

Code for Fjord oceanographic dynamics provide refuge for critically endangered *Pycnopodia helianthoides*

2025-02-28

Fjord oceanographic dynamics provide refuge for critically endangered *Pycnopodia helianthoides*

Alyssa-Lois M. Gehman¹, Ondine Pontier¹, Derek VanMaanen¹, Tristan Blaine², Tyrel Froese¹, Gillian Sadler-Brown¹, Angeleen M. Olson¹, Zachary L. Monteith¹, Krystal Bachen¹, Carolyn Prentice¹, Margot Hessing-Lewis¹, Jennifer M. Jackson¹

¹Hakai Institute, End of Kwakshua Channel, Canada. ²Central Coast Indigenous Resource Alliance, Canada

Keywords: sea star wasting disease, infection, marine, refuge habitat

Abstract: Disease outbreaks as a driver of wildlife mass mortality events have increased in magnitude and frequency since the 1940s. Remnant populations, composed of individuals that survived mass mortality events, could provide insight into disease dynamics, and species recovery. The sea star wasting disease (SSWD) epidemic led to the rapid >90% decline of the sunflower star *Pycnopodia helianthoides*. We surveyed the biomass density of *P. helianthoides* on the central British Columbia coast before, during and after the arrival of SSWD by conducting expert diver surveys in shallow subtidal habitats from 2013-2023. We found a rapid decline in biomass density following the onset of SSWD in 2015. Despite consistent recruitment post-outbreak to sites associated with outer islands, we found repeated loss of large adult individuals over multiple years. Within nearby fjord habitats, we found remnant populations composed of large adult *P. helianthoides*. The interaction of temperature and salinity with *P. helianthoides* biomass density varied by location, with high biomass density associated with higher temperatures in the outer islands, and associated with lower temperatures and higher salinity in the fjords. These patterns suggest that fjords provide refuge from consequences of SSWD and protecting these populations could be imperative for the species.

load and clean data

```
knitr::opts_chunk$set(echo = TRUE)
library(tidyrr)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

library(readr)
library(ggplot2)
library(purrr)
library(grid) #to plot multiple graphs in output
library(PNWColors)
library(stringr)
```

```

library(purrr)
library(lubridate)

##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
library(RColorBrewer)
library(mgcv)

## Loading required package: nlme
##
## Attaching package: 'nlme'
## The following object is masked from 'package:dplyr':
##
##     collapse
## This is mgcv 1.9-1. For overview type 'help("mgcv-package")'.
library(tidymv)

## tidymv has been superseded by tidygam. The package tidymv is no longer maintained but will be
## kept on CRAN to ensure reproducibility of older analyses. Users should
## use the replacement package tidygam for new analyses, which is available on
## CRAN and GitHub (https://github.com/stefanocoretta/tidygam).
library(cowplot)

##
## Attaching package: 'cowplot'
## The following object is masked from 'package:lubridate':
##
##     stamp
library(stringr)
library(ggdist) #for fancy plots with distributions
library(visreg)
library(quantreg)

## Loading required package: SparseM
library(gratia)

##
## Attaching package: 'gratia'
## The following object is masked from 'package:stringr':
##
##     boundary
library(sjPlot)

## Install package "strengjacke" from GitHub (`devtools::install_github("strengjacke/strengjacke")`)
##
## Attaching package: 'sjPlot'

```

```
## The following objects are masked from 'package:cowplot':
##
##   plot_grid, save_plot
library(vegan)

## Loading required package: permute
## Loading required package: lattice
setwd("~/Documents/GitHub/nearshore-ProcB_2024")

## full dataset, all sea star species and with each line an individual size
sswdag <- read_csv("data/20250113_sw_aggregate_2023end.csv",
  col_types = cols(radius = col_number(), transect_id = col_character(), unit_size = col_character(),
    transect_latitude=col_number(), transect_longitude=col_number(),
    transect_substrate=col_character(), unit_area_2D = col_number(),
    unit_area_3D = col_number(), site_depth = col_number()))

## New names:
## * `` -> `...1`
## * `...1` -> `...2`

## biomass density data at repeated measures sites
ssbio<-read_csv("data/ssbio_sswd_data_15May24.csv")

## Rows: 37998 Columns: 30
## -- Column specification -----
## Delimiter: ","
## chr  (14): unit_id_depth, latin_name, project, survey, site_id, site_habitat...
## dbl  (14): site_latitude, site_longitude, site_depth, transect_latitude, tra...
## lgl   (1): unit_area_3D
## date  (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

##biomass data aligned with oceanography data
pyc_aligned<-read_csv("data/pyc_biology_ocean_27Feb25.csv")

## Rows: 673 Columns: 43
## -- Column specification -----
## Delimiter: ","
## chr  (15): unit_id_depth, latin_name, project, survey, site_id, site_habitat...
## dbl  (25): ...1, site_latitude, site_longitude, site_depth, transect_latitud...
## lgl   (1): unit_area_3D
## dtm   (1): measurement.time
## date  (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

Oceanography data - Fig 1, S1 and S2 Oceanography data can be visualized at https://data.hakai.org/oceanography/ and downloaded from https://catalogue.hakai.org/erddap/index.html

Fig. 2B and S4
```

```
#####
##### Fjord size distributions #####
#####

### data split by location catagory

## correct fjord and outer delineation
size_loc.cat<- sswdag %>% dplyr::filter(latin_name=="Pycnopodia helianthoides") %>%
  mutate(loc.cat=ifelse(site_id=="Kelpie Pt" | site_id=="Mercury" | site_id=="Odlum Pt" | s

struct3<-size_loc.cat %>% group_by(year, loc.cat) %>% count(site_id) %>% count(loc.cat)

size_loc.cat1<- size_loc.cat %>% left_join(struct3)

## Joining with `by = join_by(year, loc.cat)`

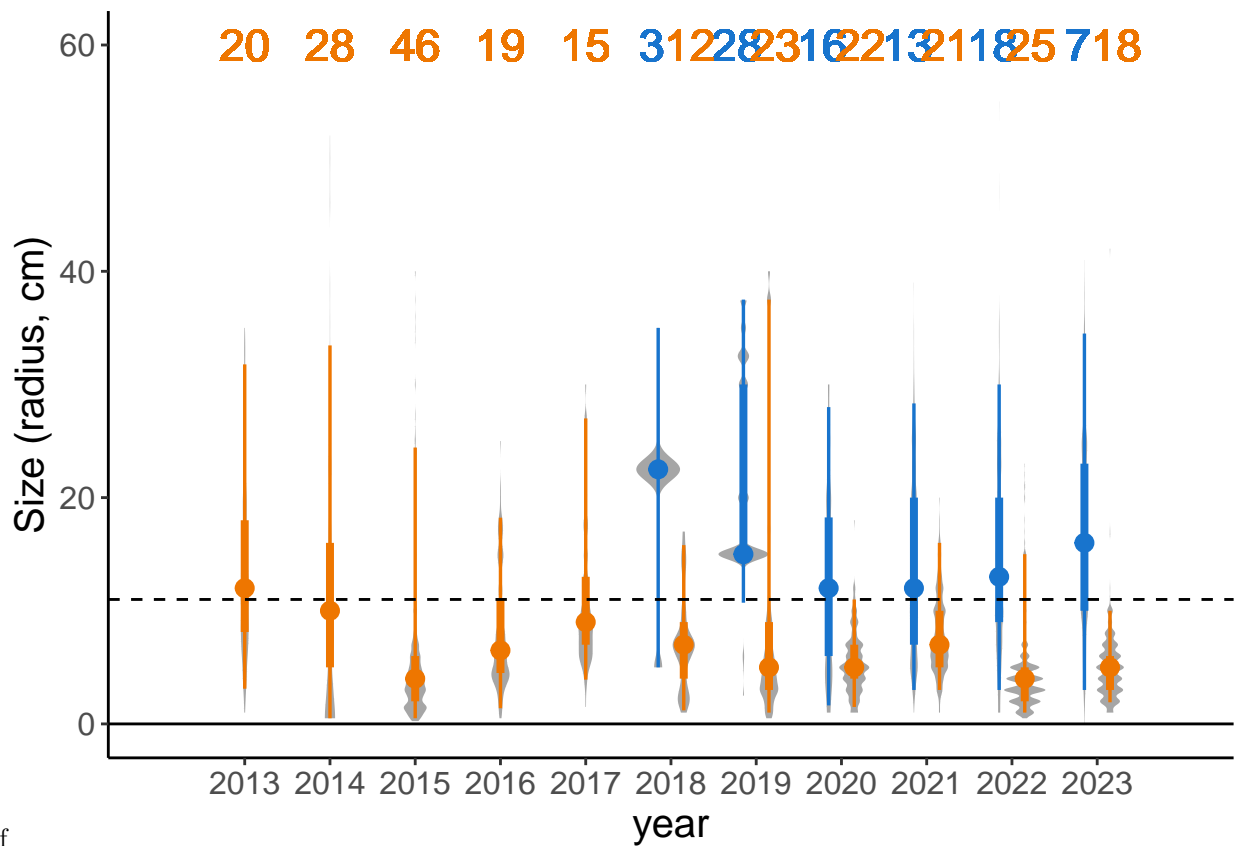
### combined all sites in one figure

all<- size_loc.cat1 %>% tidyr::drop_na(radius, year) %>%
  ggplot(aes(year, radius, color=loc.cat), fill=loc.cat)+
  ggdist::stat_eye(
    adjust = .5,
    width = 1.3,
    ## set slab interval to show IQR and 95% data range https://cran.r-project.org/web/packages/ggdist/
    .width = c(.5, 0.95), position=position_dodge(width=0.6)) +
  #geom_text(aes(x=year, y=60, label=n), position = position_dodge(width=0.9), size=6)+
  ylab("Size (radius, cm)") +
  scale_x_continuous(limits= c(2012,2024), breaks=seq(2013,2023,1)) +
  scale_y_continuous(limits = c(0,60)) +
  scale_color_manual(values=c("dodgerblue3","darkorange2")) +
  geom_text(aes(x=year, y=60, label=n), position = position_dodge(width=0.9), size=6)+
  geom_hline(yintercept=11, linetype=c("dashed"))+
  geom_hline(yintercept=0)+
  theme_bw() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=15),
        legend.position = "none",
        axis.line = element_line(colour = "black"))
#pdf("Fig2_28Feb25.pdf", height=5, width=10)
#png("Pycno_size_loc.cat_rainplot_2Nov23.pdf", height=480, width=960)
all
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`stat_slabinterval()`).
```

```
## Warning in bandwidth_dpi(): Bandwidth calculation failed.
## > Falling back to `bandwidth_nrd0()`.
## i This often occurs when a sample contains many duplicates, which suggests that
## a dotplot (e.g., `geom_dots()`) or histogram (e.g., `density_histogram()`,
## `stat_slab(density = 'histogram')`, or `stat_histinterval()`) may better
## represent the data.
```

```
## Caused by error in `bw.SJ()`:
## ! sample is too sparse to find TD
## Warning: `position_dodge()` requires non-overlapping x intervals.
```



2B and S4-1.pdf

```
#dev.off()
```

```
### All years nMDS #####
```

```
# first make data with metadata in columns, then columns for size distributions and years in the rows w
```

```
size_nmds<- size_loc.cat %>%
```

```
  select(year, loc.cat, month, radius) %>%
```

```
  mutate(count=1) %>%
```

```
  pivot_wider(names_from=radius, values_from = count, values_fn=sum, values_fill = 0)
```

```
### standardize and only numbers
```

```
size_nmds2<-size_nmds[,5:50]
```

```
size_nmds3<-decostand(size_nmds2, method="normalize")
```

```
## calculate distance matrix
```

```
size_distmat <- vegdist(size_nmds2, method = "bray")
```

```
size_NMS <-metaMDS(size_distmat,
```

```
  distance = "bray",
```

```
  k = 3,
```

```
  maxit = 999,
```

```
trymax = 500,  
wascores = TRUE)
```

```
## Run 0 stress 0.1134929  
## Run 1 stress 0.1162408  
## Run 2 stress 0.1141932  
## Run 3 stress 0.1137428  
## ... Procrustes: rmse 0.02191327 max resid 0.1184937  
## Run 4 stress 0.1141915  
## Run 5 stress 0.1133509  
## ... New best solution  
## ... Procrustes: rmse 0.05985904 max resid 0.2200149  
## Run 6 stress 0.114384  
## Run 7 stress 0.1133529  
## ... Procrustes: rmse 0.04195281 max resid 0.2820011  
## Run 8 stress 0.1133875  
## ... Procrustes: rmse 0.07296308 max resid 0.2837303  
## Run 9 stress 0.1133948  
## ... Procrustes: rmse 0.07420756 max resid 0.2834057  
## Run 10 stress 0.1135565  
## ... Procrustes: rmse 0.06696862 max resid 0.2867678  
## Run 11 stress 0.1134766  
## ... Procrustes: rmse 0.06176827 max resid 0.2241933  
## Run 12 stress 0.1133374  
## ... New best solution  
## ... Procrustes: rmse 0.03732193 max resid 0.2745882  
## Run 13 stress 0.1135562  
## ... Procrustes: rmse 0.06792508 max resid 0.2829199  
## Run 14 stress 0.113058  
## ... New best solution  
## ... Procrustes: rmse 0.04048499 max resid 0.2728937  
## Run 15 stress 0.1138364  
## Run 16 stress 0.11342  
## ... Procrustes: rmse 0.04332312 max resid 0.2619933  
## Run 17 stress 0.1140209  
## Run 18 stress 0.1138497  
## Run 19 stress 0.1140144  
## Run 20 stress 0.1141325  
## Run 21 stress 0.1136356  
## Run 22 stress 0.1156917  
## Run 23 stress 0.1143173  
## Run 24 stress 0.1137362  
## Run 25 stress 0.1142141  
## Run 26 stress 0.1144854  
## Run 27 stress 0.113427  
## ... Procrustes: rmse 0.04530746 max resid 0.2520976  
## Run 28 stress 0.1133912  
## ... Procrustes: rmse 0.07222323 max resid 0.2856579  
## Run 29 stress 0.113467  
## ... Procrustes: rmse 0.04064297 max resid 0.1739566  
## Run 30 stress 0.1149519  
## Run 31 stress 0.1141908  
## Run 32 stress 0.1149525  
## Run 33 stress 0.1149537
```

```

## Run 34 stress 0.1137448
## Run 35 stress 0.1142605
## Run 36 stress 0.1134761
## ... Procrustes: rmse 0.06290357  max resid 0.2209756
## Run 37 stress 0.1178985
## Run 38 stress 0.1152342
## Run 39 stress 0.1136245
## Run 40 stress 0.1141895
## Run 41 stress 0.1181044
## Run 42 stress 0.1133917
## ... Procrustes: rmse 0.07495844  max resid 0.2856815
## Run 43 stress 0.1160187
## Run 44 stress 0.1138965
## Run 45 stress 0.1139426
## Run 46 stress 0.1137346
## Run 47 stress 0.1141316
## Run 48 stress 0.1155896
## Run 49 stress 0.1138362
## Run 50 stress 0.1136459
## Run 51 stress 0.1133943
## ... Procrustes: rmse 0.07347454  max resid 0.2852525
## Run 52 stress 0.1142596
## Run 53 stress 0.115232
## Run 54 stress 0.1160159
## Run 55 stress 0.1148604
## Run 56 stress 0.1137438
## Run 57 stress 0.115158
## Run 58 stress 0.1142594
## Run 59 stress 0.1155903
## Run 60 stress 0.1130562
## ... New best solution
## ... Procrustes: rmse 0.004192401  max resid 0.02603341
## Run 61 stress 0.1137222
## Run 62 stress 0.1149524
## Run 63 stress 0.1155906
## Run 64 stress 0.1131089
## ... Procrustes: rmse 0.02048077  max resid 0.1256623
## Run 65 stress 0.1176636
## Run 66 stress 0.1176869
## Run 67 stress 0.1137337
## Run 68 stress 0.113352
## ... Procrustes: rmse 0.02314122  max resid 0.1189517
## Run 69 stress 0.1174893
## Run 70 stress 0.1134753
## ... Procrustes: rmse 0.06413731  max resid 0.2203712
## Run 71 stress 0.1155359
## Run 72 stress 0.1141574
## Run 73 stress 0.1161841
## Run 74 stress 0.1134763
## ... Procrustes: rmse 0.06425741  max resid 0.2205257
## Run 75 stress 0.1161096
## Run 76 stress 0.1136277
## Run 77 stress 0.116344
## Run 78 stress 0.1176671

