

Supplemental Tables and Figures for:

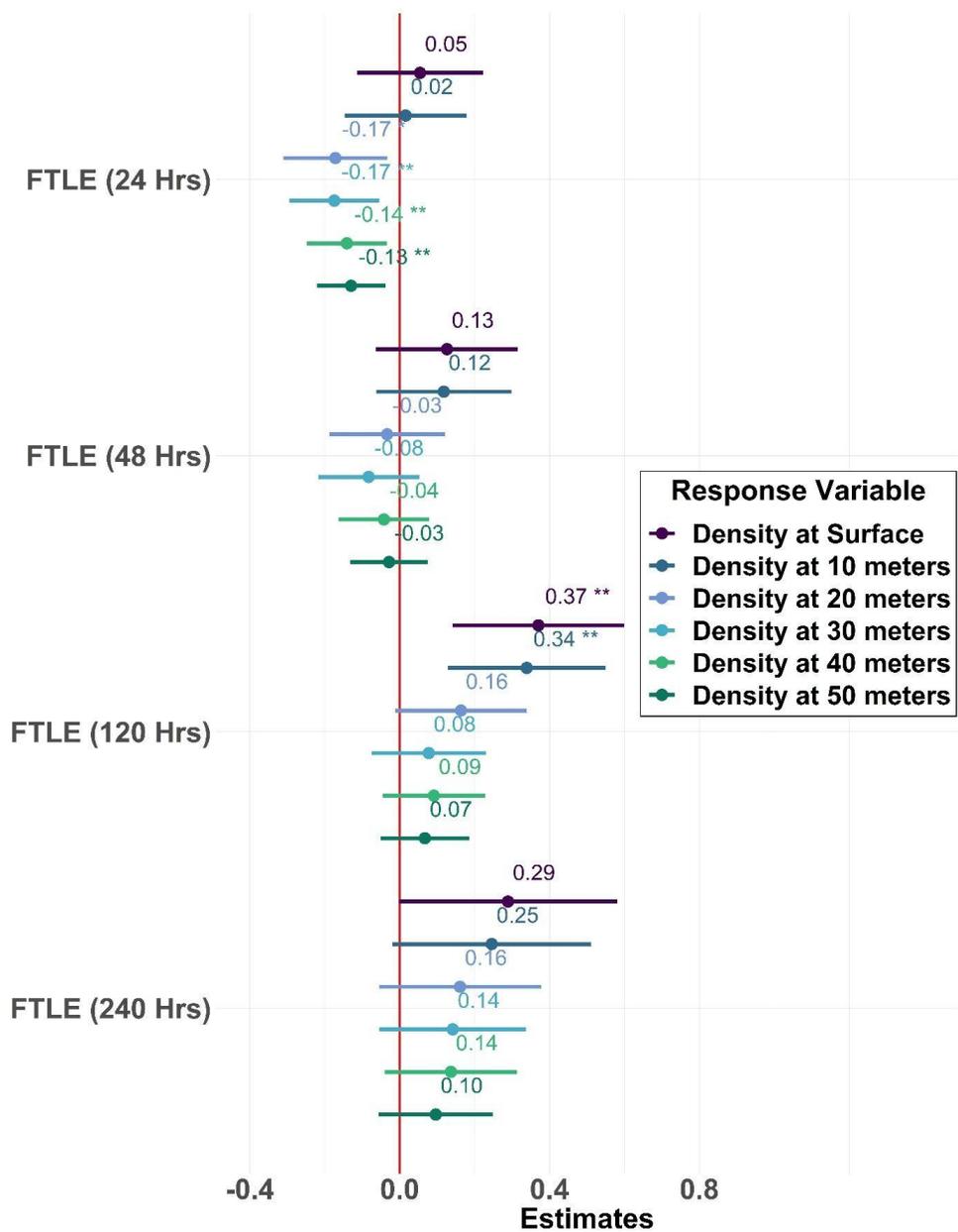
Submesoscale coupling of krill and whales revealed by aggregative Lagrangian coherent structures

James A. Fahlbusch, David E. Cade, Elliott L. Hazen, Meredith L. Elliott, Benjamin T. Saenz, Jeremy A. Goldbogen and Jaime Jahncke

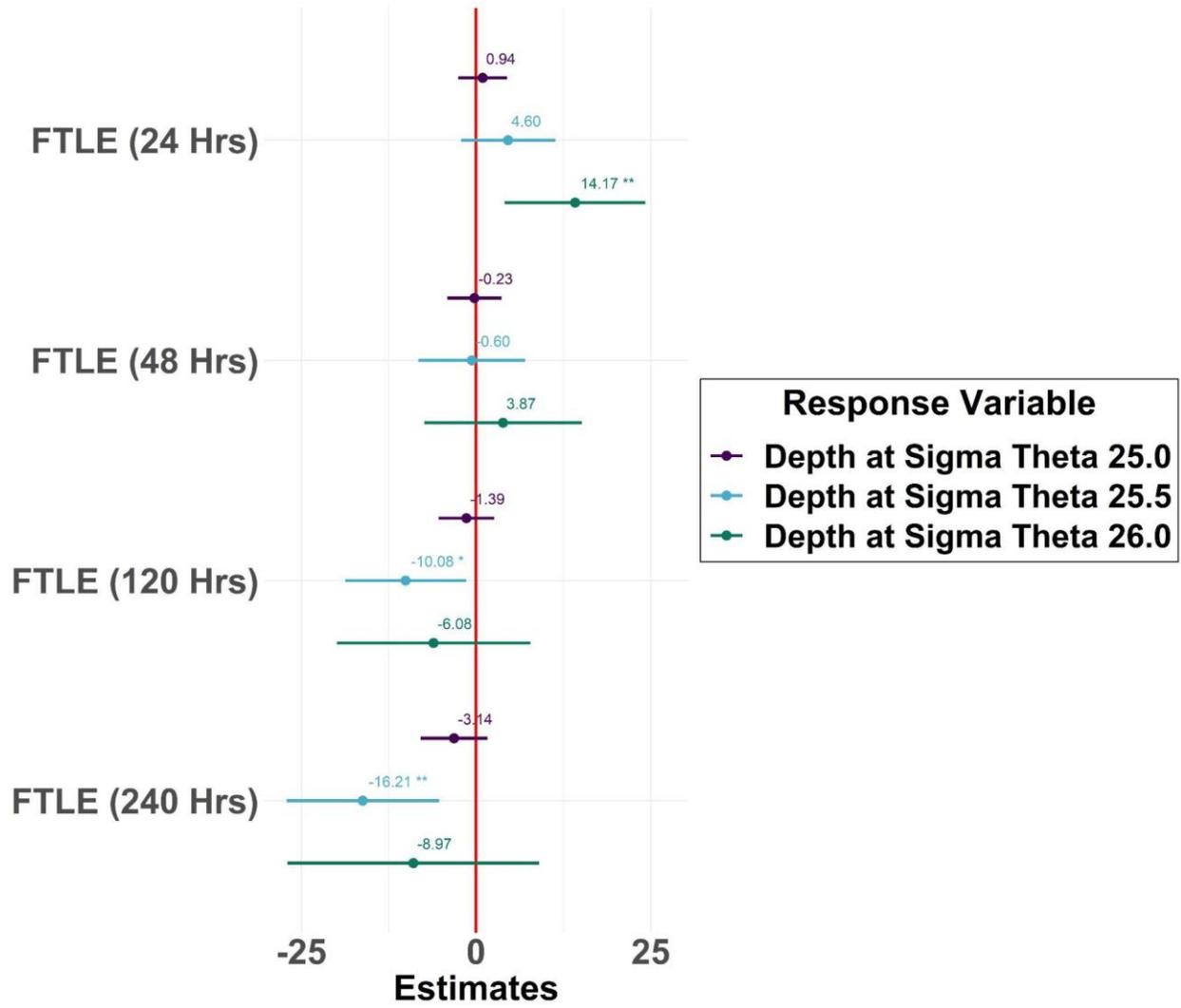
Proceedings of the Royal Society B, DOI: 10.1098/rspb.2023.2461

