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 Sadlier-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.

```r
library(rprojroot)
```

```
## Warning: package 'rprojroot' was built under R version 4.4.3
```

```r
knitr::opts_knit$set(root.dir = rprojroot::find_rstudio_root_file())
```

load and clean data

```r
knitr::opts_chunk$set(echo = TRUE)
library(tidyr)
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
```

```r
library(readr)
library(ggplot2)
library(purrr)
```

```
## Warning: package ’purrr’ was built under R version 4.4.3
```

```r
library(grid) #to plot multiple graphs in output
library(PNWColors)
library(stringr)
library(purrr)
library(lubridate)
```

```
## Warning: package ’lubridate’ was built under R version 4.4.3
```

```
##
## Attaching package: ’lubridate’
```

```
## The following objects are masked from ’package:base’:
##
##     date, intersect, setdiff, union
```

```r
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")’.
```

```r
#library(tidymv)
library(cowplot)
```

```
##
## Attaching package: ’cowplot’
```

```
## The following object is masked from ’package:lubridate’:
##
##     stamp
```

```r
library(stringr)
library(ggdist) #for fancy plots with distribtuions
```

```
## Warning: package ’ggdist’ was built under R version 4.4.2
```

```r
library(visreg)
library(quantreg)
```

```
## Warning: package ’quantreg’ was built under R version 4.4.3
```

```
## Loading required package: SparseM
```

```r
library(gratia)
```

```
## Warning: package ’gratia’ was built under R version 4.4.2
```

```
##
## Attaching package: ’gratia’
```

```
## The following object is masked from ’package:stringr’:
##
##     boundary
```

```r
library(sjPlot)
```

```
## Warning: package ’sjPlot’ was built under R version 4.4.3
```

```
## #refugeeswelcome
```

```
##
## Attaching package: ’sjPlot’
```

```
## The following objects are masked from ’package:cowplot’:
##
##     plot_grid, save_plot
```

```r
library(vegan)
```

```
## Warning: package ’vegan’ was built under R version 4.4.3
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```r
#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‘
```

```r
## 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.
```

```r
##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
## dttm  (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

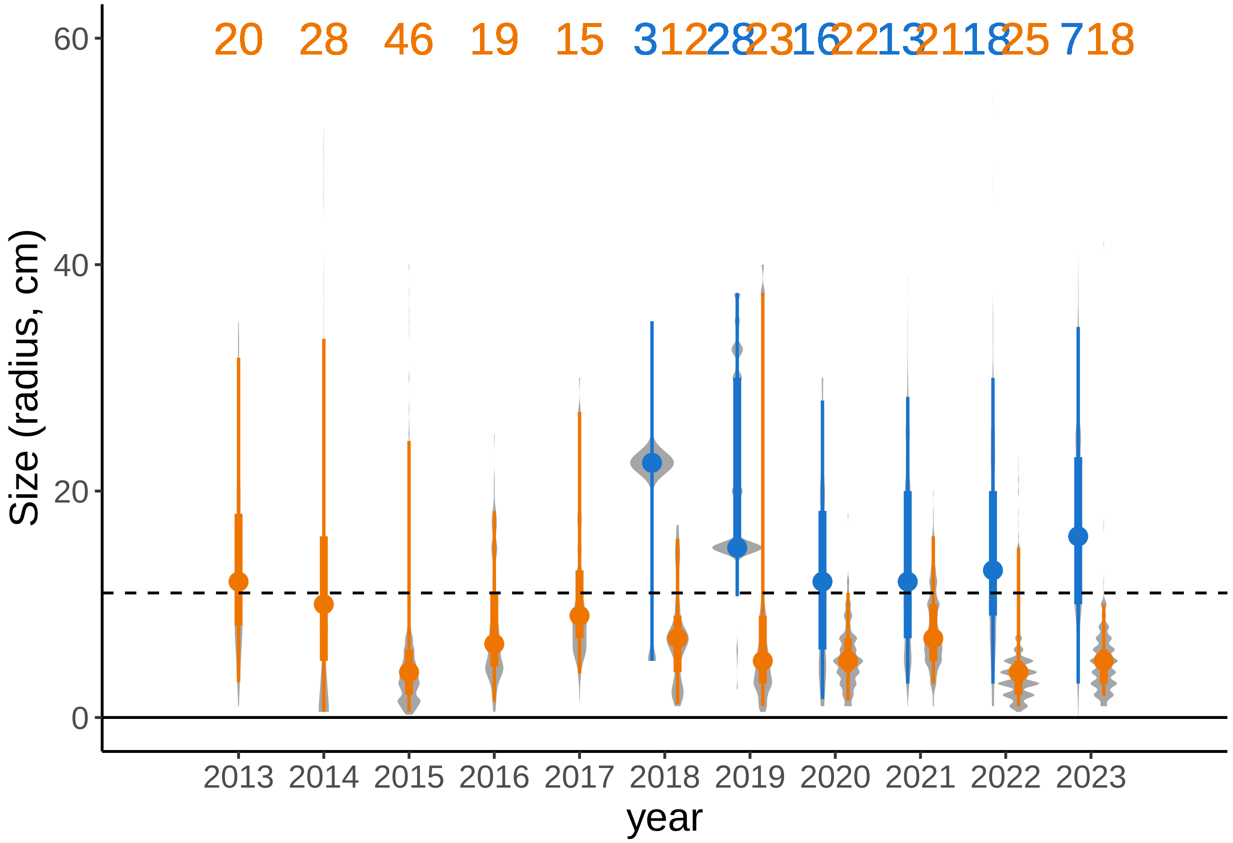
```r
################################################################################
############################ Fjord size distributions #####################
################################################################################
```

```
### 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" | si

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.
```



```
#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,
```

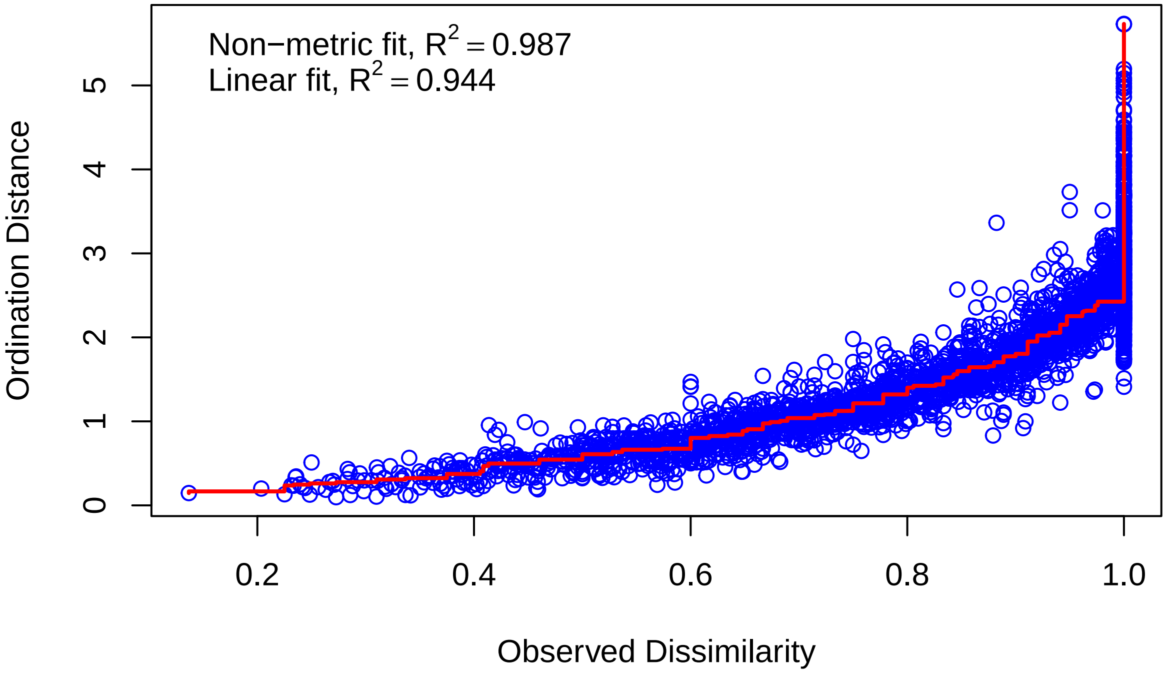
```r
          trymax = 500,
          wascores = TRUE)
```