```

```

## Run 79 stress 0.1134234
## ... Procrustes: rmse 0.04622442 max resid 0.2499099
## Run 80 stress 0.1137457
## Run 81 stress 0.1134155
## ... Procrustes: rmse 0.04505047 max resid 0.2533027
## Run 82 stress 0.1133896
## ... Procrustes: rmse 0.0759856 max resid 0.2872847
## Run 83 stress 0.1141892
## Run 84 stress 0.1133898
## ... Procrustes: rmse 0.07389187 max resid 0.2873197
## Run 85 stress 0.1134184
## ... Procrustes: rmse 0.07368274 max resid 0.2853124
## Run 86 stress 0.115922
## Run 87 stress 0.114168
## Run 88 stress 0.1141905
## Run 89 stress 0.1171021
## Run 90 stress 0.1149534
## Run 91 stress 0.1134187
## ... Procrustes: rmse 0.04561938 max resid 0.2522211
## Run 92 stress 0.1169493
## Run 93 stress 0.1198616
## Run 94 stress 0.1134177
## ... Procrustes: rmse 0.07207994 max resid 0.2829489
## Run 95 stress 0.1134116
## ... Procrustes: rmse 0.07255036 max resid 0.2836751
## Run 96 stress 0.1143842
## Run 97 stress 0.1142686
## Run 98 stress 0.1142043
## Run 99 stress 0.1144877
## Run 100 stress 0.1137646
## Run 101 stress 0.1141901
## Run 102 stress 0.1136642
## Run 103 stress 0.1145634
## Run 104 stress 0.1141916
## Run 105 stress 0.1133936
## ... Procrustes: rmse 0.07641129 max resid 0.2869839
## Run 106 stress 0.1144598
## Run 107 stress 0.1167045
## Run 108 stress 0.1138451
## Run 109 stress 0.113391
## ... Procrustes: rmse 0.076197 max resid 0.286975
## Run 110 stress 0.1161825
## Run 111 stress 0.1175022
## Run 112 stress 0.113476
## ... Procrustes: rmse 0.06416604 max resid 0.2202505
## Run 113 stress 0.1148557
## Run 114 stress 0.1166695
## Run 115 stress 0.1135885
## Run 116 stress 0.1141912
## Run 117 stress 0.1194048
## Run 118 stress 0.115589
## Run 119 stress 0.119093
## Run 120 stress 0.1133508
## ... Procrustes: rmse 0.03627056 max resid 0.2828676

```



```

## Run 121 stress 0.1174649
## Run 122 stress 0.1139408
## Run 123 stress 0.1171387
## Run 124 stress 0.1153143
## Run 125 stress 0.1134161
## ... Procrustes: rmse 0.04482205 max resid 0.256414
## Run 126 stress 0.1135509
## ... Procrustes: rmse 0.07211625 max resid 0.2905404
## Run 127 stress 0.1138747
## Run 128 stress 0.113584
## Run 129 stress 0.1173988
## Run 130 stress 0.1187612
## Run 131 stress 0.1134948
## ... Procrustes: rmse 0.06214737 max resid 0.2194386
## Run 132 stress 0.1155689
## Run 133 stress 0.1133882
## ... Procrustes: rmse 0.07571631 max resid 0.2872913
## Run 134 stress 0.1138675
## Run 135 stress 0.1152372
## Run 136 stress 0.1196855
## Run 137 stress 0.113387
## ... Procrustes: rmse 0.0755216 max resid 0.2873563
## Run 138 stress 0.1168354
## Run 139 stress 0.1153341
## Run 140 stress 0.1135559
## ... Procrustes: rmse 0.06995759 max resid 0.291148
## Run 141 stress 0.1160015
## Run 142 stress 0.1143905
## Run 143 stress 0.1203878
## Run 144 stress 0.1151572
## Run 145 stress 0.116698
## Run 146 stress 0.1172477
## Run 147 stress 0.1155896
## Run 148 stress 0.1133905
## ... Procrustes: rmse 0.07480962 max resid 0.2875065
## Run 149 stress 0.1172091
## Run 150 stress 0.1134199
## ... Procrustes: rmse 0.07221091 max resid 0.28279
## Run 151 stress 0.1133836
## ... Procrustes: rmse 0.02938964 max resid 0.1435413
## Run 152 stress 0.1135843
## Run 153 stress 0.1137334
## Run 154 stress 0.1135846
## Run 155 stress 0.1149529
## Run 156 stress 0.1144594
## Run 157 stress 0.1162971
## Run 158 stress 0.1133359
## ... Procrustes: rmse 0.04053238 max resid 0.2699538
## Run 159 stress 0.1167176
## Run 160 stress 0.1133919
## ... Procrustes: rmse 0.07600507 max resid 0.287473
## Run 161 stress 0.1171876
## Run 162 stress 0.1144297
## Run 163 stress 0.115158

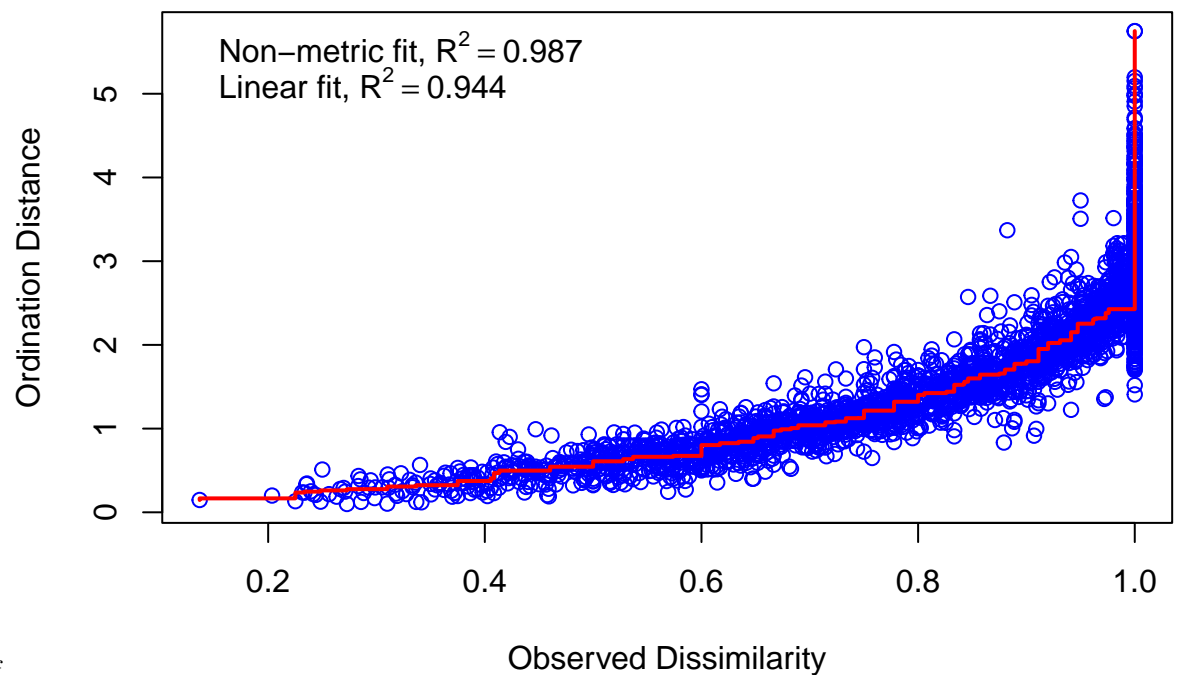
```

```
## Run 164 stress 0.1135524
## ... Procrustes: rmse 0.07028997 max resid 0.2902524
## Run 165 stress 0.1143759
## Run 166 stress 0.1133377
## ... Procrustes: rmse 0.04084447 max resid 0.2691965
## Run 167 stress 0.113477
## ... Procrustes: rmse 0.06411605 max resid 0.220921
## Run 168 stress 0.1137374
## Run 169 stress 0.1130586
## ... Procrustes: rmse 0.001041878 max resid 0.007407465
## ... Similar to previous best
## *** Best solution repeated 1 times
```

```
goodness(size_NMS)
```

```
## [1] 0.019344419 0.015039398 0.015037685 0.011008190 0.010731239 0.015674920
## [7] 0.011210071 0.017014807 0.018789264 0.020580179 0.017573124 0.017383653
## [13] 0.014925001 0.021920228 0.018154618 0.012429893 0.018242729 0.014868107
## [19] 0.020387060 0.011364241 0.008958086 0.010392235 0.014270281 0.016878110
## [25] 0.012647564 0.011297994 0.017876432 0.012591292 0.011005268 0.008404875
## [31] 0.012522524 0.009526689 0.009505352 0.011318288 0.013408960 0.014801043
## [37] 0.016392876 0.007799640 0.012990327 0.009688445 0.018522523 0.013474680
## [43] 0.017539726 0.007353003 0.010863055 0.010825233 0.014088646 0.011354700
## [49] 0.014397610 0.008617682 0.009135269 0.013276219 0.016451741 0.013594479
## [55] 0.009443336 0.006837369 0.010307436 0.010373836 0.010008299 0.007022029
## [61] 0.011710353 0.007389543 0.013941946 0.007363824 0.009374097 0.009398617
## [67] 0.006098101 0.009063657 0.010304569 0.012537582 0.008406760 0.011113194
## [73] 0.008061668 0.009471835
```

```
stressplot(size_NMS)
```



2B and S4-2.pdf

```
### pull back the metadata
size_nmds4<- size_nmds[,1:4]
```

```

## output the nmbs scores with the variables
data.scores = as.data.frame(scores(size_NMS))

size_nmds5<-cbind(size_nmds4, data.scores)

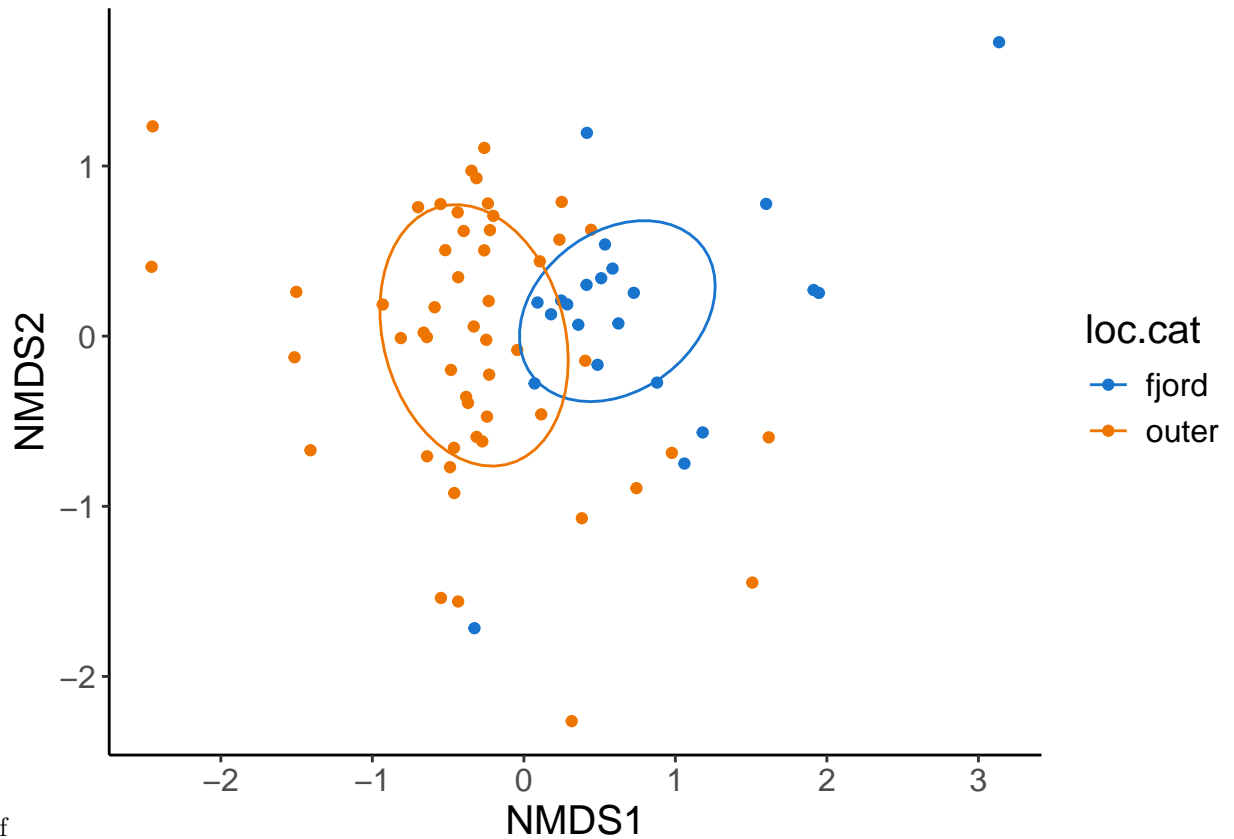
## adonis analysis - better then anosim, basically anova
size_adonis<-adonis2(size_nmds3~loc.cat*year, data=size_nmds4)

size_adonis

## Permutation test for adonis under reduced model
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = size_nmds3 ~ loc.cat * year, data = size_nmds4)
##           Df SumOfSqs      R2      F Pr(>F)
## Model      3   4.2587 0.19973 5.8236 0.001 ***
## Residual  70  17.0634 0.80027
## Total     73  21.3221 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#Figures S4
## size community and location catagory
#png("FigS4_28Feb25.png", height=10, width=12, units='cm', res=600)
ggplot(size_nmds5, aes(NMDS1, NMDS2, colour=loc.cat))+geom_point()+
  stat_ellipse(aes(x=NMDs1,y=NMDs2,colour=loc.cat),level = 0.50) +
  scale_color_manual(values=c("dodgerblue3","darkorange2")) +
  theme_bw() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=15),
        #legend.position = "none",
        axis.line = element_line(colour = "black"))

