METU AQUACOSM 2019 experiment - Available code

Caldero-Pascual et al.

April 2021

{r} –> final report with both code and results {r eval = FALSE} –> final report only copy of code (without running it) {r echo = FALS} –> final report only copy of results (without code)

Load packages

```
library(ggplot2)
library(cowplot)
library(gridExtra)
library(tidyverse)
library(dplyr)

#set working directory to input files
setwd("C:/Users/calderom/OneDrive - Dundalk Institute of
Technology/Aquacosm_TA/HydrobiologiaJournal/Code")
```

Fig. 1

Water quality dissolved, seston and total C, N, and P fractions during the mesocosm day of experiment (DOE; gaps when data not available): a) DOC; b) seston C; c) colour; d) DIN; e) seston N; f) TN; g) SRP; h) seston P; and i) TP. Each point represents an observation (4 replicates per treatment per DOE). Treatments are represented in different shapes and colours: controls (Cntl) blue points, HuminFeed (HF) grey triangles, leaf leachate (L) yellow diamonds, and combination of sources (HFL) red squares. Black dashed vertical line indicates when the allo-OM pulse event occurred. Following grey dotted vertical lines separate each showed sampling day.

Input data:

```
dat <- read.csv("Fig1.csv", stringsAsFactors = T)
dat$DOE <- as.factor(dat$DOE)
str(dat)
dat$Treatment <- ordered(dat$Treatment, levels=c("Cntl", "HF", "L","HFL"))
```

Plot:

```
#DOC
P1 <- ggplot(dat, aes(x = DOE, y = DOC, fill=Treatment, shape = Treatment)) +
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
```

```
  ylab("DOC (mgC/L)")+
  ylim(0,12)+
  ggtitle("DISSOLVED") +
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#SestonC
P2 <- ggplot(dat, aes(x = DOE, y = SestonC, fill=Treatment, shape =
Treatment)) +
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("Seston C (mgC/L)")+
  ylim(0,2.5)+
  ggtitle("SESTON") +
  theme(text = element_text(size = 8),axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#Colour
P3 <- ggplot(dat, aes(x = DOE, y = Colour, fill=Treatment, shape =
Treatment)) +
  scale_fill_manual(name = "Treatment",values =
```

```
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("Colour 420nm (AU/cm)")+
  ggtitle("TOTAL") +
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#DIN
P4 <- ggplot(dat, aes(x = DOE, y = DIN, fill = Treatment, shape = Treatment))
+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("DIN (ugN/L)")+
  ylim(0,250)+
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#SestonN
P5 <- ggplot(dat, aes(x = DOE, y = SestonN, fill = Treatment, shape =
Treatment)) +
  scale_fill_manual(name = "Treatment",values =
```

```
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("Seston N (ugN/L)")+
  ylim(0,2000)+
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#TN
P6 <- ggplot(dat, aes(x = DOE, y = TN, fill = Treatment, shape = Treatment))
+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("TN (ugN/L)")+
  ylim(0,800)+
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#SRP
P7 <- ggplot(dat, aes(x = DOE, y = SRP, fill = Treatment, shape = Treatment))
+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
```

```
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("SRP (ugP/L)")+
  ylim(0,40)+
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#SestonP
P8 <- ggplot(dat, aes(x = DOE, y = SestonP, fill = Treatment, shape =
Treatment)) +
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("Seston P (ugP/L)")+
  ylim(0,80)+
  theme(text = element_text(size = 8),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#TP
P9 <- ggplot(dat, aes(x = DOE, y = TP, fill = Treatment, shape = Treatment))
+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 2, position = position_jitterdodge( jitter.width = 0,
dodge.width = 1.1)) +
  ylab("TP (ugP/L)")+
  theme(text = element_text(size = 8), axis.title.x=element_blank(),
```

```
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=1.5, linetype=4, size=1, colour="black")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=3.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=4.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=5.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=6.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=7.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=8.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=9.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=10.5, linetype="dotted", size=0.5, colour="grey60")
#Final Plot
Fig1 <- plot_grid(P1, P2, P3,
          P4, P5, P6,
          P7, P8, P9,
          align="hv", axis="tblr",
          ncol = 3,
          labels =c("a","b","c","d","e","f", "g", "h", "i"))
#Fig1
ggsave("Fig1.jpeg", width = 24, height = 12, units = "cm")
```

Fig. 2

Distribution of the intensity, in Raman Units (RU), of different PARAFAC components (C3, C1, C4, C2) in the four different treatments (Cntl: controls, HF: HuminFeed, L: leaf leachate and HFL: combination) from day 4 to day 24 of the experiment (n = 24 per treatment). The order and the colours of each PARAFAC component are based on the information provided in Table 1. From high recalcitrant humic-like component (dark orange: C3), less recalcitrant terrestrial fulvic-like component (orange: C1), microbial derived humic-like component (yellow: C4), to more labile protein-like component (green: C2). Box = 25th and 75th percentiles, whiskers = 1.5*inter-quartile range. Black line = median.

Input data:

```
parafac <- read.csv("Fig2.csv", stringsAsFactors = T)
dat_parafac <- parafac %>%
  select(Treatment, C3.RU,  C1.RU, C4.RU, C2.RU)
pivot.parafac <- dat_parafac %>%
  pivot_longer(-Treatment,
               names_to = "variable",
               values_to = "value")
pivot.parafac$Treatment  <- ordered(pivot.parafac$Treatment, levels=c("Cntl",
"HF", "L","HFL"))
pivot.parafac$variable  <- ordered(pivot.parafac$variable, levels=c("C3.RU",
"C1.RU", "C4.RU","C2.RU"))
```

Plot:

```
Fig2 <- ggplot(pivot.parafac, aes(x=Treatment, y=value, fill=variable)) +
  geom_boxplot() +
  geom_point(pch = 21, position = position_jitterdodge(jitter.width = 0,
dodge.width = 0.75))+
  scale_fill_manual(values =
c("orangered3","orange3","yellow3","olivedrab3"))+
  facet_wrap(~variable, scale="free")+
  ylab("Intensity (RU)")+
  theme(text = element_text(size = 10),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")
Fig2
ggsave("Fig2.jpeg", width = 15, height = 10, units = "cm")
```

Fig. 3

Two-dimensional NMDS of the PARAFAC components based on 4th root transformed relative intensity of the mean of replicates per day of experiment. Treatment observations are represented in different colours and shapes: controls (Cntl) blue points, HuminFeed? (HF) grey triangles, leaf leachate (L) yellow diamonds, and combination (HFL) red squares. Specific day of experiment (4, 8, 12, 16, 20 and 24) is detailed in each observation point. The stress of the ordination is 0.026. Significant differences among treatments (pairwise adonis test with p-values < 0.05)