Cruise	8mm	9mm	10mm	11mm	12mm	13mm	14mm	15mm	16mm	17mm	18mm	19mm	20mm	21mm	22mm	23mm	24mm	25mm	26mm	27mm
ACC1206	0.01133	0.07082	0.10387	0.21058	0.06988	0.13975	0.10104	0.08687	0.02266	0.02172	0.02077	0.01794	0.03588	0.01322	0.04721	0.02266	0.00378	0.00000	0.00000	0.00000
ACC1207	0.26431	0.08298	0.13830	0.19593	0.06838	0.04956	0.04533	0.00461	0.02766	0.00999	0.00269	0.01652	0.02497	0.04764	0.01498	0.00615	0.00000	0.00000	0.00000	0.00000
ACC1209	0.25932	0.23335	0.19927	0.13571	0.05766	0.06817	0.00255	0.02803	0.00510	0.00510	0.00064	0.00127	0.00255	0.00000	0.00127	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1305	0.09960	0.09190	0.16716	0.12419	0.17983	0.14704	0.04198	0.02782	0.01689	0.02186	0.00497	0.01217	0.01739	0.01987	0.01018	0.01391	0.00124	0.00199	0.00000	0.00000
ACC1307	0.22934	0.18946	0.12821	0.05271	0.06553	0.03704	0.10541	0.01994	0.03134	0.04843	0.00570	0.04843	0.02279	0.00997	0.00000	0.00570	0.00000	0.00000	0.00000	0.00000
ACC1309	0.27785	0.19523	0.17013	0.05386	0.06671	0.02081	0.05508	0.05386	0.03978	0.04284	0.01040	0.00796	0.00551	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1406	0.38339	0.30990	0.18530	0.05751	0.03834	0.00639	0.00639	0.00000	0.00000	0.00000	0.00000	0.00000	0.01278	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1407	0.34628	0.09699	0.10328	0.25589	0.01416	0.08530	0.08919	0.00127	0.00509	0.00127	0.00127	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1409	0.45738	0.23878	0.22186	0.06897	0.00716	0.00000	0.00130	0.00455	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1506	0.31008	0.22932	0.07602	0.06663	0.07772	0.06589	0.05898	0.03975	0.03239	0.01605	0.00658	0.00976	0.00592	0.00148	0.00344	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1507	0.31298	0.22317	0.18609	0.11068	0.05896	0.05126	0.02772	0.00766	0.01497	0.00424	0.00000	0.00000	0.00000	0.00226	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1509	0.07924	0.08954	0.14987	0.12665	0.14173	0.14484	0.11492	0.08283	0.05123	0.01532	0.00000	0.00000	0.00383	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1605	0.06582	0.06351	0.08277	0.10276	0.08084	0.07484	0.07533	0.07262	0.06296	0.05170	0.06129	0.04241	0.04004	0.05113	0.03815	0.01797	0.01385	0.00201	0.00000	0.00000
ACC1607	0.04859	0.04303	0.08734	0.11068	0.09751	0.09425	0.09425	0.06406	0.05133	0.06829	0.04712	0.06001	0.05177	0.01873	0.01726	0.02852	0.01726	0.00000	0.00000	0.00000
ACC1609	0.05319	0.07345	0.09604	0.09797	0.10735	0.10975	0.10463	0.08333	0.06088	0.07125	0.05676	0.06277	0.01280	0.00225	0.00759	0.00000	0.00000	0.00000	0.00000	0.00000
ACC1705	0.04771	0.08227	0.11182	0.11315	0.13058	0.09099	0.06736	0.06071	0.03058	0.04225	0.05524	0.03545	0.05495	0.03013	0.02230	0.01211	0.01064	0.00177	0.00000	0.00000
ACC1707	0.29357	0.17950	0.17169	0.03272	0.05848	0.04835	0.05269	0.03185	0.02721	0.03243	0.02606	0.01448	0.01563	0.00753	0.00492	0.00058	0.00232	0.00000	0.00000	0.00000
ACC1709	0.12595	0.20263	0.32359	0.13022	0.07396	0.03221	0.00817	0.02042	0.00926	0.02931	0.00354	0.01670	0.01697	0.00290	0.00236	0.00054	0.00036	0.00054	0.00036	0.00000
ACC1805	0.00019	0.00011	0.03309	0.02128	0.05303	0.03791	0.04893	0.06145	0.08571	0.17142	0.07779	0.08499	0.06626	0.10443	0.08499	0.03962	0.01945	0.00936	0.00000	0.00000
ACC1807	0.18191	0.10654	0.07322	0.05597	0.04060	0.04412	0.04498	0.05839	0.07171	0.06626	0.05065	0.04398	0.04587	0.05211	0.03385	0.01848	0.00703	0.00297	0.00110	0.00025
ACC1809	0.11410	0.00055	0.08418	0.12927	0.11487	0.21833	0.14630	0.09341	0.04855	0.02531	0.00701	0.00950	0.00555	0.00305	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

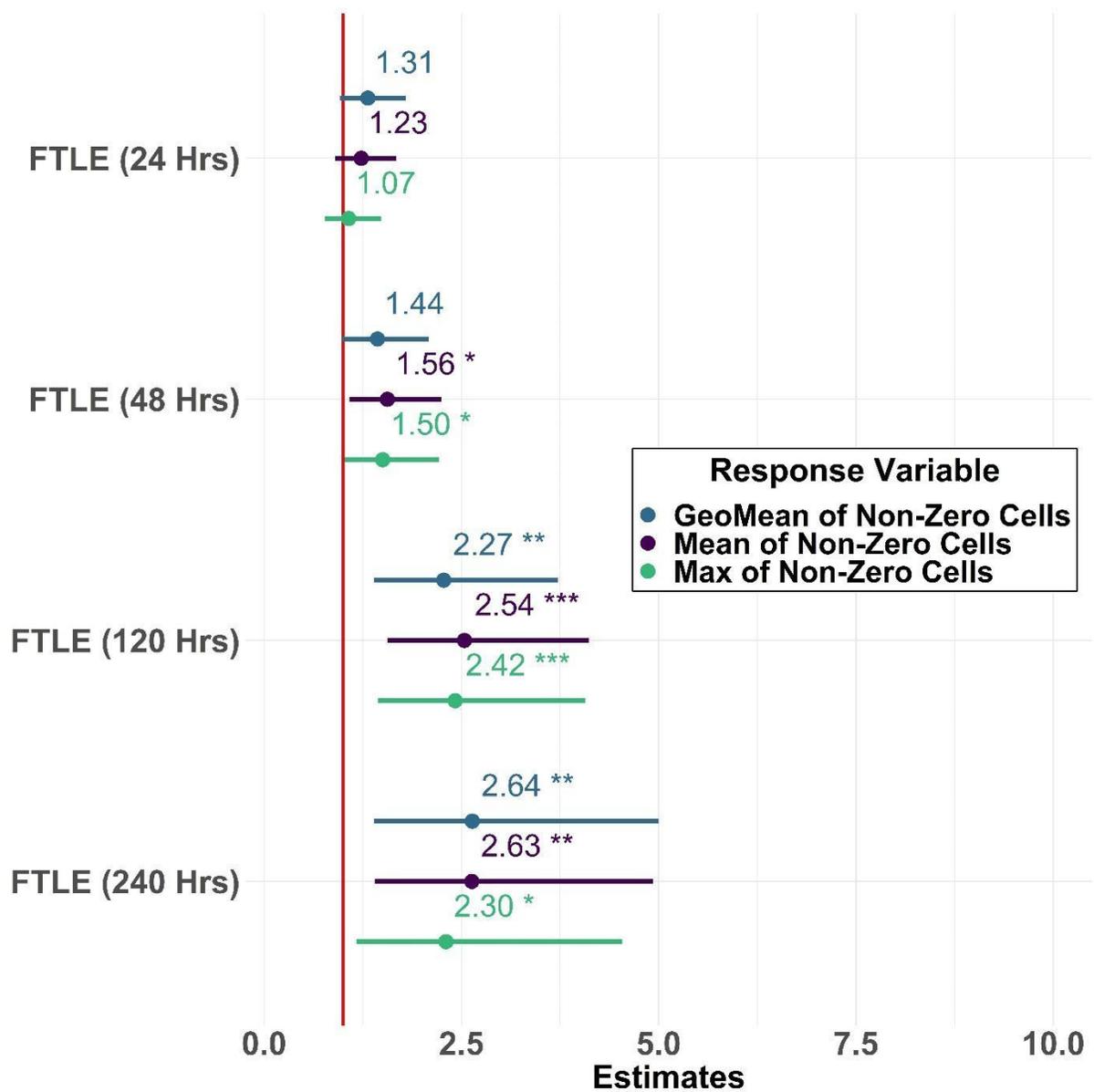
Supplemental Table 1 Length-frequency distributions of krill (8-30 mm) sampled during each cruise, which are used to estimate the average density of krill (grams of krill/m³) for each 200 m horizontal by 5 m vertical grid cell. Note: No krill greater than 27 mm were sampled on any of the cruises and those columns are omitted here.



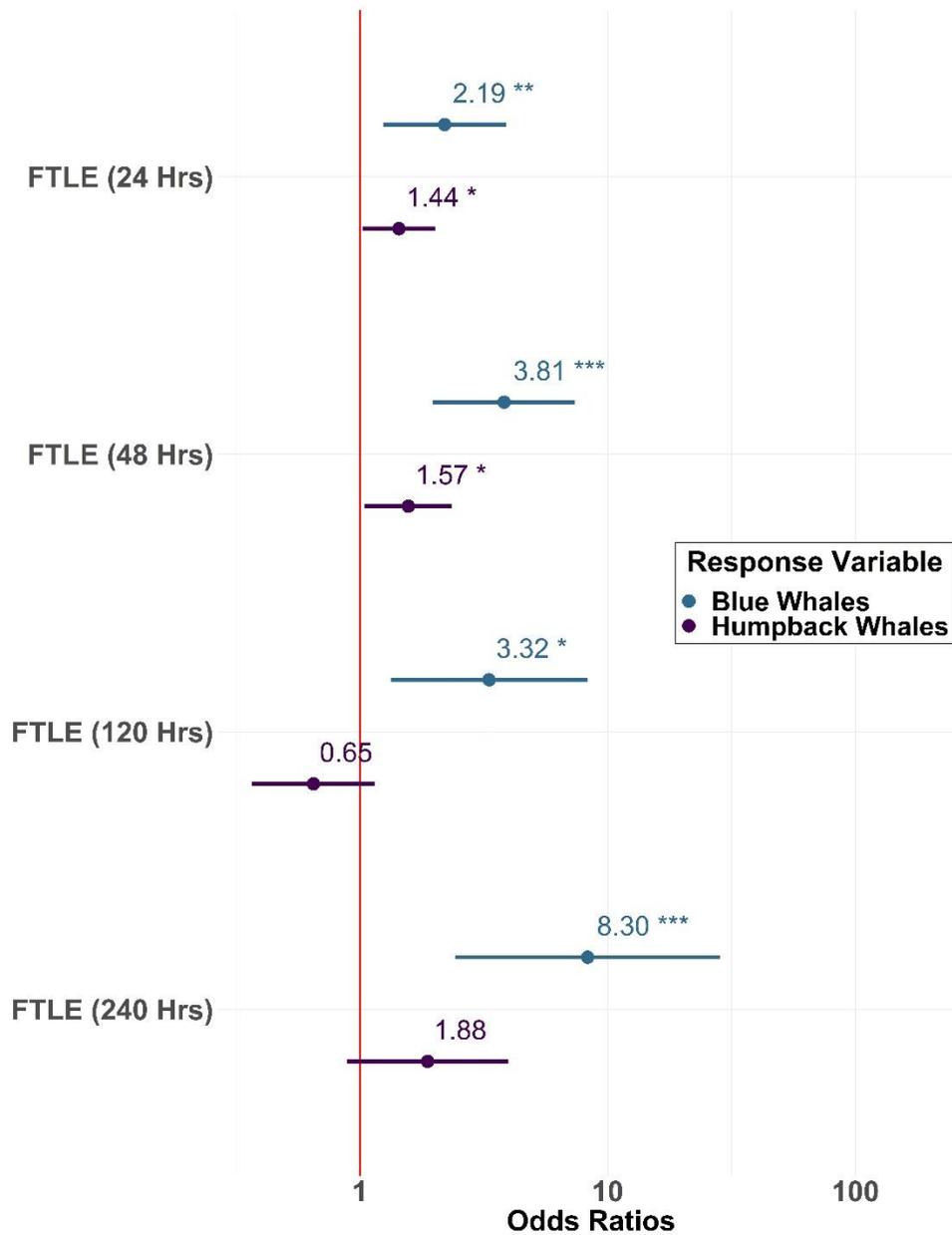
Supplemental Figure 1 Model estimates for the relationship between FTLE and seawater density at 6 different depths (Surface, 10, 20, 30, 40 and 50-meters) at each of the 4 FTLE integration windows (24, 48, 120 and 240-hours). Dots indicate estimated coefficients with their numeric value labeled above; Whiskers correspond to the confidence interval of the coefficient estimates; Red line indicates zero effect; Asterisks indicate the significance level of estimates (* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$).



