Combining Species Presence/Absence with Environmental Data

Eleanor (Ella) Crotty

Contents

Setup

```
# Warnings and startup messages suppressed
library(tidyverse)
library(patchwork)
library(scales)
library(ggrepel)
library(readxl)
library(here)
```

```
# Import data
SpeciesDetections <- read_csv(here("OCNMS_Project", "Outputs",
↪  "SOI_IDs_Species10Detections.csv")) %>%
  filter(year(Date_UTC) != 2023)
```

```
## Rows: 532 Columns: 30
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (19): Family, Genus, Species, JV_Sample_Name, Barcode.x, Barcode_mod, S...
## dbl   (8): Biological_Replicate, Technical_Replicate, Depth_m, Lat_dec, Lon_...
## lgl   (1): Present
## dttm  (2): Date_UTC, Date_local
##
## 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.
```

```
EnvData1 <- read_csv(here("OCNMS_Project", "Data", "EnvironmentalDataset1.csv")) %>%  #
↪  Using EnvironmentalDataset1 because the satellite + NEMO data in
↪  EnvironmentalDataset2 didn't turn out to be good at gap filling
  filter(year != 2023) # Ignoring 2023 due to gaps for now
```

```
## New names:
## Rows: 38938 Columns: 11
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): source dbl (8): ...1, year, temperature, DO, salinity, potential_density,
## pres, cond dttm (2): date, sampleID
## 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.
## * `` -> `...1`
```

```
SamplingDates2 <- read_csv(here("OCNMS_Project", "Data", "SamplingDates2.csv")) %>%  #
↪  Exported from EnvironmentalDataxSampleDates.Rmd, simple dataframe of all datetimes of
↪  samples
  filter(year(Date) != 2023) %>% # Ignoring 2023 due to gaps for now
  mutate(Source = case_when(Source == "Bottle_DNA" ~ "Bottle_DNA_Sampled", Source ==
   ↪  "PPS_DNA" ~ "Automated_DNA_Sampler"))
```

```
## New names:
## Rows: 109 Columns: 3
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): Source dbl (1): ...1 dttm (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.
## * `` -> `...1`
```

Date check

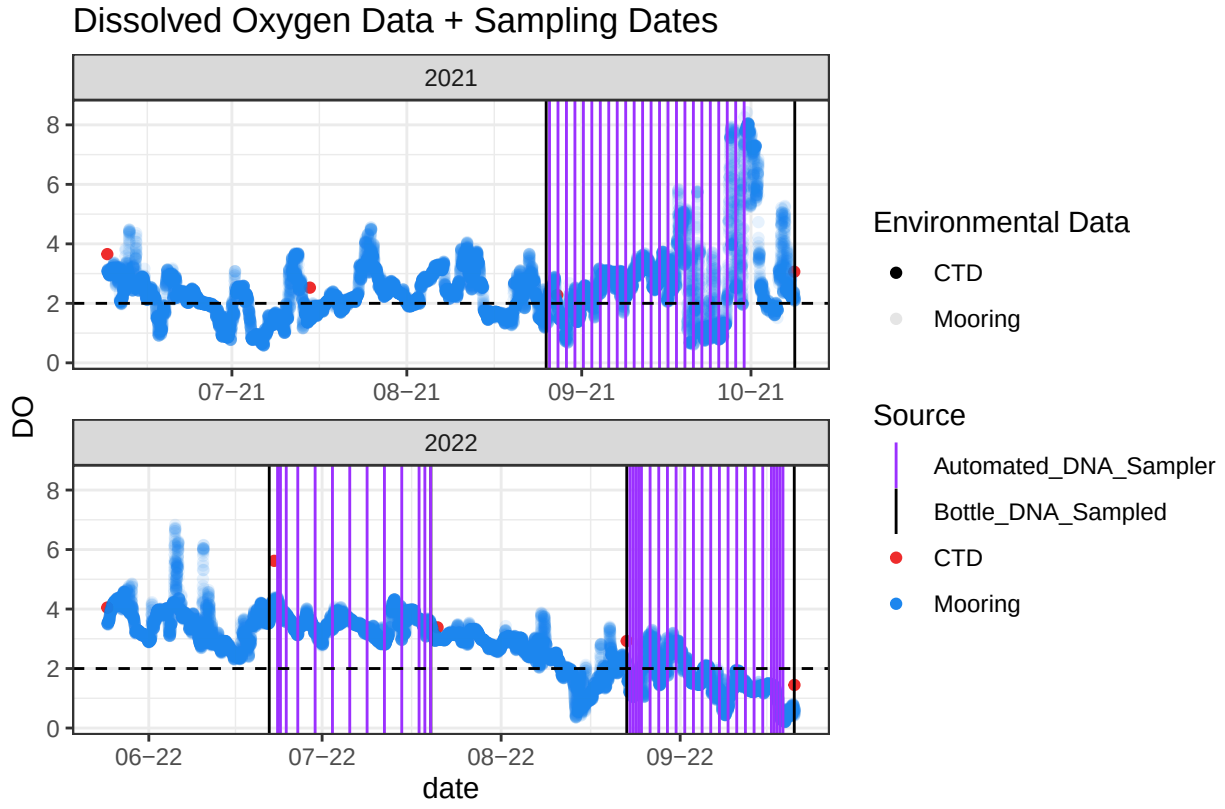
```
# Let's take a quick look at whether the dates match up
sampdates <- data.frame(Date = SamplingDates2$Date, Source = "Metadata")
detectdates <- data.frame(Date = SpeciesDetections$Date_UTC, Source = "Detections")
datescomp <- rbind(sampdates,detectdates)

ggplot() +
  geom_vline(data = datescomp, mapping = aes(xintercept = Date, color = Source, linetype
   ↪  = Source)) +
  scale_linetype_manual(values = c(2,1)) +
  theme_bw() +
  facet_wrap(facets = vars(year(Date)), scales = "free_x", ncol = 1) #+
  #scale_x_datetime(date_breaks = "4 days", date_labels = "%y-%b-%d") # This is fucked
   ↪  but tbh I don't need it
```

