C14

Denise Ong

4/2/2020

```
library(dplyr)
library(tidyverse)
library(ggplot2)
library(broom)
```

Denise edits on 11 May 2020

1. Read C14 data

```
C14_DPM <- readxl::read_excel("TAN1810 Adriana samples complete _final_AGR_1.0.xlsx", sheet = "Sheet1")
  dplyr::select (Cycle, EXP, STN, DEPTH, SAMPLE, `Vial code`, `sorting population`, `Cells sorted`, DPM
  # Can add more variables latter
  # station and depth for adding updated DIC values
     rename(cycle = Cycle,
            exp = EXP,
            station = STN,
            depth = DEPTH,
            sample = SAMPLE,
            vial = `Vial code`,
            population = `sorting population`,
            cells_sorted = `Cells sorted`,
            dpm = DPM1,
            SA = `SA in DPM (aproximation)`) %>%
    filter(population != "Nano")

C14_DPM
```

```
## # A tibble: 385 x 10
##    cycle   exp station depth sample vial  population cells_sorted   dpm       SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>             <dbl> <dbl>    <dbl>
##  1     1     1      15    12 SUR    A     Pico               2000    62   1.20e7
##  2     1     1      15    12 SUR    A     Pico               4000    93   1.20e7
##  3     1     1      15    12 SUR    A     Pico              10000   179   1.20e7
##  4     1     1      15    12 SUR    A     Syn                2000    75   1.20e7
##  5     1     1      15    12 SUR    A     Syn                4000    48   1.20e7
##  6     1     1      15    12 SUR    A     Syn               10000   140   1.20e7
##  7     1     1      15    12 SUR    B     Pico               2000    61   1.20e7
##  8     1     1      15    12 SUR    B     Pico               4000    90   1.20e7
##  9     1     1      15    12 SUR    B     Pico              10000   184   1.20e7
## 10     1     1      15    12 SUR    B     Syn                2000    54   1.20e7
## # ... with 375 more rows
```

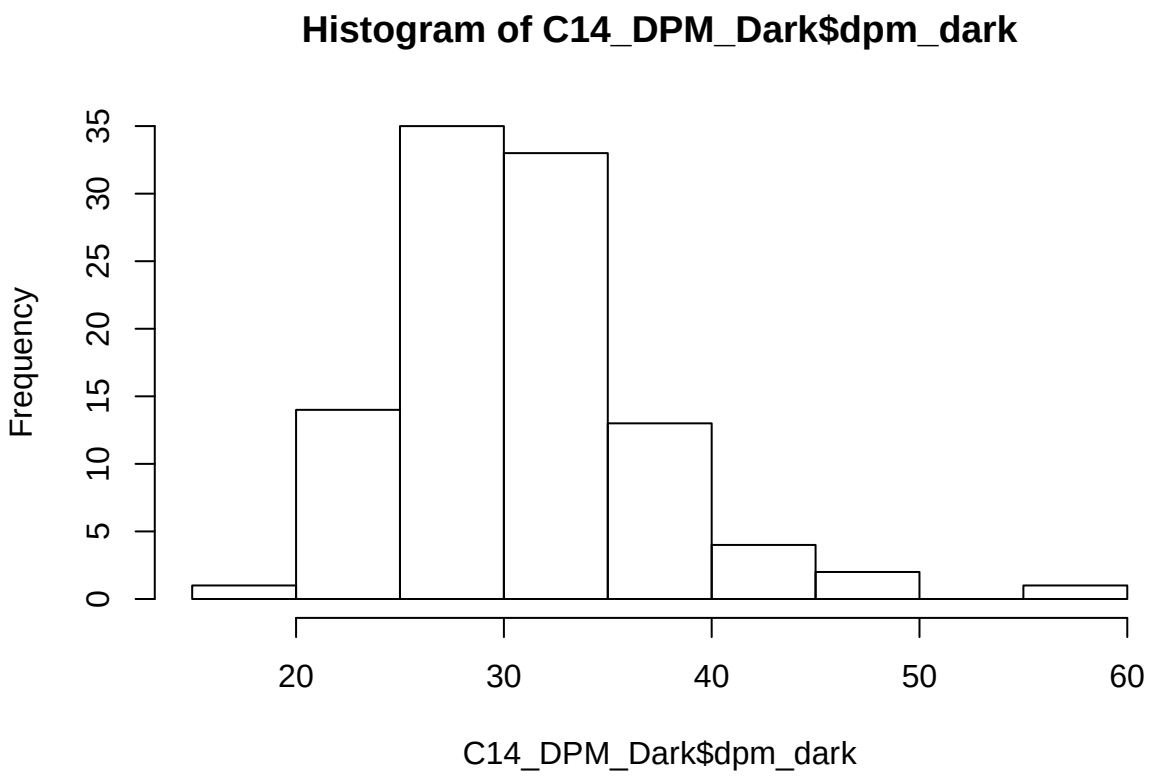
Substract the dark DPM

Join tables based on common variables

```
 C14_DPM_Dark <- C14_DPM %>%
  filter(vial == "D") %>%
  rename(dpm_dark = dpm) %>%
  dplyr::select(-vial) # Remove the vial column
C14_DPM_Dark
```

```
## # A tibble: 103 x 9
##    cycle   exp station depth sample population cells_sorted dpm_dark        SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr>              <dbl>    <dbl>     <dbl>
##  1     1     1      15    12 SUR    Pico                2000       28 12000000.
##  2     1     1      15    12 SUR    Pico                4000       28 12000000.
##  3     1     1      15    12 SUR    Pico               10000       35 12000000.
##  4     1     1      15    12 SUR    Syn                 2000       26 12000000.
##  5     1     1      15    12 SUR    Syn                 4000       30 12000000.
##  6     1     1      15    12 SUR    Syn                10000       37 12000000.
##  7     1     2      24    12 SUR    Pico                2000       25 12000000.
##  8     1     2      24    12 SUR    Pico                4000       28 12000000.
##  9     1     2      24    12 SUR    Pico               10000       29 12000000.
## 10     1     2      24    12 SUR    Syn                 2000       29 12000000.
## # ... with 93 more rows
```

```
# Check the values of dpm dark with histogram. The values are normally distributed, concentrated around
hist(C14_DPM_Dark$dpm_dark)
```




```
mean(C14_DPM_Dark$dpm_dark)
```

```
## [1] 31.06796
```

```
C14_DPM_corrected <- left_join(C14_DPM,C14_DPM_Dark ) %>%
  filter (vial != "D") %>%
  # DV - The next line replace dpm_dark to average dpm_dark if no value. Checked with histogram above.
  mutate(dpm_dark = case_when (is.na(dpm_dark) ~ mean(dpm_dark, na.rm = TRUE),
                               TRUE ~ dpm_dark))  %>%
  # DV - The next line set dpm_correct to zero if negative, as negative values are all close to zero.
  mutate(dpm_corrected = case_when (dpm >= dpm_dark ~ dpm - dpm_dark,
                                    TRUE ~0)) %>%
  filter(!(exp == 2 & sample == "DCM" & vial == "A")) # I removed this row as there was only one value
  #filter(!(exp == 6 & sample == 'SUR' & vial == 'C' & population == 'Pico' & cells_sorted == 2000)) %>
  #filter(!(exp == 6 & sample == 'SUR' & vial == 'C' & population == 'Syn' & cells_sorted == 2000)) # T

C14_DPM_corrected
```

```
## # A tibble: 279 x 12
##    cycle   exp station depth sample vial  population cells_sorted   dpm     SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>             <dbl> <dbl>  <dbl>
##  1     1     1      15    12 SUR    A     Pico               2000    62 1.20e7
##  2     1     1      15    12 SUR    A     Pico               4000    93 1.20e7
##  3     1     1      15    12 SUR    A     Pico              10000   179 1.20e7
##  4     1     1      15    12 SUR    A     Syn                2000    75 1.20e7
##  5     1     1      15    12 SUR    A     Syn                4000    48 1.20e7
##  6     1     1      15    12 SUR    A     Syn               10000   140 1.20e7
##  7     1     1      15    12 SUR    B     Pico               2000    61 1.20e7
##  8     1     1      15    12 SUR    B     Pico               4000    90 1.20e7
##  9     1     1      15    12 SUR    B     Pico              10000   184 1.20e7
## 10     1     1      15    12 SUR    B     Syn                2000    54 1.20e7
## # ... with 269 more rows, and 2 more variables: dpm_dark <dbl>,
## #   dpm_corrected <dbl>
```

2. Include DIC

Import DIC data from Andres. I will use this data to join with the DPM model output. The corrected data frame will have the DIC. The data will join based on the common columns: station and depth.