Supplemental Figure 2 Model estimates for the relationship between FTLE and the depth of seawater density at 3 different values of Sigma Theta (25.0, 25.5 and 26.0 kg/m³) at each of the 4 FTLE integration windows (24, 48, 120 and 240-hours). Dots indicate estimated coefficients with their numeric value labeled above; Whiskers correspond to the confidence interval of the coefficient estimates; Red line indicates zero effect; Asterisks indicate the significance level of estimates (* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$).



Supplemental Figure 3 Model estimates for the relationship between FTLE and 3 metrics of krill density, the mean krill density of cells containing krill (Mean of Non-Zero Krill Density, purple markers), the geometric mean krill density of cells containing krill (GeoMean of Non-Zero Krill Density, blue markers), and the maximum krill density within a column (Max of Non-Zero Krill Density, green markers) at each of the 4 FTLE integration windows. Note that the estimate represents the multiplicative effect of krill density associated with a one-unit increase in FTLE. Dots indicate estimated coefficients with their numeric value labeled above; Whiskers correspond to the confidence interval of the coefficient estimates; Red line indicates zero effect; Asterisks indicate the significance level of estimates (* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$).



Supplemental Figure 4 Model estimates for the relationship between FTLE and the probability of sighting Blue whales (blue markers) and Humpback whales (purple markers) at each of the 4 FTLE integration windows. Note that the estimate represents the odds ratio associated with a one-unit increase in FTLE. Dots indicate estimated coefficients with their numeric value labeled above; Whiskers correspond to the confidence interval of the coefficient estimates; Red line indicates zero effect; Asterisks indicate the significance level of estimates (* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$).

Supplemental Methods for:

Submesoscale coupling of krill and whales revealed by aggregative Lagrangian coherent structures

Proceedings of the Royal Society B, DOI: 10.1098/rspb.2023.2461

Krill and Lagrangian Features

Methods and Analysis

James A. Fahlbusch David E. Cade Elliott L. Hazen Meredith L. Elliott
Benjamin T. Saenz Jeremy A. Goldbogen Jaime Jahncke

30 October, 2023

Contents

Overview	7
Research Questions	7
Data Summary	7
Temporal Overview	7
Spatial Overview	8
Methods	10
FTLE	10
Krill	21
Cetacean Data	26
Oceanographic Data	28
Analysis	30
Oceanographic Expression of FTLE Features	30
1. Do surface FTLE features correspond to density differences and if so, to what depth? . . .	30
GLMM (Seawater Density ~ FTLE)	30
Density at Surface	31
Density at 10 meters	32
Density at 20 meters	33
Density at 30 meters	34
Density at 40 meters	35
Density at 50 meters	36
Supplemental Figure 1	38
Figure 3A	41
GLMM (Depth of Sigma Theta ~ FTLE)	42
Depth of Sigma Theta 25	42
Depth of Sigma Theta 25.5	43
Depth of Sigma Theta 26	44
Supplemental Figure 2	46
FTLE and Krill	47
2. Does Krill presence increase with increasing FTLE?	48
GLMM (Krill Presence ~ FTLE)	48
3. Does Krill density increase with increasing FTLE?	50
GLMM (Krill Density ~ FTLE)	51
Geometric Mean Non Zero	51
Mean Non Zero	52
Max Density	54

Supplemental Figure 3	57
Figure 3B	60
FTLE and Cetacean Sightings	61
4. Are Cetacean sightings more likely in areas with higher FTLE?	61
GLMM (Whale Presence ~ FTLE)	61
Blue Whale Sightings	61
Figure 3C	64
Humpback Whale Sightings	65
Supplemental Figure 4	67
Results Table	68

References **70**

Load Packages and Data

```
#### Load Libraries and Functions ####
pkgTest <- function(x)
{
  if (!require(x,character.only = TRUE))
  {
    install.packages(x,dep=TRUE)
    if(!require(x,character.only = TRUE)) stop("Package not found")
  }
}

pkgTest("tidyverse")
pkgTest("lubridate")
pkgTest("ggpubr") # for ggarrange
pkgTest("rnaturalearth") # for map base layers
pkgTest("paletteer")
pkgTest("metR") # for plotting bathy contours
pkgTest("ggspatial") # for plotting map annotations
pkgTest("kableExtra")
pkgTest("raster")
pkgTest("geodist") # for calculating distances

# Stats Analysis
pkgTest("lme4")
pkgTest("MASS")
pkgTest("mgcv")
pkgTest("stats")
pkgTest("sjPlot")
pkgTest("sjlabelled")
pkgTest("gridExtra") # for grid.arrange
pkgTest("AICcmodavg")

# For FTLE Plots
pkgTest("viridis")
pkgTest("ncdf4")
pkgTest("marmap")
pkgTest("sf")
pkgTest("ggnewscale")

# Process netCDF file dates and times
source("../functions/ncdate.R")
# Note: getNOAA.bathy was not working,
```

```

# but an alternate version on github is used here
source('./functions/getNOAAbathy.R')
# Get Sunrise and sunset times
source("./functions/find_Astronomical.R")

#Function to convert between Logit and Probability
logit2prob <- function(logit){
  odds <- exp(logit)
  prob <- odds / (1 + odds)
  return(prob)
}
# Function to select items Not In a group
`%notin%` <- Negate(`%in%`)

## Load Data
world <- ne_countries(scale = "large", returnclass = "sf")
load("./Figure_Files/Map_Data/bf.RData")
load("./Figure_Files/Map_Data/bf_Sub.RData")
load("./Figure_Files/Map_Data/HFRadarCoverageMCP.RData")
load("./Figure_Files/Map_Data/hfRadarStations.RData")
# HF Radar Coverage Area and Station Locations
hfRadarStns$ID <- "HF Radar Station"
hfRadar_DF <- as.data.frame(hfRadarStns)
hfRadar_DF$Long <- st_coordinates(hfRadarStns$geometry)[,1]
hfRadar_DF$Lat <- st_coordinates(hfRadarStns$geometry)[,2]

## Cetacean Data
load("./dataProcessed/cetacean_DF_FTLE.RData")

## Krill Data
# 600m Krill Data
load("./dataProcessed/acousticsFTLE600cols_Bathy.RData")
# Full 200m X 5m Krill dataset
load("./dataProcessed/acoustics_DF_FTLE_Line_w600.RData")

## CTD Data
load("./dataProcessed/castSummary_FTLE.RData")

## Load previously saved Models
# Krill Presence/Absence
if (file.exists("./Models/krill_glmmPQL_CorExp_24_Hrs.rds")) {
  krill_glmmPQL_CorExp_24_Hrs <-
    readRDS("./Models/krill_glmmPQL_CorExp_24_Hrs.rds")
} else {
  cat("Could not find ./Models/krill_glmmPQL_CorExp_24_Hrs.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krill_glmmPQL_CorExp_48_Hrs.rds")) {
  krill_glmmPQL_CorExp_48_Hrs <-
    readRDS("./Models/krill_glmmPQL_CorExp_48_Hrs.rds")
} else {
  cat("Could not find ./Models/krill_glmmPQL_CorExp_48_Hrs.rds.")
  cat("Re-create below")
}

```

```

}
if (file.exists("./Models/krill_glmmPQL_CorExp_120_Hrs.rds")) {
  krill_glmmPQL_CorExp_120_Hrs <-
    readRDS("./Models/krill_glmmPQL_CorExp_120_Hrs.rds")
} else {
  cat("Could not find ./Models/krill_glmmPQL_CorExp_120_Hrs.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krill_glmmPQL_CorExp_240_Hrs.rds")) {
  krill_glmmPQL_CorExp_240_Hrs <-
    readRDS("./Models/krill_glmmPQL_CorExp_240_Hrs.rds")
} else {
  cat("Could not find ./Models/krill_glmmPQL_CorExp_240_Hrs.rds.")
  cat("Re-create below")
}

# Krill Mean Density Log-Link
if (file.exists("./Models/krillDensity_MNZGLMMPQLLogLink24.rds")) {
  krillDensity_MNZGLMMPQLLogLink24 <-
    readRDS("./Models/krillDensity_MNZGLMMPQLLogLink24.rds")
} else {
  cat("Could not find ./Models/krillDensity_MNZGLMMPQLLogLink24.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MNZGLMMPQLLogLink48.rds")) {
  krillDensity_MNZGLMMPQLLogLink48 <-
    readRDS("./Models/krillDensity_MNZGLMMPQLLogLink48.rds")
} else {
  cat("Could not find ./Models/krillDensity_MNZGLMMPQLLogLink48.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MNZGLMMPQLLogLink120.rds")) {
  krillDensity_MNZGLMMPQLLogLink120 <-
    readRDS("./Models/krillDensity_MNZGLMMPQLLogLink120.rds")
} else {
  cat("Could not find ./Models/krillDensity_MNZGLMMPQLLogLink120.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MNZGLMMPQLLogLink240.rds")) {
  krillDensity_MNZGLMMPQLLogLink240 <-
    readRDS("./Models/krillDensity_MNZGLMMPQLLogLink240.rds")
} else {
  cat("Could not find ./Models/krillDensity_MNZGLMMPQLLogLink240.rds.")
  cat("Re-create below")
}

# Krill GeoMean Density Log-Link
if (file.exists("./Models/krillDensity_gMNZGLMMPQLLogLink24.rds")) {
  krillDensity_gMNZGLMMPQLLogLink24 <-
    readRDS("./Models/krillDensity_gMNZGLMMPQLLogLink24.rds")
} else {
  cat("Could not find ./Models/krillDensity_gMNZGLMMPQLLogLink24.rds.")
  cat("Re-create below")
}