```
## Run 0 stress 0.1134931
## Run 1 stress 0.1155886
## Run 2 stress 0.1176199
## Run 3 stress 0.1149533
## Run 4 stress 0.1149517
## Run 5 stress 0.1149521
## Run 6 stress 0.1134756
## ... New best solution
## ... Procrustes: rmse 0.00923299  max resid 0.06693638
## Run 7 stress 0.1216668
## Run 8 stress 0.1133879
## ... New best solution
## ... Procrustes: rmse 0.034197  max resid 0.2643028
## Run 9 stress 0.11374
## ... Procrustes: rmse 0.0403829  max resid 0.2650268
## Run 10 stress 0.1130561
## ... New best solution
## ... Procrustes: rmse 0.0756261  max resid 0.2906937
## Run 11 stress 0.1137404
## Run 12 stress 0.1174695
## Run 13 stress 0.1149529
## Run 14 stress 0.1141913
## Run 15 stress 0.1133524
## ... Procrustes: rmse 0.02315796  max resid 0.1149862
## Run 16 stress 0.1217577
## Run 17 stress 0.1174865
## Run 18 stress 0.1142365
## Run 19 stress 0.1135568
## Run 20 stress 0.1135868
## Run 21 stress 0.113622
## Run 22 stress 0.1148597
## Run 23 stress 0.1187637
## Run 24 stress 0.1149522
## Run 25 stress 0.1153351
## Run 26 stress 0.1141706
## Run 27 stress 0.1136546
## Run 28 stress 0.113738
## Run 29 stress 0.1137698
## Run 30 stress 0.1159226
## Run 31 stress 0.1135536
## ... Procrustes: rmse 0.07029276  max resid 0.291058
## Run 32 stress 0.1133826
## ... Procrustes: rmse 0.02980356  max resid 0.1423005
## Run 33 stress 0.1133907
## ... Procrustes: rmse 0.07386637  max resid 0.2873192
## Run 34 stress 0.1141637
## Run 35 stress 0.1134149
## ... Procrustes: rmse 0.0712307  max resid 0.2840234
## Run 36 stress 0.1144354
## Run 37 stress 0.1160159
## Run 38 stress 0.1148595
## Run 39 stress 0.1134231
## ... Procrustes: rmse 0.07066135  max resid 0.2824639
## Run 40 stress 0.113581
## Run 41 stress 0.1134262
## ... Procrustes: rmse 0.04669447  max resid 0.2489773
## Run 42 stress 0.1162237
## Run 43 stress 0.1141658
## Run 44 stress 0.1136326
## Run 45 stress 0.11339
## ... Procrustes: rmse 0.07485629  max resid 0.2873572
## Run 46 stress 0.1133535
## ... Procrustes: rmse 0.03615473  max resid 0.281394
## Run 47 stress 0.1137247
## Run 48 stress 0.1138727
## Run 49 stress 0.1138979
## Run 50 stress 0.1205339
## Run 51 stress 0.1146012
## Run 52 stress 0.1134219
## ... Procrustes: rmse 0.07081793  max resid 0.2823018
## Run 53 stress 0.1134247
## ... Procrustes: rmse 0.07132446  max resid 0.2814983
## Run 54 stress 0.1191807
## Run 55 stress 0.1138967
## Run 56 stress 0.1164006
## Run 57 stress 0.1141192
## Run 58 stress 0.1138675
## Run 59 stress 0.1197577
## Run 60 stress 0.1137427
## Run 61 stress 0.1148604
## Run 62 stress 0.1134758
## ... Procrustes: rmse 0.0643928  max resid 0.221043
## Run 63 stress 0.1133516
## ... Procrustes: rmse 0.03625818  max resid 0.2830701
## Run 64 stress 0.1143079
## Run 65 stress 0.1159888
## Run 66 stress 0.1134753
## ... Procrustes: rmse 0.06445014  max resid 0.2208684
## Run 67 stress 0.1136257
## Run 68 stress 0.1139005
## Run 69 stress 0.1141683
## Run 70 stress 0.1139961
## Run 71 stress 0.1176943
## Run 72 stress 0.1134764
## ... Procrustes: rmse 0.06439258  max resid 0.2209435
## Run 73 stress 0.1164586
## Run 74 stress 0.1135886
## Run 75 stress 0.1172455
## Run 76 stress 0.1159226
## Run 77 stress 0.1134224
## ... Procrustes: rmse 0.07178008  max resid 0.2818612
## Run 78 stress 0.1138678
## Run 79 stress 0.115335
## Run 80 stress 0.1172557
## Run 81 stress 0.1134192
## ... Procrustes: rmse 0.07199526  max resid 0.2823638
## Run 82 stress 0.1139403
## Run 83 stress 0.1174855
## Run 84 stress 0.1137442
## Run 85 stress 0.1136789
## Run 86 stress 0.1134109
## ... Procrustes: rmse 0.0730511  max resid 0.2842214
## Run 87 stress 0.1143779
## Run 88 stress 0.1137384
## Run 89 stress 0.1137336
## Run 90 stress 0.1136256
## Run 91 stress 0.1141907
## Run 92 stress 0.1171396
## Run 93 stress 0.1141917
## Run 94 stress 0.1136395
## Run 95 stress 0.114311
## Run 96 stress 0.1143775
## Run 97 stress 0.1140181
## Run 98 stress 0.1135579
## Run 99 stress 0.1133902
## ... Procrustes: rmse 0.07634623  max resid 0.2871689
## Run 100 stress 0.113476
## ... Procrustes: rmse 0.06440406  max resid 0.2215473
## Run 101 stress 0.1150339
## Run 102 stress 0.1134765
## ... Procrustes: rmse 0.06439925  max resid 0.2207882
## Run 103 stress 0.1145639
## Run 104 stress 0.1144597
## Run 105 stress 0.113392
## ... Procrustes: rmse 0.07579366  max resid 0.2873168
## Run 106 stress 0.1189829
## Run 107 stress 0.113389
## ... Procrustes: rmse 0.07490337  max resid 0.2872918
## Run 108 stress 0.1162207
## Run 109 stress 0.1137405
## Run 110 stress 0.1144603
## Run 111 stress 0.1138537
## Run 112 stress 0.1134764
## ... Procrustes: rmse 0.06461075  max resid 0.220218
## Run 113 stress 0.1145638
## Run 114 stress 0.1140132
## Run 115 stress 0.1135577
## Run 116 stress 0.1136614
## Run 117 stress 0.1142203
## Run 118 stress 0.1137387
## Run 119 stress 0.1155907
## Run 120 stress 0.1143152
## Run 121 stress 0.1144164
## Run 122 stress 0.1134761
## ... Procrustes: rmse 0.06457647  max resid 0.2204966
## Run 123 stress 0.117615
## Run 124 stress 0.1216499
## Run 125 stress 0.1133892
## ... Procrustes: rmse 0.0762343  max resid 0.2870287
## Run 126 stress 0.1189647
## Run 127 stress 0.1155889
## Run 128 stress 0.11486
## Run 129 stress 0.1154932
## Run 130 stress 0.1138362
## Run 131 stress 0.1164053
## Run 132 stress 0.1143458
## Run 133 stress 0.1144271
## Run 134 stress 0.114859
## Run 135 stress 0.1193837
## Run 136 stress 0.1133928
## ... Procrustes: rmse 0.07475728  max resid 0.2872267
## Run 137 stress 0.117686
## Run 138 stress 0.1133903
## ... Procrustes: rmse 0.07525834  max resid 0.2875017
## Run 139 stress 0.1134122
## ... Procrustes: rmse 0.07132191  max resid 0.2835103
## Run 140 stress 0.1144584
## Run 141 stress 0.1163145
## Run 142 stress 0.1134765
## ... Procrustes: rmse 0.06458381  max resid 0.2211195
## Run 143 stress 0.113576
## Run 144 stress 0.1133873
## ... Procrustes: rmse 0.0754978  max resid 0.2874306
## Run 145 stress 0.1134642
## ... Procrustes: rmse 0.04105936  max resid 0.1823042
## Run 146 stress 0.113475
## ... Procrustes: rmse 0.06439146  max resid 0.2206627
## Run 147 stress 0.1148592
## Run 148 stress 0.11559
## Run 149 stress 0.113351
## ... Procrustes: rmse 0.03619357  max resid 0.2825125
## Run 150 stress 0.1142698
## Run 151 stress 0.1141909
## Run 152 stress 0.1137607
## Run 153 stress 0.1130568
## ... Procrustes: rmse 0.0005531809  max resid 0.003520844
## ... Similar to previous best
## *** Best solution repeated 1 times
```