Extra required packages

```
#for NMDS statistics
library(MASS)
library(vegan)
#to plot NMDS
library(ggrepel)
library(grid)
```

Input data and NMDS:

```
bio <- read.csv("Fig3.csv", head = TRUE, check.names = FALSE, row.names = 1)

set.seed(50) #always at the beginning before any random numbers generating
function to get a reproducible random result
            #for example, when metaMDS using permutation tests!

bio1 <- subset(bio, select=-c(Treatment,DOE))
str(bio1)

#-c(x) --> removes column of the input file
#I did it here because these columns will be helpful to plot groups but
#it is not possible to run the metaMDS function with these group columns

fm <- metaMDS(bio1, autotransform = FALSE)
```

```
#stress value
fm

#Significant differences among treatments
library(pairwiseAdonis)
pairwise.adonis(bio[,1:4], bio$Treatment)
```

Plot with ggplot:

```
#data.scores
data.scores <- as.data.frame(scores(fm))  #Using the scores function from
vegan to extract the site scores and convert to a data.frame
data.scores$site <- rownames(data.scores)  # create a column of site names,
from the rownames of data.scores
data.scores$Treatment <- bio$Treatment  #  add the grp variable created
earlier
data.scores$doe <- bio$DOE  #  add the group variable created earlier
head(data.scores)  #look at the data
#species.scores
species.scores <- as.data.frame(scores(fm, "species"))  #Using the scores
function from vegan to extract the species scores and convert to a data.frame
species.scores$species <- rownames(species.scores)  # create a column of
species, from the rownames of species.scores
head(species.scores)  #look at the data

data.scores$Treatment <- ordered(data.scores$Treatment, levels=c("Cntl",
"HF", "L","HFL"))

Fig3 <- ggplot() +
  geom_point(data=data.scores,aes(x=NMDS1,y=NMDS2,fill=Treatment,
shape=Treatment),size=6, alpha=1) + # add the point markers
  scale_fill_manual(values=c("Cntl" = "turquoise", "HF" =
"grey50","L"="yellow", "HFL"="red")) + #manual colours for date points
  scale_shape_manual(values =c(21,24,23,22))+
  geom_hline(yintercept = 0, lty = 2) +
  geom_vline(xintercept = 0, lty = 2) +
  xlim(-0.2, 0.15)+
  ylim(-0.1,0.1)+

geom_text(data=species.scores,aes(x=NMDS1,y=NMDS2,label=species),fontface="bo
ld", size=6) +  # add the species labels
  geom_text(data=data.scores,aes(x=NMDS1,y=NMDS2,label=doe),size=4,vjust=2,
alpha = 1) +  # add the date labels
  coord_equal() + #important for the dimension of the NMDS
  theme_bw()+
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),  #remove major-grid labels
        panel.grid.minor = element_blank(),  #remove minor-grid labels
        plot.background = element_blank(),
        legend.position="right")
```

```
Fig3
ggsave("Fig3.jpeg", width = 15, height = 10, units = "cm")
```

Fig. S1

Two-dimensional NMDS of the PARAFAC components based on 4th root transformed relative intensity of the four replicates per day of experiment. Treatment observations are represented in different colours and shapes: controls (Cntl) blue points, HuminFeed? (HF) grey triangles, leaf leachate (L) yellow diamonds, and combination (HFL) red squares. Specific day of experiment (4, 8, 12, 16, 20 and 24) is detailed in each observation point (overlapping labels have been removed). The stress of the ordination is 0.025. Significant differences among treatments (pairwise adonis test with p-values < 0.05)

Input data and NMDS:

```
bio <- read.csv("FigS1.csv", head = TRUE, check.names = FALSE, row.names = 1)

set.seed(50) #always at the beginning before any random numbers generating
function to get a reproducible random result
            #for example, when metaMDS using permutation tests!

bio1 <- subset(bio, select=-c(Treatment,DOE))
str(bio1)

#-c(x) --> removes column of the input file
#I did it here because these columns will be helpful to plot groups but
#it is not possible to run the metaMDS function with these group columns

fm <- metaMDS(bio1, autotransform = FALSE)

#stress value
fm

#Significant differences among treatments
library(pairwiseAdonis)
pairwise.adonis(bio[,1:4], bio$Treatment)
```

Plot with ggplot:

```
#data.scores
data.scores <- as.data.frame(scores(fm))  #Using the scores function from
vegan to extract the site scores and convert to a data.frame
data.scores$site <- rownames(data.scores)  # create a column of site names,
from the rownames of data.scores
data.scores$Treatment <- bio$Treatment  #  add the grp variable created
earlier
data.scores$doe <- bio$DOE  #  add the group variable created earlier
head(data.scores)  #look at the data
#species.scores
species.scores <- as.data.frame(scores(fm, "species"))  #Using the scores
```

```
function from vegan to extract the species scores and convert to a data.frame
species.scores$species <- rownames(species.scores)  # create a column of
species, from the rownames of species.scores
head(species.scores)  #look at the data

data.scores$Treatment <- ordered(data.scores$Treatment, levels=c("Cntl",
"HF", "L","HFL"))

FigS1 <- ggplot() +
  geom_point(data=data.scores,aes(x=NMDS1,y=NMDS2,fill=Treatment,
shape=Treatment),size=6, alpha=1) + # add the point markers
  scale_fill_manual(values=c("Cntl" = "turquoise", "HF" =
"grey50","L"="yellow", "HFL"="red")) + #manual colours for date points
  scale_shape_manual(values =c(21,24,23,22))+
  geom_hline(yintercept = 0, lty = 2) +
  geom_vline(xintercept = 0, lty = 2) +
  xlim(-0.8, 0.6)+
  ylim(-0.3,0.4)+

geom_text(data=species.scores,aes(x=NMDS1,y=NMDS2,label=species),fontface="bo
ld", size=6) +  # add the species labels
  geom_text(data=data.scores,aes(x=NMDS1,y=NMDS2,label=doe),size=4,vjust=2,
alpha = 0.5, check_overlap = T) +  # add the date labels
  coord_equal() + #important for the dimension of the NMDS
  theme_bw()+
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),  #remove major-grid labels
        panel.grid.minor = element_blank(),  #remove minor-grid labels
        plot.background = element_blank(),
        legend.position="right")
FigS1
ggsave("FigS1.jpeg", width = 15, height = 8, units = "cm")
```

Fig. 4, 5 & 8

Ratios - GAMM/LOESS outputs

Fig. 4 GAMM results showing temporal trends of ln-transformed seston C:P ratios during the days of the experiment (DOE). Solid lines are 'loess' smoothers and points are specific observations for each respective treatment (Cntl: blue dots; HF: grey triangles; L: yellow diamonds; HFL: red squares). Green dashed horizontal line indicates the Redfield ratio (lnC:P = 4.66) and pink one the threshold elemental ratio for seston P nutrient deficiency, lnC:P > 5.56 (Healey and Hendzel, 1980). Black dashed vertical line indicates when the allo-OM pulse event occurred.

