FigS12 stats

Colleen Kellogg

2025-06-27

Spatiotemporal senstivity analysis

In response to reviewer #1's concerns regarding spatio-temporal variability and Argo data we added a thorough statistical analysis regarding spatial variability. A new section was added to the Methods, titled "Spatiotemporal sensitivity analyses", which details the approach and results. To back up the statements and observations made in Fig. S12, below are several statistical tests to support the statements made surrounding S12a, S12c, and S12c

Stats for Fig S12a:

Correlation test to support the statement that "Profiles collected within three days of each other showed a strong positive correlation in integrated Chl over the upper 100m".

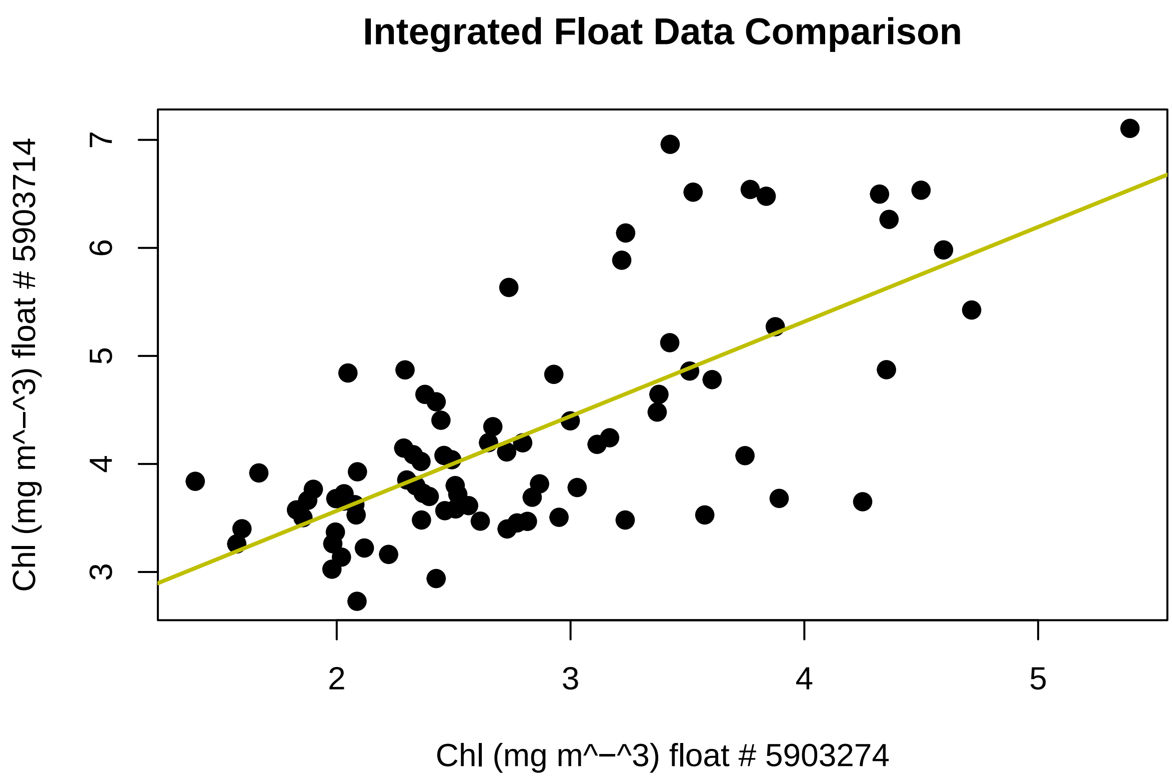
```
# Stats for fig S12a
# library to read matlab data formats into R

# read in our data
chlmat <- readMat("chla_NEPac_processed.mat")

# check out data structure
str(chlmat)
```

```
## List of 21
##  $ bbr           : num [1:14, 1] 0.00742 0.00742 0.00728 0.00754 0.00685 ...
##  $ chla1         : num [1:550, 1:681] 0.248 0.248 0.248 0.248 0.248 ...
##  $ chla.big      : num [1:550, 1:681] -0.0037 -0.0037 -0.0037 -0.0037 -0.0037 ...
##  $ chla.movmax   : num [1:550, 1:681] 0.248 0.248 0.248 0.248 0.248 ...
##  $ chla.movmin   : num [1:550, 1:681] 0.248 0.248 0.248 0.248 0.248 ...
##  $ chla.small    : num [1:550, 1:681] 0.244 0.244 0.244 0.244 0.244 ...
##  $ chla.smallwblk: num [1:550, 1:681] 0.252 0.252 0.252 0.252 0.252 ...
##  $ chla.total    : num [1:550, 1:681] 0.24 0.24 0.24 0.24 0.24 ...
##  $ date.tseries  : num [1:550, 1:681] 734308 734308 734308 734308 734308 ...
##  $ datet         : num [1, 1:681] 734308 734313 734318 734323 734328 ...
##  $ floatn        : num [1, 1:681] 5903274 5903274 5903274 5903274 5903274 ...
##  $ lat           : num [1, 1:681] 50 50 50 49.9 49.9 ...
##  $ lon           : num [1, 1:681] 215 215 215 215 215 ...
##  $ medianmld     : num [1, 1] 87
##  $ mld           : num [1, 1:681] 34.9 13.4 45.6 33.3 31.1 ...
##  $ noise         : num [1:550, 1:681] 0.0037 0.0037 0.0037 0.0037 0.0037 ...
##  $ oxyc          : num [1:550, 1:681] NaN NaN 306 NaN NaN ...
##  $ press         : num [1:550, 1:681] 4.28 6.08 7.68 8.08 10.08 ...
```

```
##  $ press1        : num [1:550, 1:681] 7.68 11.68 16.48 21.58 26.68 ...
##  $ sal           : num [1:550, 1:681] 32.7 32.7 32.7 32.7 32.7 ...
##  $ temp          : num [1:550, 1:681] 7.73 7.72 7.7 7.69 7.69 ...
##  - attr(*, "header")=List of 3
##   ..$ description: chr "MATLAB 5.0 MAT-file, Platform: MACI64, Created on: Fri Jan 20 09:46:09 2023
##   ..$ version    : chr "5"
##   ..$ endian     : chr "little"
```

```r
chla_small <- chlmat$chla1
press1 <- chlmat$press1
datet <- chlmat$datet
floatn <- chlmat$floatn

# Find overlapping dates within 3 days for different floats
overlap_indices <- matrix(ncol = 2, nrow = 0)
for (i in 1:length(datet)) {
  for (j in (i+1):length(datet)) {
    if (j <= length(datet) && abs(datet[i] - datet[j]) <= 3 && floatn[i] != floatn[j]) {
      overlap_indices <- rbind(overlap_indices, c(i, j))
    }
  }
}

# Extract overlapping data
float1_data <- chla_small[, overlap_indices[, 1]]
float2_data <- chla_small[, overlap_indices[, 2]]
press1_data <- press1[, overlap_indices[, 1]]
press2_data <- press1[, overlap_indices[, 2]]

# Initialize vectors to store integrated values
integrated_float1 <- c()
integrated_float2 <- c()

# Integrate float1_data and float2_data values for rows where press1_data < 100
for (col in 1:ncol(press1_data)) {
  rows_to_integrate_float1 <- which(press1_data[, col] < 100)
  rows_to_integrate_float2 <- which(press2_data[, col] < 100)

# Use pracma package for trapz function, or implement trapezoidal rule
# If pracma is available: library(pracma)
integrated_float1 <- c(integrated_float1, trapz(1:length(rows_to_integrate_float1), float1_data[rows_to
integrated_float2 <- c(integrated_float2, trapz(1:length(rows_to_integrate_float2), float2_data[rows_to

}

# Get the corresponding float numbers for labeling
float_num_1 <- floatn[overlap_indices[1, 1]]
float_num_2 <- floatn[overlap_indices[1, 2]]

# Create figure with scatter plot and linear fit


# Sanity check - Scatter plot 1: Integrated Float Data with linear fit.
plot(integrated_float1, integrated_float2,
```

```
     pch = 19, cex = 1.2, col = "black",
     xlab = paste("Chl (mg m^-^3) float #", float_num_1),
     ylab = paste("Chl (mg m^-^3) float #", float_num_2),
     main = "Integrated Float Data Comparison")


# Add linear fit line
fit <- lm(integrated_float2 ~ integrated_float1)
abline(fit, col = rgb(0.75, 0.75, 0), lwd = 2)  # Dark yellow line
```




```
#correlation

tocompare_mat<-cbind(integrated_float1,integrated_float2)
s12a_mat_corr<-rcorr(tocompare_mat, type = "pearson")
s12a_mat_corr
```

```
##                   integrated_float1 integrated_float2
## integrated_float1               1.0               0.7
## integrated_float2               0.7               1.0
##
## n= 83
##
##
## P
##                   integrated_float1 integrated_float2
```

```
## integrated_float1                    0
## integrated_float2  0
```

```
s12a_mat_corr$P
```

```
##                   integrated_float1 integrated_float2
## integrated_float1                NA      1.256772e-13
## integrated_float2      1.256772e-13                NA
```

```
s12a_mat_corr$r
```

```
##                   integrated_float1 integrated_float2
## integrated_float1         1.0000000         0.7031125
## integrated_float2         0.7031125         1.0000000
```

```
#Pearson correlation results:  (r(84) = 0.70, P< 0.0001)
```

