Standards Analysis

Sean Nguyen

```
library(tidyverse)
library(cowplot)
library(broom)
```

Import data

```
setwd("~/GitHub/time-course/data")

V2standards <- "day4_V2_std_reanalysis-ExperimentSummary.csv"
v2std <- read_csv(V2standards)
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   particle_size = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
v2std
```

```
## # A tibble: 1,000 × 73
##    particle_size std_after_125_4_1_00 std_after_125_4_1_01
##            <dbl>                <int>                <int>
## 1            0.5                    0                    0
## 2            1.5                    0                    0
## 3            2.5                    0                    0
## 4            3.5                    0                    0
## 5            4.5                    0                    0
## 6            5.5                    0                    0
## 7            6.5                    0                    0
## 8            7.5                    0                    0
## 9            8.5                    0                    0
## 10           9.5                    0                    0
## # ... with 990 more rows, and 70 more variables:
## #   std_after_125_4_1_02 <int>, std_after_125_4_2_00 <int>,
## #   std_after_125_4_2_01 <int>, std_after_125_4_2_02 <int>,
## #   std_before_125_4_1_00 <int>, std_before_125_4_1_01 <int>,
## #   std_before_125_4_1_02 <int>, std_before_125_4_2_00 <int>,
## #   std_before_125_4_2_01 <int>, std_before_125_4_2_02 <int>,
## #   std_after_125_5_1_00 <int>, std_after_125_5_1_01 <int>,
## #   std_after_125_5_1_02 <int>, std_after_125_5_2_00 <int>,
## #   std_after_125_5_2_01 <int>, std_after_125_5_2_02 <int>,
## #   std_before_125_5_1_00 <int>, std_before_125_5_1_01 <int>,
## #   std_before_125_5_1_02 <int>, std_before_125_5_2_00 <int>,
## #   std_before_125_5_2_01 <int>, std_before_125_5_2_02 <int>,
## #   std_after_125_1_1_00 <int>, std_after_125_1_1_01 <int>,
## #   std_after_125_1_1_02 <int>, std_after_125_1_2_00 <int>,
## #   std_after_125_1_2_01 <int>, std_after_125_1_2_02 <int>,
```

```
## #   std_before_125_1_1_00 <int>, std_before_125_1_1_01 <int>,
## #   std_before_125_1_1_02 <int>, std_before_125_1_2_00 <int>,
## #   std_before_125_1_2_01 <int>, std_before_125_1_2_02 <int>,
## #   std_after_125_6_1_00 <int>, std_after_125_6_1_01 <int>,
## #   std_after_125_6_1_02 <int>, std_after_125_6_2_00 <int>,
## #   std_after_125_6_2_01 <int>, std_after_125_6_2_02 <int>,
## #   std_before_125_6_1_00 <int>, std_before_125_6_1_01 <int>,
## #   std_before_125_6_1_02 <int>, std_before_125_6_2_00 <int>,
## #   std_before_125_6_2_01 <int>, std_before_125_6_2_02 <int>,
## #   std_after_125_2_1_00 <int>, std_after_125_2_1_01 <int>,
## #   std_after_125_2_1_02 <int>, std_after_125_2_2_00 <int>,
## #   std_after_125_2_2_01 <int>, std_after_125_2_2_02 <int>,
## #   std_after_125_3_1_00 <int>, std_after_125_3_1_01 <int>,
## #   std_after_125_3_1_02 <int>, std_after_125_3_2_00 <int>,
## #   std_after_125_3_2_01 <int>, std_after_125_3_2_02 <int>,
## #   std_before_125_2_1_00 <int>, std_before_125_2_1_01 <int>,
## #   std_before_125_2_1_02 <int>, std_before_125_2_2_00 <int>,
## #   std_before_125_2_2_01 <int>, std_before_125_2_2_02 <int>,
## #   std_before_125_3_1_00 <int>, std_before_125_3_1_01 <int>,
## #   std_before_125_3_1_02 <int>, std_before_125_3_2_00 <int>,
## #   std_before_125_3_2_01 <int>, std_before_125_3_2_02 <int>
```

Convert data from 'wide' to 'long' format and factor columns to 'categorica' variables