Sampling x environmental data graph

```
ggplot(EnvData1, aes(x = date, y = DO, color = source, alpha = source)) +
  geom_point() +
  scale_alpha_manual(values = c(1,0.1)) + # doesn't currently do anything
  scale_x_datetime(date_breaks = "1 month", date_labels = "%m-%y") +
```

```
  theme_bw() +
  facet_wrap(facets = vars(year), scales = "free_x", ncol = 1) +
  geom_vline(data = SamplingDates2, aes(xintercept = Date, color = Source)) +
  scale_color_manual(values = c("purple1", "black", "firebrick2", "dodgerblue2")) +
  geom_hline(aes(yintercept = 2), color = "black", linetype = "dashed") +
  labs(title = "Dissolved Oxygen Data + Sampling Dates", caption = "Dotted line = hypoxic
  ↪  threshold of 2 mg/L dissolved oxygen, vertical lines = DNA sampling times", alpha =
  ↪  "Environmental Data", color = "Source")
```

```
## Warning: Removed 236 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



oxic threshold of 2 mg/L dissolved oxygen, vertical lines = DNA sampling times

Data Join

```
messyjoin <- full_join(EnvData1, SpeciesDetections, by = join_by(date == Date_UTC)) # did
↪  not work, unsurprised
EnvRd <- EnvData1 %>%
  mutate(DateMatch = round_date(date, unit = "10 minutes")) # well at least it didn't
  ↪  immediately explode. However, this still has many datapoints per hour. Maybe round
  ↪  to the nearest 10 minutes?
DetectRd <- SpeciesDetections %>%
  mutate(DateMatch = round_date(Date_UTC, unit = "10 minutes"), Date_local_hr =
  ↪  round_date(Date_local, unit = "hour")) # Spot check - looks good.

messyrdjoin <- full_join(EnvRd, DetectRd, by = join_by(DateMatch)) # Many to many is
↪  probably expected - There's several detections per time point
```

```
# Ok, that was pretty good! To try and avoid many-to-many, perhaps a specific join. Don't
↪  care about EnvHr without DetectHr matches, so make DetectHr x and do a left join (A
↪  left_join() keeps all observations in x.)

eDNAxEnvData <- left_join(DetectRd, EnvRd, by = join_by(DateMatch))

investigate <- eDNAxEnvData %>% select(Species, DateMatch, Date_UTC, Date_local_hr,
↪  source, temperature, DO, SampleId, Rosette_position, Amplicon)
```

Joined data graphs

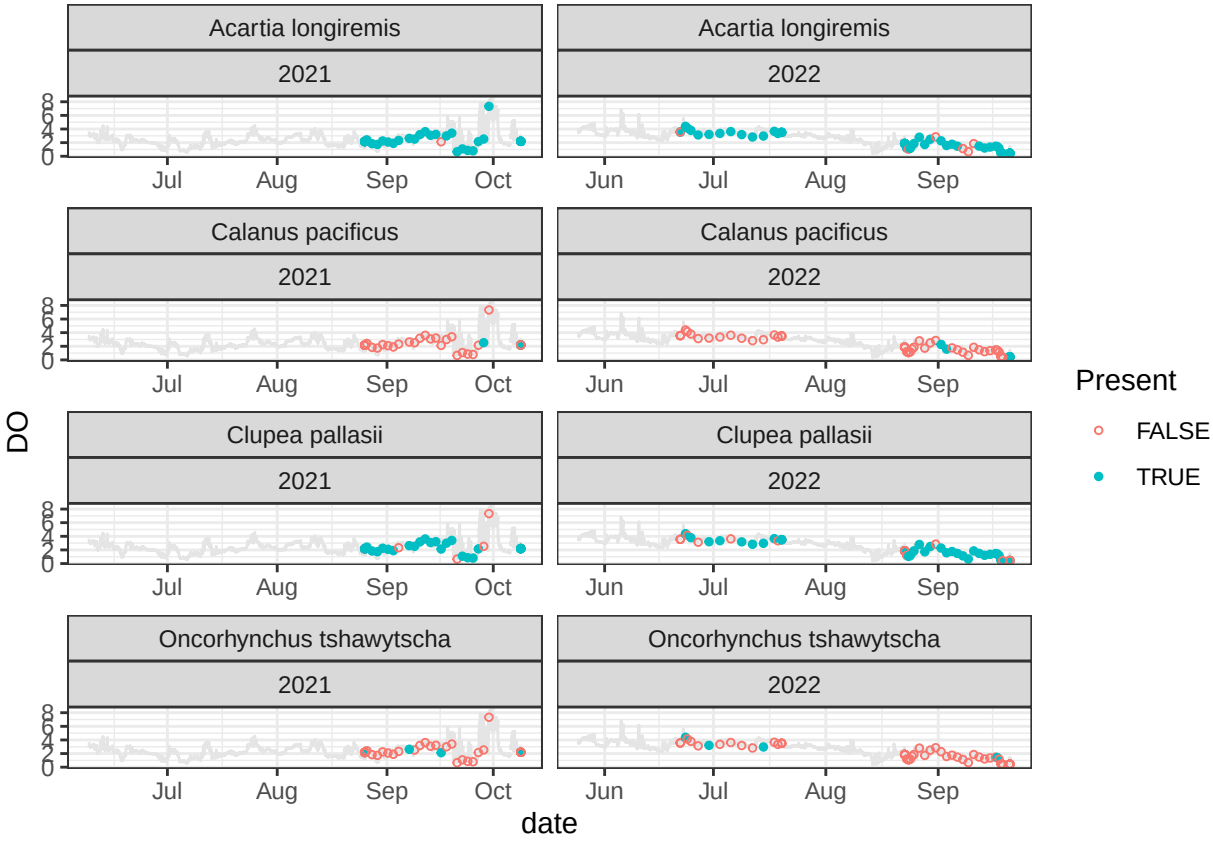
```
ggplot(eDNAxEnvData, aes(x = DateMatch, y = DO, shape = Present, size = Present, color =
↪  Present)) +
  scale_shape_manual(values = c(1, 19)) +
  scale_size_manual(values = c(1,1)) +
  geom_point() +
  theme_bw() +
  facet_wrap(facets = vars(Species))
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```


```
ggplot(data = EnvData1, aes(x = date, y = DO)) +
  geom_line(color = "gray90") +
```