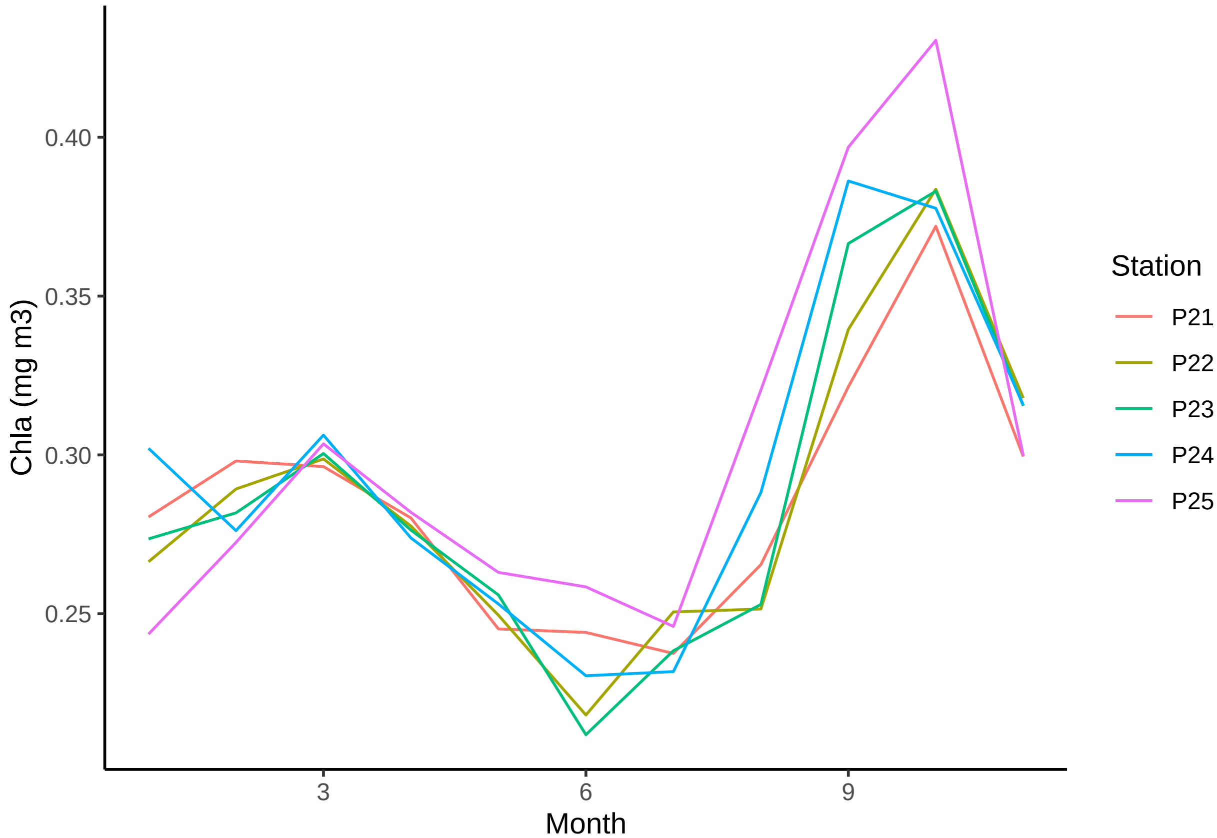
Stats for Fig S12c:

To support the statement "we examined spatial variability in surface chlorophyll using monthly mean concentrations from the Aqua-MODIS satellite product (4 km resolution, 2008-2023) across four Line P stations located within the float trajectories." we used both a pearson correlation analysis as well as a repeated measures anova to determine if how well the satellite data corresponded across all locations and whether it differed significantly within a time point, across all stations.

```
# Stats for fig S12c
# load data
monthly<-read_csv("climate_monthly_2008-2023.csv")
```

```
## Rows: 55 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Station
## dbl (7): lon, lat, Month, poc, sst, chlor_a, CAFE
##
## 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.
```

```
#sanity check: make a plot and ensure it looks like what matlab code produced.
ggplot(monthly, aes(x = Month, y = chlor_a, group = Station, color = Station)) +
  geom_line() + labs(color = "Station", y = "Chla (mg m3)") + theme_classic()
```




```
#prepare for stats

monthly$Month = as.factor(monthly$Month)
monthly$Station = as.factor(monthly$Station)

monthly
```

```
## # A tibble: 55 x 8
##    Station   lon   lat Month   poc   sst chlor_a  CAFE
##    <fct>   <dbl> <dbl> <fct> <dbl> <dbl>   <dbl> <dbl>
##  1 P21     -139.  49.4 1      80.9  7.26   0.280  289.
##  2 P21     -139.  49.4 2      78.2  6.94   0.298  307.
##  3 P21     -139.  49.4 3      79.7  6.61   0.296  362.
##  4 P21     -139.  49.4 4      78.3  7.01   0.280  430.
##  5 P21     -139.  49.4 5      72.2  8.28   0.245  546.
##  6 P21     -139.  49.4 6      72.7 10.1    0.244  604.
##  7 P21     -139.  49.4 7      74.1 13.0    0.237  568.
##  8 P21     -139.  49.4 8      78.1 14.8    0.265  513.
##  9 P21     -139.  49.4 9      83.6 14.5    0.321  482.
## 10 P21     -139.  49.4 10     89.5 12.5    0.372  420.
## # i 45 more rows
```

```
#first lets do a correlation matrix
s12c_subset<-monthly %>% select(Station, Month, chlor_a) %>% pivot_wider(values_from = chlor_a, names_f
s12c_corr<-rcorr(as.matrix(s12c_subset[2:6]), type = c("pearson","spearman"))
```

```r
# Extract the correlation coefficients
s12c_corr$r
```