```

```

}
if (file.exists("./Models/krillDensity_gMNZGLMMPQLLogLink48.rds")) {
  krillDensity_gMNZGLMMPQLLogLink48 <-
    readRDS("./Models/krillDensity_gMNZGLMMPQLLogLink48.rds")
} else {
  cat("Could not find ./Models/krillDensity_gMNZGLMMPQLLogLink48.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_gMNZGLMMPQLLogLink120.rds")) {
  krillDensity_gMNZGLMMPQLLogLink120 <-
    readRDS("./Models/krillDensity_gMNZGLMMPQLLogLink120.rds")
} else {
  cat("Could not find ./Models/krillDensity_gMNZGLMMPQLLogLink120.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_gMNZGLMMPQLLogLink240.rds")) {
  krillDensity_gMNZGLMMPQLLogLink240 <-
    readRDS("./Models/krillDensity_gMNZGLMMPQLLogLink240.rds")
} else {
  cat("Could not find ./Models/krillDensity_gMNZGLMMPQLLogLink240.rds.")
  cat("Re-create below")
}

# Krill Maximum Density Log-Link
if (file.exists("./Models/krillDensity_MaxGLMMPQLLogLink24.rds")) {
  krillDensity_MaxGLMMPQLLogLink24 <-
    readRDS("./Models/krillDensity_MaxGLMMPQLLogLink24.rds")
} else {
  cat("Could not find ./Models/krillDensity_MaxGLMMPQLLogLink24.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MaxGLMMPQLLogLink48.rds")) {
  krillDensity_MaxGLMMPQLLogLink48 <-
    readRDS("./Models/krillDensity_MaxGLMMPQLLogLink48.rds")
} else {
  cat("Could not find ./Models/krillDensity_MaxGLMMPQLLogLink48.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MaxGLMMPQLLogLink120.rds")) {
  krillDensity_MaxGLMMPQLLogLink120 <-
    readRDS("./Models/krillDensity_MaxGLMMPQLLogLink120.rds")
} else {
  cat("Could not find ./Models/krillDensity_MaxGLMMPQLLogLink120.rds.")
  cat("Re-create below")
}
if (file.exists("./Models/krillDensity_MaxGLMMPQLLogLink240.rds")) {
  krillDensity_MaxGLMMPQLLogLink240 <-
    readRDS("./Models/krillDensity_MaxGLMMPQLLogLink240.rds")
} else {
  cat("Could not find ./Models/krillDensity_MaxGLMMPQLLogLink240.rds.")
  cat("Re-create below")
}

```

```

## Process Cetacean Data for Analysis
# Create a subset dataset that removes Minke and Gray Whales,
# and removes distant sighting outliers
cetacean_Sub <- cetacean_DF_FTLE %>%
  dplyr::filter(Type == "ACC") %>%
  dplyr::filter(SpCode %notin% c('MIWH','GRWH','UNWH','FIWH'),
                DisFrShipNm < 0.5) # 0.5nm is approximately 926m

# Create a Ship-Track Cetacean Sightings dataset to join to Acoustics
cetacean_Ship <- cetacean_Sub %>%
  dplyr::filter(!is.na(Col600)) %>%
  group_by(Cruise, Line, Col600) %>%
  summarize(BLWH = ifelse(length(SpCode[SpCode == "BLWH"])>0, 1,0),
            HUWH = ifelse(length(SpCode[SpCode == "HUWH"])>0, 1,0)
            )

# Extract CTD Cast Locations for Plotting
ctd_Stations <- castSummary_FTLE %>%
  dplyr::filter(Cruise %notin% c("ACD1807", "ACM1705")) %>%
  group_by(Lat, Long) %>% summarize(Lat = first(Lat),Long=first(Long))
# Create a summary of CTD casts
dataCTD <- castSummary_FTLE %>%
  # Limit to Gulf of the Farallones only.
  dplyr::filter(Cruise %notin% c("ACD1807", "ACM1705")) %>%
  # Remove shallow water casts
  dplyr::filter(maxDepth >= 50) %>%
  mutate(Year = year(dttz),
         mo = month(dttz),
         Season = if_else(mo<7,"Spring",if_else(mo==7,"Summer","Fall" )))
# Save data for use in figure creation
save(dataCTD,file="./Figure_Files/dataCTD.RData")

# Process Krill data for analysis
dataGamm <- acousticsFTLE600cols_Bathy %>%
  # Add Cetacean Sightings
  left_join(cetacean_Ship, by = c("Cruise", "Line", "Col600")) %>%
  mutate(BLWH = ifelse(is.na(BLWH), 0, BLWH),
         HUWH = ifelse(is.na(HUWH), 0, HUWH)
         ) %>%
  # Limit to Gulf of the Farallones only.
  dplyr::filter(Type == "ACC",
                # Subset Data to only include daytime observations
                astronomical == "day") %>%
  # Create Categorization Columns for Analysis
  mutate(Year = year(dttz),
         Season = if_else(month(dttz)<7,"Spring",
                          if_else(month(dttz)==7,"Summer","Fall" )),
         KrillPresence = if_else(KrillBiomass == 0, 0, 1)
         ) %>%
  arrange(Cruise,Line,dttz)
# Save data for use in figure creation
save(dataGamm,file=paste0("./Figure_Files/dataGamm.RData"))

```

Overview

Exploring the scale of surface oceanographic processes as they relate to krill density and distribution, cetacean distribution and oceanographic features.

Research Questions

- **Oceanography-Related**
 1. Is there an oceanographic signature at FTLE features?
 - Do surface FTLE features correspond to density differences at the surface, 10, 20, 30, 40 or 50 meters?
 - * Linear Regression (GLMM using CTD data)
- **Krill-Related**
 2. Is Krill presence more likely in areas with higher FTLE?
 - Is FTLE for areas with Krill higher than areas without Krill? (And if so, at what temporal scale is the relationship most prevalent?)
 - * Logistic Regression (GLMM)
 3. Does Krill Density increase with increasing FTLE?
 - For areas that have krill, does krill density increase with increasing FTLE? (And if so, at what temporal scale is the relationship most prevalent?)
 - * We look at several metrics for Density (Geometric Mean of Non Zero Cells, Mean Density of Non Zero Cells, Max Density)
 - * Linear Regression (GLMM)
- **Whale-Related**
 4. Are Cetacean sightings more likely in areas with higher FTLE?
 - How does this differ between species and across FTLE Integration times?
 - * Logistic Regression (GLMM)

Data Summary

Temporal Overview

ACCESS completed 21 Cruises in the Gulf of the Farallones Region between 2012 and 2018 with 3 Cruises each year (Spring, Summer and Fall). Most of the cruises collected CTD data (with several exceptions) and we included only survey data from daytime hours.

```
CTD_Sum <- dataCTD %>%
  group_by(Cruise) %>%
  summarize(CTD_Casts = n())

# Calculate the length of each transect line
lineDistances <- dataGamm %>%
  # Limit to standard ACCESS cruise lines only
  dplyr::filter(Type == "ACC") %>%
  group_by(Cruise, Line) %>%
  summarize(Start = round_date(min(dttz),unit="hours"),
            End = round_date(max(dttz),unit="hours"),
            Duration = round(difftime(End,Start,units="hours"),2),
            TransectDistance =
              round(as.numeric(geodist::geodist_vec(x1 = first(Long),
                                                    x2 = last(Long),
                                                    y1 = first(Lat),
                                                    y2 = last(Lat)))/1000,1)
            )
# Calculate the length of all lines in each cruise
```

Table 1:

Cruise	Start	End	Duration	Lines	CTD_Casts	Transect_Distance
ACC1206	2012-06-19	2012-06-24	5 days	11	18	220.6
ACC1207	2012-07-24	2012-07-29	5 days	11	18	269.2
ACC1209	2012-09-13	2012-09-19	6 days	15	19	384.2
ACC1305	2013-05-25	2013-05-30	5 days	11	13	239.2
ACC1307	2013-07-24	2013-07-30	6 days	13	0	376.1
ACC1309	2013-09-26	2013-10-01	5 days	9	15	261.4
ACC1406	2014-06-21	2014-06-26	5 days	3	9	129.2
ACC1407	2014-07-17	2014-07-24	7 days	16	20	588.7
ACC1409	2014-09-20	2014-09-27	7 days	12	25	357.1
ACC1506	2015-06-21	2015-06-26	5 days	4	8	109.5
ACC1507	2015-07-19	2015-07-25	6 days	13	24	273.6
ACC1509	2015-09-23	2015-09-25	2 days	3	15	83.1
ACC1605	2016-05-15	2016-05-22	7 days	19	30	487.5
ACC1607	2016-07-18	2016-07-22	4 days	10	13	224.9
ACC1609	2016-09-18	2016-09-21	3 days	8	13	193.5
ACC1705	2017-05-26	2017-05-29	3 days	7	15	181.4
ACC1707	2017-07-22	2017-07-28	6 days	15	0	332.1
ACC1709	2017-09-23	2017-09-29	6 days	14	19	358.0
ACC1805	2018-05-25	2018-05-29	4 days	9	14	177.0
ACC1807	2018-07-04	2018-07-10	6 days	21	33	771.6
ACC1809	2018-09-21	2018-09-28	7 days	16	22	420.8

```

lineDistanceSum <- lineDistances %>%
  group_by(Cruise) %>%
  summarize(Transect_Distance = sum(TransectDistance)
            )

# Summary of Cruise start and end times
table1 <- dataGamm %>%
  # Limit to standard ACCESS cruise lines only
  dplyr::filter(Type == "ACC") %>%
  group_by(Cruise) %>%
  summarize(Start = round_date(min(dttz),unit="days"),
            End = round_date(max(dttz),unit="days"),
            Duration = round(difftime(End,Start,units="days"),2),
            Lines = length(unique(Line))) %>%
  left_join(CTD_Sum) %>%
  mutate(CTD_Casts = if_else(is.na(CTD_Casts),as.integer(0),CTD_Casts)) %>%
  left_join(lineDistanceSum)
# Save to CSV
write_csv(table1,"./Output/Table1-CruiseSummary.csv" )
kable(x=table1,caption = "")