Create DIC table

```
DIC_data <- readxl::read_excel("Chla NPP Raw TAN1810 Sept.xlsx", sheet = "Compiled TAN1810 NPP data", sk
  select(c(1:34), -c(1,27))

# select for station, depth, DIC, SA. Change all columns to integer to join with DPM output.
DIC_data_corrected <- select(DIC_data, Station, `Depth m`, DIC) %>%
  unique() %>% # delete repeated rows
  rename(station = Station,
         depth = `Depth m`) %>%
  filter(depth != "35/40?") %>% # filter out character to change to integer
  mutate_if(is.character, as.double) # change to double (same format as DPM dataset)

DIC_data_corrected
```

```
## # A tibble: 100 x 3
##    station depth   DIC
##      <dbl> <dbl> <dbl>
```

```
##  1      15     5  25.9
##  2      15    12  25.9
##  3      24     5  25.9
##  4      24    12  25.7
##  5      24    20  25.9
##  6      24    30  25.9
##  7      24    40  26.2
##  8      24    50  26.1
##  9      39     5  25.7
## 10      39    12  25.8
## # ... with 90 more rows
```

3. Calculations for Nano (average)

Data points to check for Nano calculations: 1. exp 7: SUR and DCM. D is higher than A, B, C. 2. exp 1: SUR. D is much higher than C 3. exp 5: SUR. No D vial, using average dark value. However, there are very high dark values that skew the mean. 4. exp 6: SUR. Two values for vial A 5. exp 8: DCM values are low. 6. to calculate lm for Nano (1000 and 2000 cells counted): exp 7: SUR and DCM exp 8: SUR

```
C14_DPM_nano <- readxl::read_excel("TAN1810 Adriana samples complete _final_AGR_1.0.xlsx", sheet = "She
  dplyr::select (Cycle, EXP, STN, DEPTH, SAMPLE, `Vial code`, `sorting population`, `Cells sorted`, DPM
  # Can add more variables latter
  # station and depth for adding updated DIC values
     rename(cycle = Cycle,
            exp = EXP,
            station = STN,
            depth = DEPTH,
            sample = SAMPLE,
            vial = `Vial code`,
            population = `sorting population`,
            cells_sorted = `Cells sorted`,
            dpm = DPM1,
            SA = `SA in DPM (aproximation)`) %>%
    filter(population == "Nano") %>%
    filter(!(exp == 2 & sample == "DCM" & vial == "A"))
C14_DPM_nano
```

```
## # A tibble: 74 x 10
##    cycle   exp station depth sample vial  population cells_sorted   dpm       SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>             <dbl> <dbl>    <dbl>
##  1     1     1      15    12 SUR    A     Nano               1000   926   1.20e7
##  2     1     1      15    12 SUR    B     Nano               1000  1162   1.20e7
##  3     1     1      15    12 SUR    C     Nano               1000    48   1.20e7
##  4     1     1      15    12 SUR    D     Nano               1000   905   1.20e7
##  5     1     2      24    12 SUR    A     Nano               1000  1814   1.20e7
##  6     1     2      24    12 SUR    C     Nano               1000  1832   1.20e7
##  7     1     2      24    12 SUR    D     Nano               1000    40   1.20e7
##  8     2     3     137    12 SUR    A     Nano               1000  1504   1.20e7
##  9     2     3     137    12 SUR    B     Nano               1000  1444   1.20e7
## 10     2     3     137    12 SUR    C     Nano               1000  1829   1.20e7
## # ... with 64 more rows
```