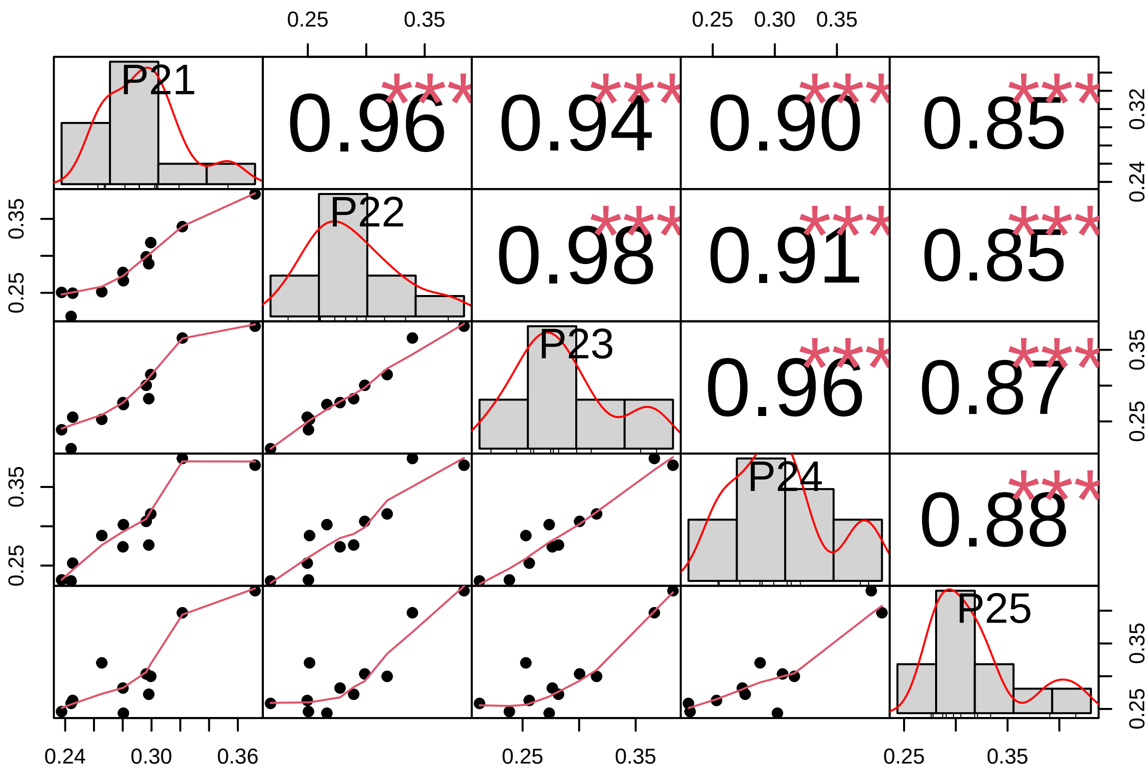
```
##           P21       P22       P23       P24       P25
## P21 1.0000000 0.9601449 0.9408566 0.9027595 0.8480850
## P22 0.9601449 1.0000000 0.9826617 0.9067281 0.8499916
## P23 0.9408566 0.9826617 1.0000000 0.9574428 0.8713965
## P24 0.9027595 0.9067281 0.9574428 1.0000000 0.8821149
## P25 0.8480850 0.8499916 0.8713965 0.8821149 1.0000000
```

```r
# Extract p-values
pvalues<-s12c_corr$P

# Example using Bonferroni correction
adjusted_p_values_bonferroni <- p.adjust(pvalues, method = "bonferroni")

library("PerformanceAnalytics")
```

```
## Loading required package: xts
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
##
## ######################### Warning from 'xts' package ##########################
## #                                                                             #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                           #
## #                                                                             #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
## # dplyr from breaking base R's lag() function.                                #
## #                                                                             #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                             #
## ###############################################################################
##
## Attaching package: 'xts'
##
## The following objects are masked from 'package:dplyr':
##
##     first, last
##
##
## Attaching package: 'PerformanceAnalytics'
##
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```r
# jpeg("figs12c-correlation-chart.jpg", width = 6, height = 7, units = "in", res = 300)
chart.Correlation(s12c_subset[2:6], histogram=TRUE, pch=19)
```




```r
# dev.off()


# results - Chlorophyll a concentrations co-vary. R2 ranges from 0.85 to 0.98 with p-values <0.001 (<0.

#But what about within a month

aov(formula = chlor_a ~ Month, data = monthly)
```

```
## Call:
##    aov(formula = chlor_a ~ Month, data = monthly)
##
## Terms:
##                     Month Residuals
## Sum of Squares  0.1165206 0.0140685
## Deg. of Freedom        10        44
##
## Residual standard error: 0.01788124
## Estimated effects may be unbalanced
```

```r
aov(formula = chlor_a ~ Month, data = monthly)
```

```
## Call:
##    aov(formula = chlor_a ~ Month, data = monthly)
##
## Terms:
##                     Month Residuals
## Sum of Squares  0.1165206 0.0140685
## Deg. of Freedom        10        44
##
## Residual standard error: 0.01788124
## Estimated effects may be unbalanced
```

```r
monthly %>%
  group_by(Station) %>%
  get_summary_stats(chlor_a, type = "mean_sd")
```

```
## # A tibble: 5 x 5
##   Station variable     n  mean    sd
##   <fct>   <fct>    <dbl> <dbl> <dbl>
## 1 P21     chlor_a     11 0.285 0.039
## 2 P22     chlor_a     11 0.286 0.047
## 3 P23     chlor_a     11 0.287 0.052
## 4 P24     chlor_a     11 0.295 0.052
## 5 P25     chlor_a     11 0.301 0.061
```

```r
#check for outliers
monthly %>%
  group_by(Station) %>%
  identify_outliers(chlor_a)
```

```
## # A tibble: 4 x 10
##   Station   lon   lat Month   poc   sst chlor_a  CAFE is.outlier is.extreme
##   <fct>   <dbl> <dbl> <fct> <dbl> <dbl>   <dbl> <dbl> <lgl>      <lgl>
## 1 P21     -139.  49.4 10     89.5  12.5   0.372  420. TRUE       FALSE
## 2 P24     -142.  49.5 9      91.6  14.1   0.386  513. TRUE       FALSE
## 3 P25     -144.  50   9      95.1  13.7   0.397  526. TRUE       FALSE
## 4 P25     -144.  50   10    101.   11.7   0.431  415. TRUE       FALSE
```

```r
monthly %>%
  group_by(Month) %>%
  identify_outliers(chlor_a)
```

```
## # A tibble: 3 x 10
##   Month Station   lon   lat   poc   sst chlor_a  CAFE is.outlier is.extreme
##   <fct> <fct>   <dbl> <dbl> <dbl> <dbl>   <dbl> <dbl> <lgl>      <lgl>
## 1 1     P24     -142.  49.5  89.9  6.98   0.302  277. TRUE       FALSE
## 2 1     P25     -144.  50    68.9  6.67   0.244  260. TRUE       FALSE
## 3 10    P25     -144.  50   101.  11.7    0.431  415. TRUE       TRUE
```