```



2B and S4-3.pdf

```
#dev.off()
```

Fig. 3

```
#set figure theme
theme_set(theme_bw() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=15),
        #legend.position = "none",
        axis.line = element_line(colour = "black"))))

### filter data to only stars that were wasting is above 0 and the summarize to count number of wasting
wp.sp<- sswdag %>% dplyr::filter(project!="Rocky Intertidal", wasting>0) %>%
  group_by(project, survey, year, day_month, date,site_id, site_habitat,latin_name, site_latitude)

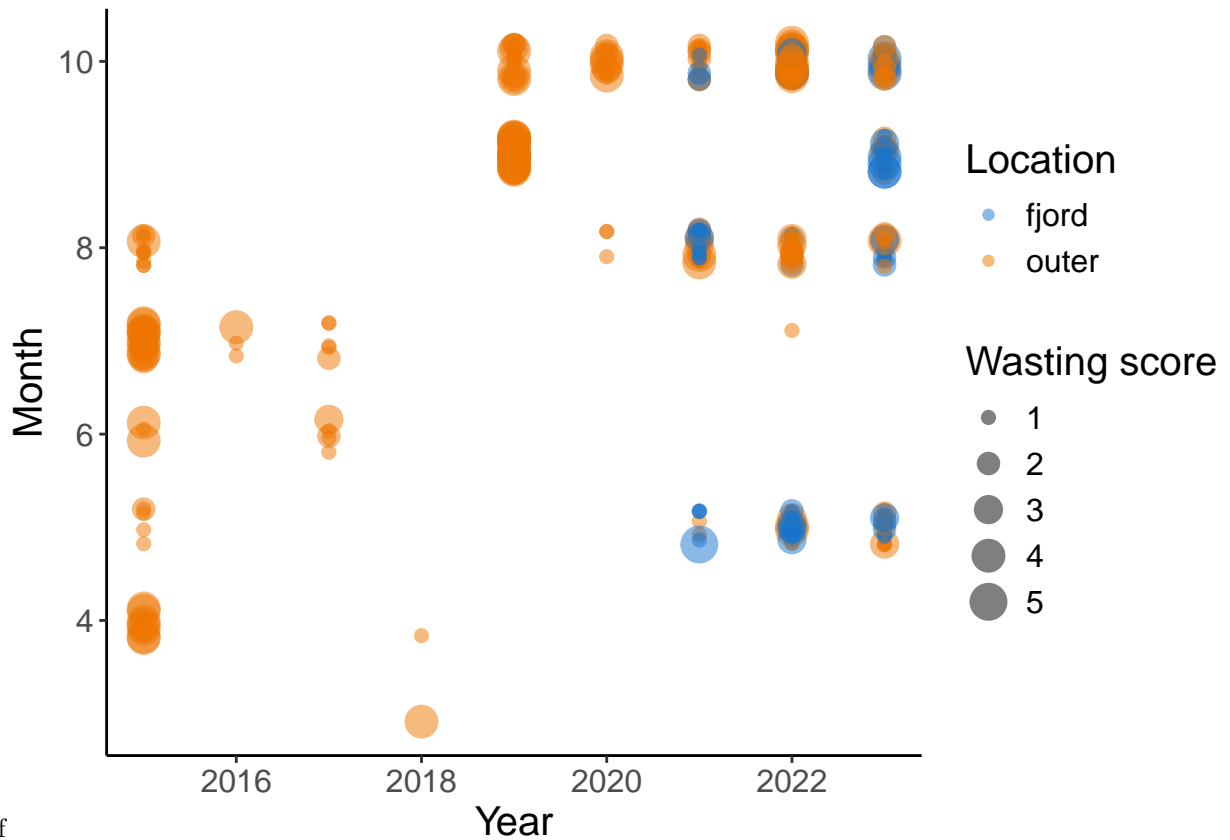
## filter data to sites that are regularly surveyed over time and split by location
wp.sp.loc<- wp.sp %>% dplyr::filter(site_id=="Kelpie Pt" | site_id=="Mercury" | site_id=="Odlum Pt" | 
  mutate(loc.cat=c(ifelse(site_id=="Kelpie Pt" | site_id=="Mercury" | site_id=="Odlum Pt" | site_id=="T

#filter to only wasting stars
wp.sp2<- sswdag %>% dplyr::filter(project!="Rocky Intertidal") %>%
  group_by(project, survey, year, day_month, date,site_id, site_habitat,latin_name, site_latitude)

## Figure 3 - wasting score by date
```

```
#png("Fig 3_28Feb25.png", height=10, width=20, units='cm', res=600)
pl2<-ggplot(wp.sp.loc)+
  geom_jitter(aes(year(date), month(date), size=factor(wasting), color=loc.cat), width = 0, height=.2,
  xlab("Year")+ ylab("Month")+
  labs(color="Location", size="Wasting score")+
  scale_color_manual(values=c("dodgerblue3","darkorange2"))
pl2
```

```
## Warning: Using size for a discrete variable is not advised.
```



3-1.pdf

```
#dev.off()
```

Fig. 4 and Table 2

```
library(itsadug)
```

```
## Loading required package: plotfunctions
##
## Attaching package: 'plotfunctions'
## The following object is masked from 'package:ggplot2':
##
##   alpha
## Loaded package itsadug 2.4 (see 'help("itsadug")' ).
##
## Attaching package: 'itsadug'
## The following objects are masked from 'package:tidymv':
```

```

##
## find_difference, get_difference, summary_data
library(ggeffects)

##
## Attaching package: 'ggeffects'
## The following object is masked from 'package:itsadug':
##
## get_predictions
## The following object is masked from 'package:cowplot':
##
## get_title
library(MapGAM)

## Loading required package: sp
## Loading required package: gam
## Loading required package: splines
## Loading required package: foreach
##
## Attaching package: 'foreach'
## The following objects are masked from 'package:purrr':
##
## accumulate, when
## Loaded gam 1.22-5
##
## Attaching package: 'gam'
## The following objects are masked from 'package:mgcv':
##
## gam, gam.control, gam.fit, s
## Loading required package: survival
##
## Attaching package: 'survival'
## The following object is masked from 'package:quantreg':
##
## untangle.specials
library(sf)

## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
library(sjPlot)

#####
##### GAMS #####
#####

## filtered to remove sites in the fjords, because not sampled before wasting

```

```

ssbio.pyc<-ssbio %>% dplyr::filter(latin_name=="Pycnopodia helianthoides"&experimental_unit!="quadrats")
dplyr::mutate(latitude= ifelse (site_latitude>0&!is.na(site_latitude), site_latitude,
longitude= ifelse(site_longitude<0&!is.na(site_longitude), site_longitude,

### gam analysis

## transform date for gam analysis
ssbio.pyc <- transform(ssbio.pyc, ndate = as.numeric(date),
nyear = as.numeric(format(date, '%Y')),
nmonth = as.numeric(format(date, '%m')),
doy = as.numeric(format(date, '%j')),
transectid=as.factor(transect_id))

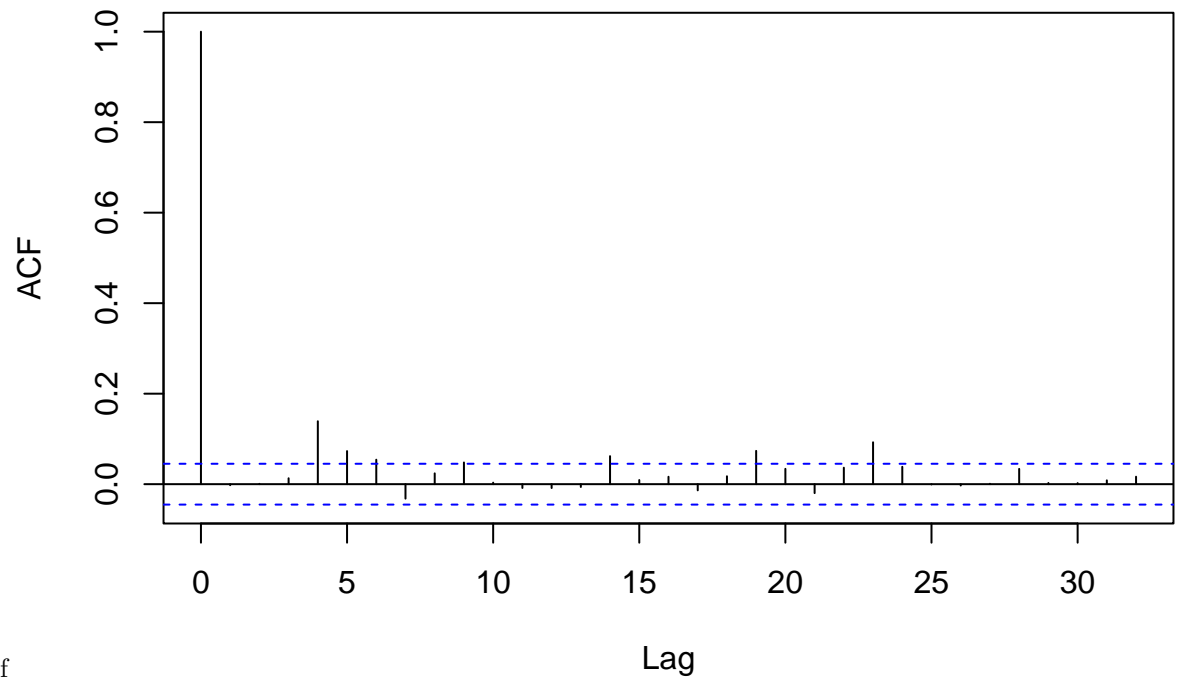
#analysis for Figure 4 and Table 2
### gam using individual response
gamm1.03<-mgcv::gamm(biomass_density~ depth_bin +
s(ndate, bs="gp", k=8) +
s(nmonth, bs="cc", k=3) +
s(transectid, bs="re")+
s(latitude, longitude, k=20),
data=ssbio.pyc, family=tw, method="REML",
correlation = corARMA(form = ~ 1|nyear, p = 0, q=3)) #tw or nb

## Warning in smooth.construct.cc.smooth.spec(object, dk$data, dk$knots): basis dimension, k, increased
##
## Maximum number of PQL iterations: 20
## iteration 1
## iteration 2
## iteration 3
## iteration 4
## iteration 5
## iteration 6
## iteration 7
## iteration 8
## iteration 9
## iteration 10
#### AIC test
AIC(gamm1.03$lme) #9448.61

## [1] 9448.61
### model evaluation
acf(resid(gamm1.03$lme, type = "normalized")) ## normalized residuals better to test

```

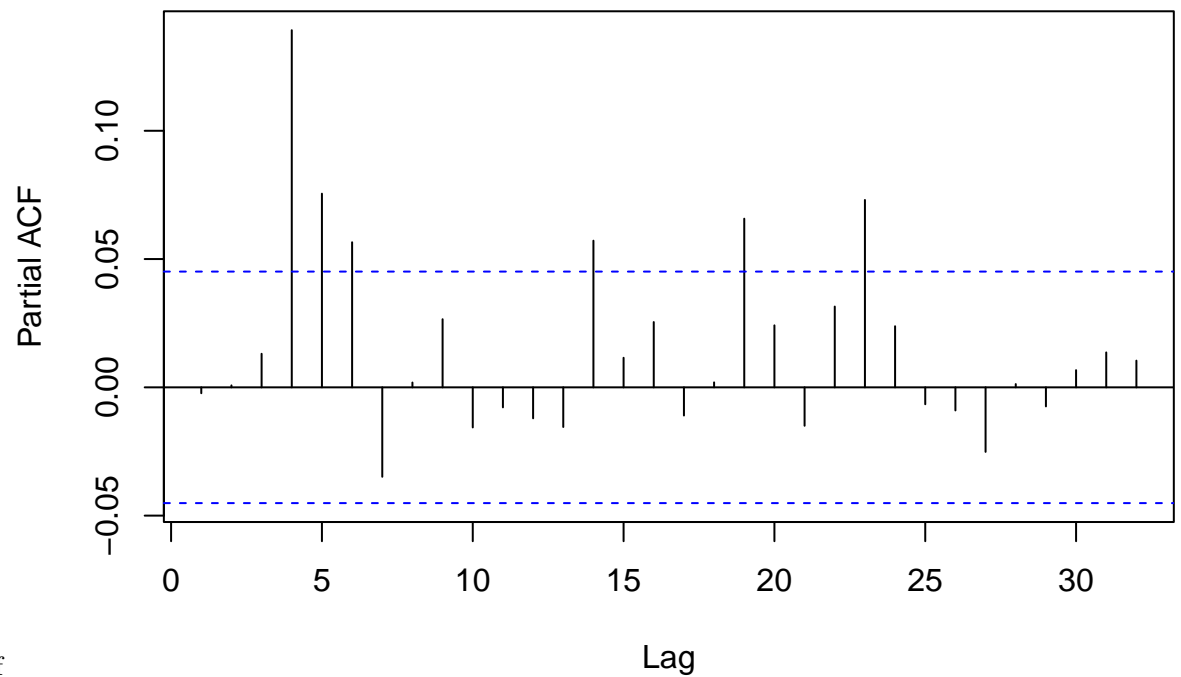
Series resid(gamm1.03\$lme, type = "normalized")