```
goodness(size_NMS)
```

```
##  [1] 0.019283825 0.015034623 0.015042157 0.010940770 0.010716815 0.015670831
##  [7] 0.011215349 0.017075790 0.018757969 0.020651060 0.017593016 0.017392972
## [13] 0.014972512 0.021896033 0.018129695 0.012426373 0.018278593 0.014801145
## [19] 0.020386679 0.011371003 0.008941744 0.010387743 0.014320882 0.016856858
## [25] 0.012651506 0.011298639 0.017843566 0.012561650 0.011011798 0.008410413
## [31] 0.012524433 0.009512165 0.009495433 0.011281031 0.013390151 0.014811840
## [37] 0.016394305 0.007793445 0.012979316 0.009695153 0.018536454 0.013487508
## [43] 0.017567425 0.007344906 0.010870281 0.010814665 0.014065355 0.011346380
## [49] 0.014414483 0.008622763 0.009149286 0.013255856 0.016461024 0.013579725
## [55] 0.009427961 0.006841688 0.010325094 0.010396028 0.010006486 0.007036236
## [61] 0.011705857 0.007388586 0.013939891 0.007379406 0.009399531 0.009398936
## [67] 0.006106390 0.009052074 0.010329482 0.012538184 0.008413592 0.011116066
## [73] 0.008073697 0.009483883
```

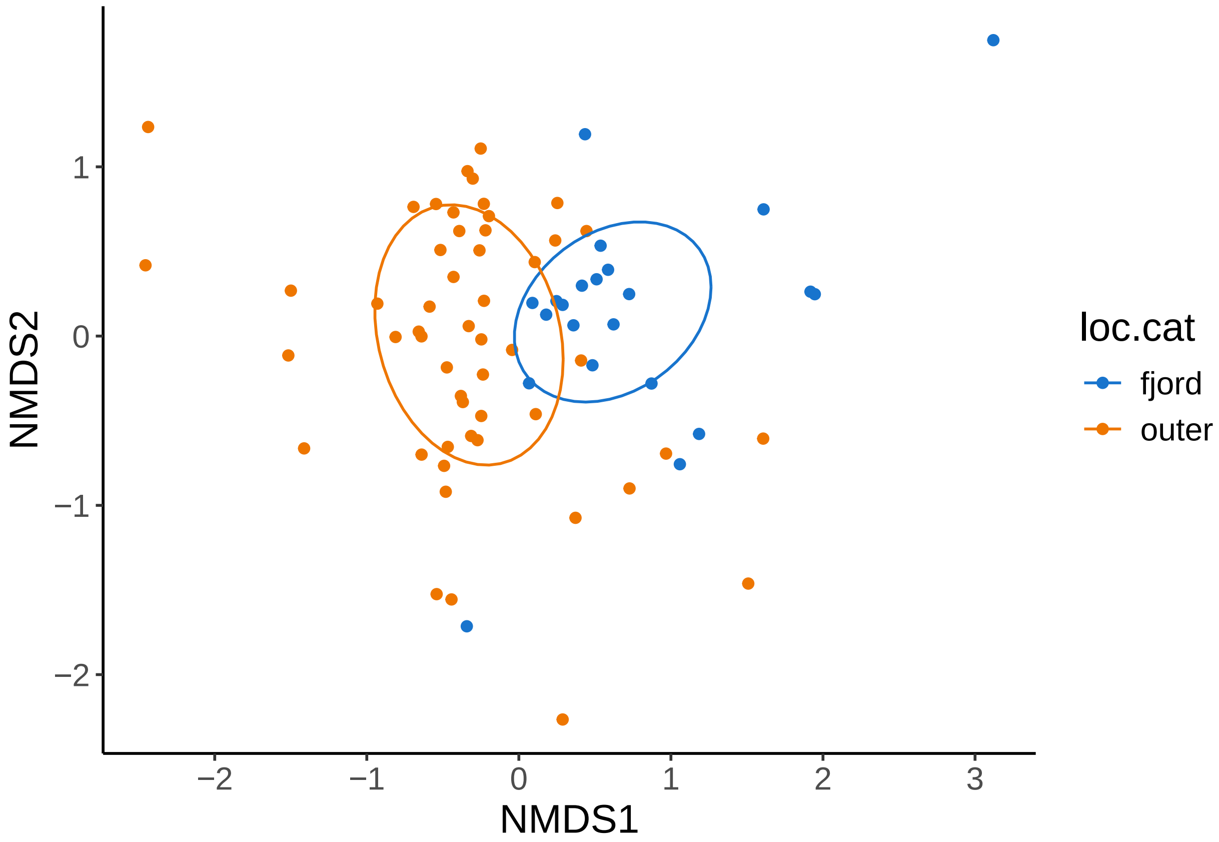
```
stressplot(size_NMS)
```



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

## output the nmds 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"))
```

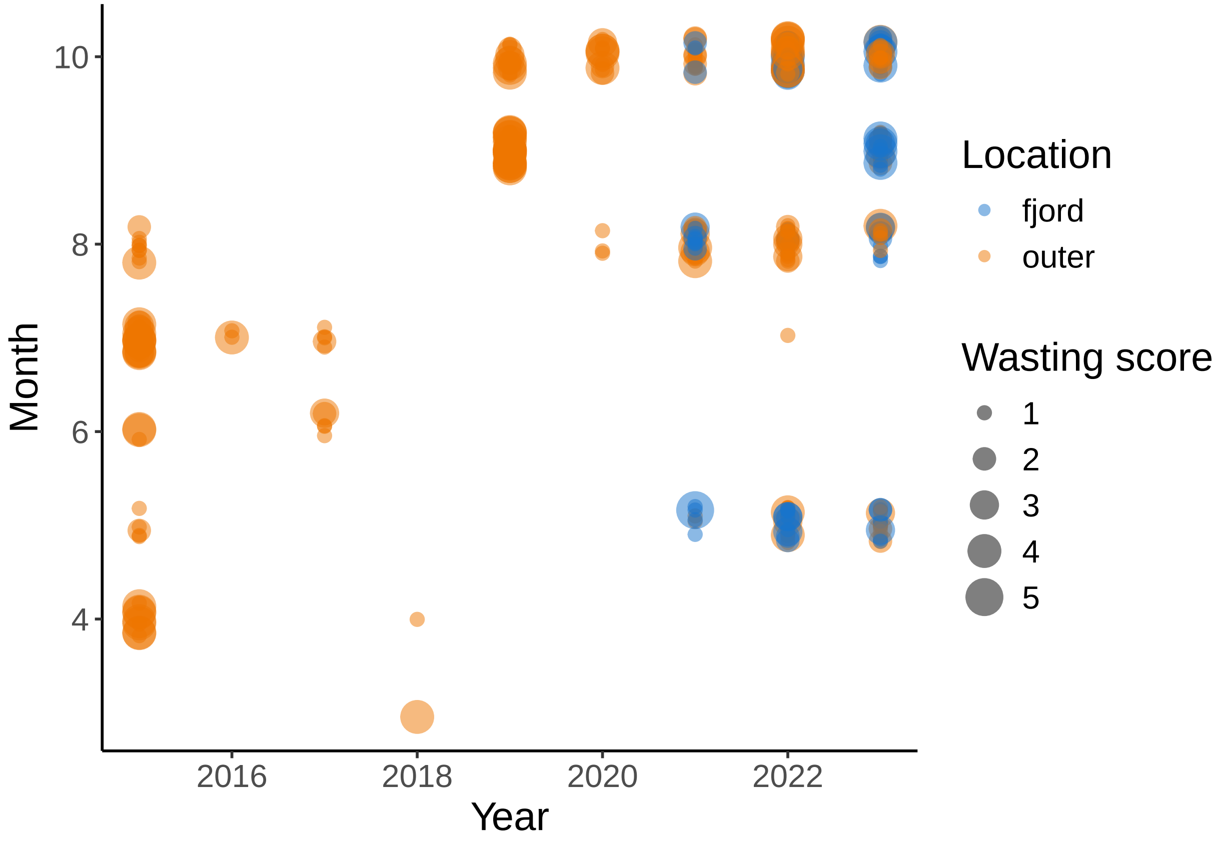


```
#dev.off()
```

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 sumarize 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" | s
  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, a
  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.
```



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

Fig. 4 and Table 2