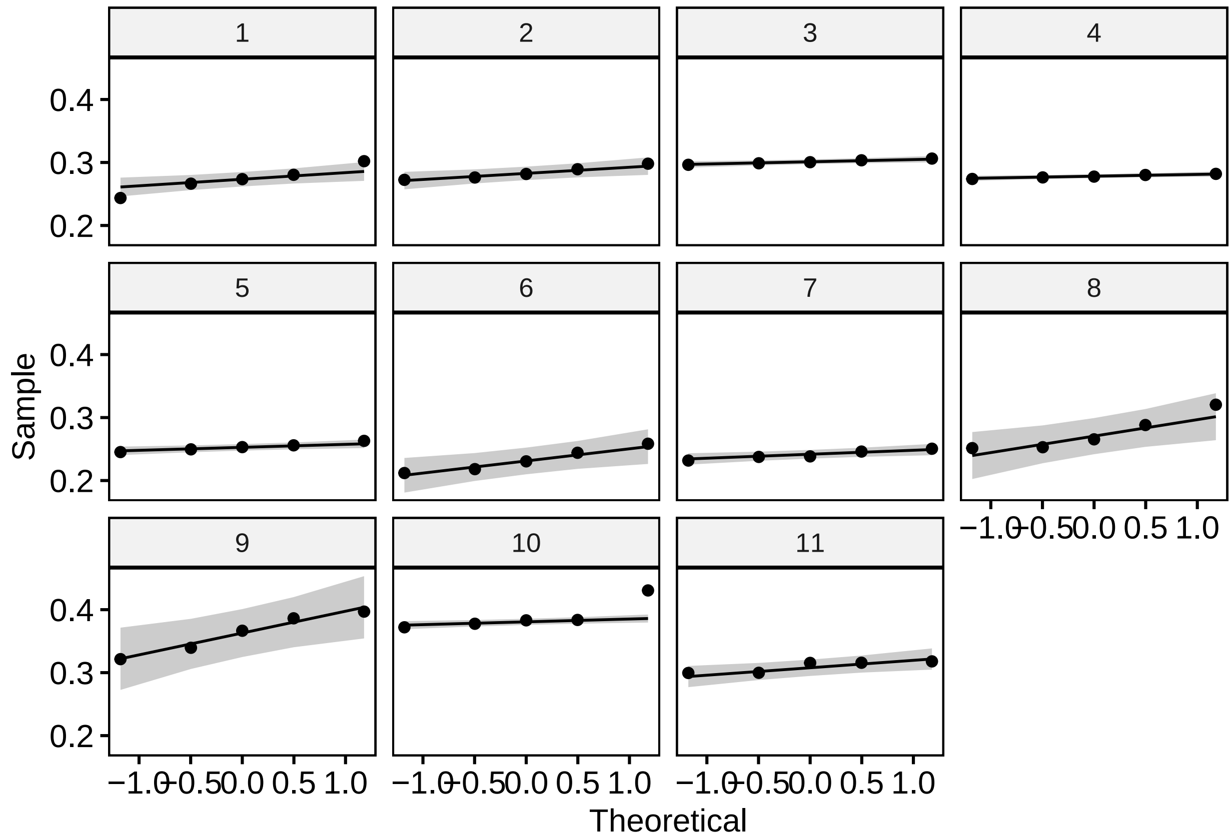
```r
#there is an extreme outlier P25, October - higher than other stations. but since we focused on spring 

#normality assumption
monthly %>%
   group_by(Station) %>%
  shapiro_test(chlor_a)
```

```
## # A tibble: 5 x 4
##   Station variable statistic      p
##   <fct>   <chr>        <dbl>  <dbl>
## 1 P21     chlor_a      0.924 0.350
## 2 P22     chlor_a      0.952 0.668
## 3 P23     chlor_a      0.941 0.537
## 4 P24     chlor_a      0.920 0.316
## 5 P25     chlor_a      0.839 0.0304
```

```r
monthly %>%
   group_by(Month) %>%
  shapiro_test(chlor_a)
```

```
## # A tibble: 11 x 4
##    Month variable statistic      p
##    <fct> <chr>        <dbl>  <dbl>
##  1 1     chlor_a      0.987 0.969
##  2 2     chlor_a      0.960 0.809
##  3 3     chlor_a      0.982 0.947
##  4 4     chlor_a      0.984 0.953
##  5 5     chlor_a      0.989 0.975
##  6 6     chlor_a      0.959 0.803
##  7 7     chlor_a      0.958 0.791
##  8 8     chlor_a      0.877 0.295
##  9 9     chlor_a      0.949 0.729
## 10 10    chlor_a      0.738 0.0232
## 11 11    chlor_a      0.763 0.0387
```

```r
#months 10 and 11 have a pvalue of 0.02 and 0.038; rejecting the assumption of normality.

ggqqplot(monthly, "chlor_a", facet.by = "Month")
```




```
#repeated measures anova

res.aov <- anova_test(data = monthly, dv = chlor_a, wid = Station, within = Month)
get_anova_table(res.aov) #significantly different at different timepoints. this makes sense.
```

```
## ANOVA Table (type III tests)
##
##   Effect DFn DFd      F        p p<.05   ges
## 1  Month  10  40 39.229 1.56e-17     * 0.892
```

```
####
# ANOVA Table (type III tests)
#
#   Effect DFn DFd      F        p p<.05   ges
# 1  Month  10  40 39.229 1.56e-17     * 0.892
####

res.aov2 <- anova_test(data = monthly, dv = chlor_a, wid = Month, within = Station)
get_anova_table(res.aov2) #but if we flip this around there is not a significant difference among stati
```

```
## ANOVA Table (type III tests)
##
##    Effect  DFn   DFd     F    p p<.05   ges
## 1 Station 2.18 21.75 1.841 0.18       0.017
```

```
#####
# ANOVA Table (type III tests)
#
#    Effect  DFn   DFd     F    p p<.05   ges
# 1 Station 2.18 21.75 1.841 0.18       0.017
#####
```

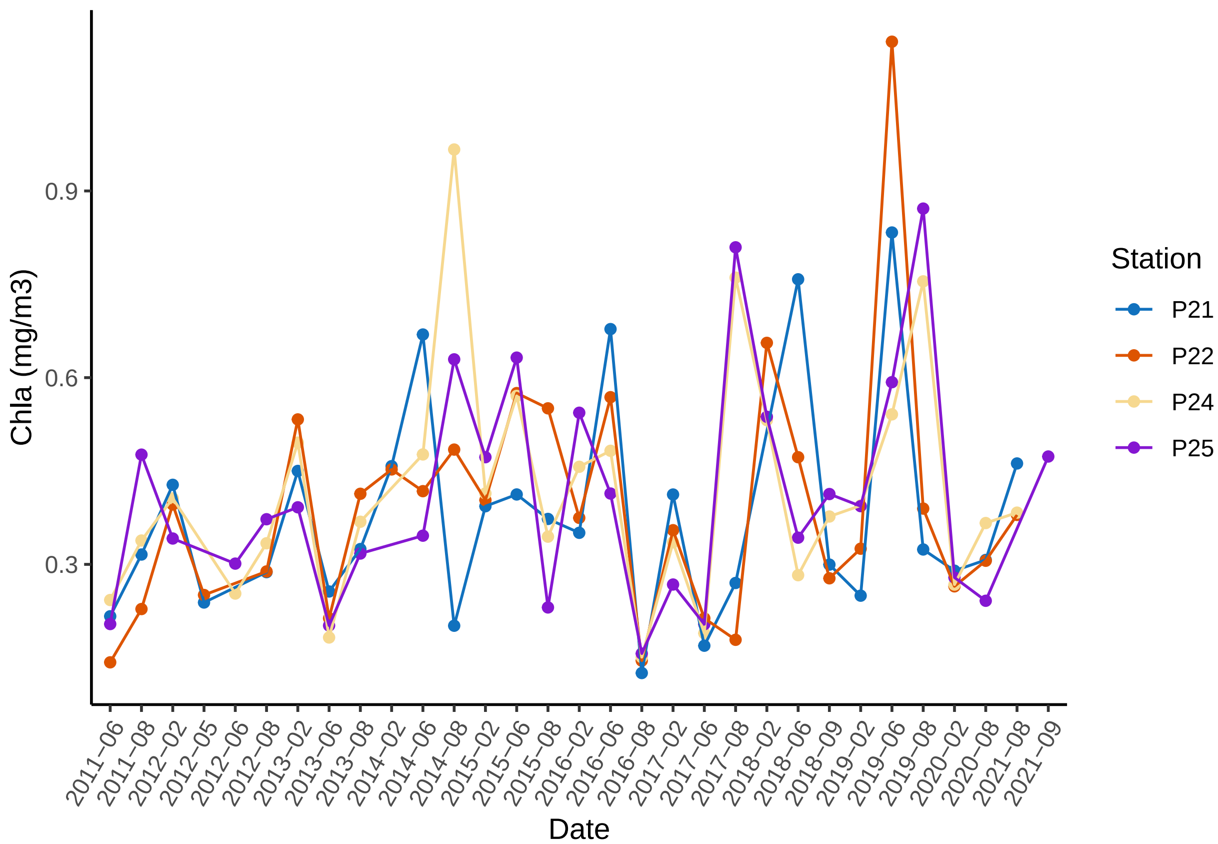
Stats for Fig S12d:

Finally, we looked at discrete Tchla samples collected on Line P cruises. We made the statement: "Discrete chlorophyll samples collected during Line P cruises further confirmed these seasonal trends (Fig. S12d)".