Back Calculate

```
re_std2 <- re_std2 %>%
  mutate(True_Count=Dilution_factor*Count)

re_std2$Nano_day <-  factor(re_std2$Nano_day, levels=c('1','2','3','4','5','6'))
re_std2$When <- factor(re_std2$When, levels=c('before','after'))
```

Summarize three technical replicates

```
re_std3 <- re_std2 %>%
  group_by(particle_size,Sample_ID,When,Dilution_factor,Nano_day,Injection) %>%
  summarise( tech_N = length(True_Count),
             tech_mean = mean(True_Count),
             tech_sd = sd(True_Count),
             tech_se = tech_sd/sqrt(tech_N))
re_std3
```

```
## Source: local data frame [24,000 x 10]
## Groups: particle_size, Sample_ID, When, Dilution_factor, Nano_day [?]
##
##    particle_size Sample_ID   When Dilution_factor Nano_day Injection
##            <dbl>    <fctr> <fctr>           <dbl>   <fctr>    <fctr>
## 1            0.5       std before             125        1         1
## 2            0.5       std before             125        1         2
## 3            0.5       std before             125        2         1
## 4            0.5       std before             125        2         2
## 5            0.5       std before             125        3         1
```

```
## 6            0.5       std before             125        3         2
## 7            0.5       std before             125        4         1
## 8            0.5       std before             125        4         2
## 9            0.5       std before             125        5         1
## 10           0.5       std before             125        5         2
## # ... with 23,990 more rows, and 4 more variables: tech_N <int>,
## #   tech_mean <dbl>, tech_sd <dbl>, tech_se <dbl>
```

Summarize by injection

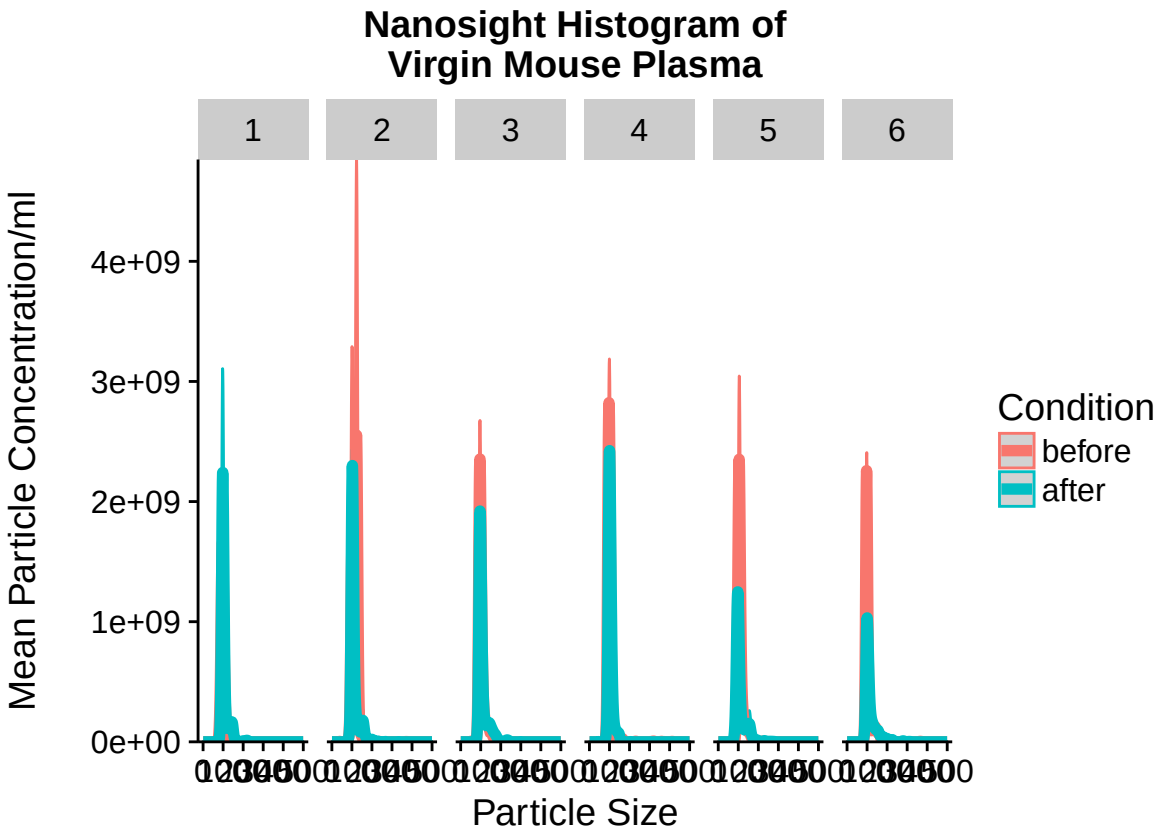
```
re_std4 <- re_std3 %>%
  group_by(Nano_day,When,particle_size) %>%
  summarise( inj_N = length(tech_mean),
             inj_mean = mean(tech_mean),
             inj_sd = sd(tech_mean),
             inj_se = inj_sd/sqrt(inj_N))
re_std4
```

```
## Source: local data frame [12,000 x 7]
## Groups: Nano_day, When [?]
##
##    Nano_day   When particle_size inj_N inj_mean inj_sd inj_se
##      <fctr> <fctr>         <dbl> <int>    <dbl>  <dbl>  <dbl>
## 1         1 before           0.5     2        0      0      0
## 2         1 before           1.5     2        0      0      0
## 3         1 before           2.5     2        0      0      0
## 4         1 before           3.5     2        0      0      0
## 5         1 before           4.5     2        0      0      0
## 6         1 before           5.5     2        0      0      0
## 7         1 before           6.5     2        0      0      0
## 8         1 before           7.5     2        0      0      0
## 9         1 before           8.5     2        0      0      0
## 10        1 before           9.5     2        0      0      0
## # ... with 11,990 more rows
```