Fig. 5 GAM results showing temporal trends of ln-transformed seston N:P ratios during the days of the experiment (DOE). Solid lines are 'loess' smoothers and points are specific observations for each respective treatment (Cntl: blue dots; HF: grey triangles; L: yellow diamonds; HFL: red squares). Green dashed horizontal line indicates the Redfield ratio

(lnN:P = 2.77) and pink one the threshold elemental ratio for seston P nutrient deficiency, lnN:P > 3.09 (Healey and Hendzel, 1980). Black dashed vertical line indicates when the allo-OM pulse event occurred.

Fig. 8 GAM results showing temporal trends of Chl-a concentrations during the days of the experiment (DOE). Solid lines are 'loess' smoothers and points are specific observations for each respective treatment (Cntl: blue dots; HF: grey triangles; L: yellow diamonds; HFL: red squares). Black dashed vertical line indicates when the allo-OM pulse event occurred.

Input data:

```
ratios <- read.csv("Fig4.5.8.csv", stringsAsFactors = T)
ratios$Treatment <- ordered(ratios$Treatment, levels=c("Cntl", "HF",
"L","HFL"))
ratios$Date <- as.Date(ratios$Date,"%d/%m/%Y")
str(ratios)
```

Plots

```
#ln sestonC:P
Fig4 <- ggplot(ratios, aes(x = DOE, y = lnCP, fill=Treatment, shape =
Treatment)) +
  geom_smooth(method = "loess", se=FALSE, aes(colour = Treatment), size = 2)+
  scale_colour_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 3, position = position_jitterdodge( jitter.width = 0,
dodge.width = 0)) +
    ylab("ln(seston C:P)")+
  scale_x_continuous(limits = c(0, 36), breaks = seq(0, 36, by = 4))+
  theme(text = element_text(size = 8),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=0.5, linetype=4, size=1, colour="black")+
  geom_hline(yintercept=5.56, linetype = "dashed", size=0.5,
colour="deeppink")+
  geom_hline(yintercept=4.66, linetype = "dashed", size=0.5, colour="green")
Fig4
ggsave("Fig4.jpeg", width = 12, height = 7, units = "cm")

#ln sestonN:P
Fig5 <- ggplot(ratios, aes(x = DOE, y = lnNP, fill=Treatment, shape =
Treatment)) +
  geom_smooth(method = "loess", se=FALSE, aes(colour = Treatment), size = 2)+
  scale_colour_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
```

```
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 3, position = position_jitterdodge( jitter.width = 0,
dodge.width = 0)) +
    ylab("ln(seston N:P)")+
  scale_x_continuous(limits = c(0, 36), breaks = seq(0, 36, by = 4))+
  theme(text = element_text(size = 8),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=0.5, linetype=4, size=1, colour="black")+
  geom_hline(yintercept=3.14, linetype = "dashed", size=0.5,
colour="deeppink")+
  geom_hline(yintercept=2.77, linetype = "dashed", size=0.5, colour="green")
Fig5
ggsave("Fig5.jpeg", width = 12, height = 7, units = "cm")

#Chl-a
Fig8 <- ggplot(ratios, aes(x = DOE, y = Chla.ugL, fill=Treatment, shape =
Treatment)) +
  geom_smooth(method = "loess", se=FALSE, aes(colour = Treatment), size = 2)+
  scale_colour_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_fill_manual(name = "Treatment",values =
c("turquoise","grey50","yellow","red"))+
  scale_shape_manual(name = "Treatment", values=c(21,24,23,22))+
  geom_point(size = 3, position = position_jitterdodge( jitter.width = 0,
dodge.width = 0)) +
    ylab("Chlorophyll-a (ugChla/L)")+
  scale_x_continuous(limits = c(0, 36), breaks = seq(0, 36, by = 4))+
  theme(text = element_text(size = 8),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "none")+
  geom_vline(xintercept=0.5, linetype=4, size=1, colour="black")
Fig8
ggsave("Fig8.jpeg", width = 12, height = 7, units = "cm")
```

Fig. 6

PCA reveals differences in treatments (A) based on different carbon and nutrient quantity and quality water chemistry parameters (B). Symbols of different colours represent different treatment observations, where blue points represent control treatment (Cntl), grey triangles HuminFeed (HF), yellow diamonds leaf leachate (L), and red squares combined treatment (HFL).

Input data:

```
ordination <- read.csv("Fig6.csv", stringsAsFactors = T, header=T, row.names
= 1)
dat_pca <- subset(ordination, select=-Treatment)
str(dat_pca)
summary(dat_pca) #no NA values = OK! #NA VALUES not accepted
```

PCA analysis:

```
pca <- prcomp(dat_pca, scale=TRUE, center = TRUE)
summary(pca) #proportion of PCs
```

Plot with ggplot:

```
# extract PC scores for first two component and add to "ordination" dataframe
ordination$pc1 <- pca$x[,1] # indexing the first column
ordination$pc2 <- pca$x[,2]  # indexing the second column
pca.vars <- pca$rotation %>% data.frame
pca.vars$vars <- rownames(pca.vars)
ordination$Treatment <- ordered(ordination$Treatment, levels=c("Cntl", "HF",
"L","HFL"))

A <- ggplot(data = ordination, aes(x = pc1, y = pc2, fill=Treatment, shape =
Treatment)) +
    geom_hline(yintercept = 0, lty = 2) +
    geom_vline(xintercept = 0, lty = 2) +
    scale_shape_manual(values =c(21,24,23,22))+
  scale_fill_manual(values = c("Cntl" = "turquoise", "HF" =
"grey50","L"="yellow", "HFL"="red")) +
  geom_point(size=5, alpha = 1)+
  ylab("PCA 2 (14.80%)")+
  xlab("PCA 1 (49.68%)")+
  theme_bw()+
  theme(panel.background = element_blank(),
        panel.grid.major =element_blank(),  #remove major-grid labels
        panel.grid.minor = element_blank(),  #remove minor-grid labels
        plot.background = element_blank())
B <- ggplot() +
  geom_text(data = pca.vars, aes(x = PC1*1.15, y =  PC2*1.15),
            label = c("DOC", "DIN", "SRP", "TN", "TP", "C seston", "N
seston","P seston",
                      "C:P","N:P", "Colour",
                      "C3","C1","C4","C2"),
            nudge_y = 0, nudge_x = 0,
            check_overlap = F, size = 4) +
  geom_segment(data = pca.vars, aes(x = 0, xend = PC1, y = 0, yend = PC2),
               arrow = arrow(length = unit(0.025,"npc"), type = "open"), lwd
= 0.5) +
  geom_hline(yintercept = 0, lty = 2) +
  geom_vline(xintercept = 0, lty = 2) +
  xlim(-0.4, 0.5)+
  theme_bw()+
```

```
  theme(panel.background = element_blank(),
        panel.grid.major =element_blank(),  #remove major-grid labels
        panel.grid.minor = element_blank(),  #remove minor-grid labels
        plot.background = element_blank(),
        axis.title.x=element_blank(),
        axis.title.y=element_blank())

Fig6 <- plot_grid(A, B, align = "h", labels = c("a", "b"),
          ncol = 2, rel_widths=c(1.2,1))
Fig6
ggsave("Fig6.jpeg", width = 18, height = 12, units = "cm")
```