```r
library(itsadug)
```

```
## Warning: package 'itsadug' was built under R version 4.4.2
```

```
## Loading required package: plotfunctions
```

```
## Warning: package 'plotfunctions' was built under R version 4.4.2
```

```
##
## 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 object is masked from 'package:gratia':
##
##     dispersion
```

```
library(ggeffects)
```

```
## Warning: package 'ggeffects' was built under R version 4.4.3

##
## 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)
```

```
## Warning: package 'MapGAM' was built under R version 4.4.2

## Loading required package: sp

## Warning: package 'sp' was built under R version 4.4.3

## Loading required package: gam

## Warning: package 'gam' was built under R version 4.4.2

## Loading required package: splines

## Loading required package: foreach

## Warning: package 'foreach' was built under R version 4.4.2

##
## 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)
```

```
## Warning: package 'sf' was built under R version 4.4.2
```

```
## Linking to GEOS 3.12.2, GDAL 3.9.3, PROJ 9.4.1; 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
```

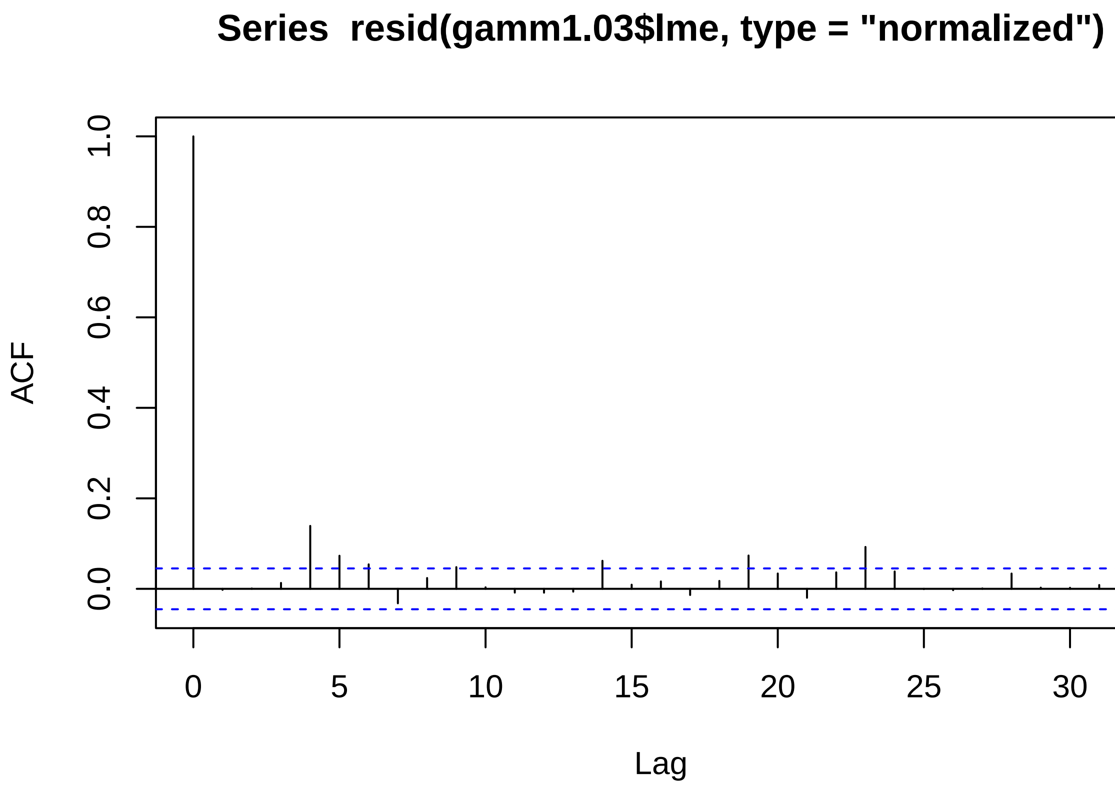
```
## iteration 9
```

```
## iteration 10
```

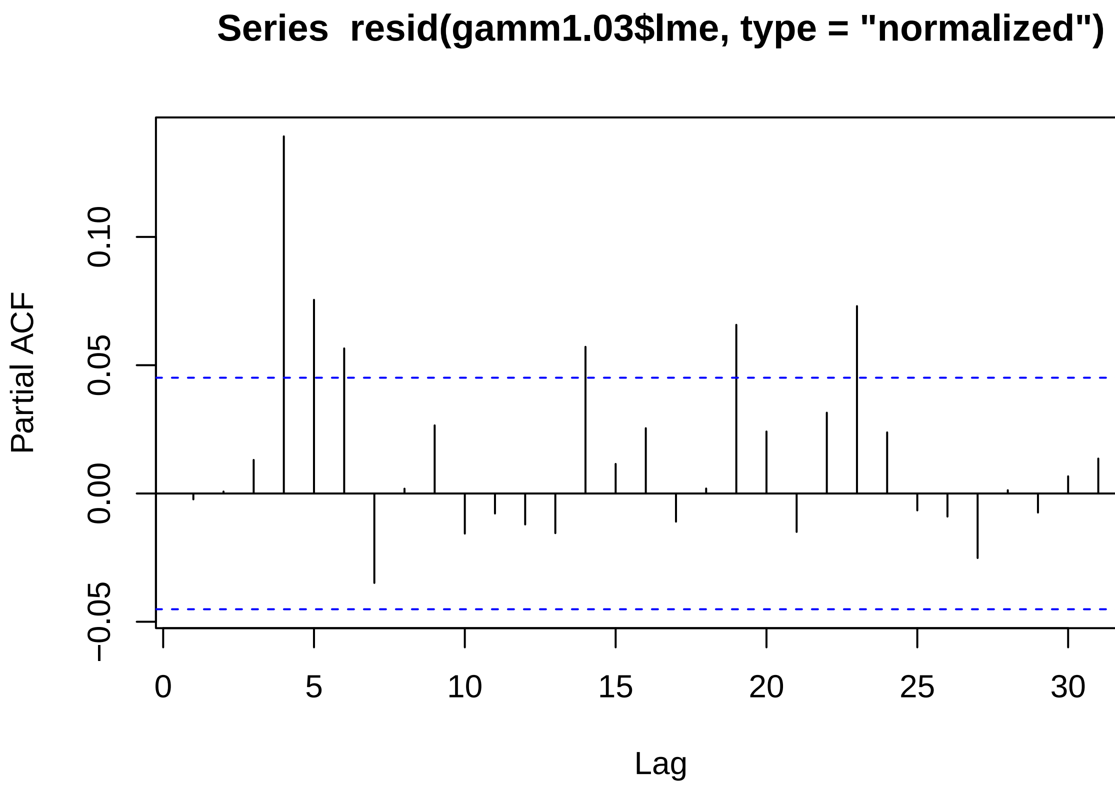
```
#### AIC test
AIC(gamm1.03$lme) #9448.61
```