4 and Table 2-1.pdf

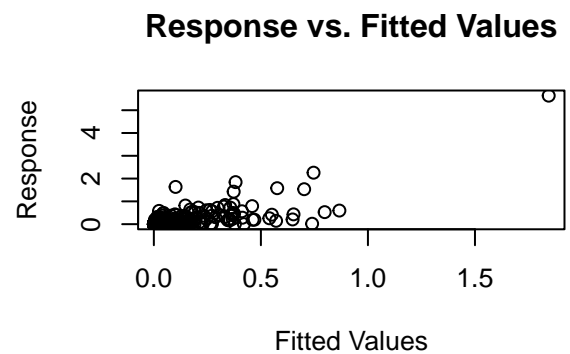
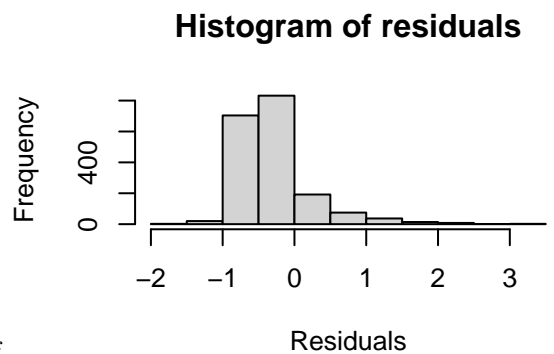
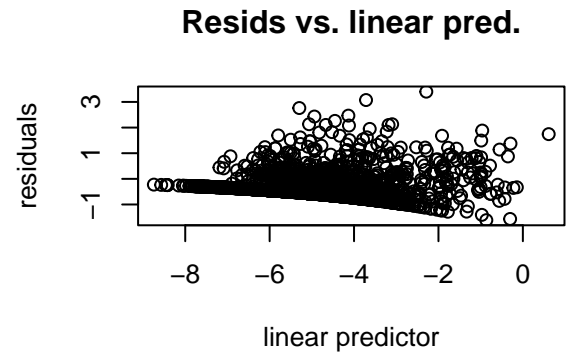
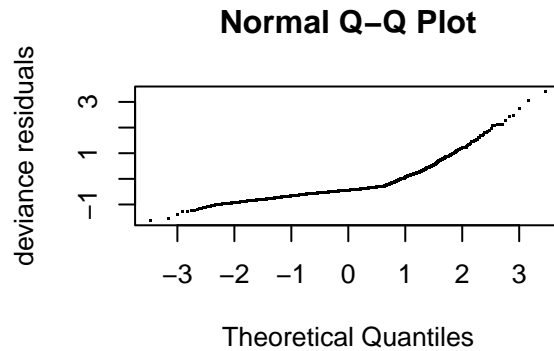
```
pacf(resid(gamm1.03$lme, type = "normalized"))
```

Series resid(gamm1.03\$lme, type = "normalized")



4 and Table 2-2.pdf

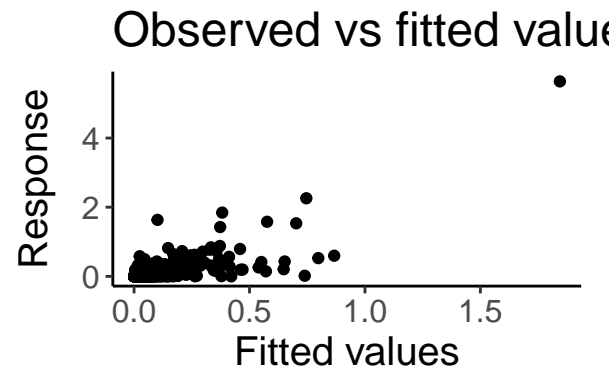
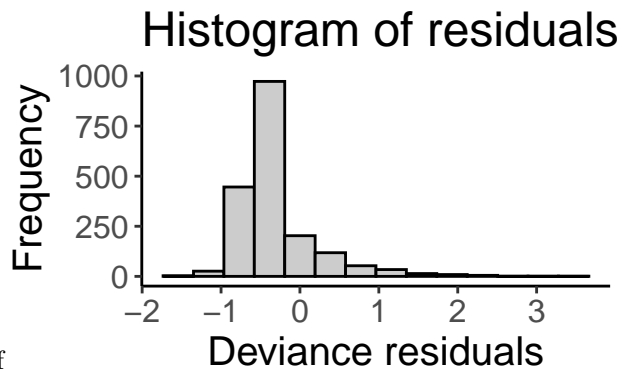
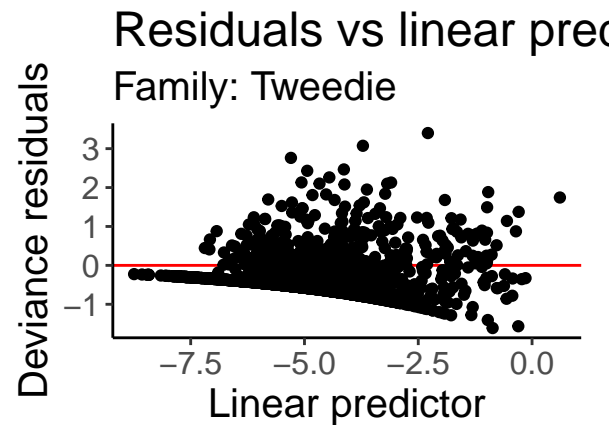
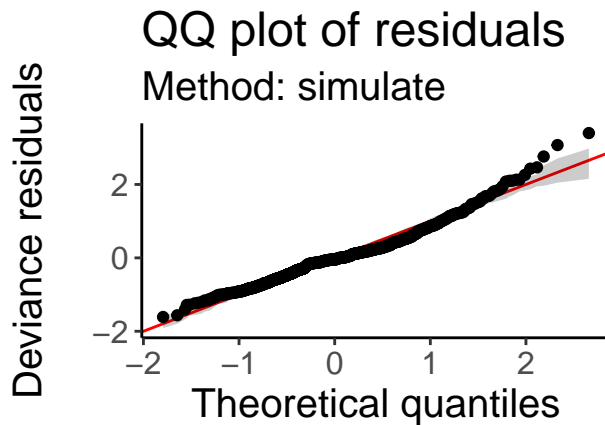

```
gam.check(gamm1.03$gam)
```



4 and Table 2-3.pdf

```
##
## 'gamm' based fit - care required with interpretation.
## Checks based on working residuals may be misleading.
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##           k'    edf k-index p-value
## s(ndate)      7.00  4.87   0.57 <2e-16 ***
## s(nmonth)     2.00  1.95   0.57 <2e-16 ***
## s(transectid) 403.00 100.34    NA      NA
## s(latitude,longitude) 19.00 12.89   0.68 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
appraise(gamm1.03$gam)
```



4 and Table 2-4.pdf

```
summary(gamm1.03$gam)
```

```
##
## Family: Tweedie
## Link function: log
##
## Formula:
## biomass_density ~ depth_bin + s(ndate, bs = "gp", k = 8) + s(nmonth,
##   bs = "cc", k = 3) + s(transectid, bs = "re") + s(latitude,
##   longitude, k = 20)
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.3770    0.1971 -27.276  <2e-16 ***
## depth_binmid   -0.3267    0.2152  -1.518   0.1292
## depth_binshallow 0.8719    0.2407   3.623   0.0003 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf   Ref.df    F  p-value
## s(ndate)       4.868     4.868 40.373  <2e-16 ***
## s(nmonth)       1.948     2.000 34.263  <2e-16 ***
## s(transectid)  100.335    400.000  0.535  <2e-16 ***
## s(latitude,longitude) 12.893    12.893  6.794  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## R-sq.(adj) = 0.457
## Scale est. = 0.66243 n = 1887
tab_model(gamm1.03$gam)
```

Dependent variable

Predictors

Estimates

CI

p

(Intercept)

0.00

0.00 – 0.01

<0.001

depth bin [mid]

0.72

0.47 – 1.10

0.129

depth bin [shallow]

2.39

1.49 – 3.83

<0.001

Smooth term (ndate)

<0.001

Smooth term (nmonth)

<0.001

Smooth term (transectid)

<0.001

Smooth term(latitude,longitude)

<0.001

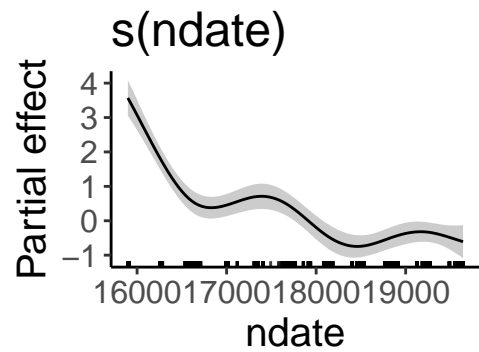
Observations

1887

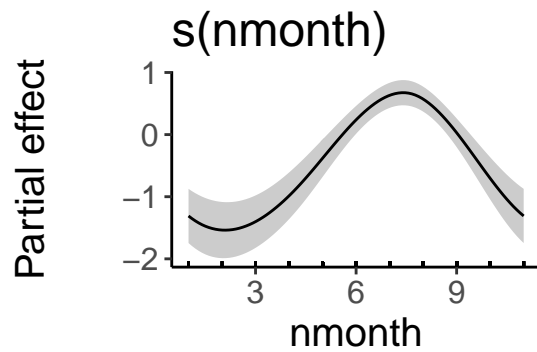
R2

0.457

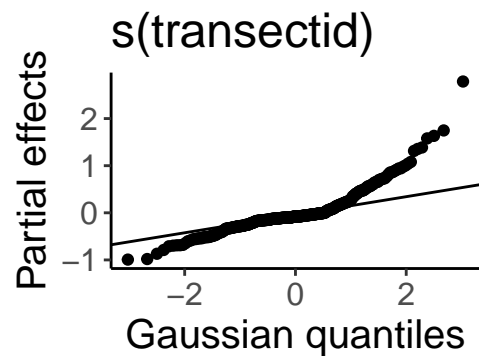
```
draw(gamm1.03$gam)
```



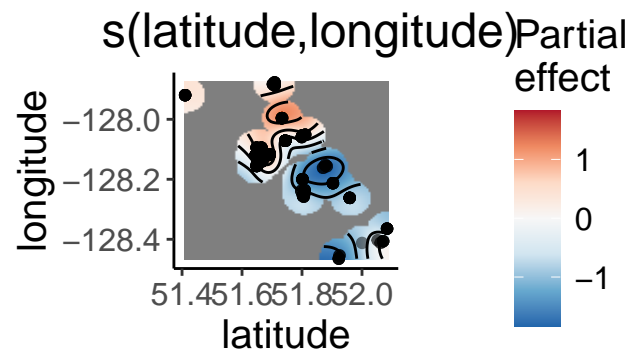
Basis: GP



Basis: Cyclic CRS



Basis: Random effect



Basis: TPRS (2d)

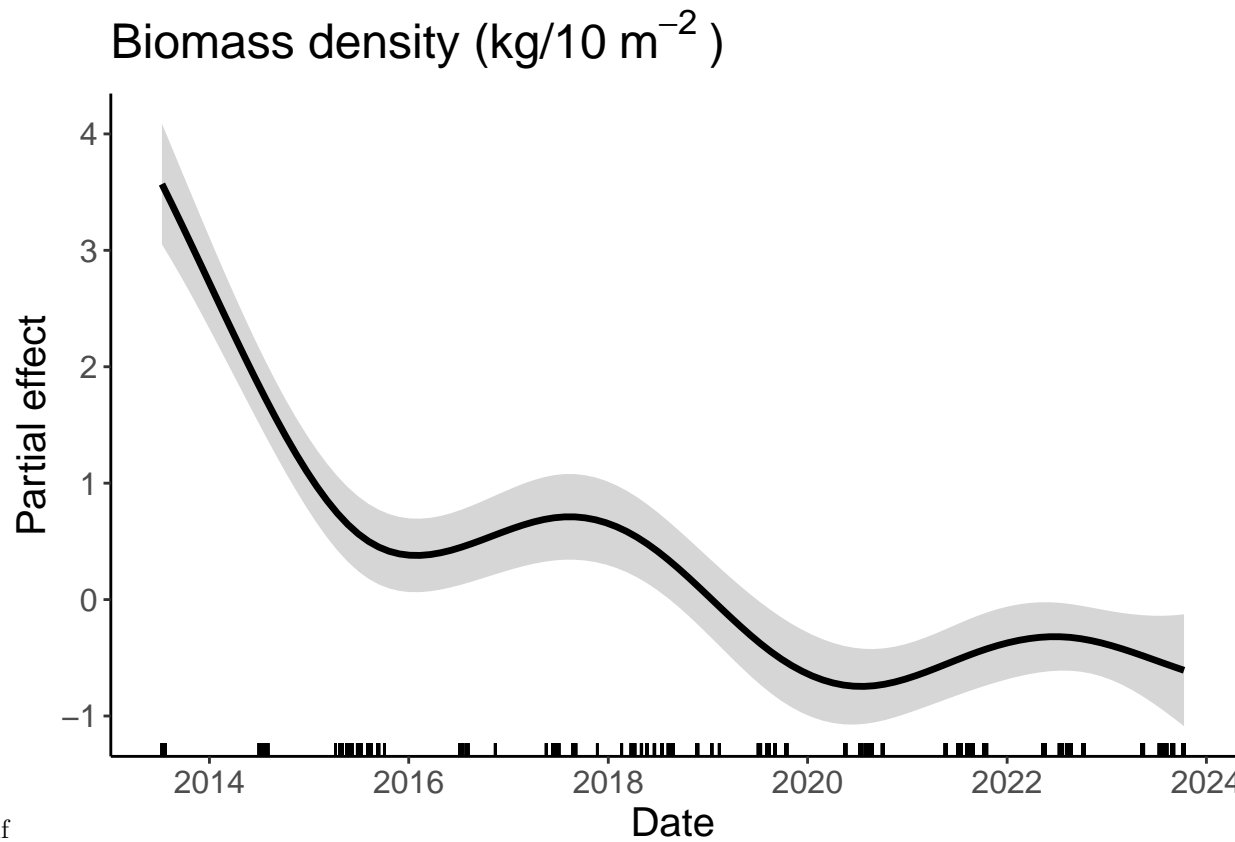
4 and Table 2-5.pdf

```
### hand draw in gratia
plotdata<-ssbio.pyc %>% select(date, ndate, nmonth, latitude, longitude, transectid, site_id, depth_bin)

sm <- smooth_estimates(gamm1.03$gam) %>%
  add_confint() %>% mutate(trans_ndate=as.Date(ndate, "1970-01-01"))

## smooth over date
cp<- sm %>%
  dplyr::filter(.smooth == "s(ndate)") %>%
  ggplot() +
  geom_rug(aes(x = trans_ndate),
    data = plotdata,
    sides = "b", length = grid::unit(0.02, "npc")) +
  geom_ribbon(aes(ymin = .lower_ci, ymax = .upper_ci, x = trans_ndate),
    alpha = 0.2) +
  geom_line(aes(x = trans_ndate, y = .estimate), lwd = 1.2) +
  labs(y = "Partial effect", x="Date", title=expression("Biomass density (kg/10^-m^-2~)"))+
  theme_bw()+
  theme(panel.grid.major.x = element_blank(),
    panel.grid.major.y = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    panel.background = element_blank(),
    text = element_text(size=15),
    #legend.position = "none",
    axis.line = element_line(colour = "black"))

#png("biomass_density_2013-2023_7Dec23.png", width=1000, height=480)
```