Fig. 7

Effect size (LRR) of the three different treatments to different water chemistry quantity and quality parameters. A) Carbon quantity: DOC, seston C and colour; B) nutrient quantity: DIN, seston N, SRP and seston P; C) DOM quality: PARAFAC components; and D) seston quality: C:P and N:P ratios. Dashed horizontal black lines at 0 indicate the general benchmark (notice than when available we used benchmark of day 0, close to 0 but not represented visually).

Input data and plots:

```
#A: Carbon quantity
dat7A <- read.csv("Fig7A.csv", stringsAsFactors = T)
dat7A$Treatment  <- ordered(dat7A$Treatment, levels=c("HF", "L", "HFL"))
dat7A$Variable  <- ordered(dat7A$Variable, levels=c("DOC", "Seston C",
"Colour"))
A <- ggplot(dat7A, aes(x = Treatment, y = Effect.Size, fill=Variable)) +
  geom_hline(yintercept=0, linetype = "dashed", size = 1, colour="black")+
  scale_fill_viridis_d() +
  geom_boxplot(width = 0.5) +
  ylab("Effect size (LRR)")+
  ylim(-0.5,3.5)+
  xlab("Treatment") +
  ggtitle("a) Carbon quantity") +
  theme(text = element_text(size = 10),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "right")+
   geom_vline(xintercept=1.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")
#B: Nutrients quantity
dat7B <- read.csv("Fig7B.csv", stringsAsFactors = T)
dat7B$Treatment  <- ordered(dat7B$Treatment, levels=c("HF", "L", "HFL"))
dat7B$Variable  <- ordered(dat7B$Variable, levels=c("DIN", "Seston N", "SRP",
"Seston P"))
B <- ggplot(dat7B, aes(x = Treatment, y = Effect.Size, fill=Variable)) +
    geom_hline(yintercept=0, linetype = "dashed", size = 1, colour="black")+
```

```
    scale_fill_viridis_d() +
    geom_boxplot(width = 0.6) +
    ylab("Effect size (LRR)")+
    ylim(-1,4)+
    xlab("Treatment") +
    ggtitle("b) Nutrient quantity") +
    theme(text = element_text(size = 10),
          plot.title = element_text(color="black",size = 10,
face="bold.italic"),
          panel.background = element_rect(fill = 'white', colour = 'black'),
          legend.position = "right")+
   geom_vline(xintercept=1.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")
#C: DOC quality --> PARAFAC components
dat7C <- read.csv("Fig7C.csv", stringsAsFactors = T)
dat7C$Treatment  <- ordered(dat7C$Treatment, levels=c("HF", "L", "HFL"))
dat7C$Variable  <- ordered(dat7C$Variable, levels=c("C3", "C1", "C4", "C2"))
C <- ggplot(dat7C, aes(x = Treatment, y = Effect.Size, fill=Variable)) +
    geom_hline(yintercept=0, linetype = "dashed", size = 1, colour="black")+
    scale_fill_viridis_d() +
    geom_boxplot(width = 0.9) +
    ylab("Effect size (LRR)")+
    ylim(-1,2.5)+
    xlab("Treatment") +
    ggtitle("c) DOM quality") +
    theme(text = element_text(size = 10),
          plot.title = element_text(color="black",size = 10,
face="bold.italic"),
          panel.background = element_rect(fill = 'white', colour = 'black'),
          legend.position = "right")+
   geom_vline(xintercept=1.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")
#D: Seston quality
dat7D <- read.csv("Fig7D.csv", stringsAsFactors = T)
dat7D$Treatment  <- ordered(dat7D$Treatment, levels=c("HF", "L", "HFL"))
D <- ggplot(dat7D, aes(x = Treatment, y = Effect.Size, fill=Variable)) +
  geom_hline(yintercept=0, linetype = "dashed", size = 1, colour="black")+
 scale_fill_viridis_d() +
  geom_boxplot(width = 0.3) +
  ylab("Effect size (LRR)")+
   ylim(-2,1)+
  xlab("Treatment") +
  ggtitle("d) Seston quality") +
  theme(text = element_text(size = 10),
        plot.title = element_text(color="black",size = 10,
face="bold.italic"),
        panel.background = element_rect(fill = 'white', colour = 'black'),
        legend.position = "right")+
   geom_vline(xintercept=1.5, linetype="dotted", size=0.5, colour="grey60")+
  geom_vline(xintercept=2.5, linetype="dotted", size=0.5, colour="grey60")
```

```
#Final plot
Fig7 <-  plot_grid(A,B,
                   C,D, align = "v", ncol = 2)
Fig7
ggsave("Fig7.jpeg", width = 20, height = 15, units = "cm")
```