```
## [1] 9448.61
```

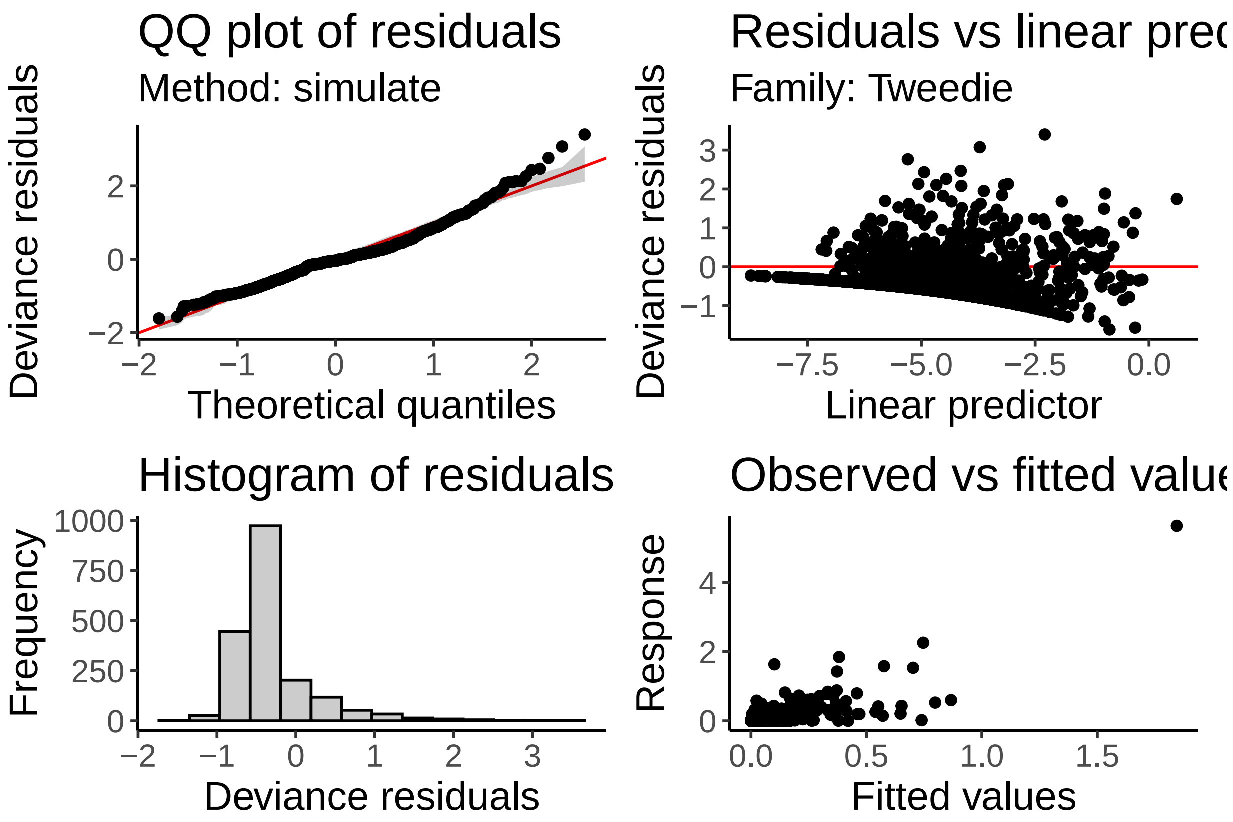
```
### model evaluation
  acf(resid(gamm1.03$lme, type = "normalized")) ## normalized residuals better to test
```



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



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



```
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

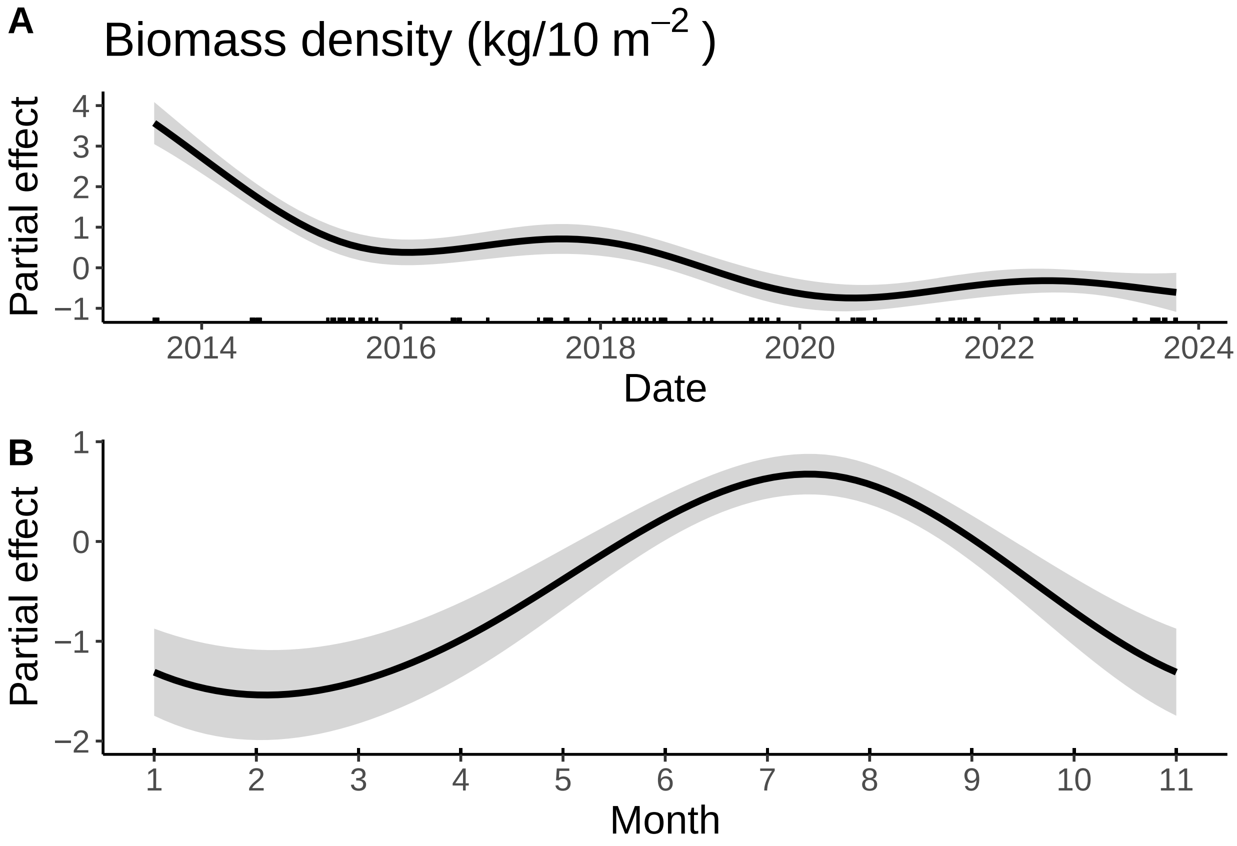
Observations

1887

R2

0.457

```r
### 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"))


### 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"))


#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"))
```



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

Fig 5 and Table 3

```r
#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
```