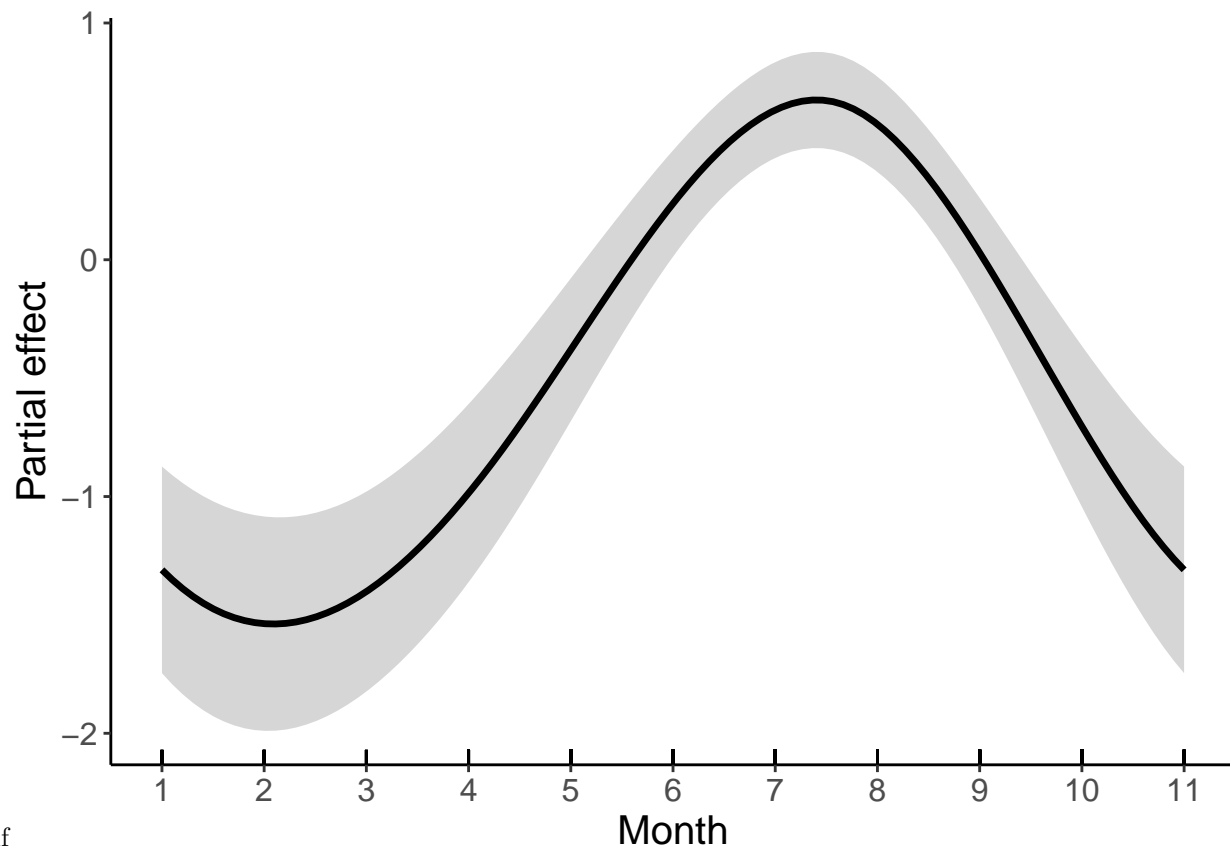
4 and Table 2-6.pdf

```
#dev.off()

### now with month ###

mp<- sm %>%
  dplyr::filter(.smooth == "s(nmonth)") %>%
  ggplot() +
  geom_rug(aes(x = nmonth),
    data = plotdata,
    sides = "b", length = grid::unit(0.02, "npc")) +
  geom_ribbon(aes(ymin = .lower_ci, ymax = .upper_ci, x = nmonth),
    alpha = 0.2) +
  geom_line(aes(x = nmonth, y = .estimate), lwd = 1.2) +
  labs(y = "Partial effect", x="Month")+theme_bw()+
  scale_x_continuous(breaks=c(1:11))+
  theme(panel.grid.major.x = element_blank(),
    panel.grid.major.y = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    panel.background = element_blank(),
    text = element_text(size=15),
    #legend.position = "none",
    axis.line = element_line(colour = "black"))
```

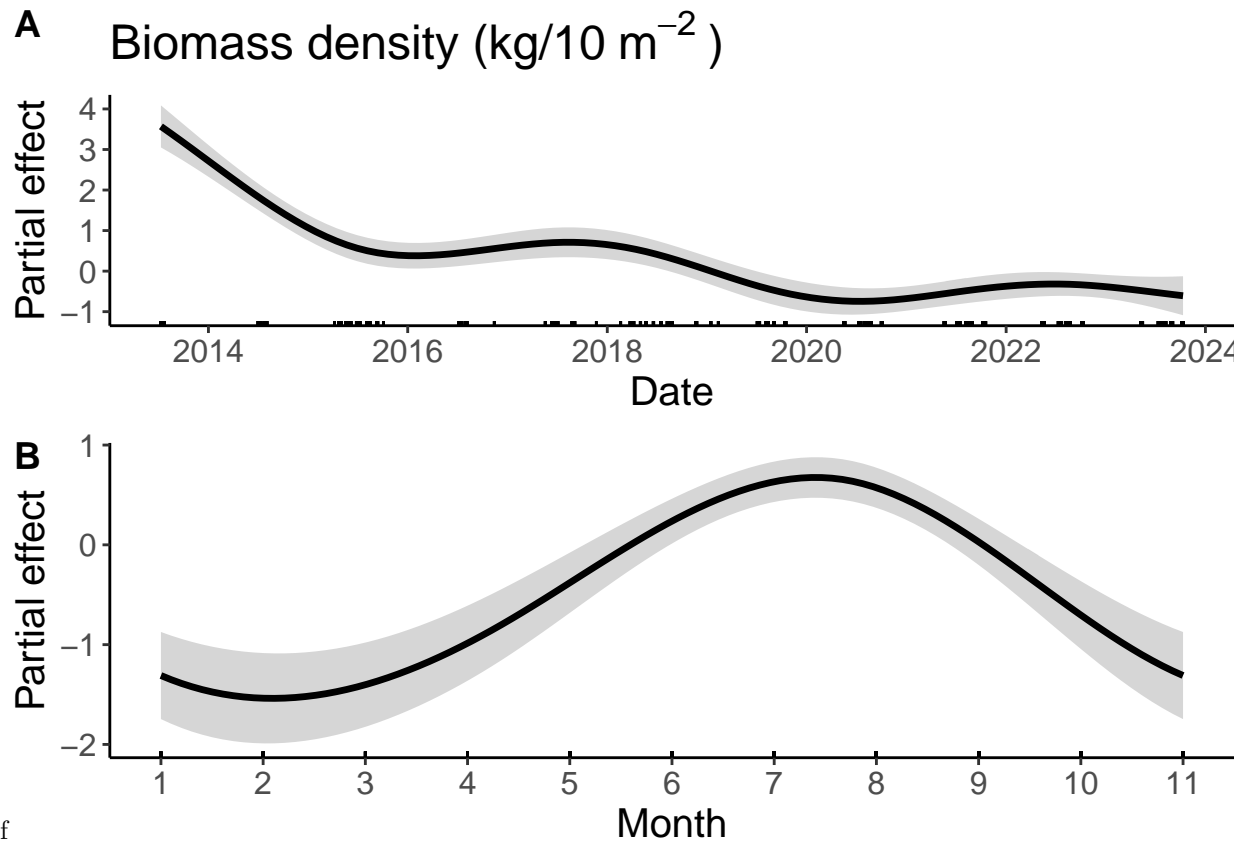
```
#png("biomass_density_bymonth_1Nov23.png", width=1000, height=480)
mp
```



4 and Table 2-7.pdf

```
#dev.off()

#Figure 4
#png("biomass_density_year_month_28Feb25.png", width=14, height=14, units = 'cm', res=600)
cowplot::plot_grid(cp, mp, nrow=2, labels = c("A", "B"))
```



4 and Table 2-8.pdf

```
#dev.off()
```

Fig 5 and Table 3

```
#transform data for gam
ssbio.pyc.o <- transform(pyc_aligned, ndate = as.numeric(date),
  nyear = as.numeric(format(date, '%Y')),
  nmonth = as.numeric(format(date, '%m')),
  doy = as.numeric(format(date, '%j')))

ssbio.pyc.o$survey<-factor(ssbio.pyc.o$survey)
ssbio.pyc.o$loc.cat<-factor(ssbio.pyc.o$loc.cat)
ssbio.pyc.o$transect_id<-factor(ssbio.pyc.o$transect_id)
ssbio.pyc.o$site_id<-factor(ssbio.pyc.o$site_id)

## cut to even data, taking out prior to 2021 and the unpaired july data
ssbio.pyc.o2<-ssbio.pyc.o %>% dplyr::filter(date>"2021-01-01") %>% dplyr::filter(nmonth!="7")

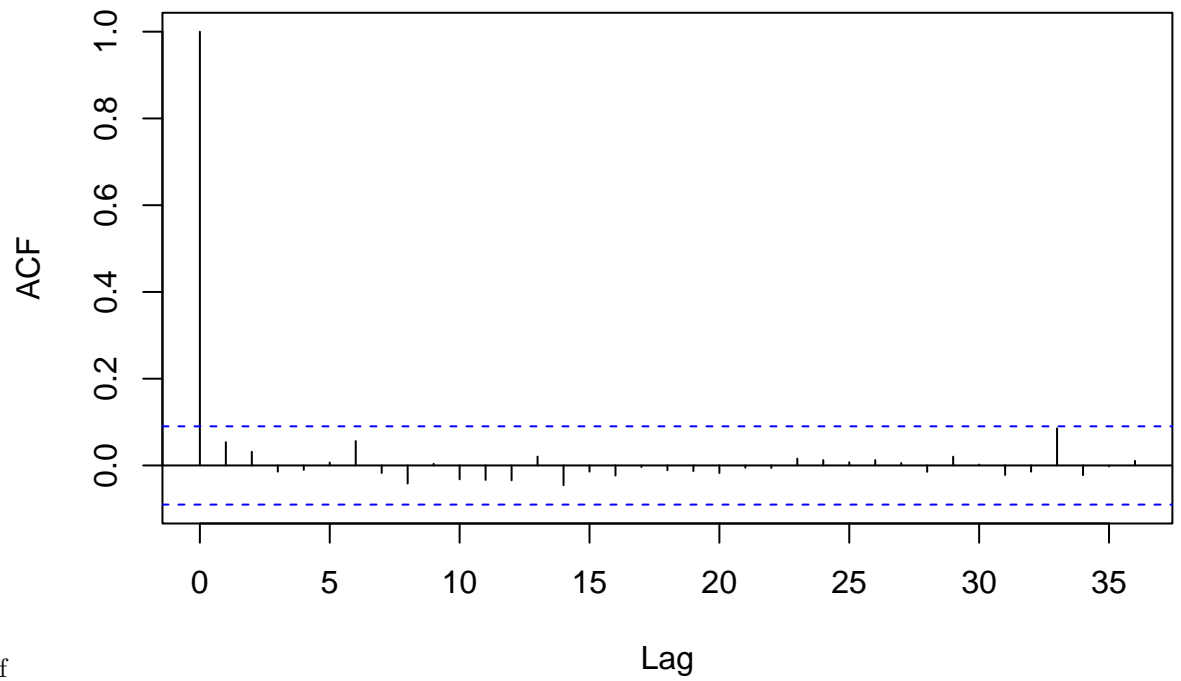
### gam using individual response
gam4<-mgcv::gamm(biomass_density~ loc.cat+
  s(ndate, bs="gp", by=loc.cat, k=10)+
  s(nmonth, bs="cc", k=4)+ #region specific smooth, and 'cc' makes it so everyt
  s(salinity.psu, temp.deg.c, by=loc.cat, bs="gp", k=10)+ ### salinity and temp
  s(site_longitude, site_latitude, k=20),
  data=ssbio.pyc.o2, family=tw,
  control = lmeControl(msMaxIter = 1000, msMaxEval = 1000),
  niterPQL=30, method="REML")#,
```

```
##
## Maximum number of PQL iterations: 30
## iteration 1
## iteration 2
## iteration 3
## iteration 4
## iteration 5
## iteration 6
## iteration 7
#model evaluation
concurvity(gam4$gam)

##           para s(ndate):loc.catfjord s(ndate):loc.catouter s(nmonth)
## worst      0.9999551           0.9899135           0.9948612 0.9964977
## observed 0.9999551           0.2716223           0.6039769 0.9623223
## estimate 0.9999551           0.2716223           0.6039769 0.9082882
##           s(salinity.psu,temp.deg.c):loc.catfjord
## worst                        0.8690961
## observed                     0.1727470
## estimate                     0.2612099
##           s(salinity.psu,temp.deg.c):loc.catouter
## worst                        0.9457620
## observed                     0.4695388
## estimate                     0.4348719
##           s(site_longitude,site_latitude)
## worst                        0.9999125
## observed                     0.8326180
## estimate                     0.4363754

res <- resid(gam4$lme, type = "normalized")
acf(res, lag.max = 36, main = "ACF - AR(2) errors")
```