SUPPLEMENTARY MATERIAL

ANOVAS for parameters presented in Fig.1

Table S1

```
library(rstatix)

Fig1pairtest <- read.csv("Fig1.csv", stringsAsFactors = T)
Fig1pairtest$DOE <- as.factor(Fig1pairtest$DOE)

Fig1pairtest.long <- Fig1pairtest %>%
  pivot_longer(cols = -c(Treatment,DOE), names_to = "variables", values_to =
"value")

D0_DOC <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="DOC")
D0_DOC.test <- D0_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_DOC.test

D1_DOC <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="DOC")
D1_DOC.test <- D1_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_DOC.test

D8_DOC <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="DOC")
D8_DOC.test <- D8_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_DOC.test

D12_DOC <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="DOC")
D12_DOC.test <- D12_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
```

```
  add_significance()
D12_DOC.test

D16_DOC <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="DOC")
D16_DOC.test <- D16_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_DOC.test

D20_DOC <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="DOC")
D20_DOC.test <- D20_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_DOC.test

D28_DOC <- Fig1pairtest.long %>%
  filter(DOE == "28" & variables =="DOC")
D28_DOC.test <- D28_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D28_DOC.test

D32_DOC <- Fig1pairtest.long %>%
  filter(DOE == "32" & variables =="DOC")
D32_DOC.test <- D32_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D32_DOC.test

D36_DOC <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="DOC")
D36_DOC.test <- D36_DOC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_DOC.test

D0_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="SestonC")
D0_sestonC.test <- D0_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
```

```
D0_sestonC.test

D1_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="SestonC")
D1_sestonC.test <- D1_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_sestonC.test

D4_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="SestonC")
D4_sestonC.test <- D4_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_sestonC.test

D8_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="SestonC")
D8_sestonC.test <- D8_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_sestonC.test

D12_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="SestonC")
D12_sestonC.test <- D12_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_sestonC.test

D16_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="SestonC")
D16_sestonC.test <- D16_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_sestonC.test

D20_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="SestonC")
D20_sestonC.test <- D20_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_sestonC.test
```

```r
D24_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="SestonC")
D24_sestonC.test <- D24_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_sestonC.test

D36_sestonC <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="SestonC")
D36_sestonC.test <- D36_sestonC %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_sestonC.test

D4_colour <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="Colour")
D4_colour.test <- D4_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_colour.test

D8_colour <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="Colour")
D8_colour.test <- D8_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_colour.test

D12_colour <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="Colour")
D12_colour.test <- D12_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_colour.test

D16_colour <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="Colour")
D16_colour.test <- D16_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_colour.test
```

```r
D20_colour <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="Colour")
D20_colour.test <- D16_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_colour.test

D24_colour <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="Colour")
D24_colour.test <- D24_colour %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_colour.test

D0_DIN <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="DIN")
D0_DIN.test <- D0_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_DIN.test

D1_DIN <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="DIN")
D1_DIN.test <- D1_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_DIN.test

D8_DIN <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="DIN")
D8_DIN.test <- D8_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_DIN.test

D12_DIN <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="DIN")
D12_DIN.test <- D12_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_DIN.test

D16_DIN <- Fig1pairtest.long %>%
```

```
  filter(DOE == "16" & variables =="DIN")
D16_DIN.test <- D16_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_DIN.test

D20_DIN <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="DIN")
D20_DIN.test <- D20_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_DIN.test

D28_DIN <- Fig1pairtest.long %>%
  filter(DOE == "28" & variables =="DIN")
D28_DIN.test <- D28_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D28_DIN.test

D32_DIN <- Fig1pairtest.long %>%
  filter(DOE == "32" & variables =="DIN")
D32_DIN.test <- D32_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D32_DIN.test

D36_DIN <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="DIN")
D36_DIN.test <- D36_DIN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_DIN.test

D0_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="SestonN")
D0_sestonN.test <- D0_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_sestonN.test

D1_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="SestonN")
```

```
D1_sestonN.test <- D1_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_sestonN.test

D4_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="SestonN")
D4_sestonN.test <- D4_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_sestonN.test

D8_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="SestonN")
D8_sestonN.test <- D8_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_sestonN.test

D12_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="SestonN")
D12_sestonN.test <- D12_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_sestonN.test

D16_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="SestonN")
D16_sestonN.test <- D16_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_sestonN.test

D20_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="SestonN")
D20_sestonN.test <- D20_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_sestonN.test

D24_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="SestonN")
D24_sestonN.test <- D24_sestonN %>%
```

```
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_sestonN.test

D36_sestonN <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="SestonN")
D36_sestonN.test <- D36_sestonN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_sestonN.test

D0_TN <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="TN")
D0_TN.test <- D0_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_TN.test

D1_TN <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="TN")
D1_TN.test <- D1_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_TN.test

D4_TN <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="TN")
D4_TN.test <- D4_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_TN.test

D8_TN <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="TN")
D8_TN.test <- D8_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_TN.test

D12_TN <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="TN")
D12_TN.test <- D12_TN %>%
  t_test(value ~ Treatment) %>%
```

```
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_TN.test

D16_TN <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="TN")
D16_TN.test <- D16_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_TN.test

D20_TN <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="TN")
D20_TN.test <- D20_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_TN.test

D24_TN <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="TN")
D24_TN.test <- D24_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_TN.test

D28_TN <- Fig1pairtest.long %>%
  filter(DOE == "28" & variables =="TN")
D28_TN.test <- D28_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D28_TN.test

D32_TN <- Fig1pairtest.long %>%
  filter(DOE == "32" & variables =="TN")
D32_TN.test <- D32_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D32_TN.test

D36_TN <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="TN")
D36_TN.test <- D36_TN %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
```

```
  add_significance()
D36_TN.test

D0_SRP <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="SRP")
D0_SRP.test <- D0_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_SRP.test

D1_SRP <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="SRP")
D1_SRP.test <- D1_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_SRP.test

D4_SRP <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="SRP")
D4_SRP.test <- D4_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_SRP.test

D8_SRP <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="SRP")
D8_SRP.test <- D8_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_SRP.test

D12_SRP <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="SRP")
D12_SRP.test <- D12_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_SRP.test

D16_SRP <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="SRP")
D16_SRP.test <- D16_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
```

```
D16_SRP.test

D20_SRP <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="SRP")
D20_SRP.test <- D20_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_SRP.test

D24_SRP <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="SRP")
D24_SRP.test <- D24_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_SRP.test

D28_SRP <- Fig1pairtest.long %>%
  filter(DOE == "28" & variables =="SRP")
D28_SRP.test <- D28_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D28_SRP.test

D32_SRP <- Fig1pairtest.long %>%
  filter(DOE == "32" & variables =="SRP")
D32_SRP.test <- D32_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D32_SRP.test

D36_SRP <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="SRP")
D36_SRP.test <- D36_SRP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_SRP.test

D0_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="SestonP")
D0_sestonP.test <- D0_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_sestonP.test
```

```r
D1_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="SestonP")
D1_sestonP.test <- D1_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_sestonP.test

D4_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="SestonP")
D4_sestonP.test <- D4_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_sestonP.test

D8_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="SestonP")
D8_sestonP.test <- D8_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_sestonP.test

D12_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "12" & variables =="SestonP")
D12_sestonP.test <- D12_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_sestonP.test

D16_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="SestonP")
D16_sestonP.test <- D16_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_sestonP.test

D20_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="SestonP")
D20_sestonP.test <- D20_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_sestonP.test
```

```r
D24_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="SestonP")
D24_sestonP.test <- D24_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_sestonP.test

D36_sestonP <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="SestonP")
D36_sestonP.test <- D36_sestonP %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_sestonP.test

D0_TP <- Fig1pairtest.long %>%
  filter(DOE == "0" & variables =="TP")
D0_TP.test <- D0_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D0_TP.test

D1_TP <- Fig1pairtest.long %>%
  filter(DOE == "1" & variables =="TP")
D1_TP.test <- D1_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D1_TP.test

D4_TP <- Fig1pairtest.long %>%
  filter(DOE == "4" & variables =="TP")
D4_TP.test <- D4_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D4_TP.test

D8_TP <- Fig1pairtest.long %>%
  filter(DOE == "8" & variables =="TP")
D8_TP.test <- D8_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D8_TP.test

D12_TP <- Fig1pairtest.long %>%
```

```
  filter(DOE == "12" & variables =="TP")
D12_TP.test <- D12_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D12_TP.test

D16_TP <- Fig1pairtest.long %>%
  filter(DOE == "16" & variables =="TP")
D16_TP.test <- D16_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D16_TP.test

D20_TP <- Fig1pairtest.long %>%
  filter(DOE == "20" & variables =="TP")
D20_TP.test <- D20_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D20_TP.test

D24_TP <- Fig1pairtest.long %>%
  filter(DOE == "24" & variables =="TP")
D24_TP.test <- D24_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D24_TP.test

D28_TP <- Fig1pairtest.long %>%
  filter(DOE == "28" & variables =="TP")
D28_TP.test <- D28_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D28_TP.test

D32_TP <- Fig1pairtest.long %>%
  filter(DOE == "32" & variables =="TP")
D32_TP.test <- D32_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D32_TP.test

D36_TP <- Fig1pairtest.long %>%
  filter(DOE == "36" & variables =="TP")
```

```
D36_TP.test <- D36_TP  %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
D36_TP.test
```