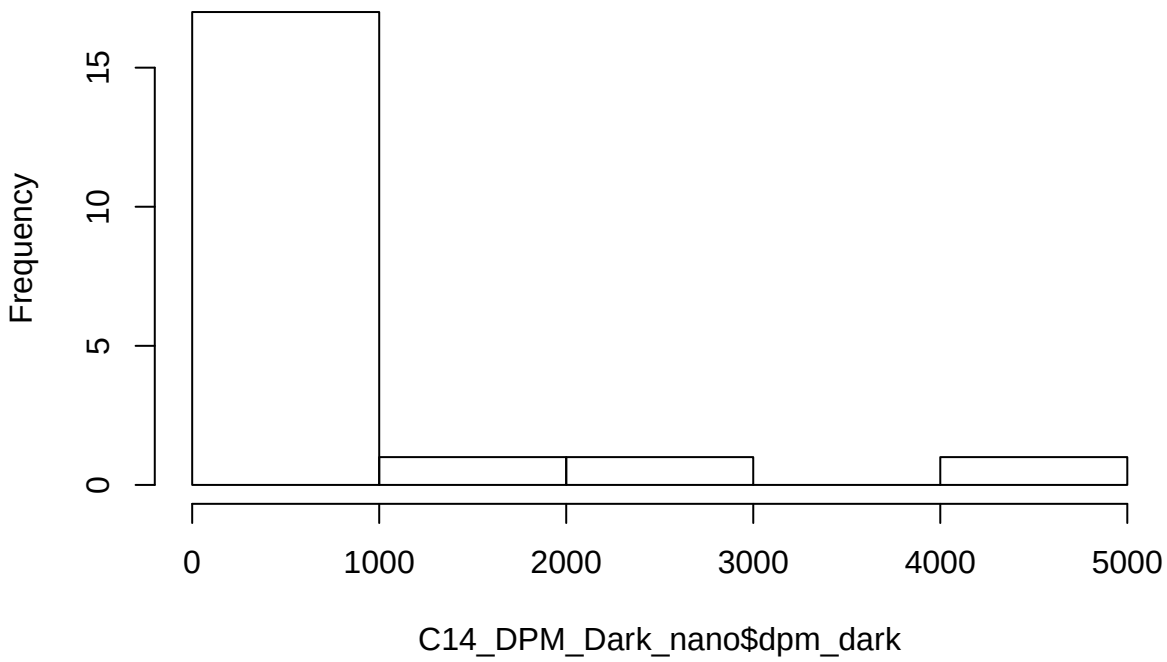
Substract the dark DPM

Join tables based on common variables

```
 C14_DPM_Dark_nano <- C14_DPM_nano %>%
  filter(vial == "D") %>%
  rename(dpm_dark = dpm) %>%
  dplyr::select(-vial)  # Remove the vial column
C14_DPM_Dark_nano
```

```
## # A tibble: 20 x 9
##    cycle   exp station depth sample population cells_sorted dpm_dark        SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr>              <dbl>    <dbl>     <dbl>
##  1     1     1      15    12 SUR    Nano                1000      905 12000000.
##  2     1     2      24    12 SUR    Nano                1000       40 12000000.
##  3     2     3     137    12 SUR    Nano                1000       40 12000000.
##  4     2     3     137    40 DCM    Nano                1000       39 12000000.
##  5     2     4     150    40 DCM    Nano                1000       58 12000000.
##  6     2     5     176    12 SUR    Nano                1000     1292 12000000.
##  7     2     6     188    12 SUR    Nano                1000       37 12000000.
##  8     3     7     207    12 SUR    Nano                1000       53 12000000.
##  9     3     7     207    12 SUR    Nano                2000       92 12000000.
## 10     3     7     207    25 DCM    Nano                1000     2437 12000000.
## 11     3     7     207    25 DCM    Nano                2000     4847 12000000.
## 12     3     8     223    12 SUR    Nano                1000       28 12000000.
## 13     3     8     223    12 SUR    Nano                2000       34 12000000.
## 14     3     8     223    40 DCM    Nano                1000       37 12000000.
## 15     4     9     266    12 SUR    Nano                1000       46 12000000.
## 16     4     9     266    30 DCM    Nano                1000       31 12000000.
## 17     4    10     283    12 SUR    Nano                1000       48 12000000.
## 18     4    10     283    40 DCM    Nano                1000       35 12000000.
## 19     5    11     324    70 DCM    Nano                1000       26 12000000.
## 20     5    11     324    12 SUR    Nano                1000       37 12000000.
```

```
# Check the values of dpm dark with histogram.
hist(C14_DPM_Dark_nano$dpm_dark)
```




```
mean(C14_DPM_Dark_nano$dpm_dark)
```

```
## [1] 508.1
```

```
C14_DPM_corrected_nano <- left_join(C14_DPM_nano,C14_DPM_Dark_nano) %>%
  filter (vial != "D") %>%
  # DV - The next line replace dpm_dark to average dpm_dark if no value. Checked with histogram above.
  mutate(dpm_dark = case_when (is.na(dpm_dark) ~ mean(dpm_dark, na.rm = TRUE),
                               TRUE ~ dpm_dark))  %>%
  # DV - The next line set dpm_correct to zero if negative, as negative values are all close to zero.
  mutate(dpm_corrected = case_when (dpm >= dpm_dark ~ dpm - dpm_dark,
                                    TRUE ~0))
C14_DPM_corrected_nano
```

```
## # A tibble: 54 x 12
##    cycle   exp station depth sample vial  population cells_sorted   dpm     SA
##    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>             <dbl> <dbl>  <dbl>
##  1     1     1      15    12 SUR    A     Nano               1000   926 1.20e7
##  2     1     1      15    12 SUR    B     Nano               1000  1162 1.20e7
##  3     1     1      15    12 SUR    C     Nano               1000    48 1.20e7
##  4     1     2      24    12 SUR    A     Nano               1000  1814 1.20e7
##  5     1     2      24    12 SUR    C     Nano               1000  1832 1.20e7
##  6     2     3     137    12 SUR    A     Nano               1000  1504 1.20e7
##  7     2     3     137    12 SUR    B     Nano               1000  1444 1.20e7
##  8     2     3     137    12 SUR    C     Nano               1000  1829 1.20e7
##  9     2     3     137    40 DCM    B     Nano               1000   633 1.20e7
## 10     2     4     150    12 SUR    A     Nano               1000  3381 1.20e7
## # ... with 44 more rows, and 2 more variables: dpm_dark <dbl>,
## #   dpm_corrected <dbl>
```

Merge DIC/SA table and DPM output, calculate mean

```
# merge tables
pp_cal_nano <- left_join(C14_DPM_corrected_nano, DIC_data_corrected) %>%
  group_by(SA, population, cycle, DIC, exp, sample) %>%
  mutate(mean_dpm_corrected=mean(dpm_corrected)) %>% #calculate mean
  select(-vial, -dpm, -dpm_dark, -station, -depth, -dpm_corrected) %>%
  distinct() #remove duplicates

pp_cal_nano
```

```
## # A tibble: 21 x 8
## # Groups:   SA, population, cycle, DIC, exp, sample [18]
##    cycle   exp sample population cells_sorted        SA   DIC mean_dpm_corrected
##    <dbl> <dbl> <chr>  <chr>             <dbl>     <dbl> <dbl>              <dbl>
##  1     1     1 SUR    Nano               1000 12000000.  25.9               92.7
##  2     1     2 SUR    Nano               1000 12000000.  25.7             1783
##  3     2     3 SUR    Nano               1000 12000000.  25.9             1552.
##  4     2     3 DCM    Nano               1000 12000000.  25.9              594
##  5     2     4 SUR    Nano               1000 12000000.  26               3042.
##  6     2     4 DCM    Nano               1000 12000000.  25.8             1700.
##  7     2     5 SUR    Nano               1000 12000000.  26                 0
##  8     2     6 SUR    Nano               1000 12000000.  26               755.
##  9     3     7 SUR    Nano               1000 12000000.  25.6               0
## 10     3     7 SUR    Nano               2000 12000000.  25.6               0
## # ... with 11 more rows
```

Calculate PP value, based on Daniel's formula found here:

https://vaulot.netlify.com/2018/05/20/compute-primary-production-based-on-single-cell-c14-uptake/

```
pp_cal_nano <- mutate(pp_cal_nano, pp = DIC*mean_dpm_corrected*(1/(SA*24*1000))*10^9*1.05) %>% #divide
  ungroup() %>%
  filter(pp >= 0) # remove negative pp values

# add missing rows for data analysis later
pp_cal_nano <- complete(pp_cal_nano, nesting(cycle, exp, sample), population) %>%
  select(-SA, -DIC)%>%
  ungroup()

pp_cal_nano
```

```
## # A tibble: 21 x 7
##    cycle   exp sample population cells_sorted mean_dpm_corrected     pp
##    <dbl> <dbl> <chr>  <chr>             <dbl>              <dbl>  <dbl>
##  1     1     1 SUR    Nano               1000               92.7   8.75
##  2     1     2 SUR    Nano               1000             1783   167.
##  3     2     3 DCM    Nano               1000              594    56.1
##  4     2     3 SUR    Nano               1000             1552.  147.
##  5     2     4 DCM    Nano               1000             1700.  160.
##  6     2     4 SUR    Nano               1000             3042.  288.
##  7     2     5 SUR    Nano               1000                0     0
##  8     2     6 SUR    Nano               1000              755.   71.5
##  9     3     7 DCM    Nano               1000                0     0
## 10     3     7 DCM    Nano               2000                0     0
```