```

Spatial Overview

The ACCESS Survey transect lines generally run along latitudinal parallels spanning the Greater Farallones and Cordell Bank National Marine Sanctuaries. These surveys sampled in 3 seasons (Spring, Summer, and Fall) and our dataset has been limited to the years 2012-2018 to overlap with HF Radar data coverage in the region.

```

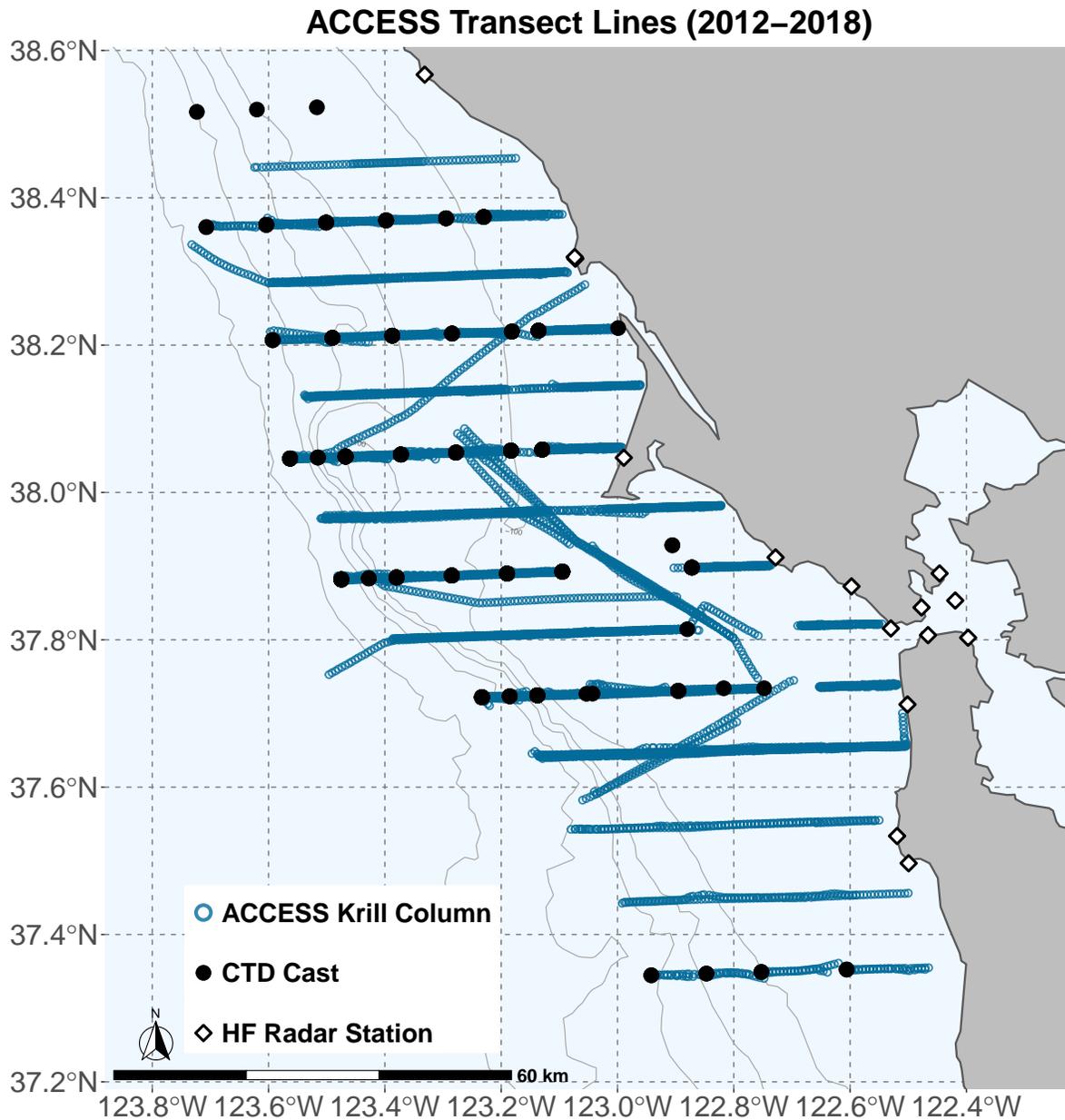
# All Lines Together
p_AllLines <- dataGamm %>%
  dplyr::select(Line, Lat, Long) %>%
  unique() %>%
  ggplot() +
  geom_sf(data = world, color=NA, fill=NA) +
  # add 200m contour
  geom_contour(data = bf,
               aes(x=x, y=y, z=z),
               breaks=c(-100,-200,-400,-800,-1600),
               size=c(0.4),
               colour="darkgrey", show.legend = FALSE) +
  geom_text_contour(data = bf, aes(x=x, y=y, z = z),
                   breaks=c(-100,-200,-400,-800,-1600),
                   show.legend = FALSE, size = 2.2,
                   alpha = .6, nudge_y = -.002) +
  geom_point(aes(Long, Lat, color="ACCESS Krill Column"),
             alpha=.8, stroke=0.9, size=2, shape=21, show.legend = TRUE) +
  geom_sf(data = world, size=.75, fill="grey") +
  geom_point(data=dataCTD, aes(x=Long, y=Lat, alpha="CTD Cast"), size=5) +
  geom_point(data=hfRadar_DF,
             aes(x=Long, y=Lat, shape = ID), color="black",
             size=3.5, stroke=1.5, fill = 'white') +
  scale_color_manual(values = c("ACCESS Krill Column"="#046C9A")) +
  scale_alpha_manual(values=c("CTD Cast"=1))+
  scale_shape_manual(values = c("HF Radar Station" = 23))+
  coord_sf(xlim = c(min(dataGamm$Long, na.rm = TRUE)-.15,
                    max(dataGamm$Long, na.rm = TRUE)+.25),
           ylim = c(min(dataGamm$Lat, na.rm = TRUE)-.15,
                    max(dataGamm$Lat, na.rm = TRUE)+.15), expand = FALSE) +
  annotation_scale(location = "bl", width_hint = 0.5, text_face = "bold",
                  text_col = "black", text_cex = 1.2) +
  annotation_north_arrow(location = "bl",
                        which_north = "true",
                        pad_x = unit(0.25, "in"),
                        pad_y = unit(0.25, "in"),
                        style = north_arrow_fancy_orienteering) +
  xlab("Longitude") +
  ylab("Latitude") +
  ggtitle("ACCESS Transect Lines (2012-2018)") +
  guides(color = guide_legend(order=1, override.aes = list(size=4, stroke=2)))+
  theme(panel.grid.major = element_line(color = gray(.5),
                                         linetype = "dashed", size = 0.5),
        panel.background = element_rect(fill = "aliceblue"),
        axis.title = element_blank(),
        axis.text = element_text( face="bold", size=22),
        plot.title = element_text(face="bold", size=25, hjust = 0.5),
        plot.subtitle = element_text( face="bold", size=18, hjust = 0.5),
        plot.caption = element_text( face="bold", size=14, hjust = 0),
        legend.position= c(0.245, 0.115),
        legend.key = element_blank(),
        legend.box.background = element_rect(color = "white", fill="white"),
        legend.background = element_rect(color = "white", fill="white"),

```

```

legend.text = element_text(face="bold", size=20),
legend.title = element_blank()
p_AllLines

```



Methods

FTLE

A recent study in the same region (Gough et al 2016) found good agreement between SST and LCS during summer months (though the relationship fell apart during strong upwelling-favorable winds in late March). Here we use the 2 integration durations used in that study (1 and 5 days), the integration used in Fahlbusch et al 2021 (2 days) as well as a longer period for comparison (10 days) to examine the interaction between