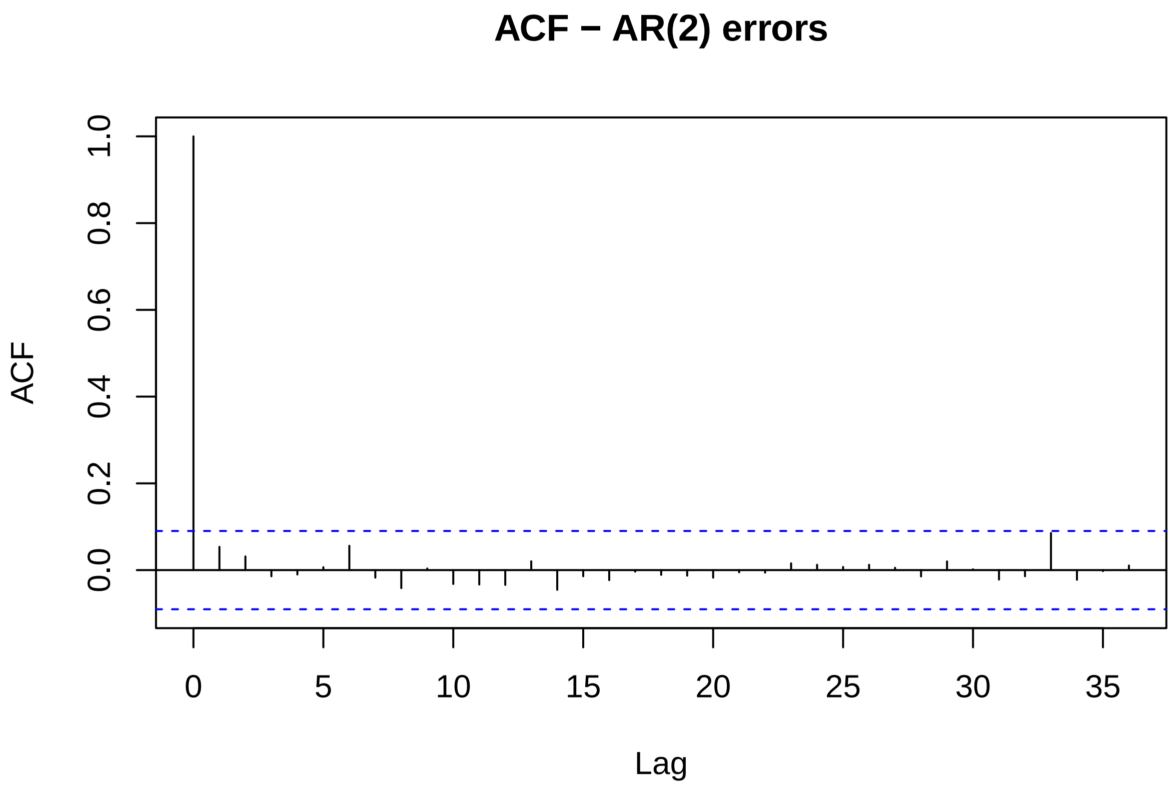
```
## iteration 4
```

```
## iteration 5
```

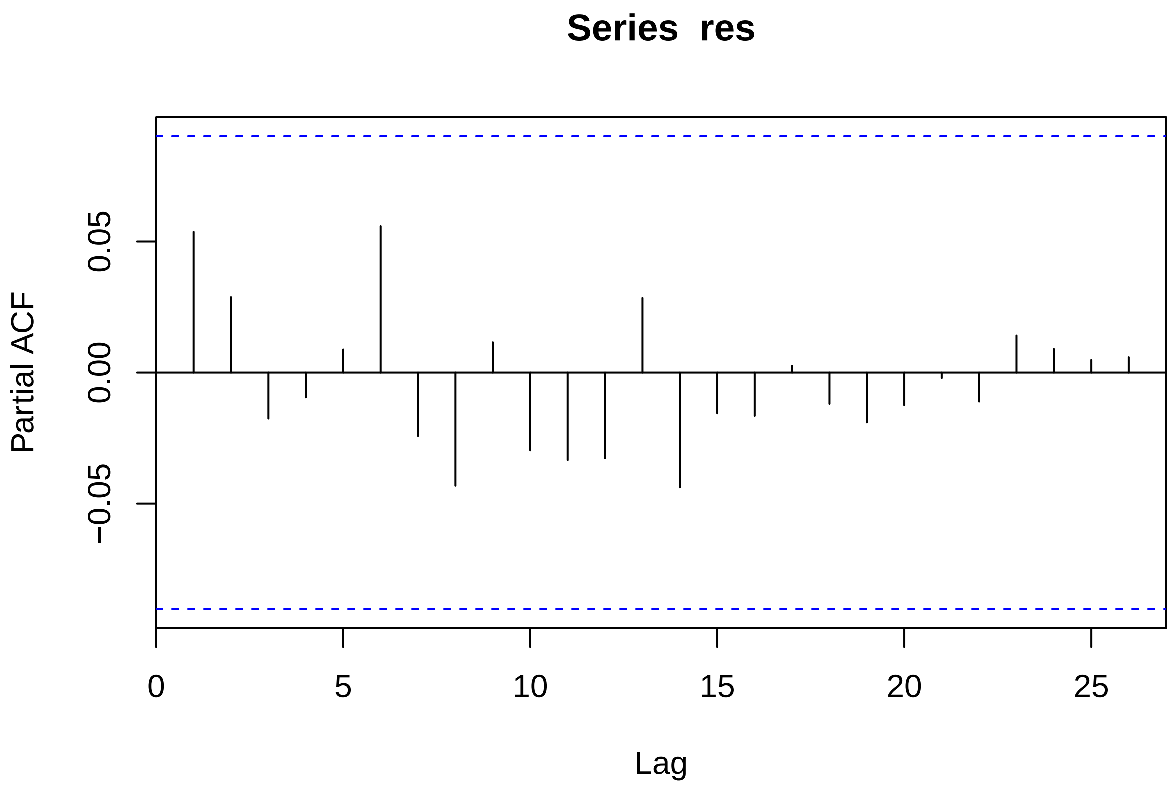
```
## iteration 6
```

```
## iteration 7
```

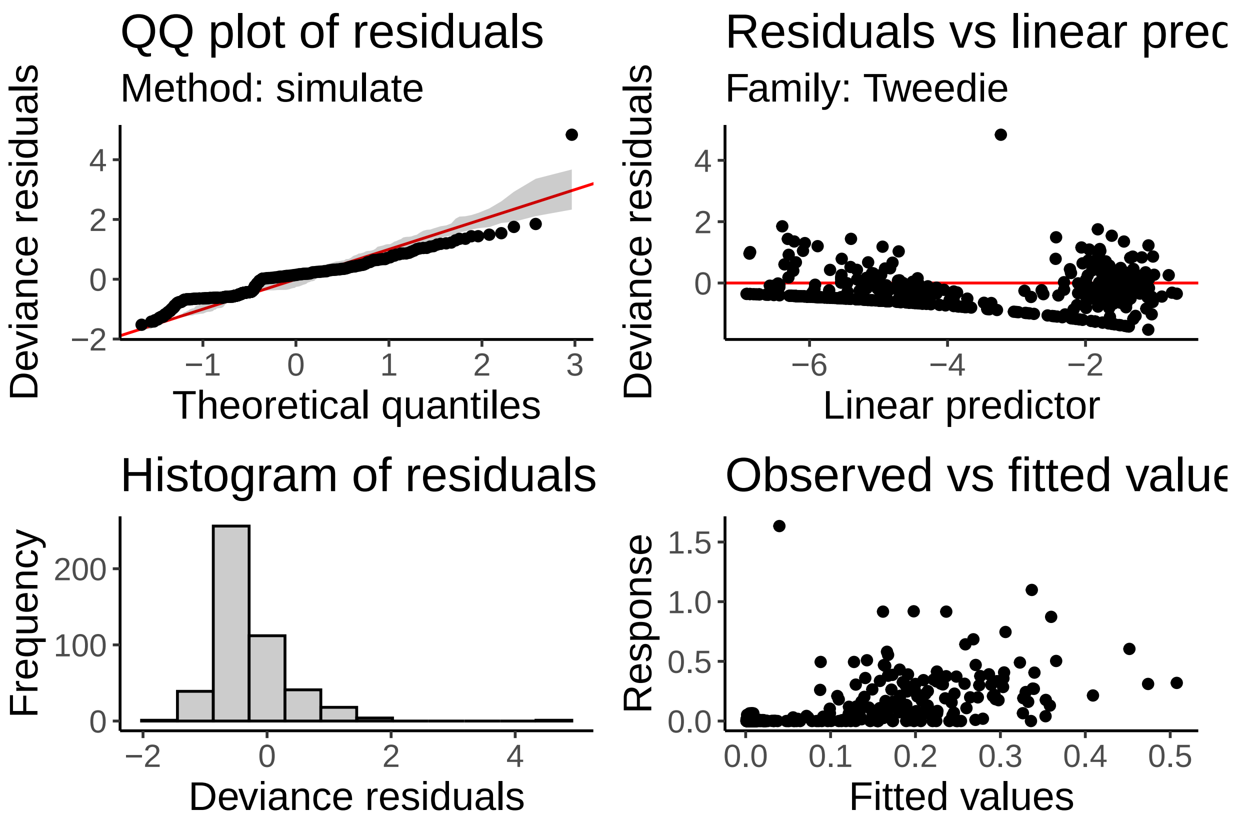
```r
#model evaluation
res <- resid(gam4$lme, type = "normalized")
acf(res, lag.max = 36, main = "ACF - AR(2) errors")
```



```r
pacf(res)
```