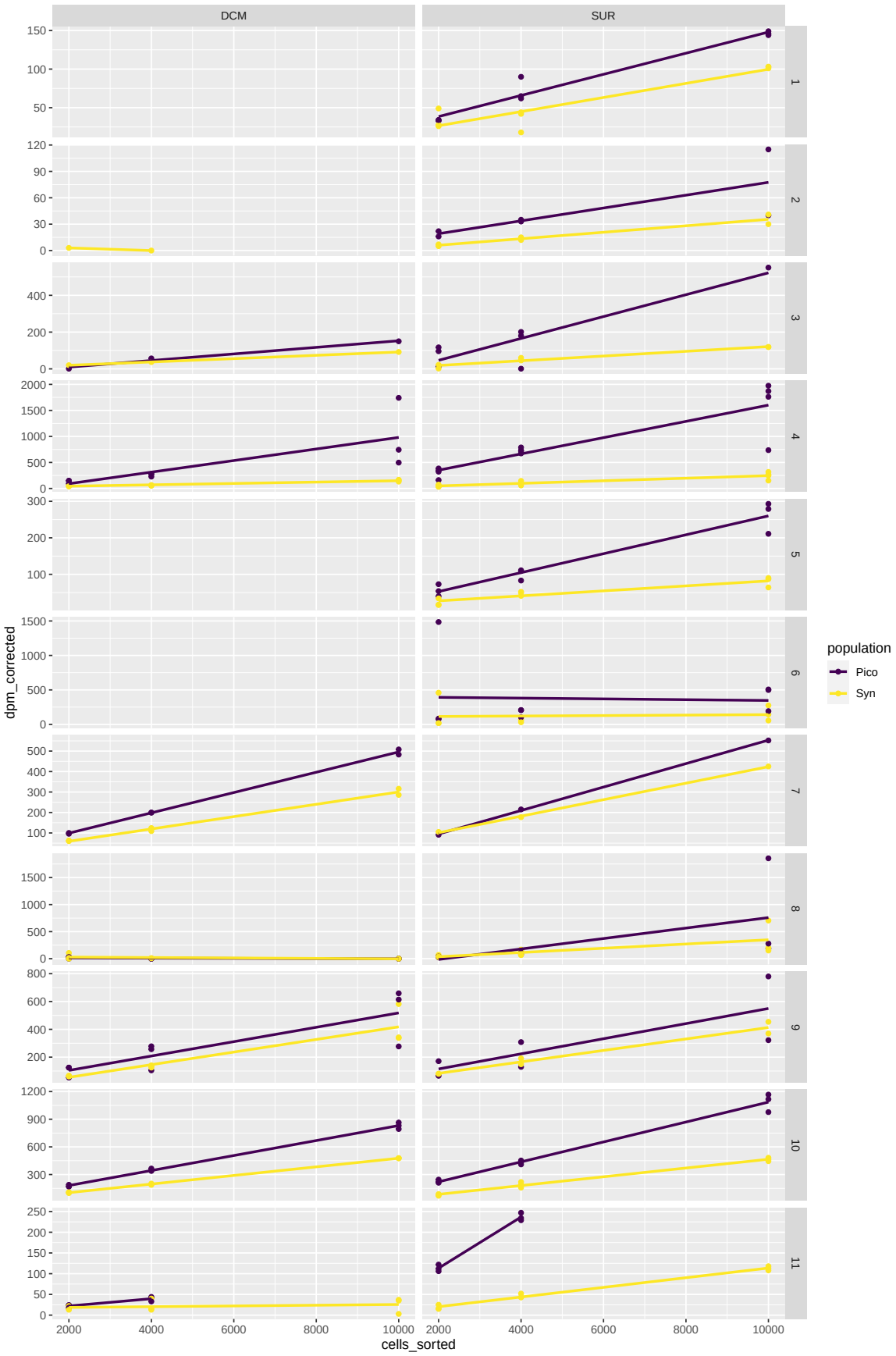
```
## # ... with 11 more rows
```

4. Compute based on Method 1 for pico and syn - Compute lm by grouping ABC together

Plots

Do plots for each group. One regression line based on the EXP, Sample and Populations

```
ggplot(data = C14_DPM_corrected, aes(x=cells_sorted, y=dpm_corrected, color=population)) +
 geom_point() + stat_smooth(method="lm", se=FALSE) +
 facet_grid(rows=vars(exp), cols=vars(sample), scales = "free_y") +
 scale_color_viridis_d()
```


DCM
SUR
1
2
3
4
5
6
7
8
9
10
11
dpm_corrected
cells_sorted
population
Pico
Syn

Data points to check: 1. exp 4: SUR: pico. repeated vial B. 2. exp 8: DCM is 0.

Do linear model

y = ax + b

See: https://cran.r-project.org/web/packages/broom/vignettes/broom_and_dplyr.html

```
C14_DPM_model_1 <- C14_DPM_corrected %>%
  group_by(SA, cycle, exp, station, depth, sample, population) %>%
  tidyr::nest() %>%
  mutate(
    fit = purrr::map(data, ~ lm(dpm_corrected ~ cells_sorted, data = .x)),
    tidied = purrr::map(fit, tidy)
  ) %>%
  unnest(tidied)

C14_DPM_model_output_1 <- C14_DPM_model_1 %>%
  select(exp:population, term, estimate) %>%
  pivot_wider(names_from="term", values_from ="estimate" ) %>%
  rename (slope = cells_sorted, intercept = `(Intercept)`)

C14_DPM_model_output_1
```

```
## # A tibble: 37 x 9
## # Groups:   cycle, exp, station, depth, sample, population, SA [37]
##    cycle        SA   exp station depth sample population intercept    slope
##    <dbl>     <dbl> <dbl>   <dbl> <dbl> <chr>  <chr>          <dbl>    <dbl>
##  1     1 12000000.     1      15    12 SUR    Pico           11.2   0.0137
##  2     1 12000000.     1      15    12 SUR    Syn             8.36  0.00914
##  3     1 12000000.     2      24    12 SUR    Pico            4.58  0.00730
##  4     1 12000000.     2      24    12 SUR    Syn            -1.31  0.00368
##  5     1 12000000.     2      24    40 DCM    Syn             6.00 -0.00150
##  6     2 12000000.     3     137    12 SUR    Pico          -72.3   0.0595
##  7     2 12000000.     3     137    12 SUR    Syn            -7.08  0.0128
##  8     2 12000000.     3     137    40 DCM    Pico          -26.2   0.0179
##  9     2 12000000.     3     137    40 DCM    Syn             1.46  0.00904
## 10     2 12000000.     4     150    12 SUR    Pico           39.3   0.156
## # ... with 27 more rows
```

Merge DIC/SA table and DPM output

```
# merge tables
pp_cal_1 <- left_join(C14_DPM_model_output_1, DIC_data_corrected)
pp_cal_1
```

```
## # A tibble: 37 x 10
## # Groups:   cycle, exp, station, depth, sample, population, SA [37]
##    cycle       SA   exp station depth sample population intercept    slope   DIC
##    <dbl>    <dbl> <dbl>   <dbl> <dbl> <chr>  <chr>          <dbl>    <dbl> <dbl>
##  1     1   1.20e7     1      15    12 SUR    Pico           11.2   0.0137   25.9
##  2     1   1.20e7     1      15    12 SUR    Syn             8.36  0.00914  25.9
##  3     1   1.20e7     2      24    12 SUR    Pico            4.58  0.00730  25.7
##  4     1   1.20e7     2      24    12 SUR    Syn            -1.31  0.00368  25.7
##  5     1   1.20e7     2      24    40 DCM    Syn             6.00 -0.00150  26.2
```

```
##  6     2   1.20e7     3     137    12 SUR    Pico          -72.3   0.0595   25.9
##  7     2   1.20e7     3     137    12 SUR    Syn            -7.08  0.0128   25.9
##  8     2   1.20e7     3     137    40 DCM    Pico          -26.2   0.0179   25.9
##  9     2   1.20e7     3     137    40 DCM    Syn             1.46  0.00904  25.9
## 10     2   1.20e7     4     150    12 SUR    Pico           39.3   0.156    26
## # ... with 27 more rows
```

Calculate PP value, based on Daniel's formula found here:

https://vaulot.netlify.com/2018/05/20/compute-primary-production-based-on-single-cell-c14-uptake/

```
pp_cal_1 <- mutate(pp_cal_1, pp = DIC*slope*(1/(SA*24))*10^9*1.05) %>%
  ungroup() %>%
  select( -c(station, depth)) %>%
  filter(pp >= 0) # remove negative pp values

# add missing rows for data analysis later
pp_cal_1 <- complete(pp_cal_1, nesting(cycle, exp, sample), population)

pp_cal_1
```

```
## # A tibble: 34 x 9
##    cycle   exp sample population        SA intercept   slope   DIC     pp
##    <dbl> <dbl> <chr>  <chr>          <dbl>     <dbl>   <dbl> <dbl>  <dbl>
##  1     1     1 SUR    Pico       12000000.      11.2  0.0137   25.9  1.29
##  2     1     1 SUR    Syn        12000000.       8.36 0.00914  25.9  0.863
##  3     1     2 SUR    Pico       12000000.       4.58 0.00730  25.7  0.684
##  4     1     2 SUR    Syn        12000000.      -1.31 0.00368  25.7  0.345
##  5     2     3 DCM    Pico       12000000.     -26.2  0.0179   25.9  1.69
##  6     2     3 DCM    Syn        12000000.       1.46 0.00904  25.9  0.853
##  7     2     3 SUR    Pico       12000000.     -72.3  0.0595   25.9  5.61
##  8     2     3 SUR    Syn        12000000.      -7.08 0.0128   25.9  1.21
##  9     2     4 DCM    Pico       12000000.    -131.   0.111    25.8 10.5
## 10     2     4 DCM    Syn        12000000.      17.6  0.0131   25.8  1.23
## # ... with 24 more rows
```