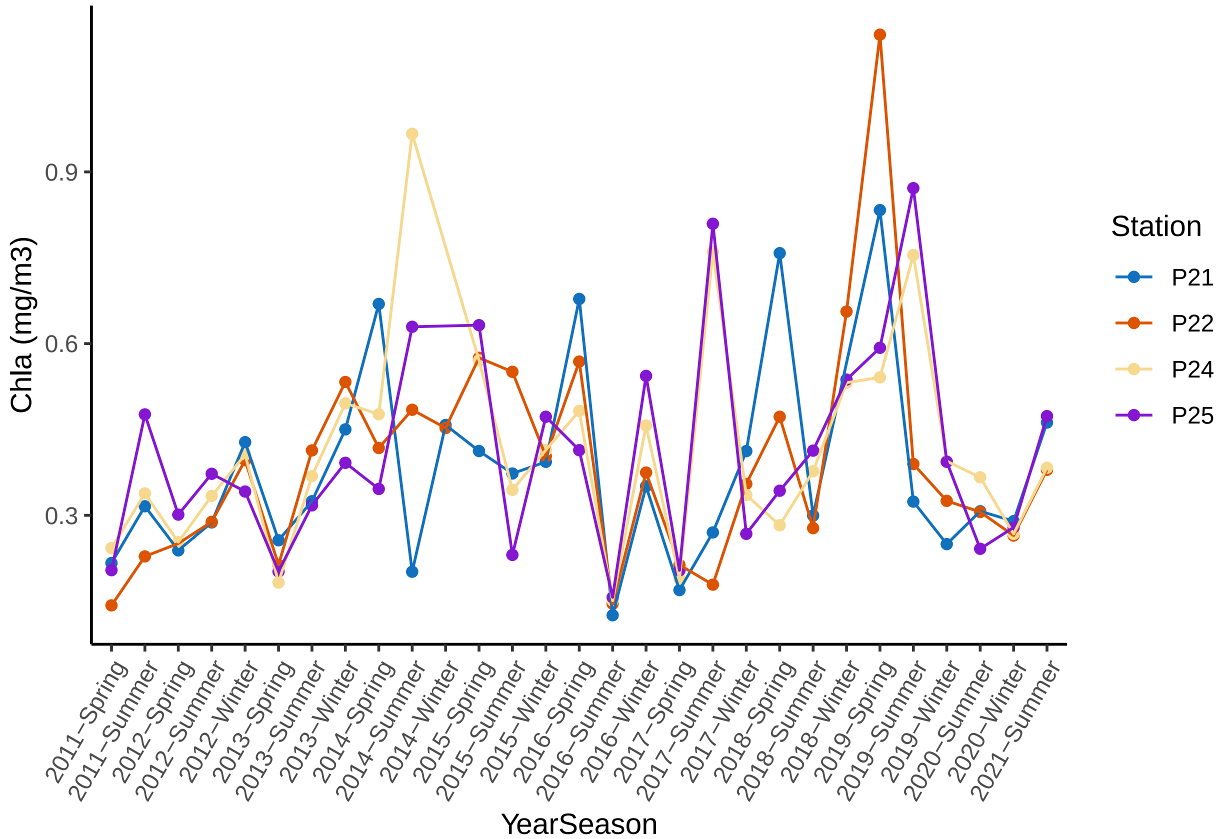
```
# Stats for fig S12d
# load data
chlbtl_all<-read_csv("forMarianaB_PhytoComposition_avg-rev-cat.csv")
```

```
## Rows: 113 Columns: 18
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (4): Cruise, TimeofYear, YearSeason, Station
## dbl  (13): Year, Month, Longitude, Latitude, Cyanobacteria, Chlorophytes, Pr...
## date  (1): Date
##
## 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.
```

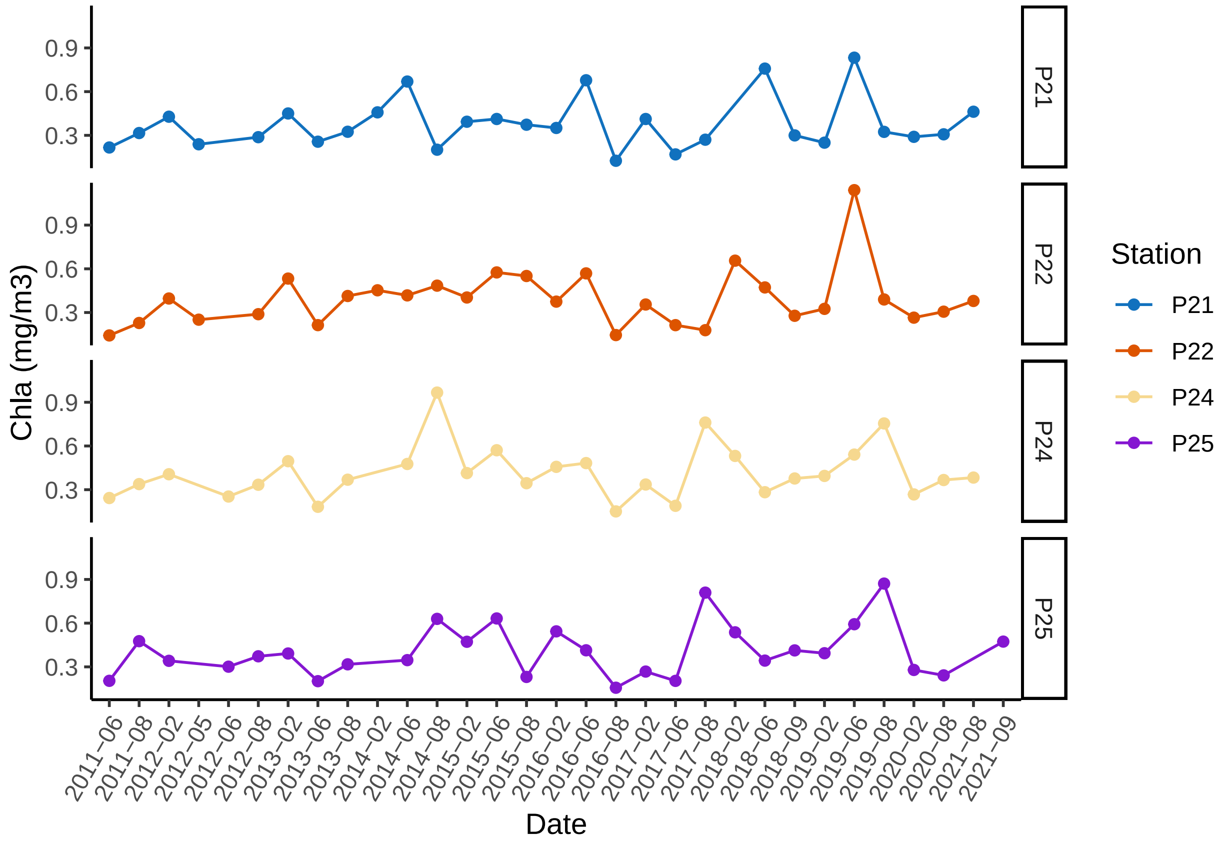
```
chlbtl<-chlbtl_all %>% subset(TimeofYear %in% c("Spring","Summer","Winter"))

chlbtl$Date <- as_date(chlbtl$Date)

chlbtl$Date <- strftime(chlbtl$Date,format="%Y-%m")

#make a plot or two or three.
#reproduce matlab version s12d
ggplot(chlbtl, aes(x = Date, y = Tchl_a, group = Station, color = Station)) +
  geom_point() + geom_line() + labs(color = "Station", y = "Chla (mg/m3)") + theme_classic() + theme(ax
```