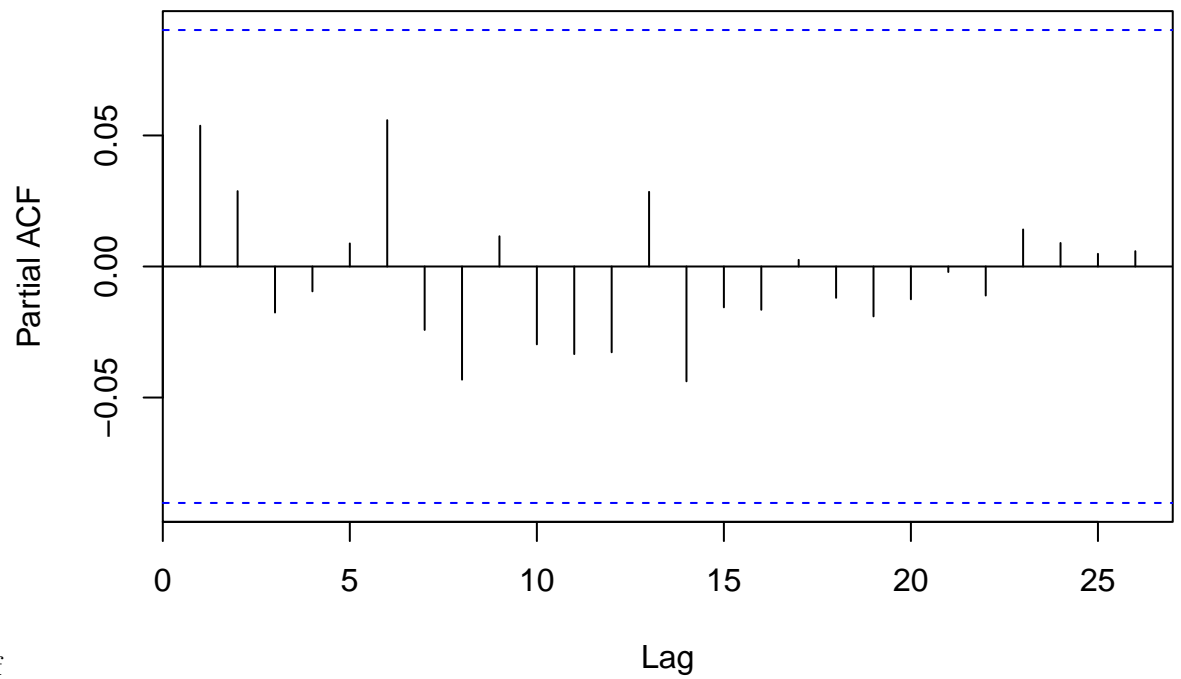

ACF – AR(2) errors



5 and Table 3-1.pdf

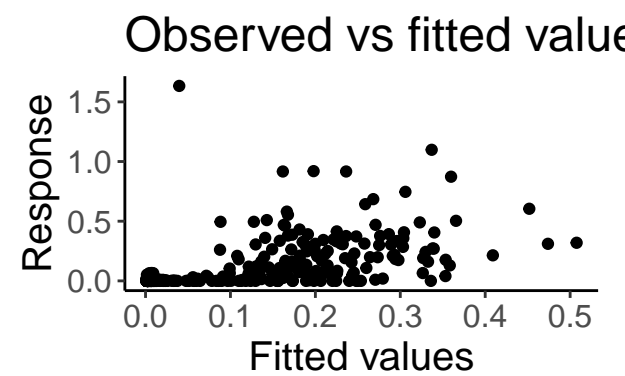
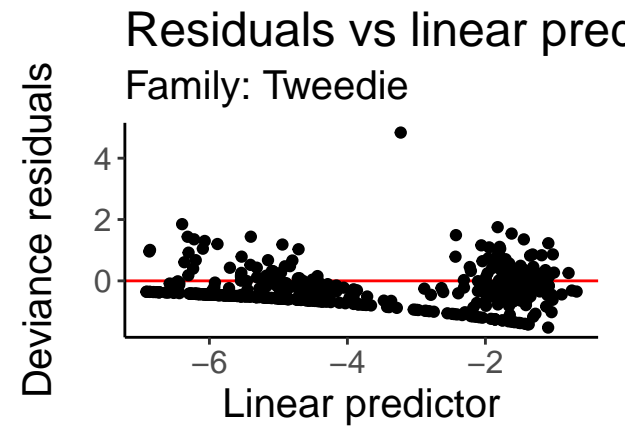
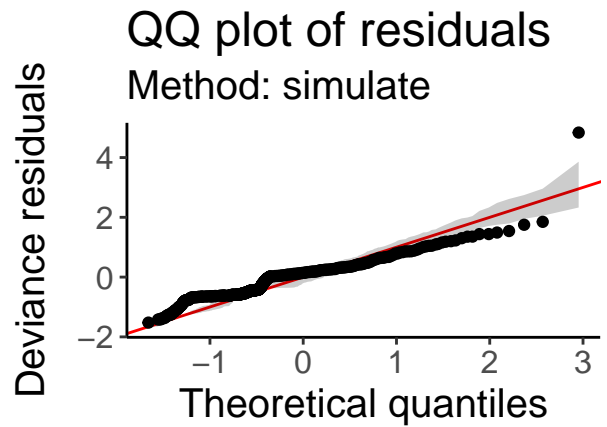
```
pacf(res)
```

Series res



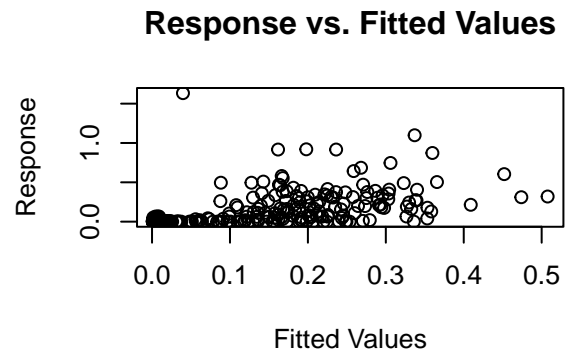
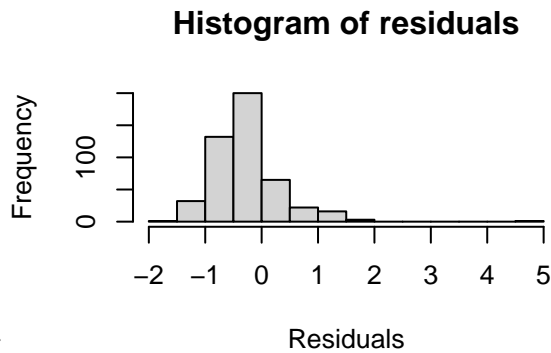
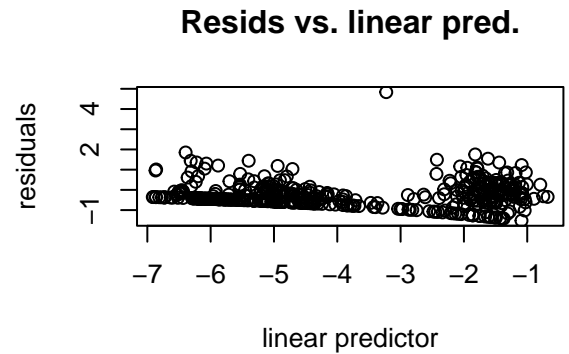
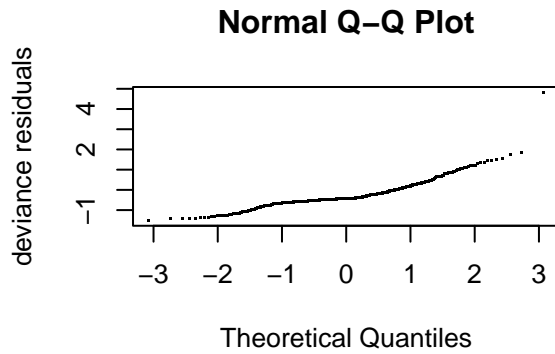
5 and Table 3-2.pdf

```
appraise(gam4$gam)
```



5 and Table 3-3.pdf

```
gam.check(gam4$gam)
```



5 and Table 3-4.pdf

```
##
## 'gamm' based fit - care required with interpretation.
## Checks based on working residuals may be misleading.
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##
```

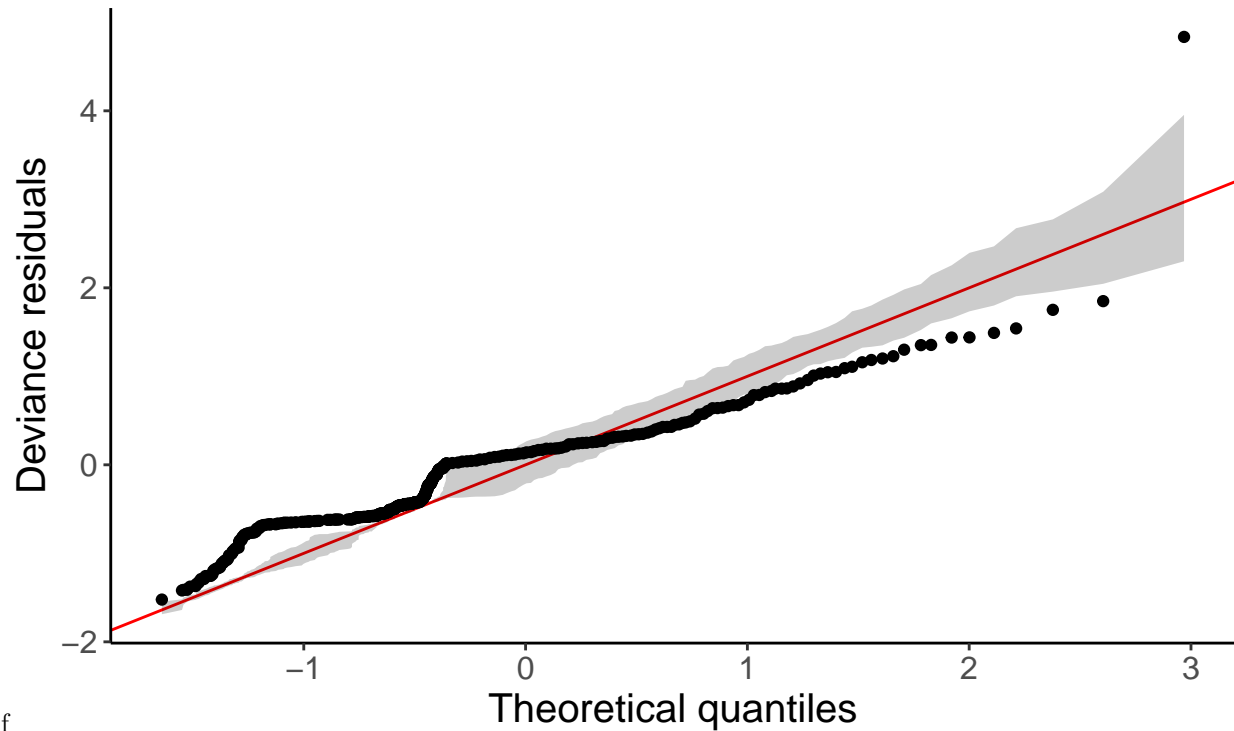
	k'	edf	k-index	p-value
## s(ndate):loc.catfjord	9.000	1.000	0.69	<2e-16 ***
## s(ndate):loc.catouter	9.000	1.000	0.69	<2e-16 ***
## s(nmonth)	2.000	0.814	0.70	<2e-16 ***
## s(salinity.psu,temp.deg.c):loc.catfjord	9.000	2.000	0.72	0.005 **
## s(salinity.psu,temp.deg.c):loc.catouter	9.000	2.000	0.72	0.005 **
## s(site_longitude,site_latitude)	19.000	2.000	0.76	0.090 .