krill density and distribution and Lagrangian Coherent Structures (LCS) during summer months. **Here we see a comparison of the 4 integration times for a single survey transect line in 2012.**

```
#### Setup ####
tzOffset <- "Etc/GMT+7" # all surveys are on the same timezone
FTLEdatapath <- "./dataRaw/FTLE/"
degreeBuffer <- .25
quantilePlot <- seq(.3,.95,by=.07)

c = "ACC1207" # Cruise
l = "6" # Line
data_Sub <- dataGamm %>%
  dplyr::filter(Cruise == c, Line == l)
# Subset full resolution acoustics data for this cruise/line
d_Sub_Line <- acoustics_DF_FTLE_W600 %>%
  dplyr::filter(Cruise == c, Line == l)
# Create a column summary for this line
lineSum <- d_Sub_Line %>%
  mutate(minCID = min(as.integer(d_Sub_Line$ColumnID))) %>%
  group_by(ColumnID) %>%
  summarise(dttz = first(dttz),
            Lat = first(Lat),
            Long = first(Long),
            ftle24=first(ftle24_L),
            ftle48=first(ftle48_L),
            ftle120=first(ftle120_L),
            ftle240=first(ftle240_L),
            botDepth = first(BotDepth),
            KrillBiomass = sum(KrillBiomass),
            numWith = sum(KrillBiomass>0),
            density = if_else(numWith > 0, mean(KrillDensity[KrillDensity>0]),0),
            logDensity = if_else(density==0, -4,log(density)),
            x = (as.integer(first(ColumnID))-first(minCID))*200)

# Load the FTLE data
FTLE_stack24 <- stack(paste0(FTLEdatapath,c,"_FTLE_24Hrs.nc"),
                     varname = "FTLE")
FTLE_stack48 <- stack(paste0(FTLEdatapath,c,"_FTLE_48Hrs.nc"),
                     varname = "FTLE")
FTLE_stack120 <- stack(paste0(FTLEdatapath,c,"_FTLE_120Hrs.nc"),
                      varname = "FTLE")
FTLE_stack240 <- stack(paste0(FTLEdatapath,c,"_FTLE_240Hrs.nc"),
                      varname = "FTLE")

# FTLE PLOT
ftle_alpha <- function(n, flatten, scrunch) {
  c(scales::rescale(exp((1:flatten)/scrunch), to = c(0.25, 1)),
    rep(1, n - flatten))
}

#Determine Quantiles for FTLE Color Scaling
qFTLE24 <- quantile(FTLE_stack24, probs = quantilePlot,na.rm=TRUE)
qSumFTLE24 <- as.data.frame(summary(qFTLE24)) %>%
  dplyr::select(-Var1) %>%
  dplyr::filter(grepl("Mean",Freq)) %>%
```

```

  group_by(Var2) %>%
  mutate(Means = unlist(str_split(Freq,":"))[2]) %>%
  ungroup()
qFTLE48 <- quantile(FTLE_stack48, probs = quantilePlot,na.rm=TRUE)
qSumFTLE48 <- as.data.frame(summary(qFTLE48)) %>%
  dplyr::select(-Var1) %>%
  dplyr::filter(grepl("Mean",Freq)) %>%
  group_by(Var2) %>%
  mutate(Means = unlist(str_split(Freq,":"))[2]) %>%
  ungroup()
qFTLE120 <- quantile(FTLE_stack120, probs = quantilePlot,na.rm=TRUE)
qSumFTLE120 <- as.data.frame(summary(qFTLE120)) %>%
  dplyr::select(-Var1) %>%
  dplyr::filter(grepl("Mean",Freq)) %>%
  group_by(Var2) %>%
  mutate(Means = unlist(str_split(Freq,":"))[2]) %>%
  ungroup()
qFTLE240 <- quantile(FTLE_stack240,probs = quantilePlot,na.rm=TRUE)
qSumFTLE240 <- as.data.frame(summary(qFTLE240)) %>%
  dplyr::select(-Var1) %>%
  dplyr::filter(grepl("Mean",Freq)) %>%
  group_by(Var2) %>%
  mutate(Means = unlist(str_split(Freq,":"))[2]) %>%
  ungroup()

#### Maps ####

# create a dataframe of each integration
FTLE_spdf24 <-
as(mean(FTLE_stack24[[first(d_Sub_Line$lineMinI):first(d_Sub_Line$lineMaxI)]]),
  "SpatialPixelsDataFrame")
FTLE_df24 <- as.data.frame(FTLE_spdf24)
colnames(FTLE_df24) <- c("FTLE", "x", "y")
FTLE_spdf48 <-
  as(mean(FTLE_stack48[[first(d_Sub_Line$lineMinI):first(d_Sub_Line$lineMaxI)]]),
    "SpatialPixelsDataFrame")
FTLE_df48 <- as.data.frame(FTLE_spdf48)
colnames(FTLE_df48) <- c("FTLE", "x", "y")
FTLE_spdf120 <-
  as(mean(FTLE_stack120[[first(d_Sub_Line$lineMinI):first(d_Sub_Line$lineMaxI)]]),
    "SpatialPixelsDataFrame")
FTLE_df120 <- as.data.frame(FTLE_spdf120)
colnames(FTLE_df120) <- c("FTLE", "x", "y")
FTLE_spdf240 <-
  as(mean(FTLE_stack240[[first(d_Sub_Line$lineMinI):first(d_Sub_Line$lineMaxI)]]),
    "SpatialPixelsDataFrame")
FTLE_df240 <- as.data.frame(FTLE_spdf240)
colnames(FTLE_df240) <- c("FTLE", "x", "y")

# 24 ####
p24 <- ggplot() +
  geom_sf(data = world,color=NA,fill=NA) +
  geom_raster(data = FTLE_df24[complete.cases(FTLE_df24),],

```

```

        mapping = aes(x, y, fill = FTLE)) +
labs(x="", y="") +
scale_fill_gradientn(colors = viridis(option = "plasma", n = 40,
                                     alpha = ftle_alpha(40, 8, 8)),
                    limits=c(0, 1.25),
                    breaks = c(0,0.6,1.2),
                    values = scales::rescale(
                      c(0,as.numeric(qSumFTLE24$Means[1:9]))),
                    oob=scales::squish,
                    na.value = NA,
                    name = expression(atop( bold('FTLE (24 hr)'),
                                           bold(day^bold({bold("-1")})))))) +

# add 100m and 200m contours
ggplot2::geom_contour(data = bf_Sub,
                     mapping = aes(x=x, y=y, z=z),
                     breaks=c(-100),
                     size=c(0.4),
                     colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
                       breaks = c(-100), show.legend = FALSE,
                       size = 2.5, alpha = .8, nudge_y = -.002,
                       color="lightgrey") +
ggplot2::geom_contour(data = bf_Sub,
                     mapping = aes(x=x, y=y, z=z),
                     breaks=c(-200),
                     size=c(0.4),
                     colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
                       breaks = c(-200),
                       label.placement = label_placement_n(5),
                       show.legend = FALSE, size = 2.5, alpha = .8,
                       nudge_y = -.002, color="lightgrey") +
geom_point(data = data_Sub,
           mapping = aes(Long,Lat, color=KrillDensity_MeanNonZero ),
           size=5,shape=15, show.legend = TRUE,
           alpha = 1, fill="black") +
geom_label(data = lineSum[c(1,length(lineSum$ColumnID) )],
           mapping = aes(Long,Lat),label=c("Start","End"),
           size=2.5,nudge_y = .02)+
scale_color_gradient(name = expression(atop(bold("Krill Density"),
                                           bold(log[e](g/m^3)) )),
                    low="white",
                    high="magenta",
                    na.value = "white",
                    limits = c(0,4),
                    breaks=seq(0,5,1),
                    oob=scales::squish) +
geom_sf(data = world, size=1.25,fill="lightgrey",color="grey") +
geom_sf(data = world, size=1.25,fill="lightgrey",color="grey") +
coord_sf(xlim = c(min(d_Sub_Line$Long,na.rm = TRUE)-degreeBuffer,
                    max(d_Sub_Line$Long,na.rm = TRUE)+degreeBuffer),
          ylim = c(min(d_Sub_Line$Lat,na.rm = TRUE)-0.35,
                    max(d_Sub_Line$Lat,na.rm = TRUE)+0.35),

```

```

    expand = FALSE) +
theme_classic() +
annotation_scale(location = "br", width_hint = 0.5,
                 text_face = "bold", text_col = "grey") +
annotation_north_arrow(location = "br",
                      which_north = "true",
                      pad_x = unit(0.25, "in"),
                      pad_y = unit(0.25, "in"),
                      style = north_arrow_fancy_orienteering(line_col = "grey",
                                                              fill = c("white",
                                                                    "black"),
                                                              text_col = "grey")) +

xlab("Longitude") +
ylab("Latitude") +
guides(color = guide_colourbar(order=1,direction = 'horizontal' ,
                              title.position = "top",
                              title.hjust = .5,
                              label.position = "bottom"),
       fill = guide_colourbar(order=2,direction = 'horizontal',
                              title.position = "top",
                              title.hjust = .5,
                              label.position = "bottom")) +
theme(panel.grid.major = element_line(color = gray(.5),
                                      linetype = "dashed", size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      axis.text = element_text(face="bold", size=10),
      axis.title = element_blank(),
      legend.text = element_text(face="bold", size=10),
      legend.position = c(.125,.2),
      legend.box = 'vertical',
      legend.title = element_text(face="bold", size=12),
      legend.box.background = element_rect(color="white",
                                           size=2.5, fill="white"),
      legend.spacing.y = unit(0.025, "cm"),
      # adds a buffer on the sides so the plot is centered
      plot.margin=margin(t = 0.5, r = 1, b = 0, l = 0.25, "cm"),
      plot.title = element_text(face="bold", size=24, hjust = 0.5),
      plot.subtitle = element_text(face="bold", size=20, hjust = 0.5))

# 48 ####
p48 <- ggplot()+
  geom_sf(data = world,color=NA,fill=NA) +
  geom_raster(data = FTLE_df48[complete.cases(FTLE_df48),],
             mapping = aes(x, y, fill = FTLE)) +
  labs(x="", y="") +
  scale_fill_gradientn(colors = viridis(option = "plasma", n = 40,
                                       alpha = ftle_alpha(40, 8, 8)),
                     values = scales::rescale(
                       c(0,as.numeric(qSumFTLE48$Means[1:9]))),
                     limits=c(0, 1),
                     breaks = c(0,0.5,1),
                     oob=scales::squish,
                     na.value = NA,

```

```

        name = expression(atop( bold('FTLE (48 hr)'),
                                bold(day^bold({bold("-1")})))) +
# add 100m and 200m contours
ggplot2::geom_contour(data = bf_Sub,
  mapping = aes(x=x, y=y, z=z),
  breaks=c(-100),
  size=c(0.4),
  colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
  breaks = c(-100), show.legend = FALSE,
  size = 2.5, alpha = .8, nudge_y = -.002,
  color="lightgrey") +
ggplot2::geom_contour(data = bf_Sub,
  mapping = aes(x=x, y=y, z=z),
  breaks=c(-200),
  size=c(0.4),
  colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
  breaks = c(-200),
  label.placement = label_placement_n(5),
  show.legend = FALSE, size = 2.5, alpha = .8,
  nudge_y = -.002, color="lightgrey") +
geom_point(data = data_Sub,
  mapping = aes(Long,Lat, color=KrillDensity_MeanNonZero ),
  size=5,shape=15, show.legend = TRUE,
  alpha = 1, fill="black") +
geom_label(data = lineSum[c(1,length(lineSum$ColumnID) )],
  mapping = aes(Long,Lat),label=c("Start","End"),
  size=2.5,nudge_y = .02)+
scale_color_gradient(name = expression(atop(bold("Krill Density"),
                                             bold(log[e](g/m^3)) )),
  low="white",
  high="magenta",
  na.value = "white",
  limits = c(0,4),
  breaks=seq(0,5,1),
  oob=scales::squish) +
geom_sf(data = world, size=1.25,fill="lightgrey",color="grey") +
coord_sf(xlim = c(min(d_Sub_Line$Long,na.rm = TRUE)-degreeBuffer,
  max(d_Sub_Line$Long,na.rm = TRUE)+degreeBuffer),
  ylim = c(min(d_Sub_Line$Lat,na.rm = TRUE)-0.35,
  max(d_Sub_Line$Lat,na.rm = TRUE)+0.35),
  expand = FALSE) +
theme_classic() +
annotation_scale(location = "br", width_hint = 0.5,
  text_face = "bold", text_col = "grey") +
annotation_north_arrow(location = "br",
  which_north = "true",
  pad_x = unit(0.25, "in"),
  pad_y = unit(0.25, "in"),
  style = north_arrow_fancy_orienteering(line_col = "grey",
  fill = c("white",
  "black"),