GAM/Ms COMBINATIONS

GAM/M model trials: https://fromthebottomoftheheap.net/2017/12/14/difference-splines-ii/

Extra required packages & import data:

```
library(mgcv)
library(gratia)

models <- read.csv("Fig4.5.8.csv", stringsAsFactors = T, header=T)
models$DOE <- as.numeric(models$DOE)
models$Date <- as.Date(models$Date, "%d/%m/%Y")
str(models)

#mutate to create ordered factors:
models <- mutate(models,oTreatment = ordered(Treatment, levels =
c("Cntl","HF","L","HFL")))
```

GAM/Ms seston C:P

Table S2

```
#Model A
CP.A <- gam(lnCP ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.A) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.A <- residuals(CP.A)
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.A
resCP.A<-Efull
ACF_CP.A<-acf(resCP.A, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

CP.A1 <- gamm(lnCP ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
              correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
```

```r
appraise(CP.A1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.A1 <- residuals(CP.A1$lme,type="normalized")
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.A1
resCP.A1<-Efull
ACF_CP.A1 <- acf(resCP.A1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected

#Output:
summary(CP.A1$gam) #R-sq.(adj) =  0.0416
AIC(CP.A1$lme) #65.49467
plot(CP.A1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)

#Model B
CP.B <- gam(lnCP ~ s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.B) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.B <- residuals(CP.B)
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.B
resCP.B<-Efull
ACF_CP.B<-acf(resCP.B, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

CP.B1 <- gamm(lnCP ~ s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
             correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.B1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.B1 <- residuals(CP.B1$lme,type="normalized")
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.B1
resCP.B1 <-Efull
ACF_CP.B1 <- acf(resCP.B1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
```

```
summary(CP.B1$gam) #R-sq.(adj) =  0.0241
AIC(CP.B1$lme) #78.47783
plot(CP.B1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(CP.B1$gam, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(CP.B1$gam, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(CP.B1$gam, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)

#Model C
CP.C <- gam(lnCP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.C) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.C <- residuals(CP.C)
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.C
resCP.C<-Efull
ACF_CP.C<-acf(resCP.C, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

CP.C1 <- gamm(lnCP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.C1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.C1 <- residuals(CP.C1$lme,type="normalized")
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.C1
resCP.C1 <-Efull
ACF_CP.C1 <- acf(resCP.C1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
summary(CP.C1$gam) #R-sq.(adj) =  0.559
AIC(CP.C1$lme) #43.98008
plot(CP.C1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)

#Model D
CP.D <- gam(lnCP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            family = gaussian,
```

```
            data = models, method = "REML")
#check assumptions:
appraise(CP.D) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.D <- residuals(CP.D)
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.D
resCP.D<-Efull
ACF_CP.D<-acf(resCP.D, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

CP.D1 <- gamm(lnCP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(CP.D1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resCP.D1 <- residuals(CP.D1$lme,type="normalized")
I1<-!is.na(models$lnCP)
Efull<-vector(length = length(models$lnCP))
Efull<-NA
Efull[I1]<-resCP.D1
resCP.D1 <-Efull
ACF_CP.D1 <- acf(resCP.D1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
summary(CP.D1$gam) #R-sq.(adj) =  0.55
AIC(CP.D1$lme) #58.42054
plot(CP.D1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(CP.D1$gam, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(CP.D1$gam, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(CP.D1$gam, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)
```

GAM/Ms seston N:P

Table S3

```
#Model A
NP.A <- gam(lnNP ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(NP.A) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.A <- residuals(NP.A)
```

```r
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.A
resNP.A<-Efull
ACF_NP.A<-acf(resNP.A, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

NP.A1 <- gamm(lnNP ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
              correlation = corAR1(form = ~ 1 | DOE),
             family = gaussian,
             data = models, method = "REML")
#check assumptions:
appraise(NP.A1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.A1 <- residuals(NP.A1$lme,type="normalized")
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.A1
resNP.A1<-Efull
ACF_NP.A1 <- acf(resNP.A1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected

#Output:
summary(NP.A1$gam) #R-sq.(adj) =  0.258
AIC(NP.A1$lme) #208.4655
plot(NP.A1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)

#Model B
NP.B <- gam(lnNP ~ s(DOE, fx = FALSE, k = 6, bs="cr")
             + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
             family = gaussian,
             data = models, method = "REML")
#check assumptions:
appraise(NP.B) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.B <- residuals(NP.B)
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.B
resNP.B<-Efull
ACF_NP.B<-acf(resNP.B, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

NP.B1 <- gamm(lnNP ~ s(DOE, fx = FALSE, k = 6, bs="cr")
             + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
              correlation = corAR1(form = ~ 1 | DOE),
```

```
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(NP.B1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.B1 <- residuals(NP.B1$lme,type="normalized")
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.B1
resNP.B1 <-Efull
ACF_NP.B1 <- acf(resNP.B1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
summary(NP.B1$gam) #R-sq.(adj) =  0.261
AIC(NP.B1$lme) #216.4121
plot(NP.B1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(NP.B1$gam, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(NP.B1$gam, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(NP.B1$gam, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)

#Model C
NP.C <- gam(lnNP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(NP.C) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.C <- residuals(NP.C)
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.C
resNP.C<-Efull
ACF_NP.C<-acf(resNP.C, main = "ACF", na.action=na.pass, lag.max=10) #temporal
correlation

NP.C1 <- gamm(lnNP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(NP.C1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.C1 <- residuals(NP.C1$lme,type="normalized")
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
```

```
Efull[I1]<-resNP.C1
resNP.C1 <-Efull
ACF_NP.C1 <- acf(resNP.C1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
summary(NP.C1$gam) #R-sq.(adj) =  0.621
AIC(NP.C1$lme) #179.1742
plot(NP.C1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)


#Model D
NP.D <- gam(lnNP ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(NP.D) #normality and heterogeneity OK
#independence of observations --> ACF plot
resNP.D <- residuals(NP.D)
I1<-!is.na(models$lnNP)
Efull<-vector(length = length(models$lnNP))
Efull<-NA
Efull[I1]<-resNP.D
resNP.D<-Efull
ACF_NP.D<-acf(resNP.D, main = "ACF", na.action=na.pass, lag.max=10) #ok
#Output:
summary(NP.D) #R-sq.(adj) =  0.689
AIC(NP.D) #142.3208
plot(NP.D, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(NP.D, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(NP.D, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(NP.D, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)
```