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



```
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)     -1.8491     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                  1.0000      1 4.222 0.04046 *
## s(ndate):loc.catouter                  1.0000      1 1.752 0.18627
## s(nmonth)                              0.8144      2 1.220 0.06737 .
## s(salinity.psu,temp.deg.c):loc.catfjord 2.0000      2 2.493 0.08381 .
## s(salinity.psu,temp.deg.c):loc.catouter 2.0000      2 5.303 0.00528 **
## s(site_longitude,site_latitude)         2.0000      2 0.147 0.86343
## ---
## 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")
```

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

```
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")
```

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

```
#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.
```

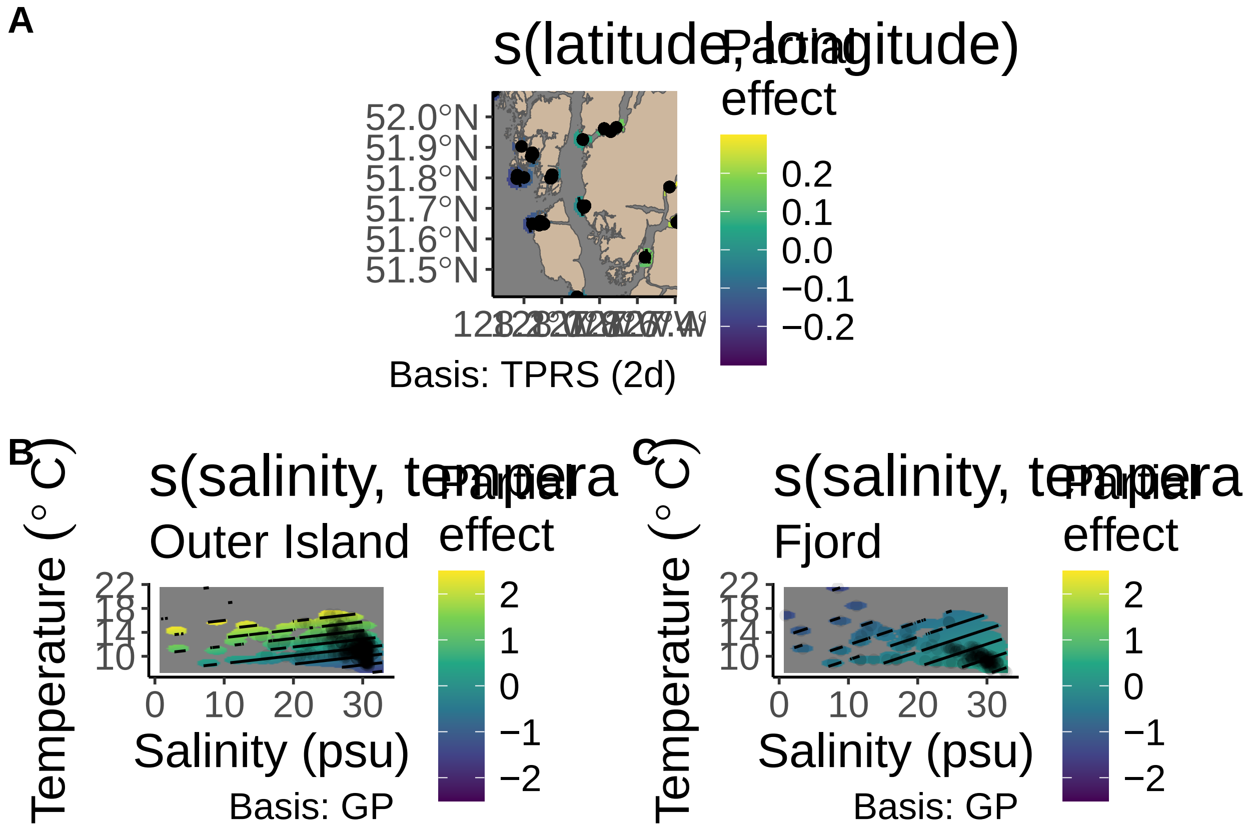
```
### Overlay map
#setwd so map can load
#setwd("~/Documents/GitHub/nearshore-ProcB_2024/data/GIS data/BC SHAPEFILE/NTS_BC_COASTLINE_POLYS_125M"
bc.sf <- './data/GIS data/BC SHAPEFILE/NTS_BC_COASTLINE_POLYS_125M/250_CST_PY_polygon.shp'

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

```
## Reading layer '250_CST_PY_polygon' from data source
##   'C:\Users\tyrel.froese\Documents\GitHub\Seastar Wasting\nearshore-ProcB_2024\data\GIS data\BC SHAP
##   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.
```

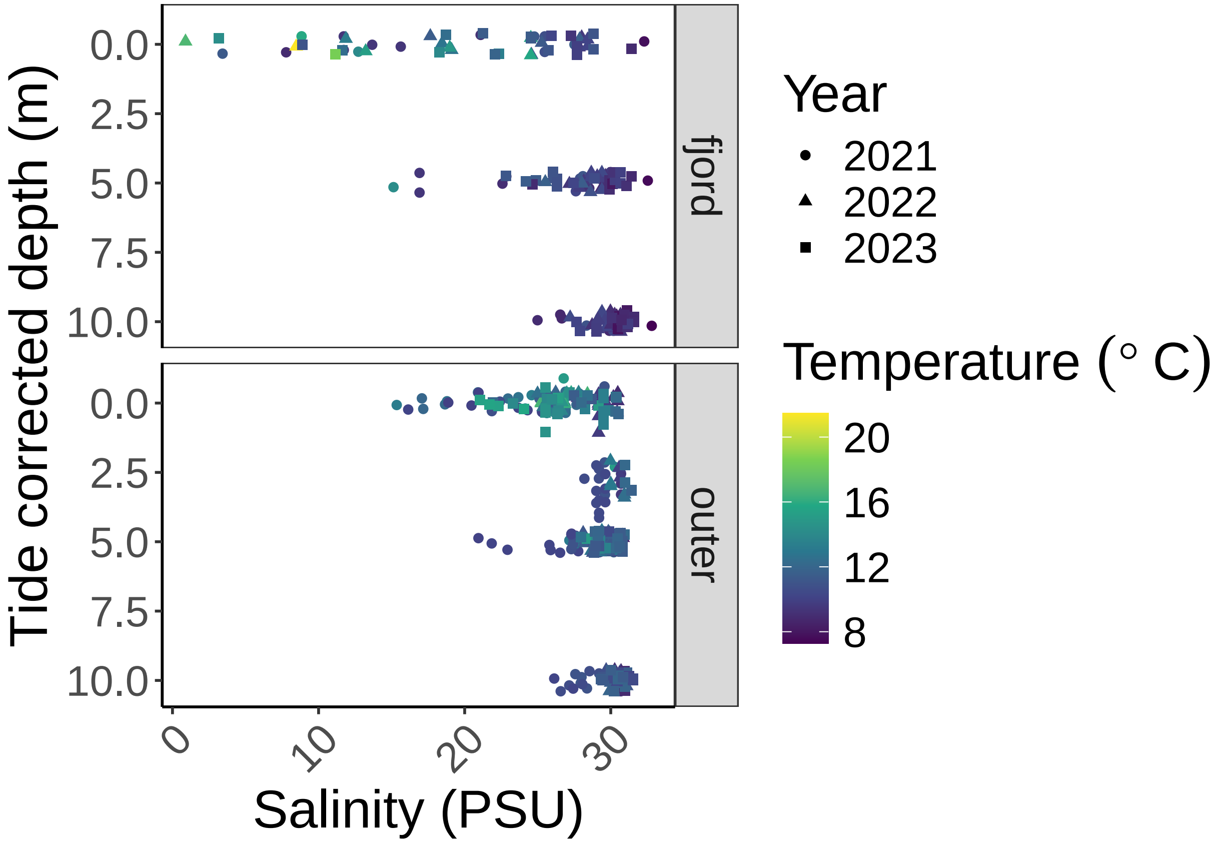
```
#reset wd back to main files
#setwd("~/Documents/GitHub/nearshore-ProcB_2024")