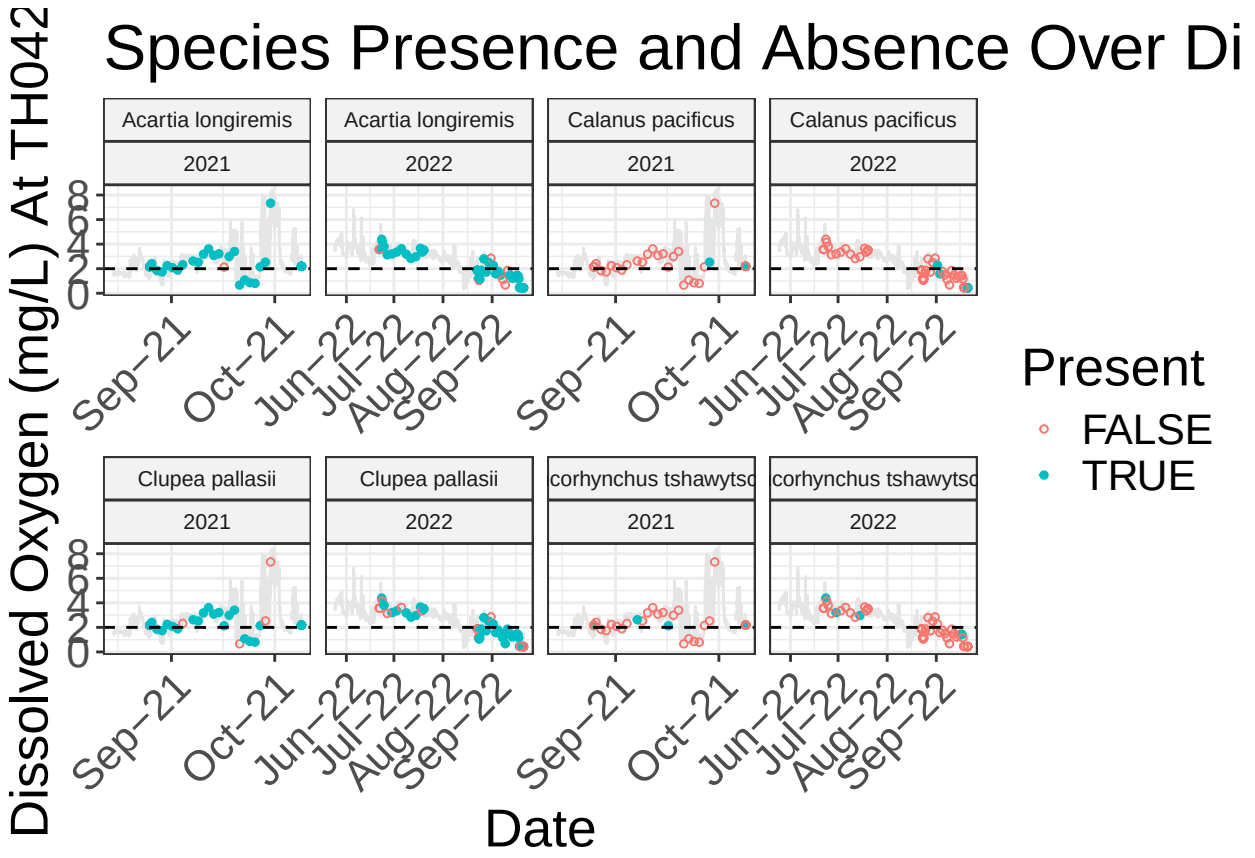
```
  geom_point(data = eDNAxEnvData, aes(x = DateMatch, y = DO, shape = Present, size =
  ↪  Present, color = Present)) +
  scale_shape_manual(values = c(1, 19)) +
  scale_size_manual(values = c(1,1)) +
  theme_bw() +
  facet_wrap(facets = vars(Species, year(date)), scales = "free_x", ncol = 2)
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
EnvDataRange <- EnvData1 %>%
  filter(date > as.POSIXct("2021-08-15 00:00:00", tz = "UTC"))

ggplot(data = EnvDataRange, aes(x = date, y = DO)) +
  geom_line(color = "gray90") +
  geom_point(data = eDNAxEnvData, aes(x = DateMatch, y = DO, shape = Present, size =
  ↪  Present, color = Present)) +
  scale_shape_manual(values = c(1, 19)) +
  scale_size_manual(values = c(1,1)) +
  theme_bw() +
  facet_wrap(facets = vars(Species, year(date)), scales = "free_x", ncol = 4) +
  scale_x_datetime(breaks = "month", date_labels = "%b-%y") +
  theme(text = element_text(size = 20), axis.text.x = element_text(angle = 45, hjust =
  ↪  1), strip.text = element_text(size = 8), strip.background = element_rect(fill =
  ↪  "gray95")) +
  geom_hline(yintercept = 2, linetype = 2) +
  labs(title = "Species Presence and Absence Over Dissolved Oxygen", x = "Date", y =
  ↪  "Dissolved Oxygen (mg/L) At TH042")
```