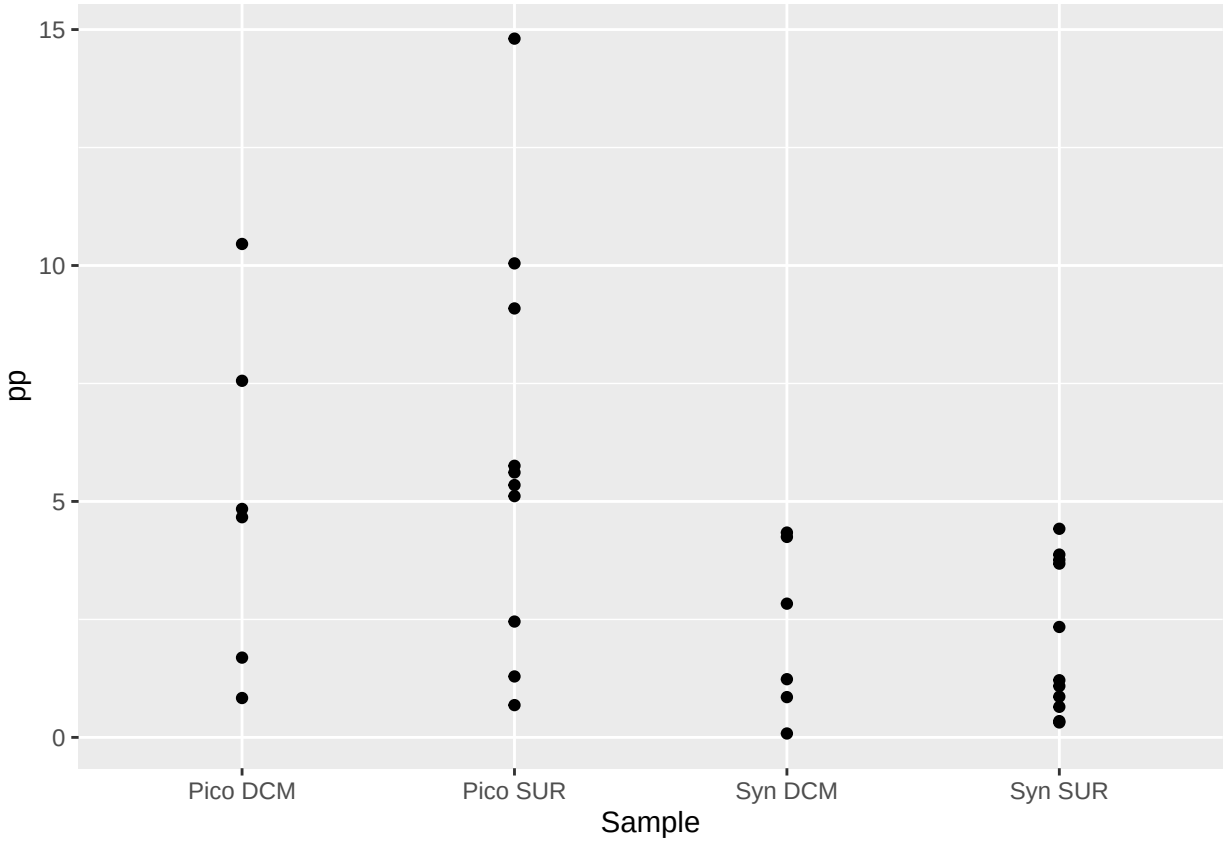
```
pp_cal_1_join<-full_join(pp_cal_1,pp_cal_nano) %>%
  select(-SA,-intercept, -slope, -DIC, -cells_sorted, -mean_dpm_corrected)%>%
  arrange(cycle, exp, sample, population)
pp_cal_1_join
```

```
## # A tibble: 55 x 5
##    cycle   exp sample population      pp
##    <dbl> <dbl> <chr>  <chr>        <dbl>
##  1     1     1 SUR    Nano         8.75
##  2     1     1 SUR    Pico         1.29
##  3     1     1 SUR    Syn          0.863
##  4     1     2 SUR    Nano       167.
##  5     1     2 SUR    Pico         0.684
##  6     1     2 SUR    Syn          0.345
##  7     2     3 DCM    Nano        56.1
##  8     2     3 DCM    Pico         1.69
##  9     2     3 DCM    Syn          0.853
## 10     2     3 SUR    Nano       147.
## # ... with 45 more rows
```

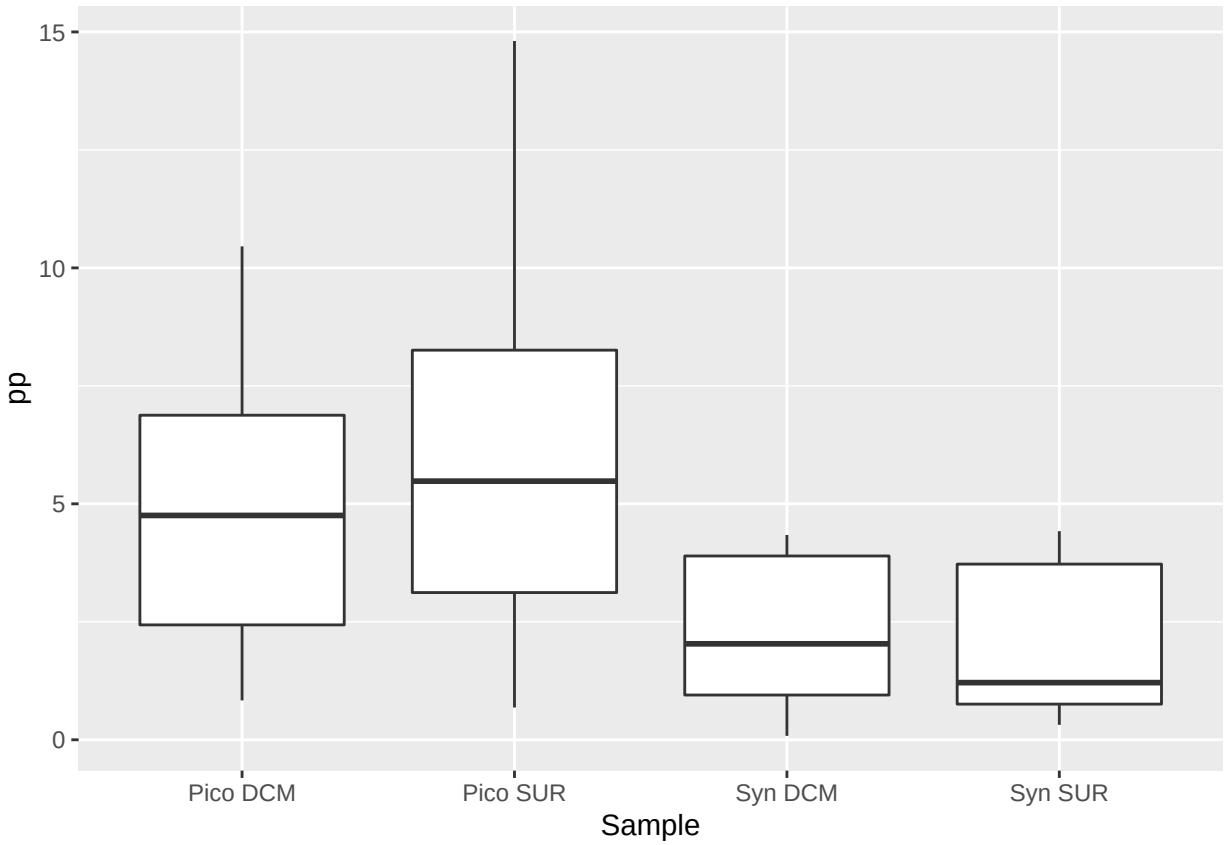
```r
pp_cal_1 %>%
  group_by(population, sample) %>%
  summarise(pp_mean = mean(pp, na.rm = TRUE))
```

```
## # A tibble: 4 x 3
## # Groups:   population [2]
##   population sample pp_mean
##   <chr>      <chr>    <dbl>
## 1 Pico       DCM       5.01
## 2 Pico       SUR       6.02
## 3 Syn        DCM       2.27
## 4 Syn        SUR       2.05
```

```r
pp_cal_1 %>%
  ggplot() +
  geom_point(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample")
```