### 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)
```



```
#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
```



```
#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=0.90, data=size15)

summary(quant, se="ker")
```

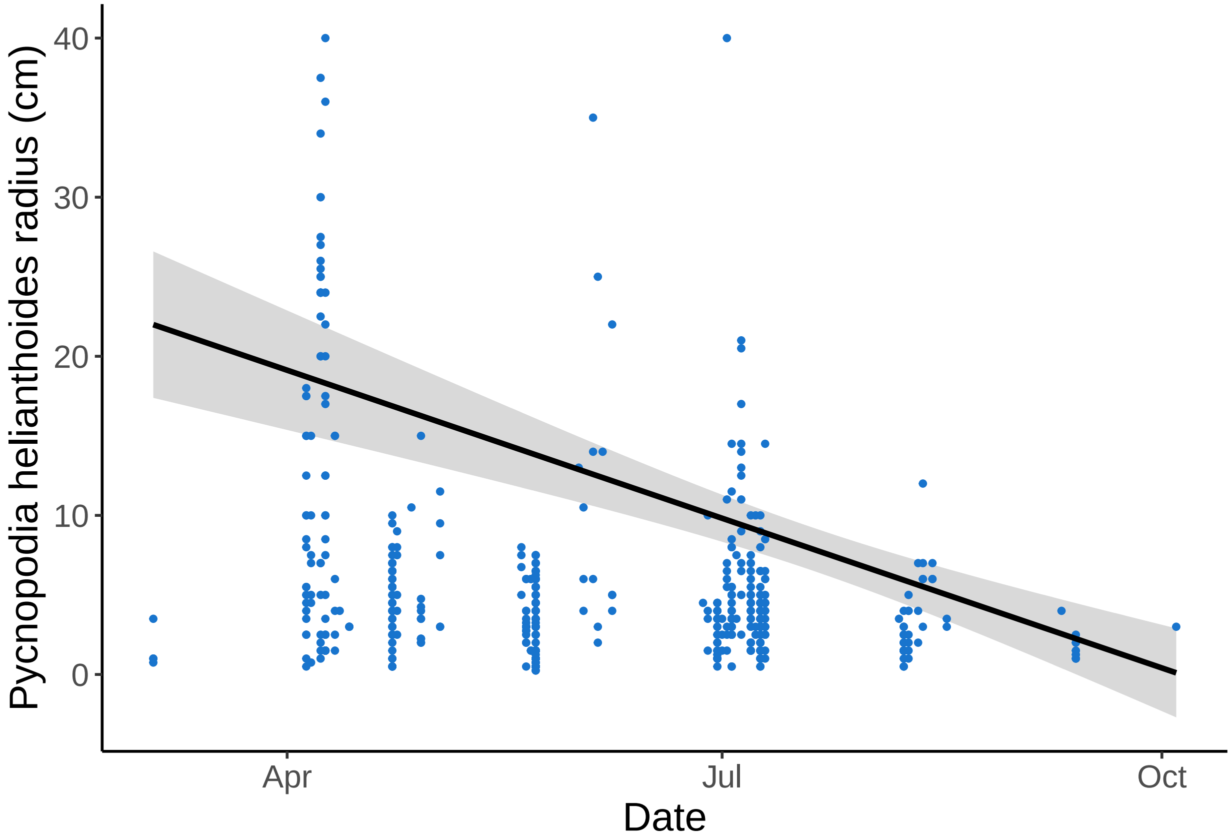
```
##
## Call: rq(formula = radius ~ date, tau = 0.9, data = size15)
##
## tau: [1] 0.9
##
## Coefficients:
##             Value      Std. Error t value    Pr(>|t|)
## (Intercept) 1709.28409  246.80496    6.92565    0.00000
## date          -0.10227    0.01482   -6.89869    0.00000
```

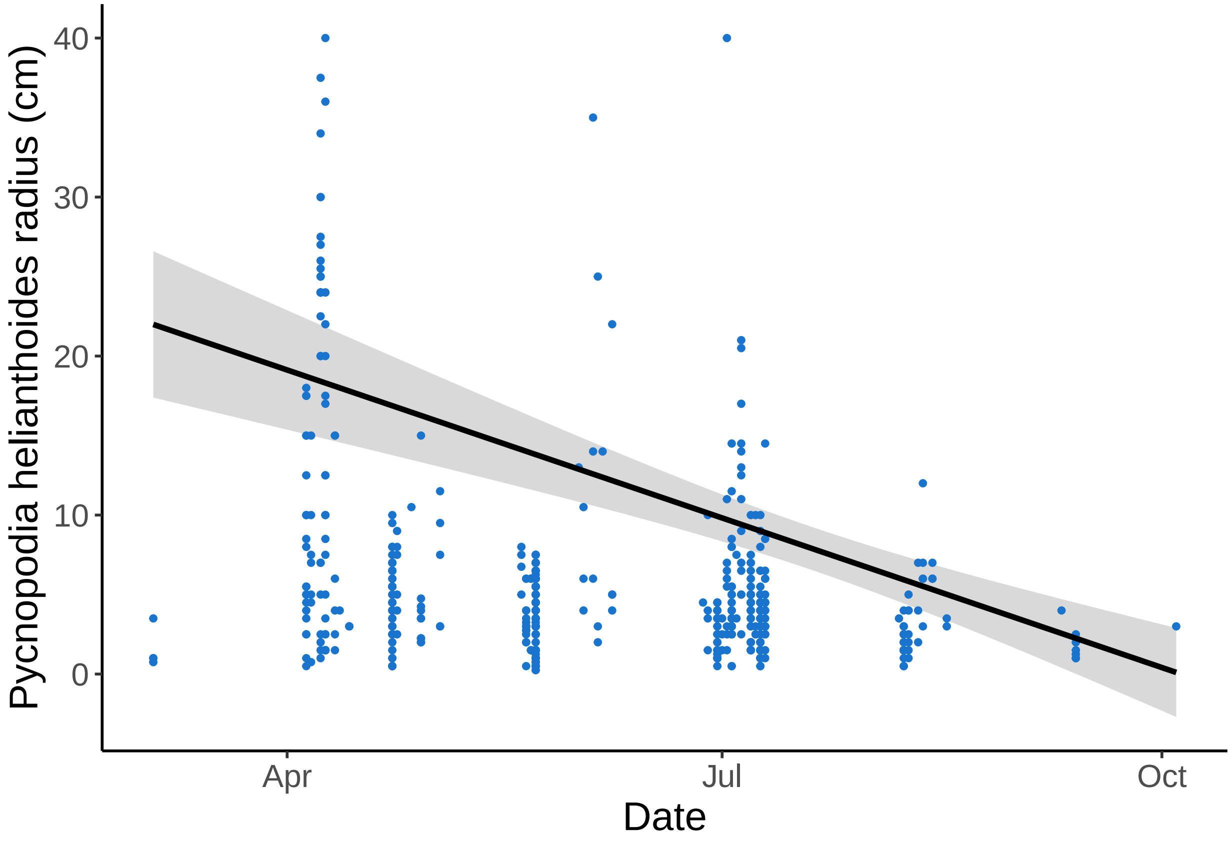
```
summary(quant10, se="ker")
```

```
##
## 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
```





```
#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 pycnopodia 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")
```