GAM/Ms Chl-a

Table S4

```
#Model A
Chla.A <- gam(Chla.ln ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.A) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.A <- residuals(Chla.A)
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.A
```

```
resChla.A<-Efull
ACF_Chla.A<-acf(resChla.A, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation

Chla.A1 <- gamm(Chla.ln ~ s(DOE, fx = FALSE, k = 6, bs="cr"),
             correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.A1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.A1 <- residuals(Chla.A1$lme,type="normalized")
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.A1
resChla.A1<-Efull
ACF_Chla.A1 <- acf(resChla.A1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected

#Output:
summary(Chla.A1$gam) #R-sq.(adj) =  0.233
AIC(Chla.A1$lme) #155.106
plot(Chla.A1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)

#Model B
Chla.B <- gam(Chla.ln ~ s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.B) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.B <- residuals(Chla.B)
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.B
resChla.B<-Efull
ACF_Chla.B<-acf(resNP.B, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation

Chla.B1 <- gamm(Chla.ln ~ s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
             correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.B1$gam) #normality and heterogeneity OK
```

```r
#independence of observations --> ACF plot
resChla.B1 <- residuals(Chla.B1$lme,type="normalized")
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.B1
resChla.B1 <-Efull
ACF_Chla.B1 <- acf(resChla.B1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
#Output:
summary(Chla.B1$gam) #R-sq.(adj) =  0.248
AIC(Chla.B1$lme) #167.2798
plot(Chla.B1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(Chla.B1$gam, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(Chla.B1$gam, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(Chla.B1$gam, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)

#Model C
Chla.C <- gam(Chla.ln ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.C) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.C <- residuals(Chla.C)
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.C
resChla.C<-Efull
ACF_Chla.C<-acf(resChla.C, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation

Chla.C1 <- gamm(Chla.ln ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr"),
            correlation = corAR1(form = ~ 1 | DOE),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.C1$gam) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.C1 <- residuals(Chla.C1$lme,type="normalized")
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.C1
resChla.C1 <-Efull
ACF_Chla.C1 <- acf(resChla.C1, main = "ACF", na.action=na.pass, lag.max=10)
#temporal correlation corrected
```

```
#Output:
summary(Chla.C1$gam) #R-sq.(adj) =  0.271
AIC(Chla.C1$lme) #165.3261
plot(Chla.C1$gam, main ="Reference", shade=F, col="black", lwd=4, select = 1)

#Model D
Chla.D <- gam(Chla.ln ~ Treatment
            + s(DOE, fx = FALSE, k = 6, bs="cr")
            + s(DOE, fx = FALSE, k = 6, bs="cr", by = oTreatment),
            family = gaussian,
            data = models, method = "REML")
#check assumptions:
appraise(Chla.D) #normality and heterogeneity OK
#independence of observations --> ACF plot
resChla.D <- residuals(Chla.D)
I1<-!is.na(models$Chla.ln)
Efull<-vector(length = length(models$Chla.ln))
Efull<-NA
Efull[I1]<-resChla.D
resChla.D<-Efull
ACF_Chla.D<-acf(resChla.D, main = "ACF", na.action=na.pass, lag.max=10) #ok
#Output:
summary(Chla.D) #R-sq.(adj) =  0.289
AIC(Chla.D) #139.4042
plot(Chla.D, main ="Reference", shade=F, col="black", lwd=4, select = 1)
plot(Chla.D, main ="Cntl - HF", shade=F, col="grey50",lwd=4, select = 2)
plot(Chla.D, main ="Cntl - L", shade=F, col="yellow",lwd=4, select = 3)
plot(Chla.D, main ="Cntl - HFL", shade=F, col="red",lwd=4, select = 4)
```

ANOVAS for LRR presented in Fig.7

```
ESpairtest <- read.csv("PairwiseTtest.EffectSizes.csv", stringsAsFactors = T)

ESpairtest.long <- ESpairtest %>%
  pivot_longer(-Treatment, names_to = "variables", values_to = "value")
str(ESpairtest.long)
```