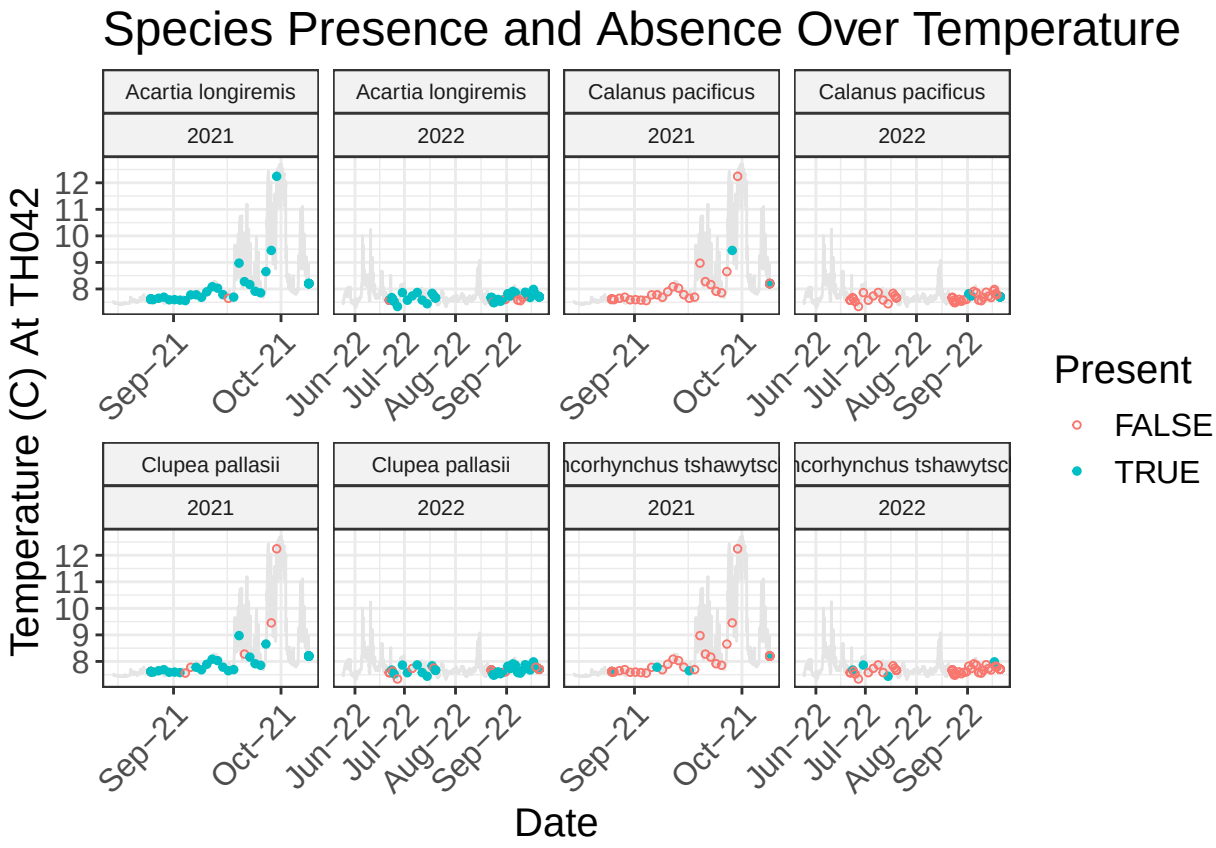
```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
ggsave(filename = here("OCNMS_Project", "Plots",
↪  "SpeciesPresence_Oxygen_Preliminary.png"), width = 2500, height = 2000, units = "px")
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
ggplot(data = EnvDataRange, aes(x = date, y = temperature)) +
  geom_line(color = "gray90") +
  geom_point(data = eDNAxEnvData, aes(x = DateMatch, y = temperature, shape = Present,
  ↪  size = Present, color = Present)) +
  scale_shape_manual(values = c(1, 19)) +
  scale_size_manual(values = c(1,1)) +
  theme_bw() +
  facet_wrap(facets = vars(Species, year(date)), scales = "free_x", ncol = 4) +
  scale_x_datetime(breaks = "month", date_labels = "%b-%y") +
  theme(text = element_text(size = 15), axis.text.x = element_text(angle = 45, hjust =
  ↪  1), strip.text = element_text(size = 8), strip.background = element_rect(fill =
  ↪  "gray95")) +
  labs(title = "Species Presence and Absence Over Temperature", x = "Date", y =
  ↪  "Temperature (C) At TH042")
```