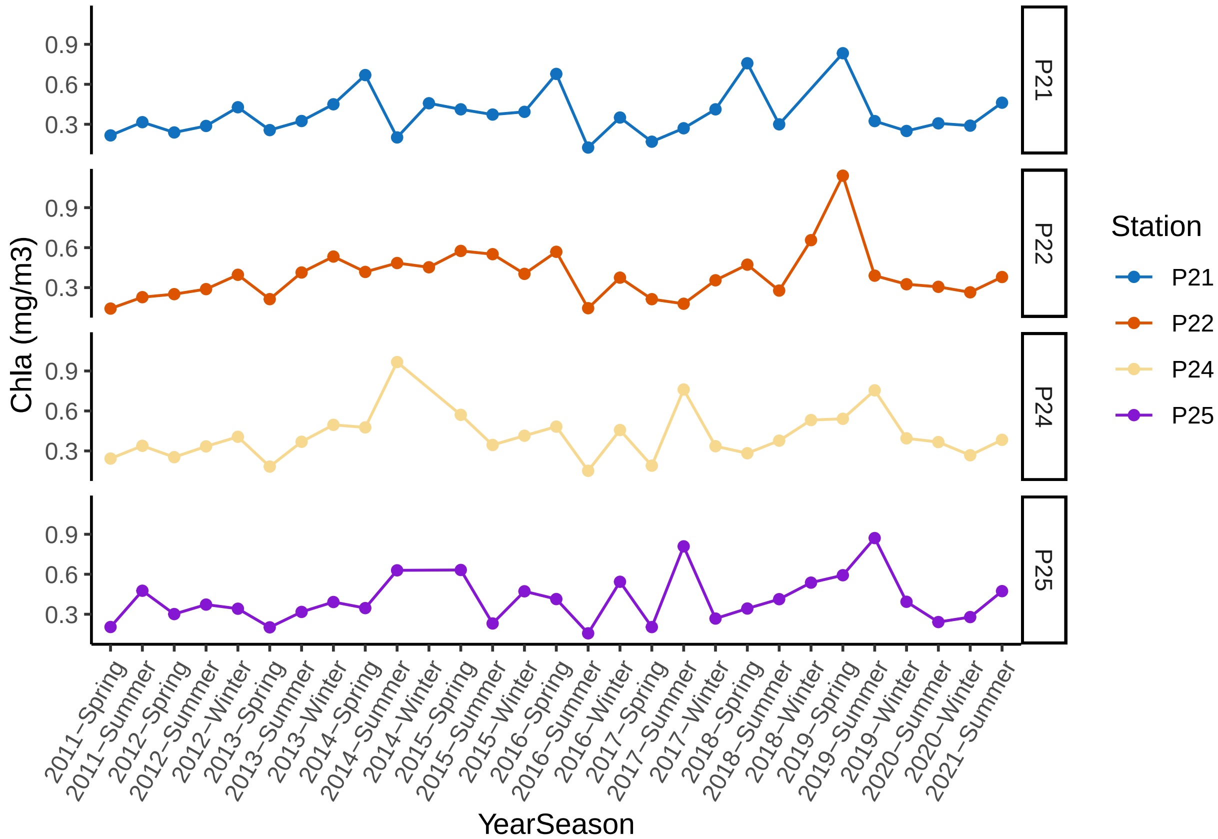

```
#but let's plot the data a few others ways to just explore.
#plotted by year-season
ggplot(chlbtl, aes(x = YearSeason, y = Tchl_a, group = Station, color = Station)) +
  geom_point() + geom_line() + labs(color = "Station", y = "Chla (mg/m3)") + theme_classic() + theme(ax
```



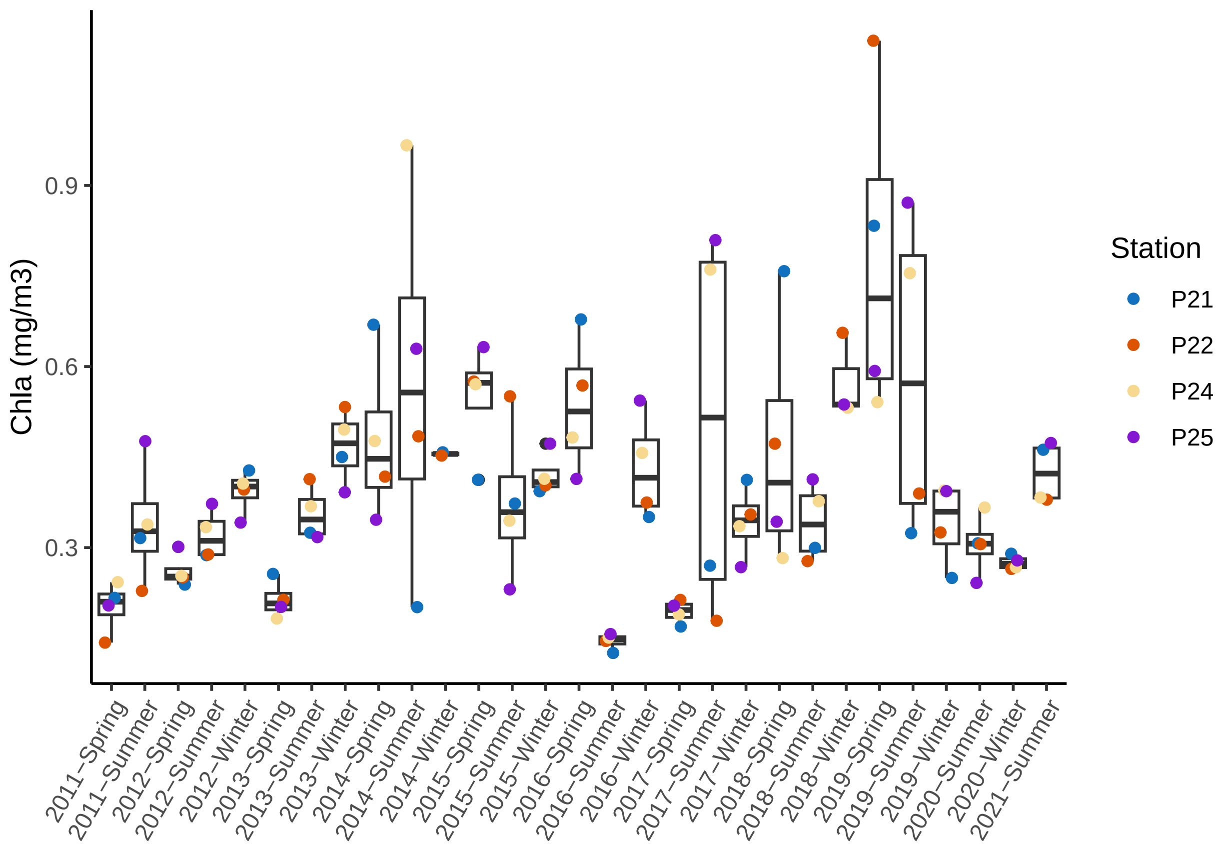

```
#stacked, date
ggplot(chlbtl, aes(x = Date, y = Tchl_a, group = Station, color = Station)) +
  geom_point() + geom_line() + labs(color = "Station", y = "Chla (mg/m3)") + theme_classic() + theme(ax
```




```
#stacked, year-season
ggplot(chlbtl, aes(x = YearSeason, y = Tchl_a, group = Station, color = Station)) +
  geom_point() + geom_line() + labs(color = "Station", y = "Chla (mg/m3)") + theme_classic() + theme(ax
```




```
#boxplot to see variability within each cruise/season a bit more clearly.
ggplot(chlbtl, aes(x = YearSeason, y = Tchl_a, group = YearSeason)) +
  geom_boxplot() + geom_jitter(aes(color = Station),width = 0.2) + labs(color = "Station", y = "Chla (mg
```




```
#prepare for stats

chlbtl$Month = as.factor(chlbtl$Month)

chlbtl
```

```
## # A tibble: 113 x 18
##    Cruise   Year Month TimeofYear Date    YearSeason  Longitude Latitude Station
##    <chr>   <dbl> <fct> <chr>      <chr>   <chr>           <dbl>    <dbl> <chr>
##  1 2011_26  2011 6     Spring     2011-06 2011-Spring     -140.     49.6 P21
##  2 2011_27  2011 8     Summer     2011-08 2011-Summer     -140.     49.6 P21
##  3 2012_01  2012 2     Winter     2012-02 2012-Winter     -140.     49.6 P21
##  4 2012_12  2012 5     Spring     2012-05 2012-Spring     -140.     49.6 P21
##  5 2012_13  2012 8     Summer     2012-08 2012-Summer     -140.     49.6 P21
##  6 2013_01  2013 2     Winter     2013-02 2013-Winter     -140.     49.6 P21
##  7 2013_17  2013 6     Spring     2013-06 2013-Spring     -140.     49.6 P21
##  8 2013_18  2013 8     Summer     2013-08 2013-Summer     -140.     49.6 P21
##  9 2014_01  2014 2     Winter     2014-02 2014-Winter     -140.     49.6 P21
## 10 2014_18  2014 6     Spring     2014-06 2014-Spring     -140.     49.6 P21
## # i 103 more rows
## # i 9 more variables: Cyanobacteria <dbl>, Chlorophytes <dbl>,
## #   Prasinophytes <dbl>, Cryptophytes <dbl>, ‘Diatom-2‘ <dbl>,
## #   ‘Dinoflage-1‘ <dbl>, Pelagophytes <dbl>, Haptophytes <dbl>, Tchl_a <dbl>
```

```r
#first lets do a correlation matrix
library(Hmisc)
s12d_subset<-chlbtl %>% select(Station, Date, TimeofYear, YearSeason,Tchl_a) %>% pivot_wider(values_from
s12d_corr_pear<-rcorr(as.matrix(s12d_subset[4:7]), type = "pearson")
s12d_corr_spear<-rcorr(as.matrix(s12d_subset[4:7]), type = "spearman")

#subset spring and summer, as that was the focus of the comparison in the paper
s12d_subset_ss<-s12d_subset %>% filter(TimeofYear %in% c("Spring","Summer"))

s12d_corr_sssp<-rcorr(as.matrix(s12d_subset_ss[4:7]), type ="spearman")
s12d_corr_ssp<-rcorr(as.matrix(s12d_subset_ss[4:7]), type ="pearson")

s12d_corr_sssp$r
```

```
##           P21       P22       P24       P25
## P21 1.0000000 0.7699248 0.3771930 0.3787410
## P22 0.7699248 1.0000000 0.5614035 0.4406605
## P24 0.3771930 0.5614035 1.0000000 0.8859649
## P25 0.3787410 0.4406605 0.8859649 1.0000000
```

```r
s12d_corr_ssp$r
```

```
##           P21       P22       P24       P25
## P21 1.0000000 0.7566742 0.1071729 0.1615848
## P22 0.7566742 1.0000000 0.3484169 0.3127959
## P24 0.1071729 0.3484169 1.0000000 0.8696302
## P25 0.1615848 0.3127959 0.8696302 1.0000000
```

```r
s12d_corr_sssp$P
```

```
##              P21          P22          P24          P25
## P21           NA 0.0000718354 1.113882e-01 1.211642e-01
## P22 0.0000718354           NA 1.238048e-02 6.720191e-02
## P24 0.1113882030 0.0123804768           NA 4.505111e-07
## P25 0.1211642259 0.0672019124 4.505111e-07           NA
```

```r
s12d_corr_ssp$P
```

```
##              P21          P22          P24          P25
## P21           NA 0.0001127839 6.623269e-01 5.218079e-01
## P22 0.0001127839           NA 1.437744e-01 2.062971e-01
## P24 0.6623268519 0.1437744113           NA 1.326838e-06
## P25 0.5218078560 0.2062971027 1.326838e-06           NA
```

```r
# Extract p-values
pvalues_12d<-s12d_corr_sssp$P
# pvalues_11d<-s11d_corr_ssp$P

# Example using Bonferroni correction
adjusted_p_values_bonferroni <- p.adjust(pvalues_12d, method = "bonferroni")
```