T test - Table S6: Effect sizes against benchmark values

```
#DOC_HF
dat_DOC_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="DOC")
dat_DOC_HF %>% t_test(value ~ 1, mu=-0.08) #p-value = 0.00014 (***) -->
significantly different from benchmark
#DOC_L
dat_DOC_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="DOC")
dat_DOC_L %>% t_test(value ~ 1, mu=0.02) #p-value = 7.05e-05 (***)-->
significantly different from 0
#DOC_HFL
dat_DOC_HFL <- ESpairtest.long %>%
```

```
  filter(Treatment == "HFL" & variables =="DOC")
dat_DOC_HFL %>% t_test(value ~ 1, mu = -0.08) #p-value = 2.789e-06 (***) -->
significantly different from 0
#SestonC_HF
dat_sestonC_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="SestonC")
dat_sestonC_HF %>% t_test(value ~ 1, mu=0.14) #p-value = 0.2583 (-)--> non-
significantly different from benchmark
#SestonC_L
dat_sestonC_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SestonC")
dat_sestonC_L %>% t_test(value ~ 1, mu=0.16) #p-value = 0.01714 (*)-->
significantly different from 0
#SestonC_HFL
dat_sestonC_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SestonC")
dat_sestonC_HFL %>% t_test(value ~ 1, mu = 0.11) #p-value = 0.002757 (**) -->
significantly different from 0
#Colour_HF
dat_colour_HF <- ESpairtest.long %>%
   filter(Treatment == "HF" & variables =="Colour")
dat_colour_HF %>% t_test(value ~ 1) #p-value = 3.447e-05 (***) -->
significantly different from 0
#Colour_L
dat_colour_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="Colour")
dat_colour_L %>% t_test(value ~ 1) #p-value = 7.883e-06 (***) -->
significantly different from 0
#Colour_HFL
dat_colour_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="Colour")
dat_colour_HFL %>% t_test(value ~ 1) #p-value = 3.189e-05 (***) -->
significantly different from 0
#DIN_HF
dat_DIN_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="DIN")
dat_DIN_HF %>% t_test(value ~ 1, mu=0.01) #p-value = 0.2566 (-) --> NON-
significantly different from benchmark
#DIN_L
dat_DIN_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="DIN")
dat_DIN_L %>% t_test(value ~ 1, mu=-0.28) #p-value = 0.002188 (**)-->
significantly different from 0
#DIN_HFL
dat_DIN_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="DIN")
dat_DIN_HFL %>% t_test(value ~ 1, mu = 0.24) #p-value = 0.0002492 (***) -->
significantly different from 0
#SestonN_HF
dat_sestonN_HF <- ESpairtest.long %>%
```

```
  filter(Treatment == "HF" & variables =="SestonN")
dat_sestonN_HF %>% t_test(value ~ 1, mu=0.19) #p-value = 0.0758 (-)--> non-
significantly different from benchmark
#SestonN_L
dat_sestonN_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SestonN")
dat_sestonN_L %>% t_test(value ~ 1, mu=0.01) #p-value = 0.05959 (-)--> NON-
significantly different from 0
#SestonN_HFL
dat_sestonN_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SestonN")
dat_sestonN_HFL %>% t_test(value ~ 1, mu = 0.06) #p-value = 0.03451 (*) -->
significantly different from 0
#SRP_HF
dat_SRP_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="SRP")
dat_SRP_HF %>% t_test(value ~ 1, mu=-0.65) #p-value = 0.004277 (**) -->
significantly different from 0
#SRP_L
dat_SRP_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SRP")
dat_SRP_L %>% t_test(value ~ 1, mu=-0.2) #p-value = 0.000257 (***) -->
significantly different from 0
#SRP_HFL
dat_SRP_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SRP")
dat_SRP_HFL %>% t_test(value ~ 1, mu=-0.63) #p-value = 5.211e-07 (***) -->
significantly different from 0
#SestonP_HF
dat_SestonP_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="SestonP")
dat_SestonP_HF %>% t_test(value ~ 1, mu=0.28) #p-value = 0.669 (-) --> non-
significantly different from 0
#SestonP_L
dat_SestonP_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SestonP")
dat_SestonP_L %>% t_test(value ~ 1, mu=0.24) #p-value = 0.000252 (***) -->
significantly different from 0
#SestonP_HFL
dat_SestonP_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SestonP")
dat_SestonP_HFL %>% t_test(value ~ 1, mu=0.18) #p-value = 0.0002858 (***) -->
significantly different from 0
#C3_HF
dat_C3_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="C3")
dat_C3_HF %>% t_test(value ~ 1) #p-value = 5.074e-08 (***) --> significantly
different from benchmark
#C3_L
dat_C3_L <- ESpairtest.long %>%
```

```r
  filter(Treatment == "L" & variables =="C3")
dat_C3_L %>% t_test(value ~ 1) #p-value = 1.392e-05 (***)--> significantly
different from 0
#C3_HFL
dat_C3_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="C3")
dat_C3_HFL %>% t_test(value ~ 1) #p-value = 1.543e-07 (***) --> significantly
different from 0
#C1_HF
dat_C1_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="C1")
dat_C1_HF %>% t_test(value ~ 1) #p-value = 7.838e-07 (***)--> significantly
different from benchmark
#C1_L
dat_C1_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="C1")
dat_C1_L %>% t_test(value ~ 1) #p-value = 0.04605 (*)--> significantly
different from 0
#C1_HFL
dat_C1_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="C1")
dat_C1_HFL %>% t_test(value ~ 1) #p-value = 1.906e-06 (***) --> significantly
different from 0
#C4_HF
dat_C4_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="C4")
dat_C4_HF %>% t_test(value ~ 1) #p-value = 4.129e-05 (***) --> significantly
different from 0
#C4_L
dat_C4_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="C4")
dat_C4_L %>% t_test(value ~ 1) #p-value = 6.809e-06 (***) --> significantly
different from 0
#C4_HFL
dat_C4_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="C4")
dat_C4_HFL %>% t_test(value ~ 1) #p-value = 8.575e-07 (***) --> significantly
different from 0
#C2_HF
dat_C2_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="C2")
dat_C2_HF %>% t_test(value ~ 1) #p-value = 0.0005862 (**) --> significantly
different from 0
#C2_L
dat_C2_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="C2")
dat_C2_L %>% t_test(value ~ 1) #p-value = 0.3163 (-) --> non-significantly
different from 0
#C2_HFL
dat_C2_HFL <- ESpairtest.long %>%
```

```
  filter(Treatment == "HFL" & variables =="C2")
dat_C2_HFL %>% t_test(value ~ 1) #p-value = 0.03566 (*) --> significantly
different from

#SestonCP_HF
dat_sestonCP_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="SestonCP")
dat_sestonCP_HF %>% t_test(value ~ 1, mu = -0.12) #p-value =  0.01164 (*) -->
significantly different from benchmark
#SestonCP_L
dat_sestonCP_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SestonCP")
dat_sestonCP_L %>% t_test(value ~ 1, mu=-0.06) #p-value = 0.0001099 (***)-->
significantly different from 0
#SestonCP_HFL
dat_sestonCP_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SestonCP")
dat_sestonCP_HFL %>% t_test(value ~ 1, mu=-0.01) #p-value = 4.346e-05 (***) -
-> significantly different from 0
#SestonNP_HF
dat_sestonNP_HF <- ESpairtest.long %>%
  filter(Treatment == "HF" & variables =="SestonNP")
dat_sestonNP_HF %>% t_test(value ~ 1, mu=-0.06) #p-value = 0.00933 (**)-->
significantly different from benchmark
#SestonNP_L
dat_sestonNP_L <- ESpairtest.long %>%
  filter(Treatment == "L" & variables =="SestonNP")
dat_sestonNP_L %>% t_test(value ~ 1, mu=-0.2) #p-value = 0.01545 (*)-->
significantly different from 0
#SestonNP_HFL
dat_sestonNP_HFL <- ESpairtest.long %>%
  filter(Treatment == "HFL" & variables =="SestonNP")
dat_sestonNP_HFL %>% t_test(value ~ 1, mu=-0.04) #p-value = 0.001194 (**) -->
significantly different from 0
```

Pairwise t test - Table S7: Effect sizes between treatments

```
stat.test <- ESpairtest.long %>%
  group_by(variables) %>%
  t_test(value ~ Treatment) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
stat.test
```