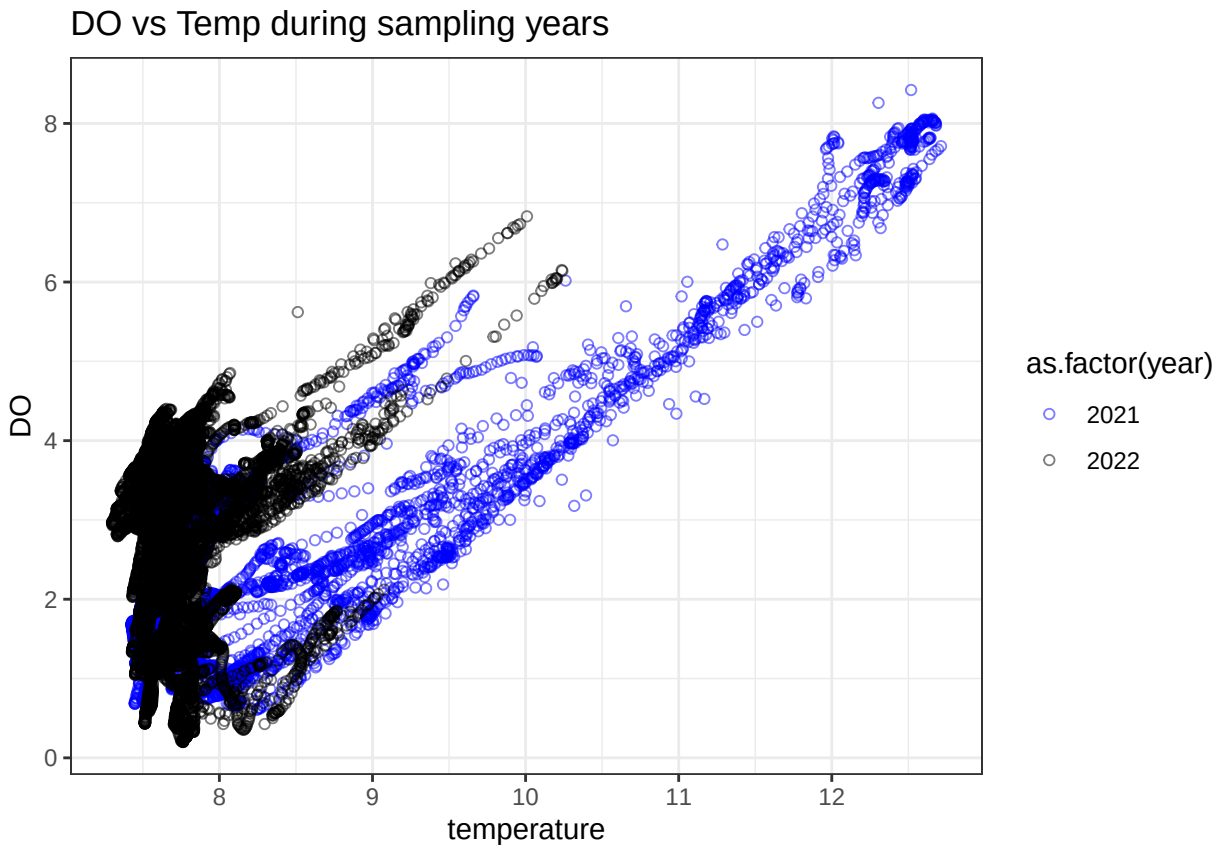



```
ggsave(filename = here("OCNMS_Project", "Plots", "SpeciesPresence_Temp_Preliminary.png"),
↪  width = 2500, height = 2000, units = "px")
```

DO x Temp graphs

```
ggplot(EnvDataRange, aes(x = temperature, y = DO, shape = source, color =
↪  as.factor(year))) +
  scale_shape_manual(values = c(4,19)) +
  scale_color_manual(values = c("blue", "black")) +
  geom_point(shape = 1, alpha = 0.5) +
  theme_bw() +
  labs(title = "DO vs Temp during sampling years")
```

```
## Warning: Removed 231 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



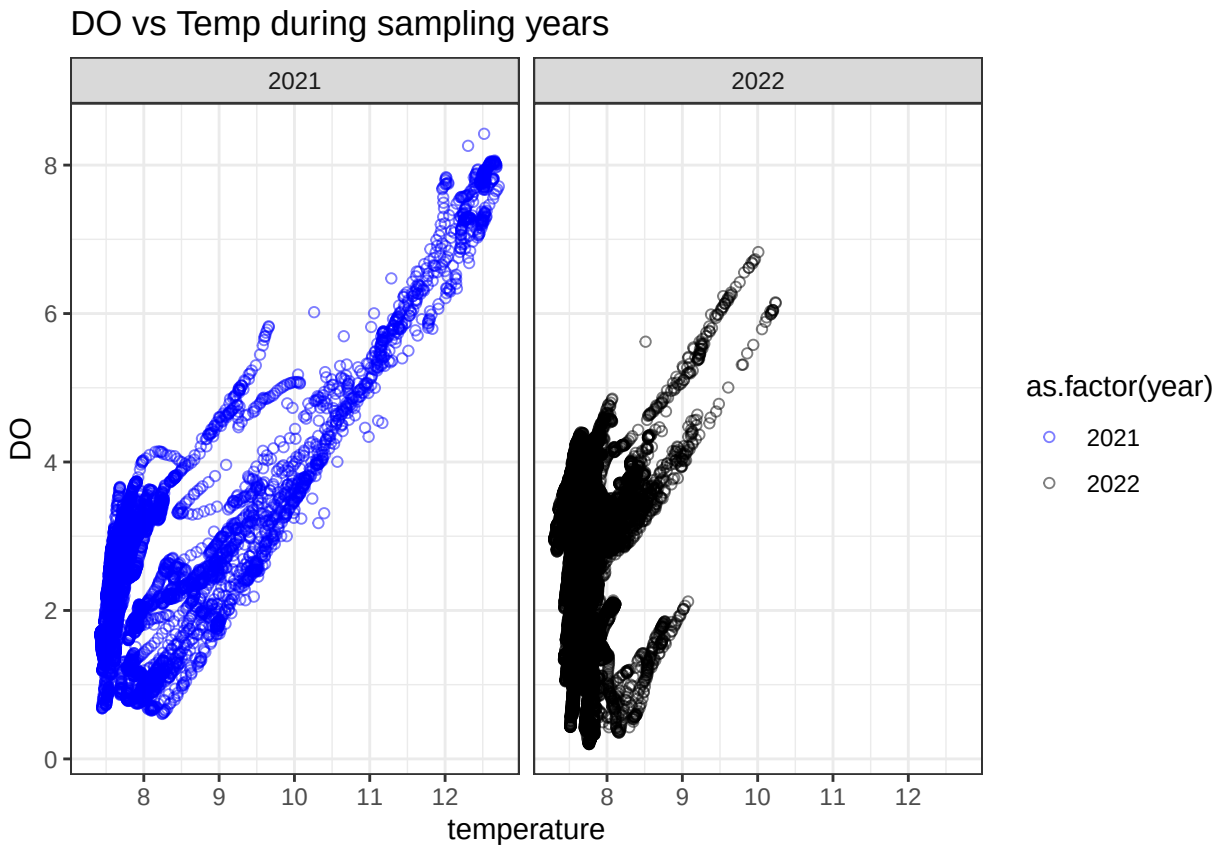

