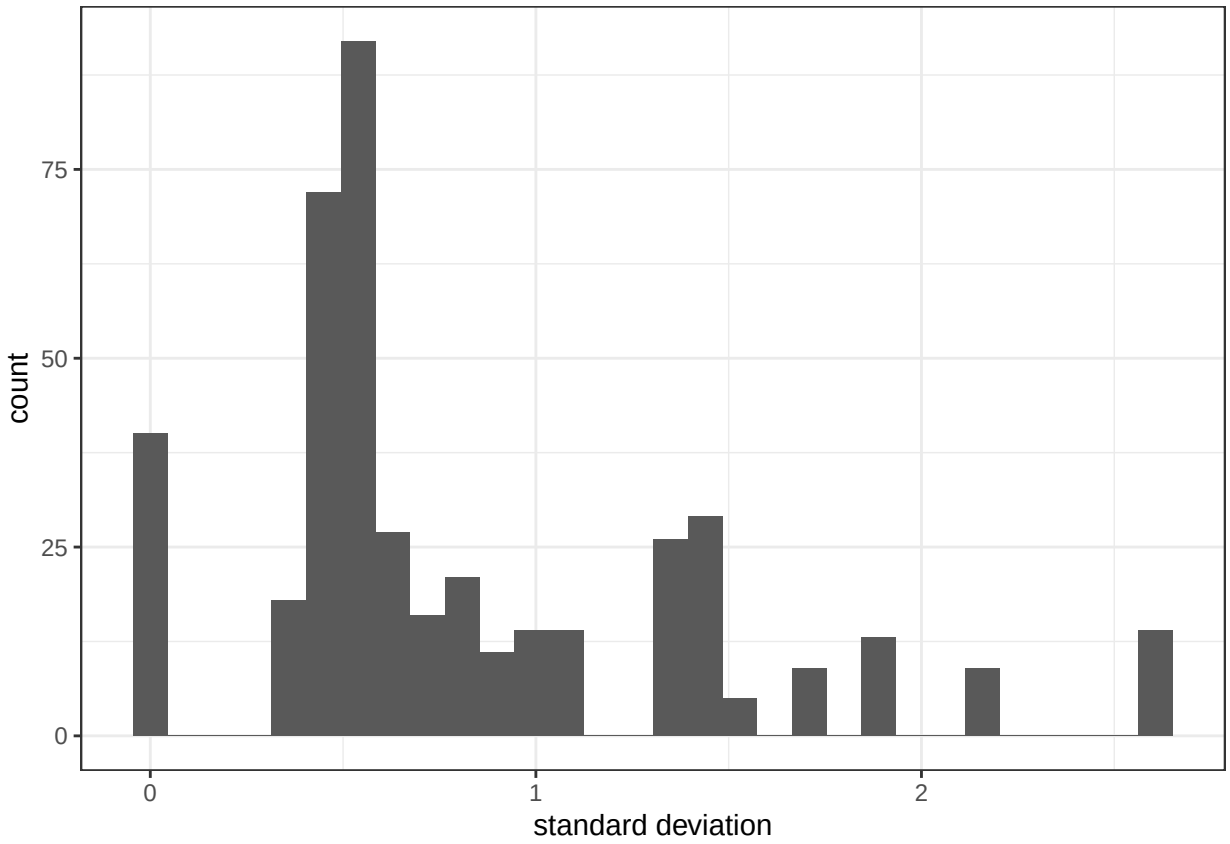



Overall, it looks like log2 titer measured in individual ferrets infected by the same virus, and then tested against the same strain vary somewhat, but most observations fall within 1 log2 unit (i.e. one two-fold dilution) of the group mean (most observations fall between the blue lines).

```
repeat_observations %>%
mutate(id = factor(paste0(virusStrain, '_', serumIsolate), levels = paste0(ordered_levels$virusStrain,
mutate(is_undetectable = grepl("<", titer),
         titer = clean_titers(titer),
         logtiter = log2(titer/10)) %>%
  group_by(id) %>%
  mutate(sd = sd(logtiter)) %>%
  ungroup() %>%
  ggplot() +
  geom_histogram(aes(x = sd))+
  xlab('standard deviation')
```

```
## Warning in ifelse(grepl("<", titer_char), as.numeric(gsub("<(\\d+)", "\\1", :
## NAs introduced by coercion
```

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




Within each virus-strain pair, the standard deviation of individual log2 titer obsercations is usually less than 1 log2 unit, indicating that we expect 95% of the observations in that group to fall within 2 log2 units (<4-fold difference in titer) from the group mean.