```

```

xlab("Longitude") +
ylab("Latitude") +
guides(color = guide_colourbar(order=1,direction = 'horizontal' ,
                                title.position = "top",
                                title.hjust = .5,
                                label.position = "bottom"),
        fill = guide_colourbar(order=2,direction = 'horizontal',
                                title.position = "top",
                                title.hjust = .5,
                                label.position = "bottom")) +
theme(panel.grid.major = element_line(color = gray(.5),
                                       linetype = "dashed", size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      axis.text = element_text(face="bold", size=10),
      axis.title = element_blank(),
      legend.text = element_text(face="bold", size=10),
      legend.position = c(.125,.2),
      legend.box = 'vertical',
      legend.title = element_text(face="bold", size=12),
      legend.box.background = element_rect(color="white",
                                           size=2.5, fill="white"),
      legend.spacing.y = unit(0.025, "cm"),
      # adds a buffer on the sides so the plot is centered
      plot.margin=margin(t = 0.5, r = 1, b = 0, l = 0.25, "cm"),
      plot.title = element_text(face="bold", size=24, hjust = 0.5),
      plot.subtitle = element_text(face="bold", size=20, hjust = 0.5))

# 120 ####
p120 <- ggplot()+
  geom_sf(data = world,color=NA,fill=NA) +
  geom_raster(data = FTLE_df120[complete.cases(FTLE_df120),],
             mapping = aes(x, y, fill = FTLE)) +
  labs(x="", y="") +
  scale_fill_gradientn(colors = viridis(option = "plasma", n = 40,
                                       alpha = ftle_alpha(40, 8, 8)),
                      values = scales::rescale(
                        c(0,as.numeric(qSumFTLE120$Means[1:9]))),
                      limits=c(0, .8),
                      breaks = c(0,.4,.8),
                      oob=scales::squish,
                      na.value = NA,
                      name = expression(atop( bold('FTLE (120 hr)'),
                                             bold(day^bold({bold("-1")})))))) +

# add 100m and 200m contours
ggplot2::geom_contour(data = bf_Sub,
                     mapping = aes(x=x, y=y, z=z),
                     breaks=c(-100),
                     size=c(0.4),
                     colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
                       breaks = c(-100), show.legend = FALSE,
                       size = 2.5, alpha = .8, nudge_y = -.002,

```

```

        color="lightgrey") +
ggplot2::geom_contour(data = bf_Sub,
  mapping = aes(x=x, y=y, z=z),
  breaks=c(-200),
  size=c(0.4),
  colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
  breaks = c(-200),
  label.placement = label_placement_n(5),
  show.legend = FALSE, size = 2.5, alpha = .8,
  nudge_y = -.002, color="lightgrey") +
geom_point(data = data_Sub,
  mapping = aes(Long,Lat, color=KrillDensity_MeanNonZero ),
  size=5,shape=15, show.legend = TRUE,
  alpha = 1, fill="black") +
geom_label(data = lineSum[c(1,length(lineSum$ColumnID) )],
  mapping = aes(Long,Lat),label=c("Start","End"),
  size=2.5,nudge_y = .02)+
scale_color_gradient(name = expression(atop(bold("Krill Density"),
  bold(log[e](g/m^3) ))),
  low="white",
  high="magenta",
  na.value = "white",
  limits = c(0,4),
  breaks=seq(0,5,1),
  oob=scales::squish) +
geom_sf(data = world, size=1.25,fill="lightgrey",color="grey") +
coord_sf(xlim = c(min(d_Sub_Line$Long,na.rm = TRUE)-degreeBuffer,
  max(d_Sub_Line$Long,na.rm = TRUE)+degreeBuffer),
  ylim = c(min(d_Sub_Line$Lat,na.rm = TRUE)-0.35,
  max(d_Sub_Line$Lat,na.rm = TRUE)+0.35),
  expand = FALSE) +
theme_classic() +
annotation_scale(location = "br", width_hint = 0.5,
  text_face = "bold", text_col = "grey") +
annotation_north_arrow(location = "br",
  which_north = "true",
  pad_x = unit(0.25, "in"),
  pad_y = unit(0.25, "in"),
  style = north_arrow_fancy_orienteering(line_col = "grey",
  fill = c("white",
  "black"),
  text_col = "grey")) +
xlab("Longitude") +
ylab("Latitude") +
guides(color = guide_colourbar(order=1,direction = 'horizontal' ,
  title.position = "top",
  title.hjust = .5,
  label.position = "bottom"),
  fill = guide_colourbar(order=2,direction = 'horizontal',
  title.position = "top",
  title.hjust = .5,
  label.position = "bottom")) +

```

```

theme(panel.grid.major = element_line(color = gray(.5),
                                     linetype = "dashed", size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      axis.text = element_text(face="bold", size=10),
      axis.title = element_blank(),
      legend.text = element_text(face="bold", size=10),
      legend.position = c(.125,.2),
      legend.box = 'vertical',
      legend.title = element_text(face="bold", size=12),
      legend.box.background = element_rect(color="white",
                                           size=2.5, fill="white"),
      legend.spacing.y = unit(0.025, "cm"),
      # adds a buffer on the sides so the plot is centered
      plot.margin=margin(t = 0.5, r = 1, b = 0, l = 0.25, "cm"),
      plot.title = element_text(face="bold", size=24, hjust = 0.5),
      plot.subtitle = element_text(face="bold", size=20, hjust = 0.5))

# 240 ####
p240 <- ggplot()+
  geom_sf(data = world,color=NA,fill=NA) +
  geom_raster(data = FTLE_df240[complete.cases(FTLE_df240),],
             mapping = aes(x, y, fill = FTLE)) +
  labs(x="", y="") +
  scale_fill_gradientn(colors = viridis(option = "plasma", n = 40,
                                       alpha = ftle_alpha(40, 8, 8)),
                      values = scales::rescale(
                        c(0,as.numeric(qSumFTLE240$Means[1:9]))),
                      limits=c(0, .6),
                      breaks = c(0,.3,.6),
                      oob=scales::squish,
                      na.value = NA,
                      name = expression(atop( bold('FTLE (240 hr)'),
                                             bold(day^bold({bold("-1")})))))) +

# add 100m and 200m contours
ggplot2::geom_contour(data = bf_Sub,
                     mapping = aes(x=x, y=y, z=z),
                     breaks=c(-100),
                     size=c(0.4),
                     colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
                       breaks = c(-100), show.legend = FALSE,
                       size = 2.5, alpha = .8, nudge_y = -.002,
                       color="lightgrey") +
ggplot2::geom_contour(data = bf_Sub,
                     mapping = aes(x=x, y=y, z=z),
                     breaks=c(-200),
                     size=c(0.4),
                     colour="darkgray", show.legend = FALSE) +
metR::geom_text_contour(data = bf_Sub, mapping = aes(x=x,y=y,z=z),
                       breaks = c(-200),
                       label.placement = label_placement_n(5),
                       show.legend = FALSE, size = 2.5, alpha = .8,
                       nudge_y = -.002, color="lightgrey") +

```

```

geom_point(data = data_Sub,
  mapping = aes(Long,Lat, color=KrillDensity_MeanNonZero ),
  size=5,shape=15, show.legend = TRUE,
  alpha = 1, fill="black") +
geom_label(data = lineSum[c(1,length(lineSum$ColumnID) )],
  mapping = aes(Long,Lat),label=c("Start","End"),
  size=2.5,nudge_y = .02)+
scale_color_gradient(name = expression(atop(bold("Krill Density"),
  bold(log[e](g/m^3)) )),
  low="white",
  high="magenta",
  na.value = "white",
  limits = c(0,4),
  breaks=seq(0,5,1),
  oob=scales::squish) +
geom_sf(data = world, size=1.25,fill="lightgrey",color="grey") +
coord_sf(xlim = c(min(d_Sub_Line$Long,na.rm = TRUE)-degreeBuffer,
  max(d_Sub_Line$Long,na.rm = TRUE)+degreeBuffer),
  ylim = c(min(d_Sub_Line$Lat,na.rm = TRUE)-0.35,
  max(d_Sub_Line$Lat,na.rm = TRUE)+0.35),
  expand = FALSE) +
theme_classic() +
annotation_scale(location = "br", width_hint = 0.5,
  text_face = "bold", text_col = "grey") +
annotation_north_arrow(location = "br",
  which_north = "true",
  pad_x = unit(0.25, "in"),
  pad_y = unit(0.25, "in"),
  style = north_arrow_fancy_orienteering(line_col = "grey",
  fill = c("white",
  "black"),
  text_col = "grey")) +
xlab("Longitude") +
ylab("Latitude") +
guides(color = guide_colourbar(order=1,direction = 'horizontal' ,
  title.position = "top",
  title.hjust = .5,
  label.position = "bottom"),
  fill = guide_colourbar(order=2,direction = 'horizontal',
  title.position = "top",
  title.hjust = .5,
  label.position = "bottom")) +
theme(panel.grid.major = element_line(color = gray(.5),
  linetype = "dashed", size = 0.5),
  panel.background = element_rect(fill = "aliceblue"),
  axis.text = element_text(face="bold", size=10),
  axis.title = element_blank(),
  legend.text = element_text(face="bold", size=10),
  legend.position = c(.125,.2),
  legend.box = 'vertical',
  legend.title = element_text(face="bold", size=12),
  legend.box.background = element_rect(color="white",
  size=2.5, fill="white"),