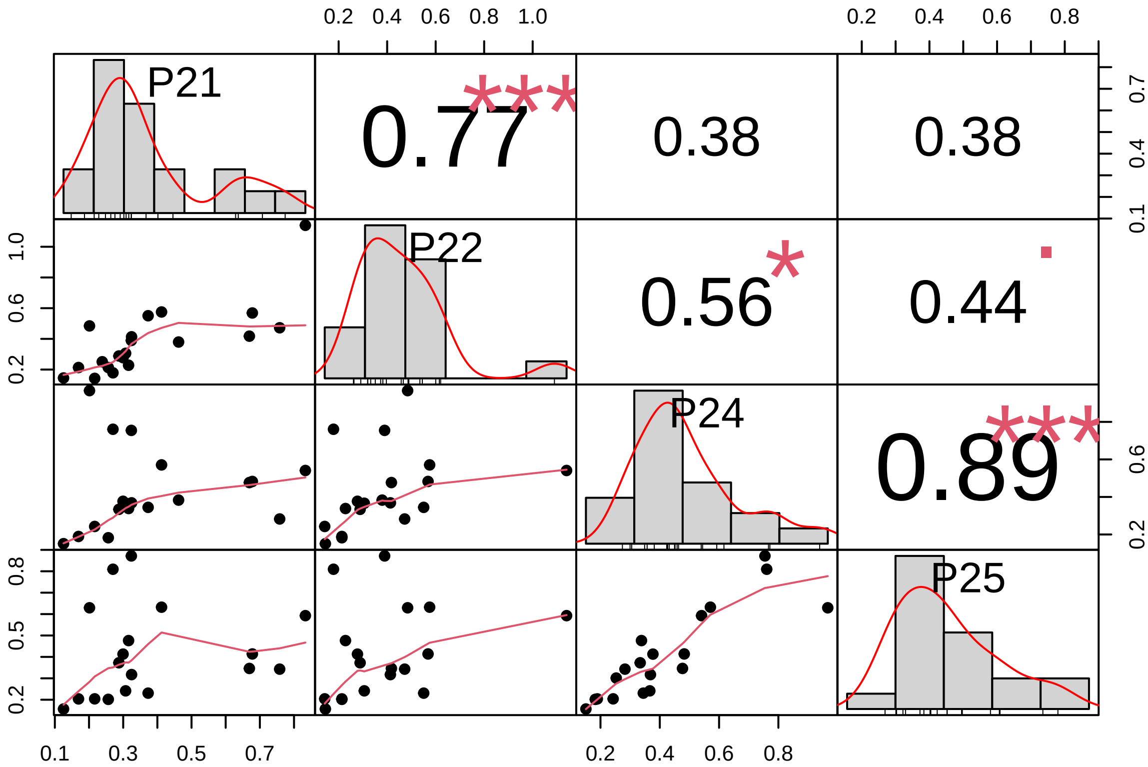
```r
#test normality assumptions
#normality assumption ; rejected with exception of P25
chlbtl %>%
   group_by(Station) %>%
  shapiro_test(Tchl_a)
```

```
## # A tibble: 4 x 4
##   Station variable statistic        p
##   <chr>   <chr>        <dbl>    <dbl>
## 1 P21     Tchl_a       0.888 0.00605
## 2 P22     Tchl_a       0.849 0.000726
## 3 P24     Tchl_a       0.909 0.0189
## 4 P25     Tchl_a       0.933 0.0718
```

```r
#normality assumption ; rejected with exception of winter
chlbtl %>%
  group_by(TimeofYear) %>%
  shapiro_test(Tchl_a)
```

```
## # A tibble: 3 x 4
##   TimeofYear variable statistic        p
##   <chr>      <chr>        <dbl>    <dbl>
## 1 Spring     Tchl_a       0.886 0.00140
## 2 Summer     Tchl_a       0.866 0.000113
## 3 Winter     Tchl_a       0.965 0.361
```

```r
#so probably best to use Spearman.

library("PerformanceAnalytics")
# chart.Correlation(s12d_subset_ss[4:7], histogram=TRUE, pch=19, method = "pearson")
chart.Correlation(s12d_subset_ss[4:7], histogram=TRUE, pch=19, method = "spearman")
```




```
#P24 and P25 correlated (0.89) and P21 and P22 correlated (0.77) P22 and P24 also, but weaker (0.56, 0.

chlbtl %>%
  group_by(Station) %>%
  get_summary_stats(Tchl_a, type = "mean_sd")
```

```
## # A tibble: 4 x 5
##   Station variable     n  mean    sd
##   <chr>   <fct>    <dbl> <dbl> <dbl>
## 1 P21     Tchl_a      28 0.377 0.174
## 2 P22     Tchl_a      29 0.393 0.197
## 3 P24     Tchl_a      28 0.417 0.184
## 4 P25     Tchl_a      28 0.409 0.18
```

```
#check for outliers

chlbtl %>%
  group_by(Station) %>%
  identify_outliers(Tchl_a)
```

```
## # A tibble: 7 x 20
##   Station Cruise    Year Month TimeofYear Date    YearSeason  Longitude Latitude
##   <chr>   <chr>    <dbl> <fct> <chr>      <chr>   <chr>           <dbl>    <dbl>
## 1 P21     2018_26   2018 6     Spring     2018-06 2018-Spring     -140.     49.6
## 2 P21     2019_006  2019 6     Spring     2019-06 2019-Spring     -140.     49.6
```

```
## 3 P22     2019_006  2019 6     Spring     2019-06 2019-Spring     -141.     49.7
## 4 P24     2014_19   2014 8     Summer     2014-08 2014-Summer     -143.     49.8
## 5 P24     2017_08   2017 8     Summer     2017-08 2017-Summer     -143.     49.8
## 6 P24     2019_008  2019 8     Summer     2019-08 2019-Summer     -143.     49.8
## 7 P25     2019_008  2019 8     Summer     2019-08 2019-Summer     -144.     50.0
## # i 11 more variables: Cyanobacteria <dbl>, Chlorophytes <dbl>,
## #   Prasinophytes <dbl>, Cryptophytes <dbl>, `Diatom-2` <dbl>,
## #   `Dinoflage-1` <dbl>, Pelagophytes <dbl>, Haptophytes <dbl>, Tchl_a <dbl>,
## #   is.outlier <lgl>, is.extreme <lgl>
```

```
#there is an extreme outlier P22, 2019-06 - higher than other stations.

chlbtl %>%
  group_by(TimeofYear) %>%
  identify_outliers(Tchl_a)
```

```
## # A tibble: 7 x 20
##   TimeofYear Cruise    Year Month Date    YearSeason  Longitude Latitude Station
##   <chr>      <chr>    <dbl> <fct> <chr>   <chr>           <dbl>    <dbl> <chr>
## 1 Spring     2019_006  2019 6     2019-06 2019-Spring     -141.     49.7 P22
## 2 Summer     2014_19   2014 8     2014-08 2014-Summer     -143.     49.8 P24
## 3 Summer     2017_08   2017 8     2017-08 2017-Summer     -143.     49.8 P24
## 4 Summer     2019_008  2019 8     2019-08 2019-Summer     -143.     49.8 P24
## 5 Summer     2017_08   2017 8     2017-08 2017-Summer     -144.     50.0 P25
## 6 Summer     2019_008  2019 8     2019-08 2019-Summer     -144.     50.0 P25
## 7 Winter     2018_01   2018 2     2018-02 2018-Winter     -141.     49.7 P22
## # i 11 more variables: Cyanobacteria <dbl>, Chlorophytes <dbl>,
## #   Prasinophytes <dbl>, Cryptophytes <dbl>, `Diatom-2` <dbl>,
## #   `Dinoflage-1` <dbl>, Pelagophytes <dbl>, Haptophytes <dbl>, Tchl_a <dbl>,
## #   is.outlier <lgl>, is.extreme <lgl>
```

```
#normality assumption ; rejected.
chlbtl %>%
   group_by(Station) %>%
  shapiro_test(Tchl_a) #only p25 is normal
```