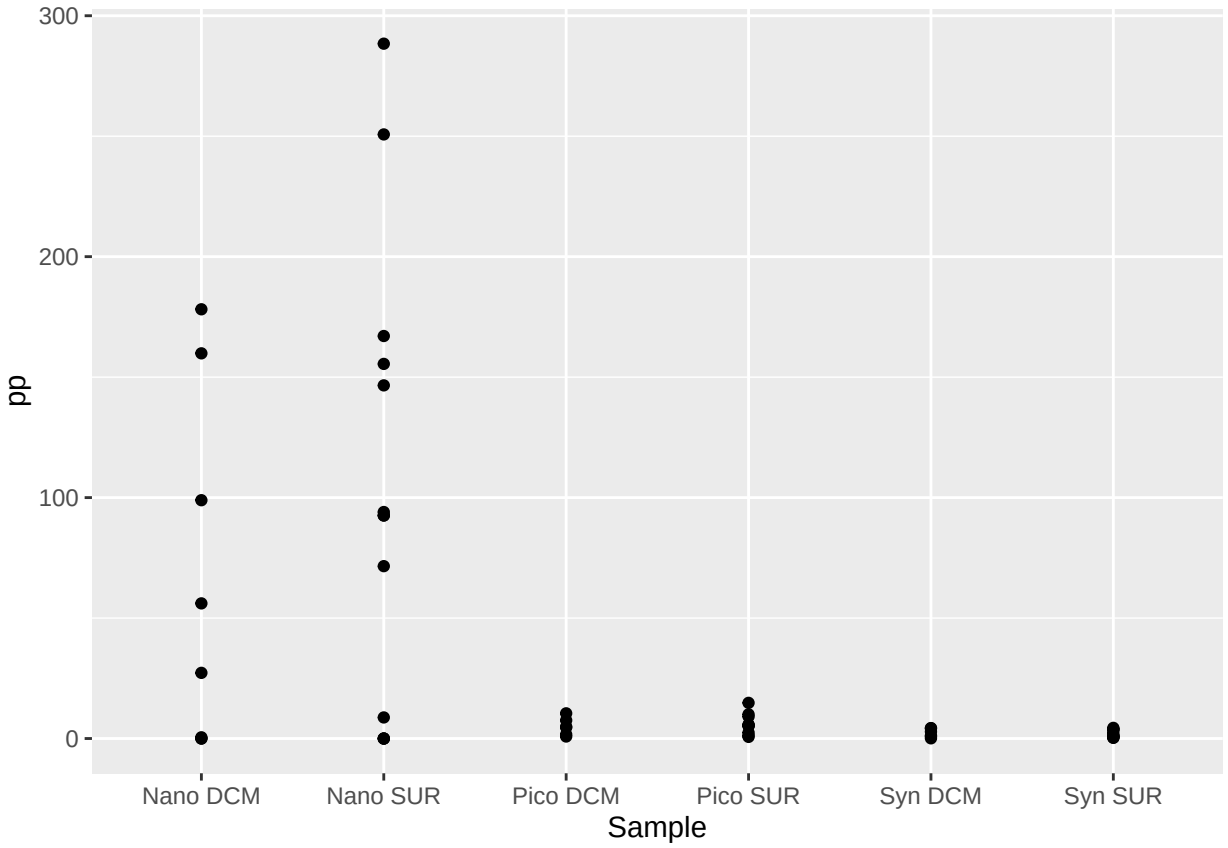

```r
pp_cal_1 %>%
  ggplot() +
  geom_boxplot(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample")
```



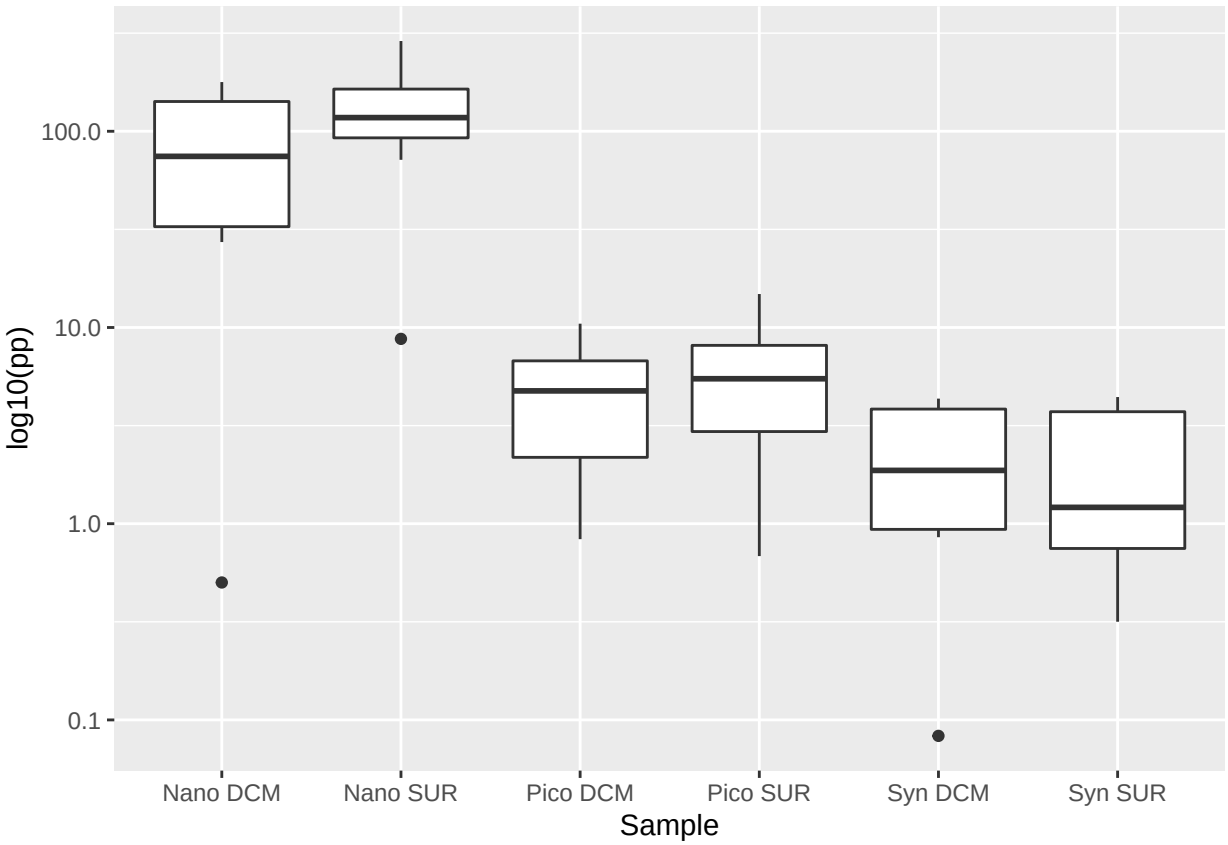

```
pp_cal_1_join %>%
  group_by(population, sample) %>%
  summarise(pp_mean = mean(pp, na.rm = TRUE))
```

```
## # A tibble: 6 x 3
## # Groups:   population [3]
##   population sample pp_mean
##   <chr>      <chr>    <dbl>
## 1 Nano       DCM       65.1
## 2 Nano       SUR      105.
## 3 Pico       DCM        5.01
## 4 Pico       SUR        6.02
## 5 Syn        DCM        2.27
## 6 Syn        SUR        2.05
```

```
pp_cal_1_join %>%
  ggplot() +
  geom_point(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample")
```




```
pp_cal_1_join %>%
  ggplot() +
  geom_boxplot(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample") +
  scale_y_continuous(trans='log10')+
  ylab("log10(pp)")
```