```
ggsave(here("OCNMS_eDNA", "Plots", "DO_vs_Temp.png"))
```

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

```
## Warning: Removed 231 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
ggplot(EnvDataRange, aes(x = temperature, y = DO, shape = source, color =
↪  as.factor(year))) +
  scale_shape_manual(values = c(4,19)) +
  scale_color_manual(values = c("blue", "black")) +
  geom_point(shape = 1, alpha = 0.5) +
  theme_bw() +
  facet_wrap(facets = vars(year)) +
  labs(title = "DO vs Temp during sampling years")
```

```
## Warning: Removed 231 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



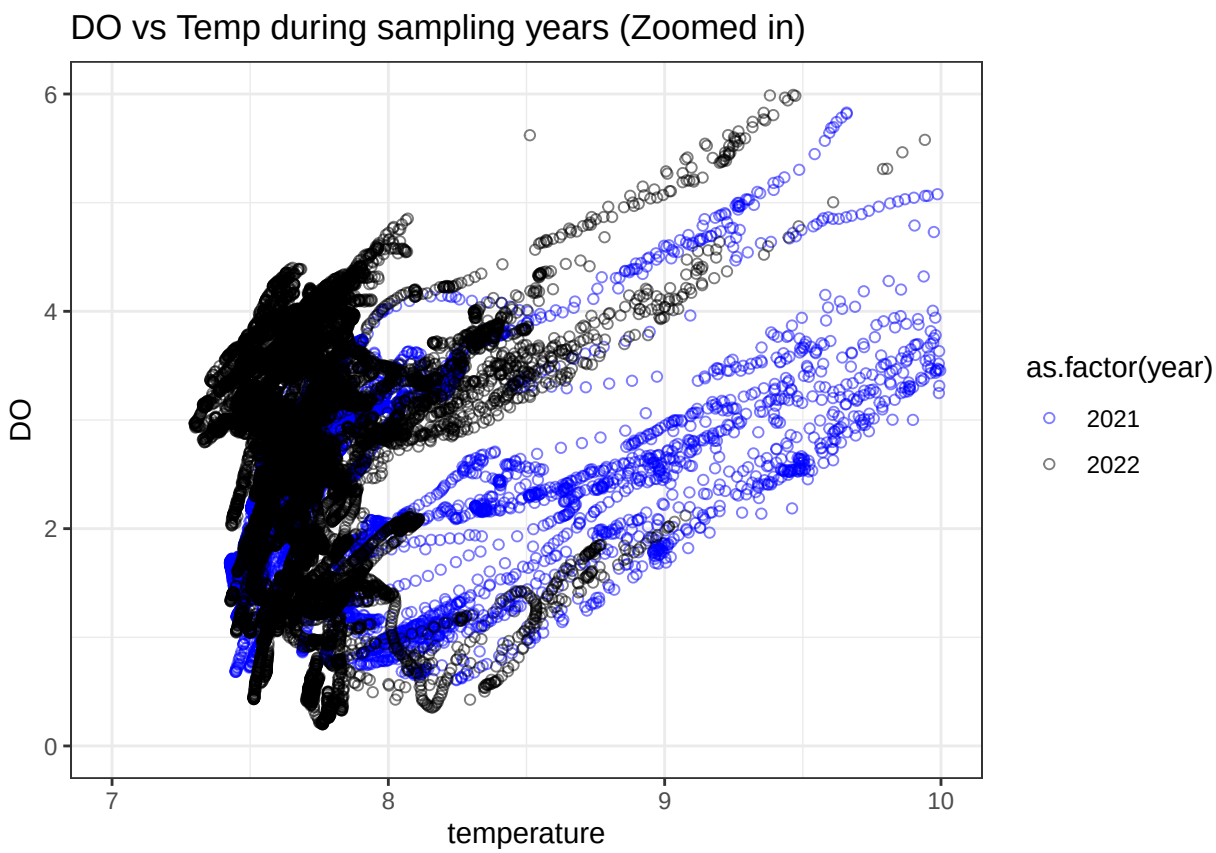

```
ggsave(here("OCNMS_eDNA", "Plots", "DO_vs_TempYear.png"))
```