Plot before and after plots, facet by experimental day

```
redo_std <- re_std4 %>%
  ggplot(aes(x=particle_size,y=inj_mean,color=When))+
  geom_ribbon(aes(ymin=inj_mean-inj_se, ymax=inj_mean+inj_se),alpha=0.2,fill = alpha('grey12', 0.2)) +
  geom_line(size=2) + xlim(0,500)+ #line size, x-axis scale
  scale_y_continuous(expand=c(0,0))+ #set bottom of graph
  xlab("Particle Size") + # X axis label
  ylab("\nMean Particle Concentration/ml\n") + # Y axis label
  ggtitle("Nanosight Histogram of\nVirgin Mouse Plasma")+ #title
  labs(color="Condition")+ #Label table title
  facet_grid(. ~ Nano_day)
redo_std
```

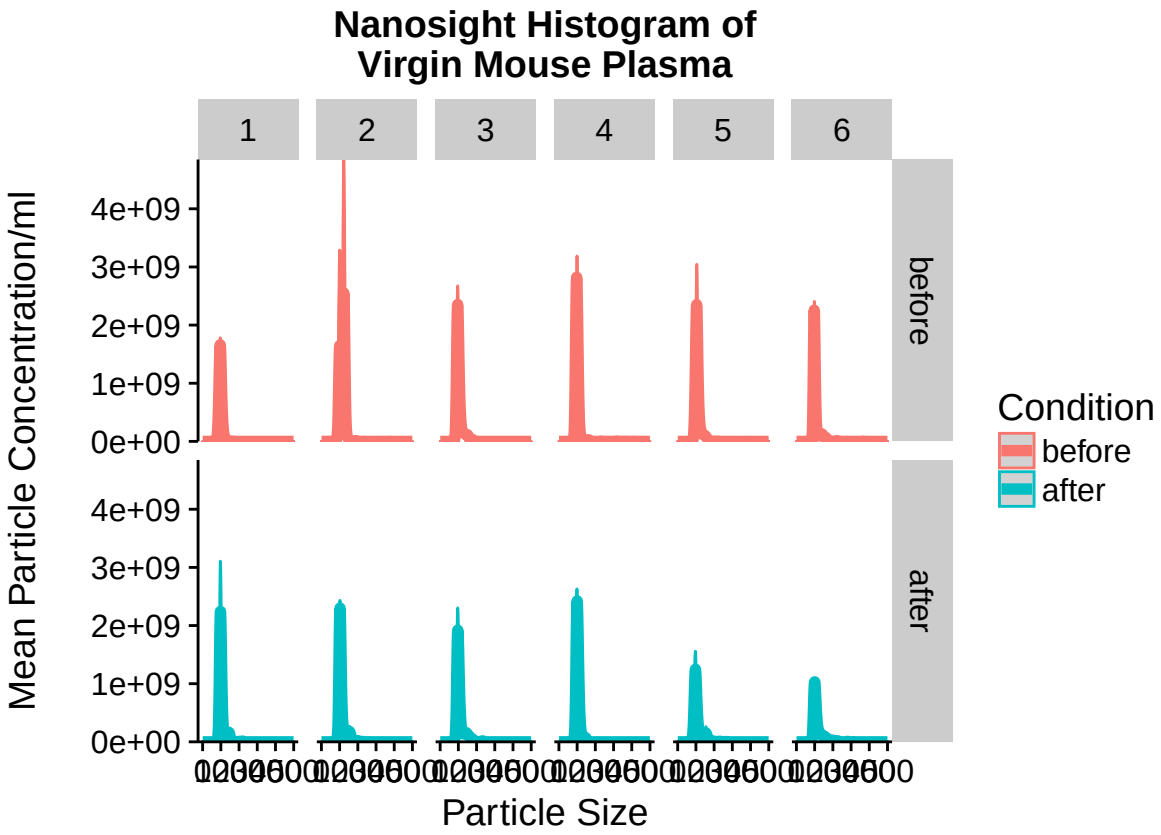
```
## Warning: Removed 1000 rows containing missing values (geom_path).
```