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
gratia::qq_plot(gam4$gam)
```

QQ plot of residuals

Method: simulate



5 and Table 3-5.pdf

check the basis number used

```
k.check(gam4$gam)
```

	k'	edf	k-index	p-value
## s(ndate):loc.catfjord	9	1.0000000	0.6942777	0.0000
## s(ndate):loc.catouter	9	1.0000000	0.6942777	0.0000
## s(nmonth)	2	0.8139834	0.6962120	0.0000
## s(salinity.psu,temp.deg.c):loc.catfjord	9	1.9999999	0.7185680	0.0025
## s(salinity.psu,temp.deg.c):loc.catouter	9	2.0000001	0.7185680	0.0150
## s(site_longitude,site_latitude)	19	1.9999987	0.7554476	0.1025

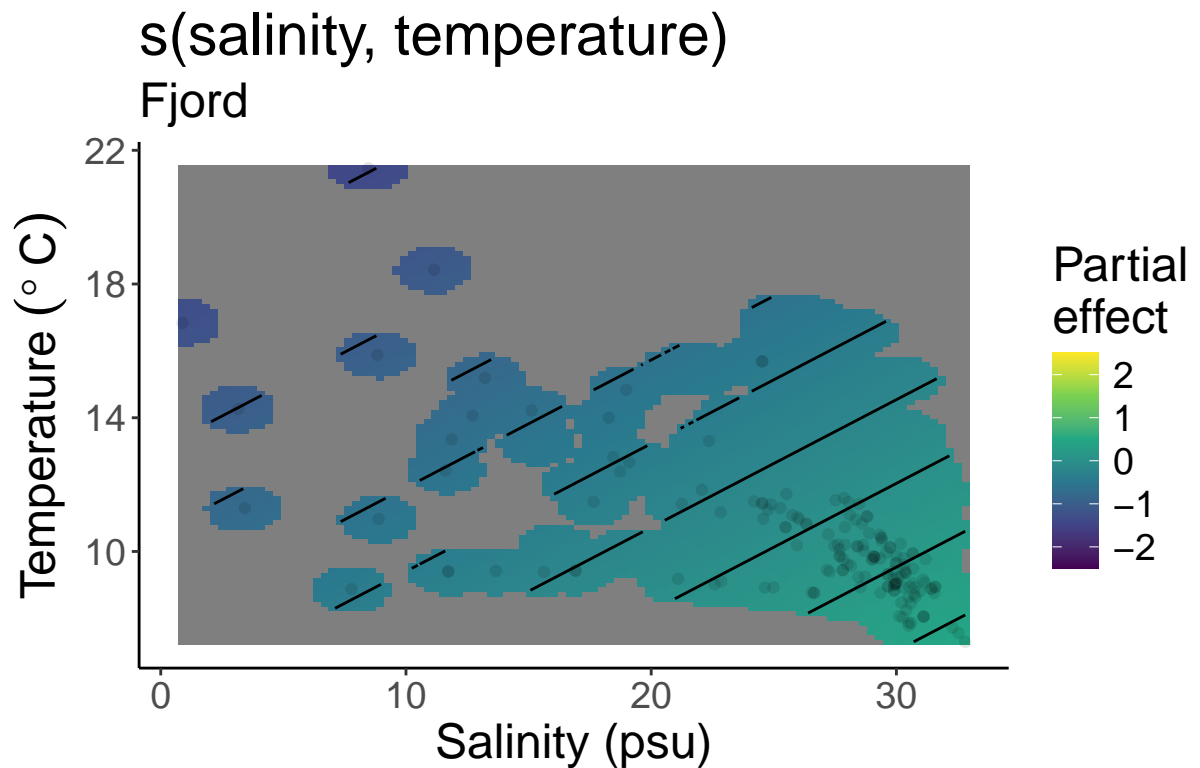
```
summary(gam4$gam)
```

```
##
## Family: Tweedie
## Link function: log
##
## Formula:
## biomass_density ~ loc.cat + s(ndate, bs = "gp", by = loc.cat,
##   k = 10) + s(nmonth, bs = "cc", k = 4) + s(salinity.psu, temp.deg.c,
##   by = loc.cat, bs = "gp", k = 10) + s(site_longitude, site_latitude,
##   k = 20)
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.8490      0.3555  -5.201 2.99e-07 ***
## loc.catouter  -3.4375      0.5542  -6.203 1.23e-09 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##
##          edf Ref.df      F p-value
## s(ndate):loc.catfjord      1.000      1 4.222 0.04047 *
## s(ndate):loc.catouter      1.000      1 1.752 0.18630
## s(nmonth)                   0.814      2 1.219 0.06736 .
## s(salinity.psu,temp.deg.c):loc.catfjord 2.000      2 2.492 0.08382 .
## s(salinity.psu,temp.deg.c):loc.catouter 2.000      2 5.305 0.00528 **
## s(site_longitude,site_latitude)      2.000      2 0.147 0.86339
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.344
##   Scale est. = 1.1685    n = 472
```

```
### plotting with gratia - interfaces between mgcv and ggplot
plot.tsf<-gratia::draw(gam4$gam, select="s(salinity.psu,temp.deg.c):loc.catfjord",
  partial_match=TRUE, residuals=T, dist=0.05)+xlab("Salinity (psu)") +
  ylab(expression("Temperature " ( degree~C))) +
  theme(text = element_text(size=18)) +
  scale_fill_continuous(type='viridis', limits=c(-2.5, 2.5)) +
  labs(title="s(salinity, temperature)", subtitle = "Fjord"); plot.tsf
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```



5 and Table 3-6.pdf

Basis: GP

```
plot.tso<-gratia::draw(gam4$gam, select="s(salinity.psu,temp.deg.c):loc.catouter", partial_match=TRUE,
  residuals=T, dist=0.05)+xlab("Salinity (psu)") +
```

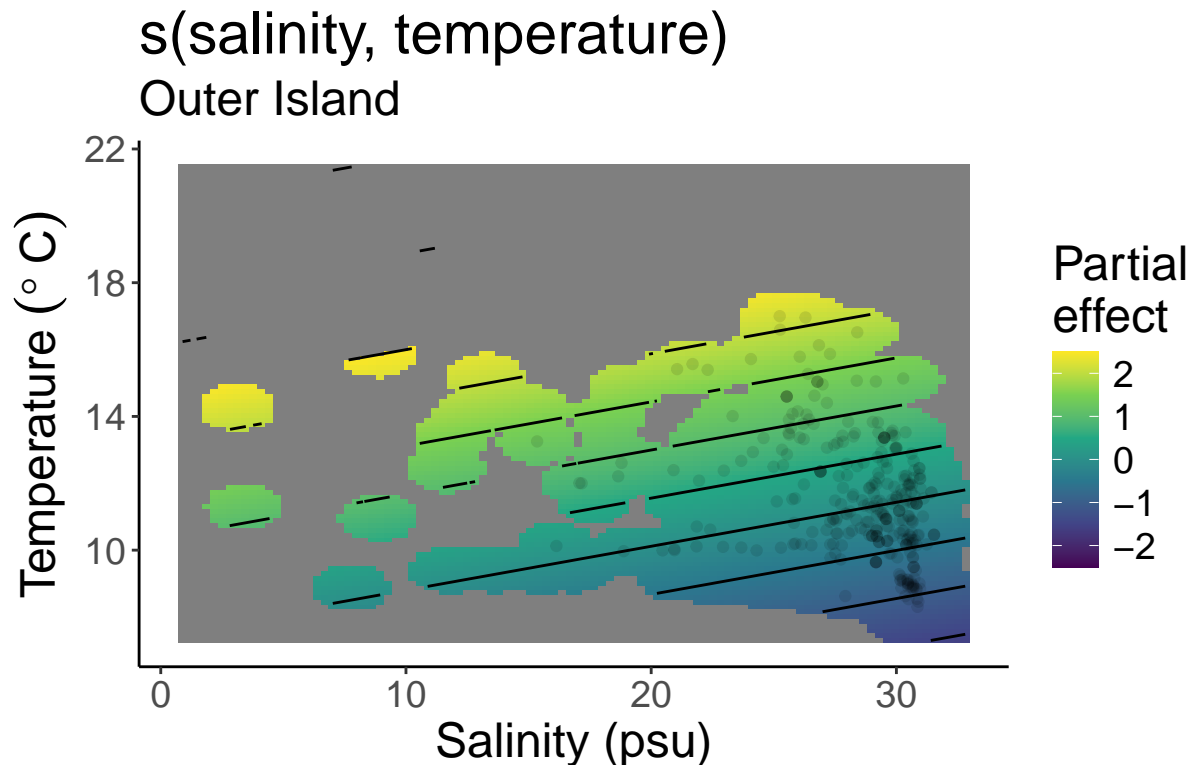
```

ylab(expression("Temperature " ( degree~C))) +
scale_fill_continuous(type='viridis', limits=c(-2.5, 2.5)) +
theme(text = element_text(size=18)) +
labs(title="s(salinity, temperature)", subtitle="Outer Island"); plot.tso

```

Scale for fill is already present.

Adding another scale for fill, which will replace the existing scale.



5 and Table 3-7.pdf

Basis: GP

```

#plot.de<-gratia::draw(gam4, select="s(unit_depth_corrected)", residuals=T)
plot.ll<-gratia::draw(gam4, select="s(site_longitude,site_latitude)", residuals=T, dist=0.05) +
  theme(text = element_text(size=18)) +
  scale_fill_continuous(type='viridis', limits=c(-.3, .3)) +
  labs(title="s(latitude, longitude)")

```

Scale for fill is already present.

Adding another scale for fill, which will replace the existing scale.

```

### Overlay map
#setwd so map can load
setwd("~/Documents/GitHub/nearshore-ProcB_2024/data/GIS data/BC SHAPEFILE/NTS_BC_COASTLINE_POLYS_125M")
bc.sf <- '250_CST_PY_polygon.shp'

bc <- st_read(bc.sf) %>%
  subset(QCST_TAG != 'OCEAN')

```

Reading layer `250_CST_PY_polygon' from data source

`~/Users/gehman/Documents/GitHub/nearshore-ProcB_2024/data/GIS data/BC SHAPEFILE/NTS_BC_COASTLINE_P

using driver `ESRI Shapefile'

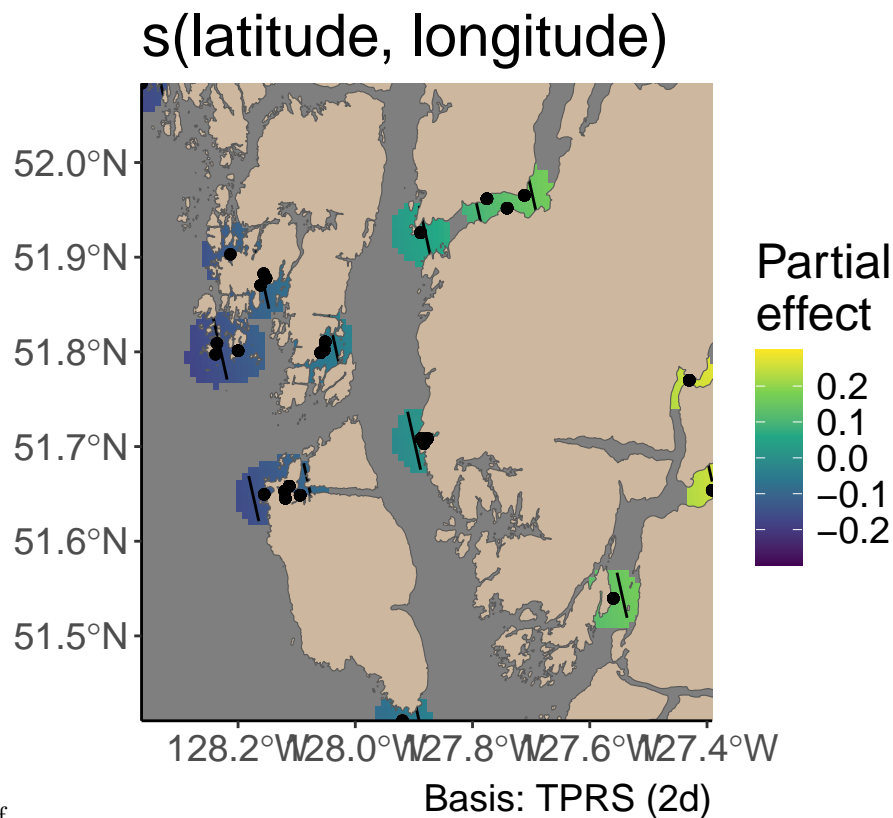
Simple feature collection with 874 features and 7 fields

```
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:   xmin: -128.7027 ymin: 51.17302 xmax: -127.0218 ymax: 52.28518
## Geodetic CRS:   NAD83

plot.ll.land <- plot.ll +
  theme(panel.grid = element_blank(),
        axis.title = element_blank()) +
  geom_sf(data = bc, fill = 'bisque3', inherit.aes = FALSE) +
  geom_point(data = ssbio.pyc.o2, aes(x = site_longitude,
                                     y = site_latitude)) +
  coord_sf(xlim = c(-127.3913, -128.3646), ylim = c(51.41033, 52.08358), expand = FALSE)

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

plot.ll.land
```



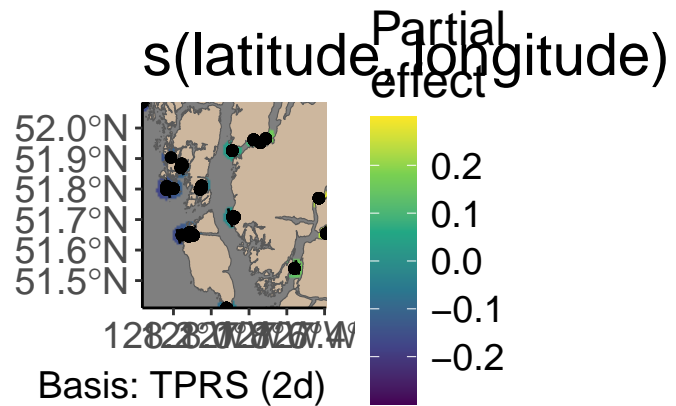
5 and Table 3-8.pdf

```
#reset wd back to main files
setwd("~/Documents/GitHub/nearshore-ProcB_2024")

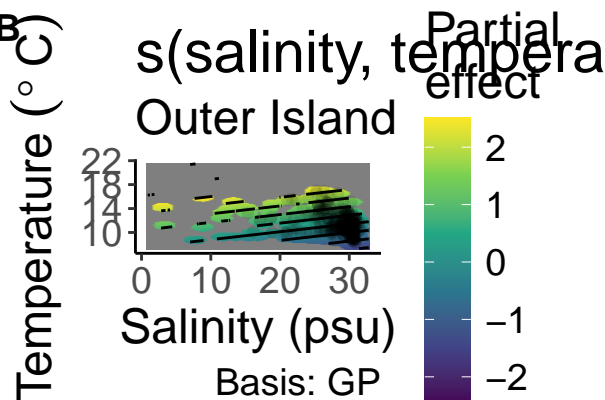
### salinity by temp

#Figure 5
## biomass density by temp/sal
#png("drivers_biomassdensity_28Feb25.png", width=20, height=20, units='cm', res=600)
#pdf("drivers_biomassdensity_28Feb24.pdf", height=12, width=12)
bottomrow<-cowplot::plot_grid(plot.tso, plot.tsf, labels=c("B", "C"), label_size=14)
cowplot::plot_grid(plot.ll.land, bottomrow, labels=c("A", " "), nrow=2, label_size=14)
```