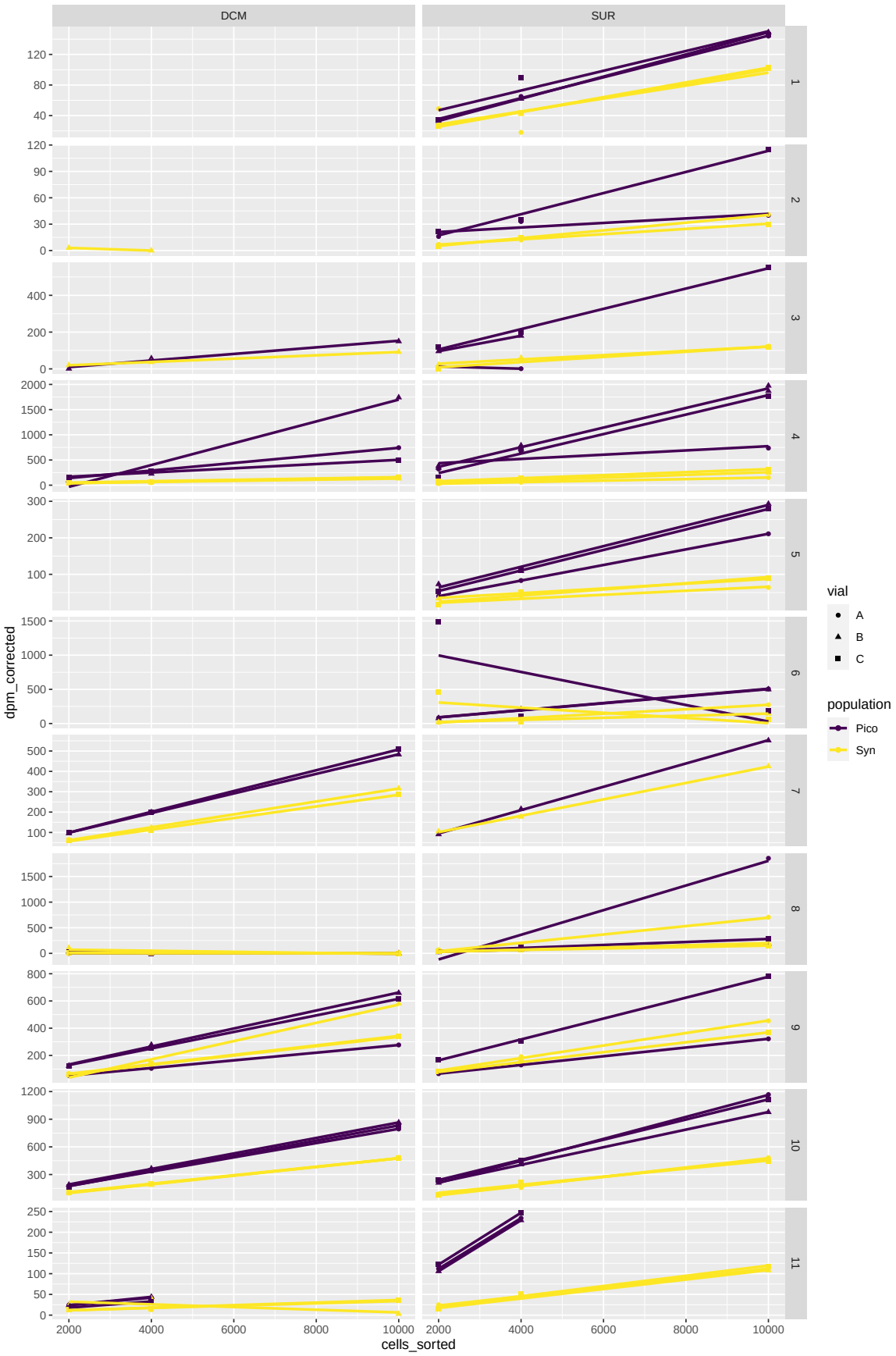


5. Compute based on Method 2 - Compute lm for ABC separately

Plots

Do plots for each group. One regression line based on the EXP, Sample and Populations

```
ggplot(data = C14_DPM_corrected, aes(x=cells_sorted, y=dpm_corrected, color=population, shape=vial)) +
 geom_point() + stat_smooth(method="lm", se=FALSE) +
 # DV - Set the y_scale to free so that graph is more easy to read
 facet_grid(rows=vars(exp), cols=vars(sample), scales = "free_y") +
 scale_color_viridis_d()
```


DCM
SUR
1
2
3
4
5
6
7
8
9
10
11
dpm_corrected
cells_sorted
2000
4000
6000
8000
10000
vial
A
B
C
population
Pico
Syn

Do linear model

$y = ax + b$

See: https://cran.r-project.org/web/packages/broom/vignettes/broom_and_dplyr.html

```
 C14_DPM_model_2 <- C14_DPM_corrected %>%
  group_by(cycle, exp, station, depth, sample, population, vial, SA) %>%
  tidyr::nest() %>%
  mutate(
    fit = purrr::map(data, ~ lm(dpm_corrected ~ cells_sorted, data = .x)),
    tidied = purrr::map(fit, tidy)
  ) %>%
  unnest(tidied)

C14_DPM_model_output_2 <- C14_DPM_model_2 %>%
  dplyr::select(exp:population, term, estimate) %>%
  pivot_wider(names_from="term", values_from ="estimate" ) %>%
  rename (slope = cells_sorted, intercept = `(Intercept)`)

C14_DPM_model_output_2
```

```
## # A tibble: 94 x 10
## # Groups:   cycle, exp, station, depth, sample, vial, population, SA [94]
##    cycle        SA   exp station depth sample vial  population intercept   slope
##    <dbl>     <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>          <dbl>   <dbl>
##  1     1 12000000.     1      15    12 SUR    A     Pico           8.38  0.0136
##  2     1 12000000.     1      15    12 SUR    A     Syn           11.5   0.00846
##  3     1 12000000.     1      15    12 SUR    B     Pico           4.    0.0145
##  4     1 12000000.     1      15    12 SUR    B     Syn            7.46  0.00929
##  5     1 12000000.     1      15    12 SUR    C     Pico          21.1   0.0129
##  6     1 12000000.     1      15    12 SUR    C     Syn            6.08  0.00967
##  7     1 12000000.     2      24    12 SUR    A     Pico          15.9   0.00258
##  8     1 12000000.     2      24    12 SUR    A     Syn           -3.38  0.00438
##  9     1 12000000.     2      24    12 SUR    C     Pico          -6.77  0.0120
## 10     1 12000000.     2      24    12 SUR    C     Syn            0.769 0.00298
## # ... with 84 more rows
```

Merge DIC/SA table and DPM output

```
# merge tables
pp_cal_2 <- left_join(C14_DPM_model_output_2, DIC_data_corrected)
pp_cal_2
```

```
## # A tibble: 94 x 11
## # Groups:   cycle, exp, station, depth, sample, vial, population, SA [94]
##    cycle     SA   exp station depth sample vial  population intercept   slope
##    <dbl>  <dbl> <dbl>   <dbl> <dbl> <chr>  <chr> <chr>          <dbl>   <dbl>
##  1     1 1.20e7     1      15    12 SUR    A     Pico           8.38  0.0136
##  2     1 1.20e7     1      15    12 SUR    A     Syn           11.5   0.00846
##  3     1 1.20e7     1      15    12 SUR    B     Pico           4.    0.0145
##  4     1 1.20e7     1      15    12 SUR    B     Syn            7.46  0.00929
##  5     1 1.20e7     1      15    12 SUR    C     Pico          21.1   0.0129
##  6     1 1.20e7     1      15    12 SUR    C     Syn            6.08  0.00967
##  7     1 1.20e7     2      24    12 SUR    A     Pico          15.9   0.00258
##  8     1 1.20e7     2      24    12 SUR    A     Syn           -3.38  0.00438
```

```
##  9     1 1.20e7     2      24    12 SUR    C     Pico          -6.77  0.0120
## 10     1 1.20e7     2      24    12 SUR    C     Syn            0.769 0.00298
## # ... with 84 more rows, and 1 more variable: DIC <dbl>
```

Calculate PP value, based on Daniel's formula found here:

https://vaulot.netlify.com/2018/05/20/compute-primary-production-based-on-single-cell-c14-uptake/

```
pp_cal_2 <- mutate(pp_cal_2, pp = DIC*slope*(1/(SA*24))*10^9*1.05) %>%
  ungroup() %>%
  select( -c(station, depth)) %>%
  filter(pp >= 0) # remove negative pp values

# add missing rows for data analysis later
pp_cal_2 <- complete(pp_cal_2, nesting(cycle, exp, sample), vial, population)

pp_cal_2
```