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

```
## Warning: Removed 231 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
ggplot(EnvDataRange, aes(x = temperature, y = DO, shape = source, color =
↪  as.factor(year))) +
  scale_shape_manual(values = c(4,19)) +
  scale_color_manual(values = c("blue", "black")) +
  geom_point(shape = 1, alpha = 0.5) +
  scale_x_continuous(limits = c(7, 10)) +
  scale_y_continuous(limits = c(0,6)) +
  theme_bw() +
  labs(title = "DO vs Temp during sampling years (Zoomed in)")
```

```
## Warning: Removed 1044 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



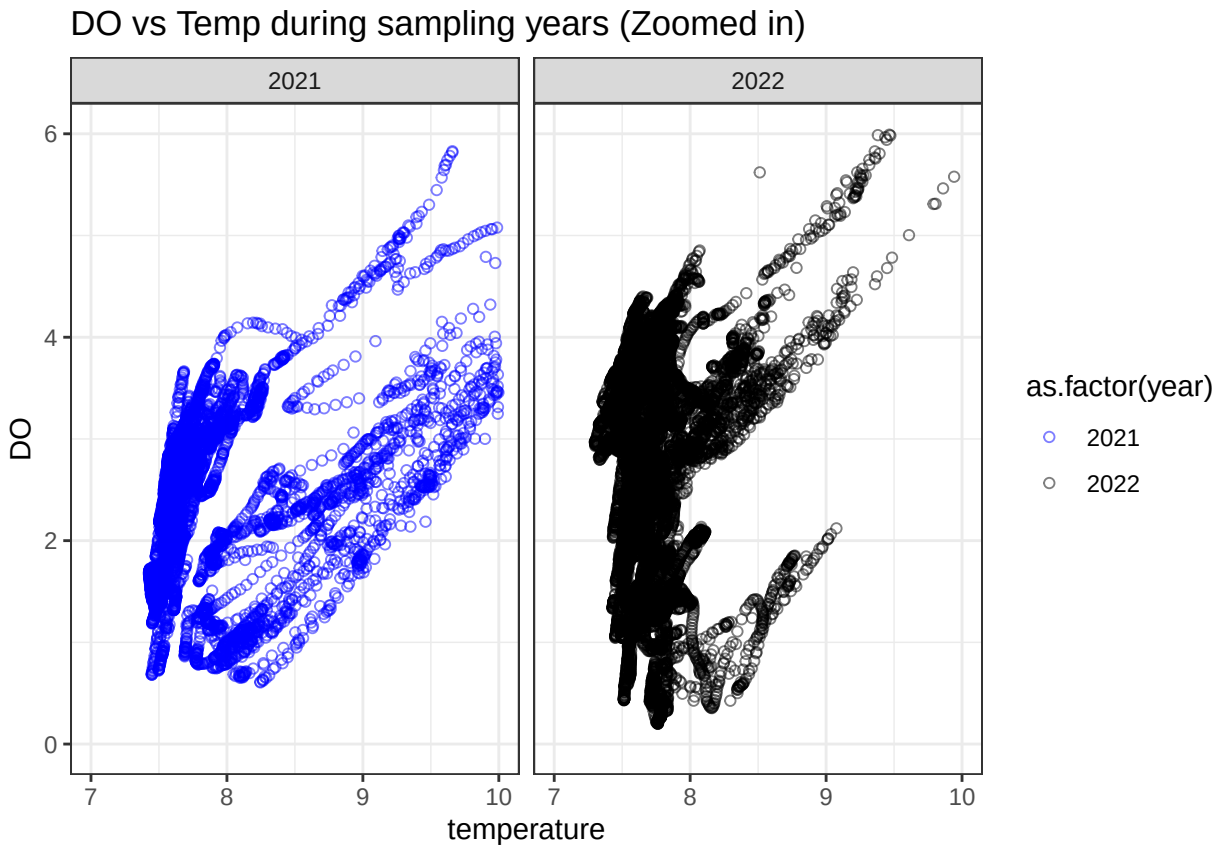

```
ggsave(here("OCNMS_eDNA", "Plots", "DO_vs_Temp_Zoomed.png"))
```

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

```
## Warning: Removed 1044 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
ggplot(EnvDataRange, aes(x = temperature, y = DO, shape = source, color =
↪  as.factor(year))) +
  scale_shape_manual(values = c(4,19)) +
  scale_color_manual(values = c("blue", "black")) +
  geom_point(shape = 1, alpha = 0.5) +
  scale_x_continuous(limits = c(7, 10)) +
  scale_y_continuous(limits = c(0,6)) +
  theme_bw() +
  facet_wrap(facets = vars(year)) +
  labs(title = "DO vs Temp during sampling years (Zoomed in)")
```

```
## Warning: Removed 1044 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
ggsave(here("OCNMS_eDNA", "Plots", "DO_vs_Temp_ZoomedYear.png"))
```

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

```
## Warning: Removed 1044 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

Binomial Regression

```
oxmodel <- glm(Present ~ DO, data = eDNAxEnvData, family = "binomial")
summary(oxmodel)
```

```
##
## Call:
## glm(formula = Present ~ DO, family = "binomial", data = eDNAxEnvData)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.07553    0.21773  -0.347    0.729
## DO          -0.04719    0.09027  -0.523    0.601
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 562.43  on 407  degrees of freedom
## Residual deviance: 562.15  on 406  degrees of freedom
##   (4 observations deleted due to missingness)
```