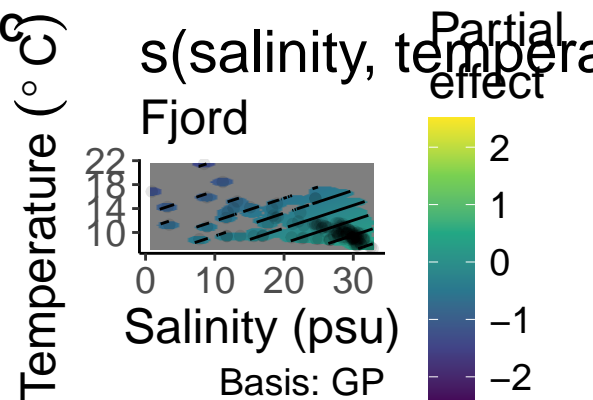
A



B



C

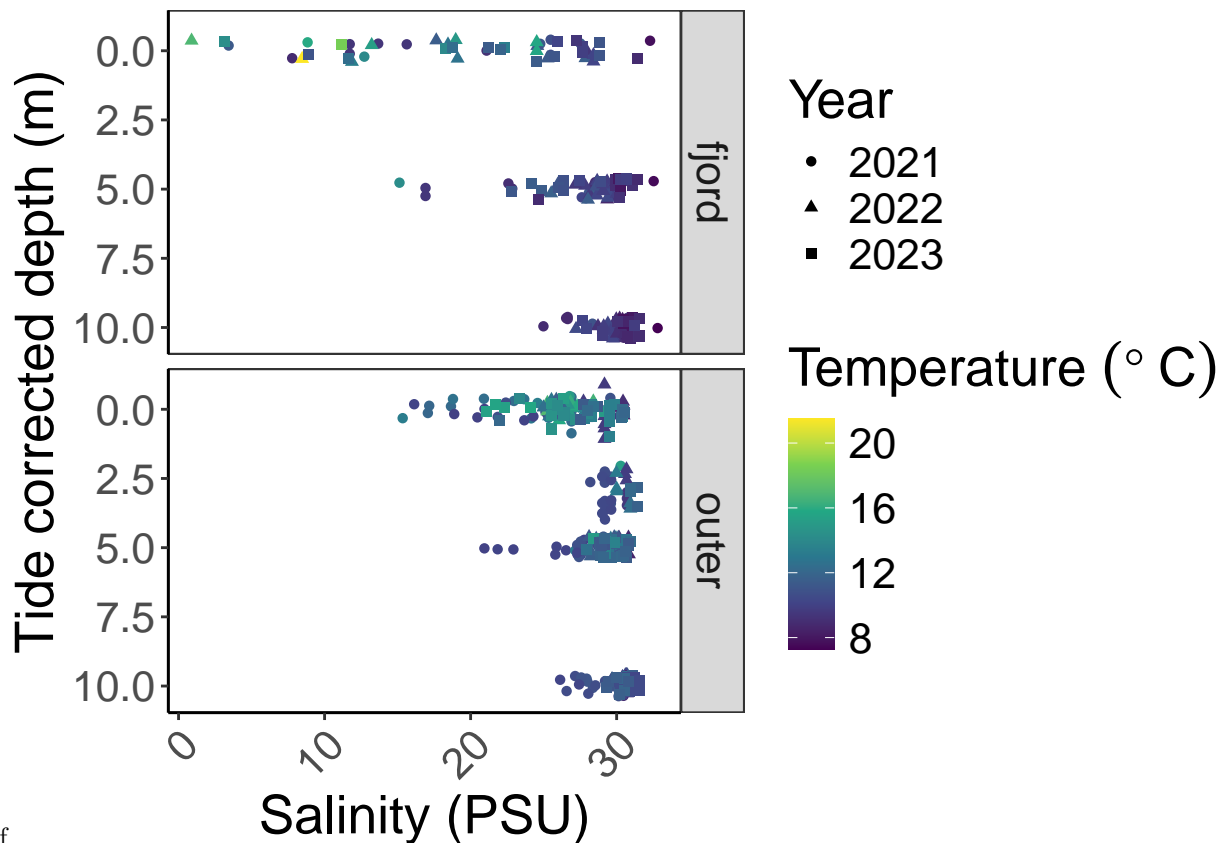


5 and Table 3-9.pdf

#dev.off()

Fig 6

```
p2o<- ssbio.pyc.o2 %>%
  ggplot(aes(salinity.psu, unit_depth_corrected, color=temp.deg.c, shape=as.factor(year(date))))+
  geom_jitter(width = 0.0001, height = 0.4)+
  scale_color_continuous(type='viridis')+
  facet_grid(rows=vars(loc.cat))+
  scale_y_continuous(trans="reverse")+
  xlab("Salinity (PSU)") + ylab("Tide corrected depth (m)") +
  labs(color=(expression("Temperature " ( degree-C))), shape="Year") +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        #panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=20),
        #legend.position = "none",
        axis.text.x = element_text(angle=45, hjust=1),
        axis.line = element_line(colour = "black"))
#ng("Fig6_28Feb25.png", width=18, height=18, units='cm', res=600)
p2o
```

6-1.pdf

```
#dev.off()
```

Fig S3

```
#set figure theme
theme_set(theme_bw() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=15),
        #legend.position = "none",
        axis.line = element_line(colour = "black")))

#### 2015 quantile regression #####

size15<-sswdag %>% dplyr::filter(year==2015 & latin_name=="Pycnopodia helianthoides")
size15$numeric_date<-as.numeric(size15$date)

quant10<-rq(radius~date, tau=0.10, data=size15)
quant<-rq(radius~date, tau=c(0.90), data=size15)

summary(quant, se="ker")

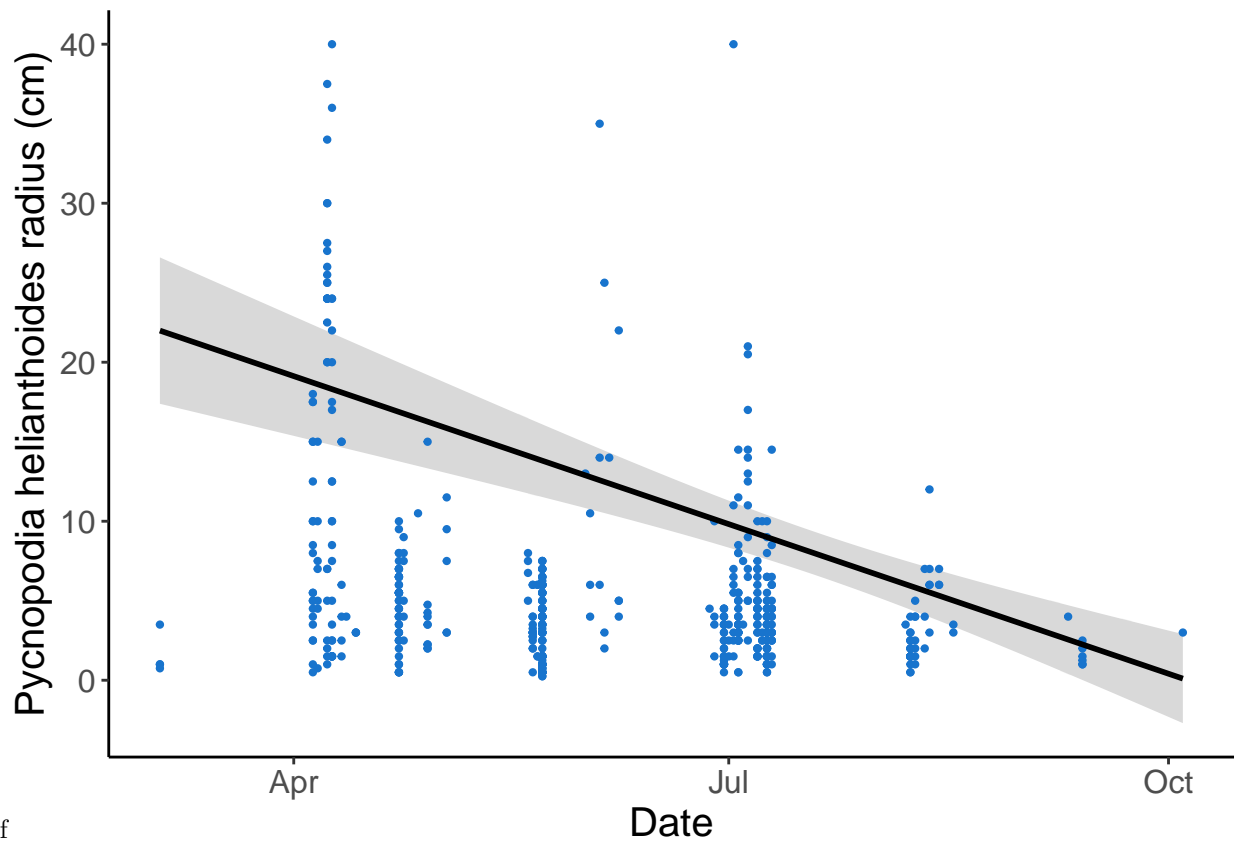
##
## Call: rq(formula = radius ~ date, tau = c(0.9), data = size15)
```

```
##
## tau: [1] 0.9
##
## Coefficients:
##           Value      Std. Error t value    Pr(>|t|)
## (Intercept) 1709.28409    246.80496     6.92565    0.00000
## date        -0.10227     0.01482    -6.89869    0.00000
summary(quant10, se="ker")

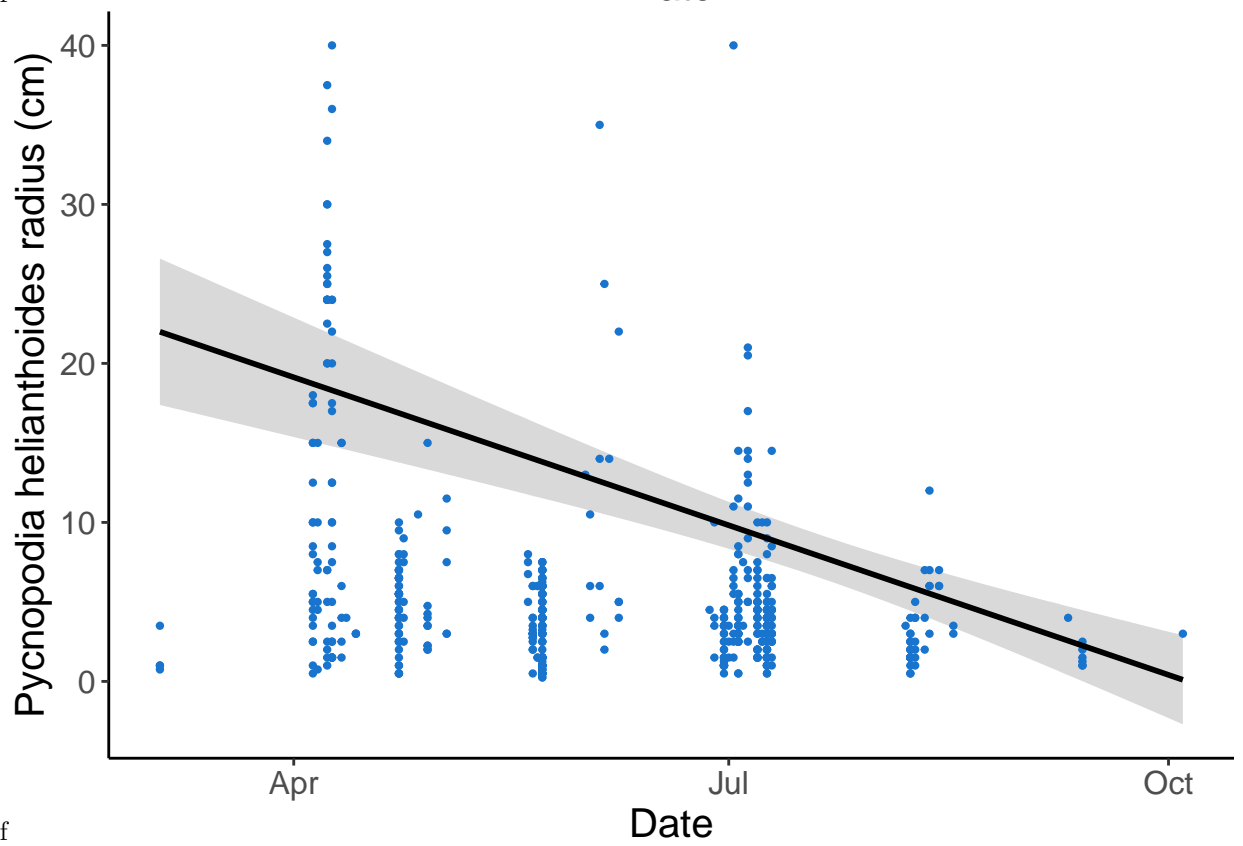
##
## Call: rq(formula = radius ~ date, tau = 0.1, data = size15)
##
## tau: [1] 0.1
##
## Coefficients:
##           Value      Std. Error t value    Pr(>|t|)
## (Intercept)  1.00000    52.07018     0.01920    0.98468
## date         0.00000     0.00314     0.00000    1.00000
### ggplot directly has quant_reg, but no error bars

gv<-visreg(quant, "date", ylab="Pycnopodia helianthoides radius (cm)", xlab="Date", bty="n",
  points=list(col="dodgerblue3"), line=list(col="black"), gg=T)

#png("FigS3_28Feb25.png", width=12, height=10, units='cm', res=600)
gv+theme_bw()+
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank(),
        text = element_text(size=15),
        #legend.position = "none",
        axis.line = element_line(colour = "black")); gv
```



S3-1.pdf



S3-2.pdf

```
#dev.off()
```

tables of data for publication (supplement Table S2 and S3)

```
## Table 2
```

```
size_data<-sswdag %>% dplyr::filter(project!="CCIRA") %>%  
  dplyr::group_by(project, survey, site_id, site_latitude, site_longitude) %>%  
  summarize(min_year=min(year), max_year=max(year))
```

```
## `summarise()` has grouped output by 'project', 'survey', 'site_id',  
## 'site_latitude'. You can override using the `.groups` argument.
```

```
## Table 3 pycnopia biomass density data used
```

```
bd_data<-pyc_aligned %>% dplyr::group_by(site_id, site_latitude, site_longitude) %>%  
  summarize(min_year=year(min(date)), max_year=year(max(date)), n())
```

```
## `summarise()` has grouped output by 'site_id', 'site_latitude'. You can  
## override using the `.groups` argument.
```

```
#write.csv(bd_data, "Fig7_Table3_dataset_26Oct23.csv")
```