###Plot facet by when and experimental day

```
redo_std2 <- re_std4 %>%
  ggplot(aes(x=particle_size,y=inj_mean,color=When))+
  geom_ribbon(aes(ymin=inj_mean-inj_se, ymax=inj_mean+inj_se),alpha=0.2,fill = alpha('grey12', 0.2)) +
  geom_line(size=2) + xlim(0,500)+ #line size, x-axis scale
  scale_y_continuous(expand=c(0,0))+ #set bottom of graph
  xlab("Particle Size") + # X axis label
  ylab("\nMean Particle Concentration/ml\n") + # Y axis label
  ggtitle("Nanosight Histogram of\nVirgin Mouse Plasma")+ #title
  labs(color="Condition")+ #Label table title
  facet_grid(When ~ Nano_day)
redo_std2
```

```
## Warning: Removed 1000 rows containing missing values (geom_path).
```



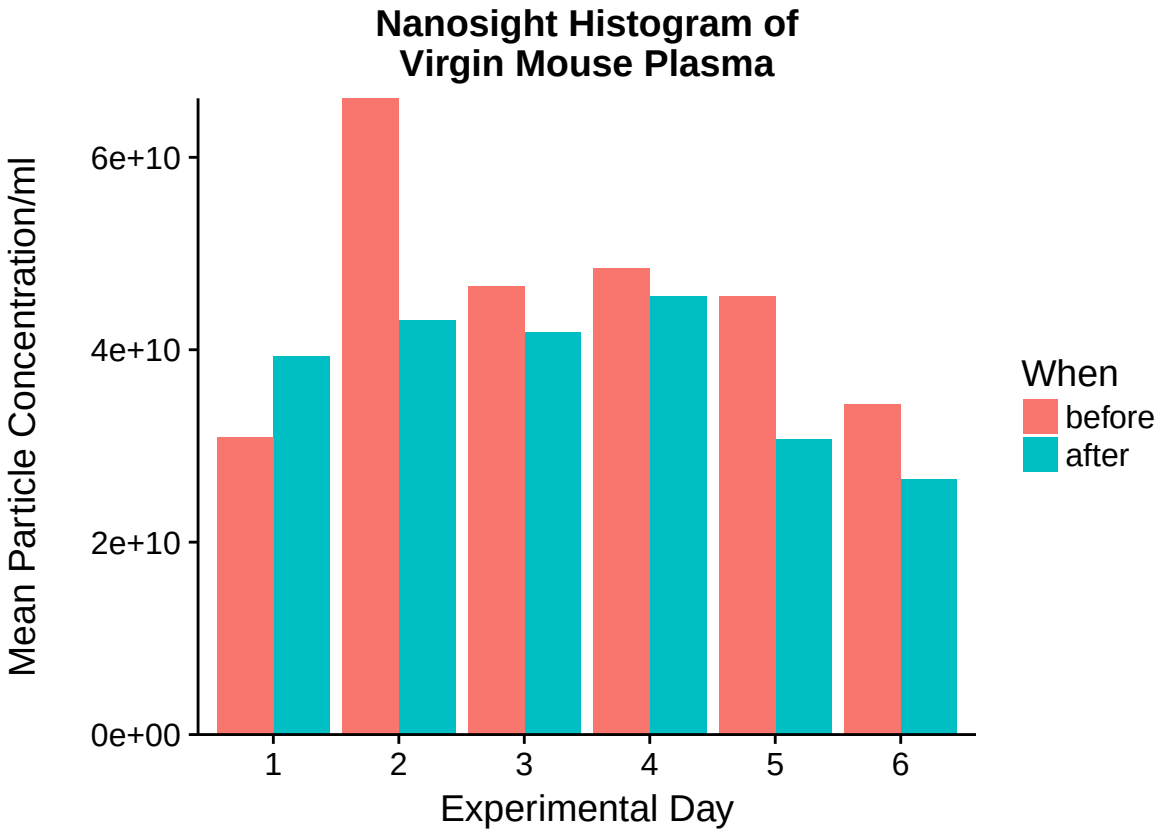
Particle concentrations from each experimental day

```
redo_std_df <- re_std4 %>%
  group_by(Nano_day,When) %>%
  summarise(total=sum(inj_mean))
redo_std_df
```

```
## Source: local data frame [12 x 3]
## Groups: Nano_day [?]
##
##    Nano_day   When       total
##       <fctr> <fctr>       <dbl>
## 1         1 before 30894192250
## 2         1  after 39336694688
## 3         2 before 66132971542
## 4         2  after 43118638104
## 5         3 before 46647311021
## 6         3  after 41797692063
## 7         4 before 48537507313
## 8         4  after 45586525625
## 9         5 before 45533836938
## 10        5  after 30708761458
## 11        6 before 34317033229
## 12        6  after 26543941000
```

Bar graph of particle concentrations

```
redo_std_df %>%
  ggplot(aes(x=Nano_day,y=total,fill=When))+
  geom_col(position="dodge")+