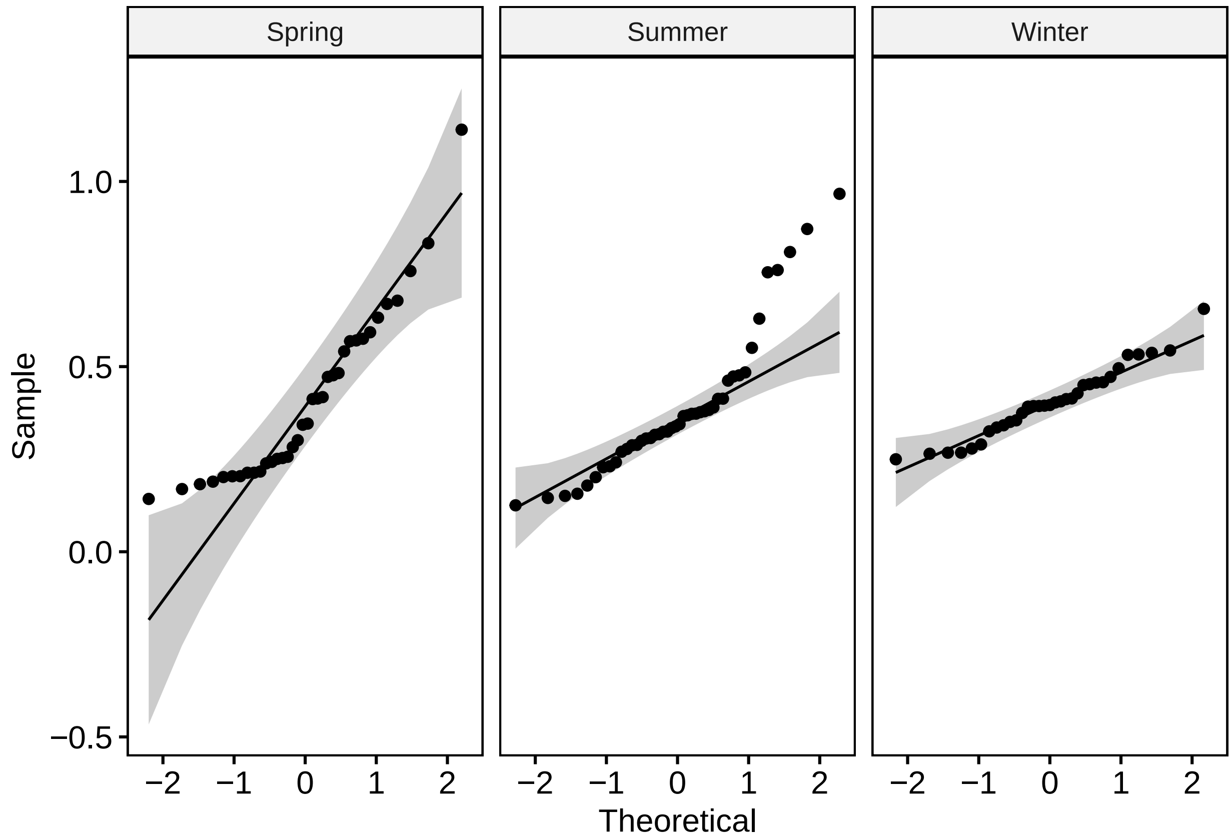
```
## # A tibble: 4 x 4
##   Station variable statistic        p
##   <chr>   <chr>        <dbl>    <dbl>
## 1 P21     Tchl_a       0.888 0.00605
## 2 P22     Tchl_a       0.849 0.000726
## 3 P24     Tchl_a       0.909 0.0189
## 4 P25     Tchl_a       0.933 0.0718
```

```
chlbtl %>%
  group_by(TimeofYear) %>%
  shapiro_test(Tchl_a) #only winter is normal
```

```
## # A tibble: 3 x 4
##   TimeofYear variable statistic       p
##   <chr>      <chr>        <dbl>   <dbl>
## 1 Spring     Tchl_a       0.886 0.00140
```

```
## 2 Summer     Tchl_a       0.866 0.000113
## 3 Winter     Tchl_a       0.965 0.361
```

```r
ggqqplot(chlbtl, "Tchl_a", facet.by = "TimeofYear")
```



Given the results above, we can say the following to inform that statement: While individual profiles were strongly correlated regionally (P21 and P22, rho(28)= 0.77, adjusted P(Bonferroni)< 0.001; P24 and P25, rho(27) = 0.89, adjusted P(Bonferroni)< 0.001), there was weaker correspondence between more distant stations, reflecting the inherently patchy nature of chlorophyll distributions, these results collectively indicate that the observed trends in POC production and accumulation were not driven by float-specific or spatial biases.