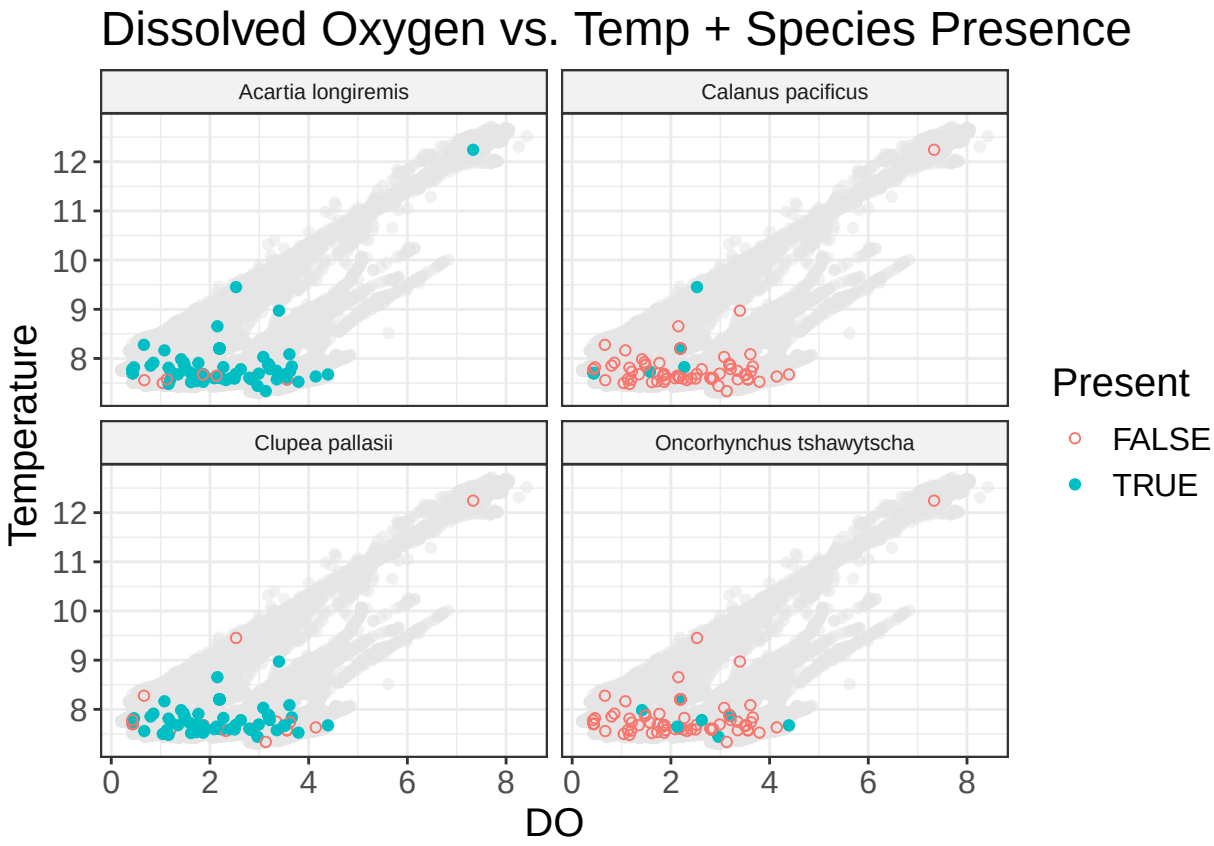
```
## AIC: 566.15
##
## Number of Fisher Scoring iterations: 3
```

Check for outliers

```
ggplot(EnvDataRange, aes(x = DO, y = temperature)) +
  geom_point(color = "gray90", alpha = 0.5) +
  geom_point(eDNAxEnvData, mapping = aes(x = DO, y = temperature, color = Present, shape
  ↪  = Present), inherit.aes = F) +
    scale_shape_manual(values = c(1, 19)) +
  theme_bw() +
  theme(text = element_text(size = 15), strip.text = element_text(size = 8),
  ↪  strip.background = element_rect(fill = "gray95")) +
  labs(title = "Dissolved Oxygen vs. Temp + Species Presence", y = "Temperature") +
  facet_wrap(facets = vars(Species))
```

```
## Warning: Removed 924 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
ggsave(filename = here("OCNMS_Project", "Plots", "SpeciesPresence_TempxDO.png"), width =
↪  2500, height = 2000, units = "px")
```

```
## Warning: Removed 924 rows containing missing values or values outside the scale range
```

```
## (`geom_point()`).
## Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

Check for linearity

"Assumption #5 involves the necessity of a linear relationship between the continuous independent variables and the logit transformation of the dependent variable. This linearity assumption implies that **for continuous independent variables** like income level, hours of exercise per week, and blood sugar levels, **there should be a linear relationship with the logit of the dependent variable**, such as the probability of developing diabetes. Various methods can be employed to assess this linearity, with one common approach being the **Box-Tidwell procedure.** This technique involves creating interaction terms between each continuous independent variable and its natural logarithm and adding these to the logistic regression model. This technique can be implemented using software like **SPSS Statistics, which offers the Binary Logistic procedure to test for this assumption.** The results of this test are then interpreted to decide the next steps in the analysis, depending on whether the linearity assumption holds or is violated. If the assumption is met, the analysis can proceed as planned. However, if the assumption is not met, adjustments to the model or alternative methods may be necessary to address the non-linearity appropriately." - Binomial Logistic Regression