  scale_y_continuous(expand=c(0,0))+ #set bottom of graph
  xlab("Experimental Day") + # X axis label
  ylab("\nMean Particle Concentration/ml\n") + # Y axis label
  ggtitle("Nanosight Histogram of\nVirgin Mouse Plasma")+ #title
  labs(color="When") #Label table title
```



###Intraassay variability

```
Intra.assay_cv <- redo_std_df %>%
  group_by(Nano_day) %>%
  summarise(Day_N = length(total),
            Day_mean = mean(total),
            Day_sd = sd(total),
            Day_se = Day_sd/sqrt(Day_N),
           Day_cv = Day_sd/Day_mean )
Intra.assay_cv
```

```
## # A tibble: 6 × 6
##   Nano_day Day_N    Day_mean      Day_sd      Day_se    Day_cv
##      <fctr> <int>       <dbl>       <dbl>       <dbl>     <dbl>
## 1        1     2 35115443469  5969750724  4221251219 0.1700036
## 2        2     2 54625804823 16273591238 11507166719 0.2979103
```

```
## 3        3     2 44222501542  3429198452  2424809479 0.0775442
## 4        4     2 47062016469  2086659162  1475490844 0.0443385
## 5        5     2 38121299198 10482911403  7412537740 0.2749883
## 6        6     2 30430487115  5496406226  3886546115 0.1806217
```

Inter assay variability

```
Inter.assay <- Intra.assay_cv %>%
 summarise(Exp_N = length(Day_mean),
             Exp_mean = mean(Day_mean),
             Exp_sd = sd(Day_mean),
             Exp_se = Exp_sd/sqrt(Exp_N),
            Exp_cv = Exp_sd/Exp_mean )
Inter.assay
```

```
## # A tibble: 1 × 5
##   Exp_N    Exp_mean     Exp_sd     Exp_se    Exp_cv
##   <int>       <dbl>      <dbl>      <dbl>     <dbl>
## 1     6 41596258769 8778285668 3583720117 0.2110355
```