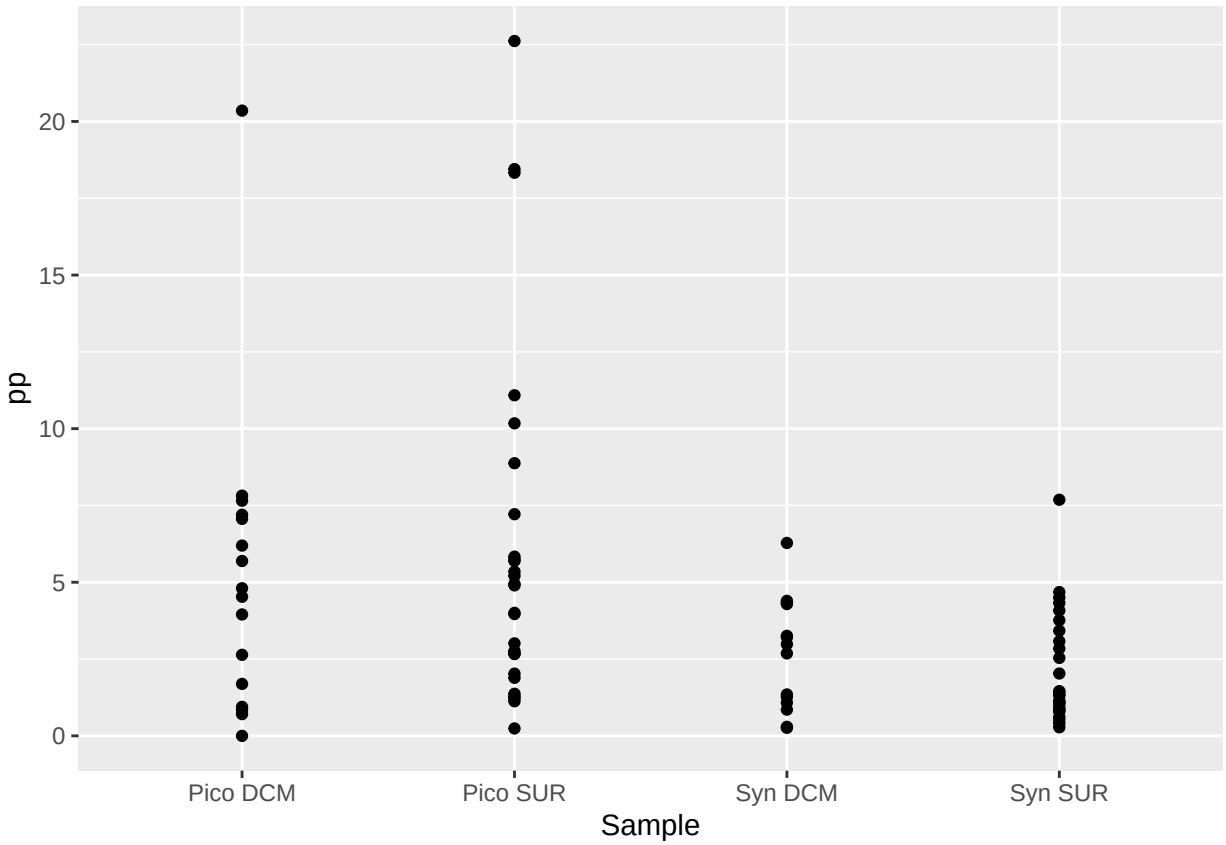
```
## # A tibble: 108 x 10
##    cycle   exp sample vial  population        SA intercept    slope   DIC     pp
##    <dbl> <dbl> <chr>  <chr> <chr>          <dbl>     <dbl>    <dbl> <dbl>  <dbl>
##  1     1     1 SUR    A     Pico       12000000.      8.38  0.0136   25.9  1.29
##  2     1     1 SUR    A     Syn        12000000.     11.5   0.00846  25.9  0.799
##  3     1     1 SUR    B     Pico       12000000.      4.    0.0145   25.9  1.37
##  4     1     1 SUR    B     Syn        12000000.      7.46  0.00929  25.9  0.877
##  5     1     1 SUR    C     Pico       12000000.     21.1   0.0129   25.9  1.22
##  6     1     1 SUR    C     Syn        12000000.      6.08  0.00967  25.9  0.913
##  7     1     2 SUR    A     Pico       12000000.     15.9   0.00258  25.7  0.241
##  8     1     2 SUR    A     Syn        12000000.     -3.38  0.00438  25.7  0.411
##  9     1     2 SUR    B     Pico              NA      NA    NA         NA   NA
## 10     1     2 SUR    B     Syn               NA      NA    NA         NA   NA
## # ... with 98 more rows
```

Compute means quickly.

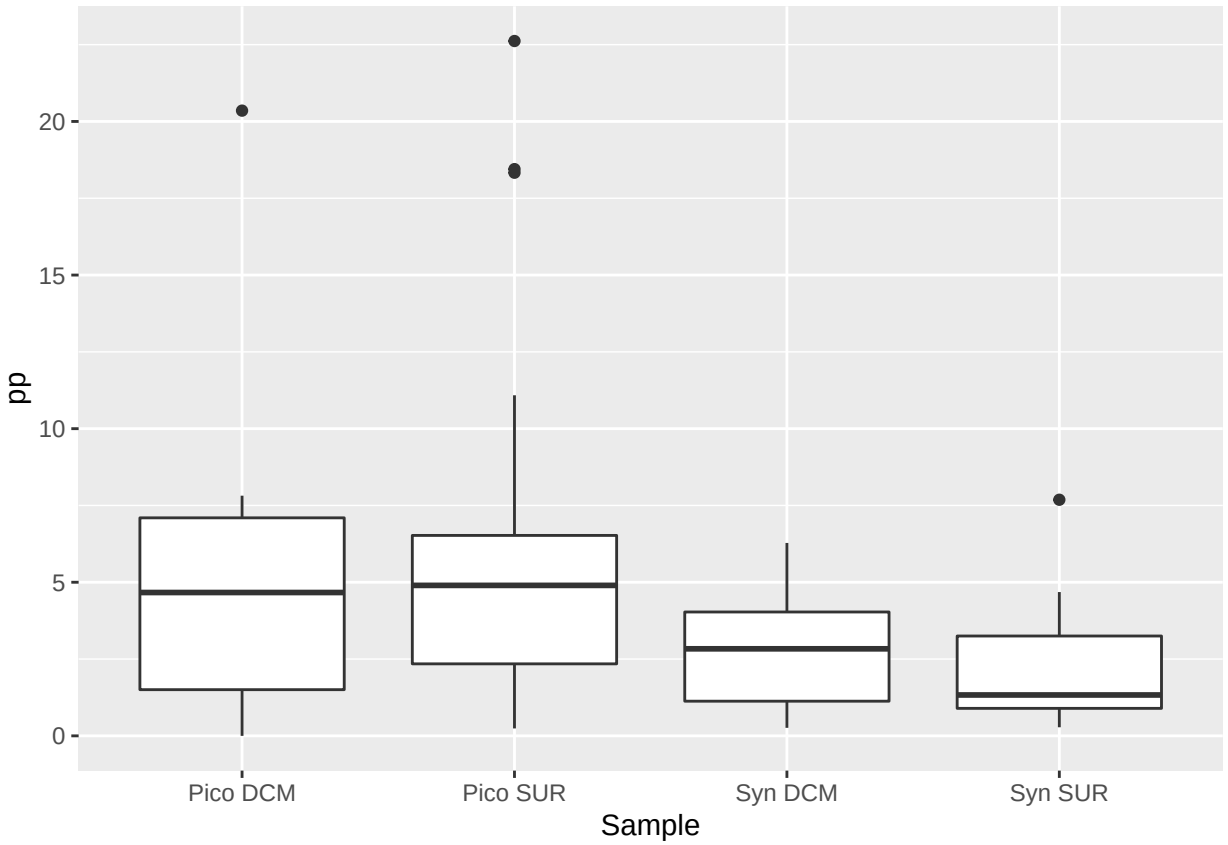
```
pp_cal_2 %>%
  group_by(population, sample) %>%
  summarise(pp_mean = mean(pp, na.rm = TRUE))
```

```
## # A tibble: 4 x 3
## # Groups:   population [2]
##   population sample pp_mean
##   <chr>      <chr>    <dbl>
## 1 Pico       DCM       5.13
## 2 Pico       SUR       6.02
## 3 Syn        DCM       2.61
## 4 Syn        SUR       2.15
```

```
pp_cal_2 %>%
  ggplot() +
  geom_point(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample")
```




```
pp_cal_2 %>%
  ggplot() +
  geom_boxplot(aes(x= str_c(population,sample,  sep=" "), y = pp)) +
  xlab("Sample")
```



The units for pp is fgC/h/cell. Rii (2016) finds 5–11 fmol C/cell/day for Syn which corresponds to 60-132 fg /day/cell ie 2.5 to 5.5 fg/h/cell.