```

```

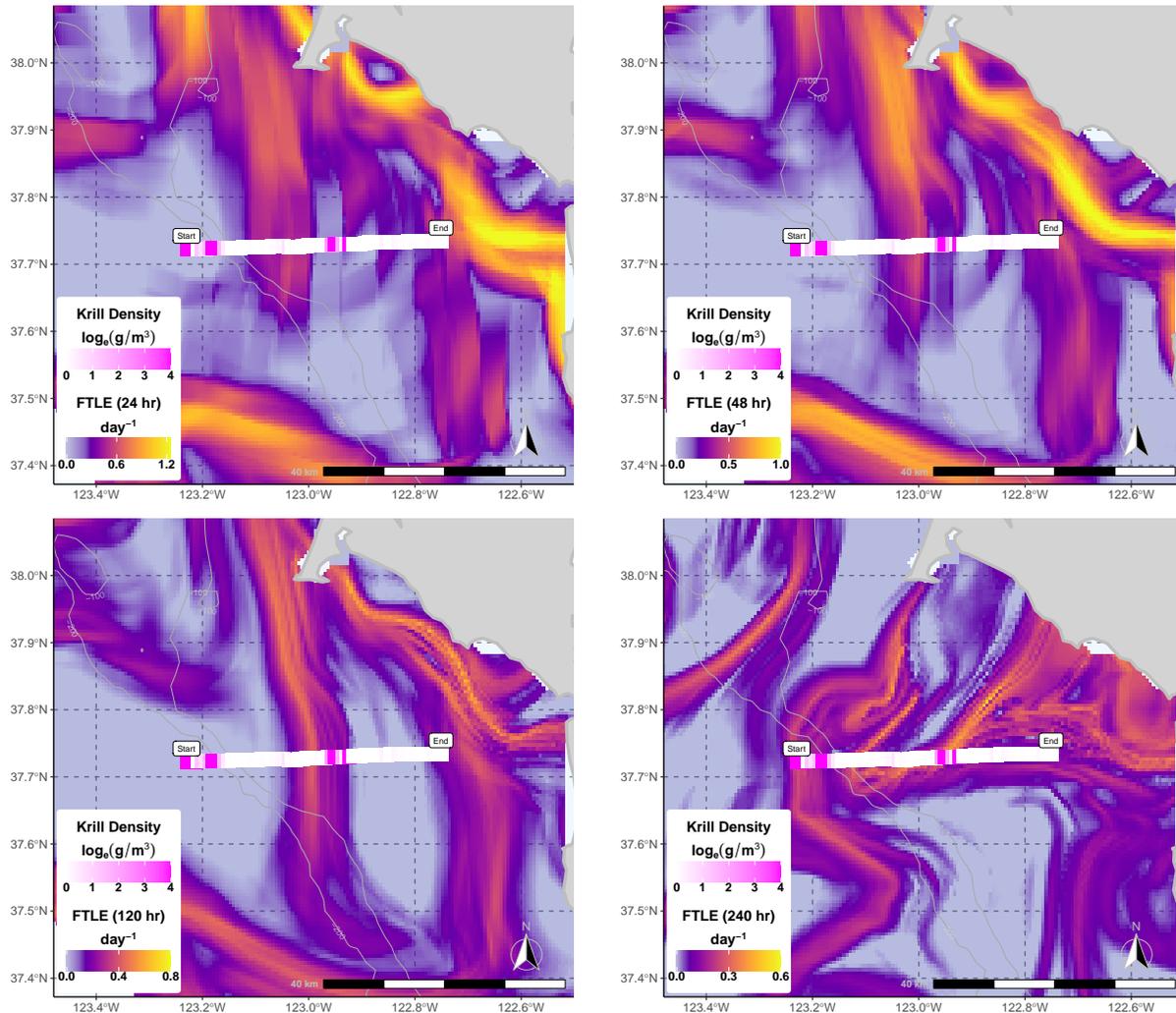
legend.spacing.y = unit(0.025, "cm"),
# adds a buffer on the sides so the plot is centered
plot.margin=margin(t = 0.5, r = 1, b = 0, l = 0.25, "cm"),
plot.title = element_text(face="bold", size=24, hjust = 0.5),
plot.subtitle = element_text(face="bold", size=20, hjust = 0.5))

```

```

# Plot All ####
pMapH <- ggarrange(p24,p48,p120,p240,
                  ncol=2, nrow = 2,
                  align = "hv")
pMapH

```



FTLE data is in hourly increments (due to the temporal resolution of the HF Radar surface current measurements), and some survey transect lines spanned multiple FTLE measurement increments (mean = 3.02, 90% have 5 or less layers). To avoid introducing artificial jumps in measurement values at hourly changes, for this analysis we calculated a spatial mean of the FTLE layers for each line.

Krill

Krill acoustic data from the ACCESS Cruises are summarized into 200m long x 5m deep cells. (Note: The first 5m are not analyzed, so the top of the most shallow cell in any column is 5m.) To match the spatial resolution of the FTLE data, we combine acoustic grid cells into 600m Columns and calculate several summary statistics of each column for analysis. These include:

- Maximum Krill Density (max in any cell within a column)
- Mean Density of Cells with Krill
- Geometric Mean Density of Cells with Krill ($10^{(\text{mean}(\log_{10}(\text{Krill Density})))}$)

Here we see the distribution of krill in relation the FTLE along the transect line for all 4 integration times. Note: this is the same transect line shown in the figure above.

```
# Create a plot showing the acoustics data and how they were processed
c = "ACC1207"
l = "6"

markerSize <- 5.5
markerBump <- 0.021
# Subset 600m resolution acoustics data for this cruise/line
data_Sub <- dataGamm %>%
  dplyr::filter(Cruise == c, Line == l)
# Subset CTD data for this cruise/line
cast_Sub <- dataCTD %>%
  dplyr::filter(Cruise == c, Line == l)
# Subset full resolution acoustics data for this cruise/line
d_Sub_Line <- acoustics_DF_FTLE_W600 %>%
  dplyr::filter(Cruise == c, Line == l)
# Create a column summary for this line
lineSum <- d_Sub_Line %>%
  mutate(minCID = min(as.integer(d_Sub_Line$ColumnID))) %>%
  group_by(ColumnID) %>%
  summarise(dttz = first(dttz),
            Lat = first(Lat),
            Long = first(Long),
            ftle24=first(ftle24_L),
            ftle48=first(ftle48_L),
            ftle120=first(ftle120_L),
            ftle240=first(ftle240_L),
            botDepth = first(BotDepth),
            KrillBiomass = sum(KrillBiomass),
            numWith = sum(KrillBiomass>0),
            density = if_else(numWith > 0, mean(KrillDensity[KrillDensity>0]),0),
            logDensity = if_else(density==0, -8,log10(density)),
            x = (as.integer(first(ColumnID))-first(minCID))*200)
#### Transect Plot ####
# Create a matrix of the krill data
x = matrix(nrow = (2+length(unique(as.integer(d_Sub_Line$CellID)))),
           ncol = length(unique(as.integer(d_Sub_Line$ColumnID))),
           byrow=FALSE)
for(ii in 1:length(unique(as.integer(d_Sub_Line$ColumnID)))){
  numNA <-
  nrow(x) -
  (2+max(as.integer(d_Sub_Line$CellID[d_Sub_Line$ColumnID ==
                                         unique(as.integer(d_Sub_Line$ColumnID))[ii]])))
```

```

if(numNA>0){
  x[,ii] <- c(NA, NA, d_Sub_Line$KrillDensity[d_Sub_Line$ColumnID ==
                                                    unique(as.integer(d_Sub_Line$ColumnID))[ii]],
            rep(NA, numNA))
} else {
  x[,ii] <- c(NA, NA, d_Sub_Line$KrillDensity[d_Sub_Line$ColumnID ==
                                                    unique(as.integer(d_Sub_Line$ColumnID))[ii]],
            rep(NA, numNA))
}
}

# Create a raster from the matrix of krill transect values
r <- raster(ncol=ncol(x),nrow=nrow(x))
raster::values(r) <- x
bb <- extent(0, (ncol(x)-1)*200,(nrow(x)-1)*-5, 0)
extent(r) <- bb
r <- setExtent(r, bb, keepres=TRUE)

rSPDF <- as(r, "SpatialPixelsDataFrame")
r_df <- as.data.frame(rSPDF)
colnames(r_df) <- c("KrillDensity", "x", "y")
r_df <- arrange(r_df, x, -y)
# add x and y back onto d_sub_Line
d_Sub_Line <- cbind(d_Sub_Line,x=r_df$x,y=r_df$y)

d_sub600 <- d_Sub_Line %>%
  group_by(Col600) %>%
  summarize(minX = min(x)-100,
            maxX= max(x)+100,
            minY = min(y)+5, # adjust to represent the top of the column
            maxY= 0,
            Krill = if_else(sum(KrillBiomass)>0,1,0))

cast_SubSum <- cast_Sub %>%
  mutate(x = c(0,4180,7763,28661))

whaleCTD <- d_sub600 %>%
  left_join(dplyr::select(data_Sub,Col600, HUWH, BLWH), by = "Col600" )

ylim.prim <- c(round(min(r_df$y))-10, 0)
ylim.sec <- c(1000,0)

b <- diff(ylim.prim)/diff(ylim.sec)
a <- ylim.prim[1] - b*ylim.sec[1]

# Main Plot

p1 <- ggplot() +
  geom_rect(data=d_sub600, aes(xmin = minX, xmax = maxX, ymin = minY-2.5, ymax = -5,
                              alpha="600m Columns"),color="black", fill="white") +
  geom_rect(data=d_sub600, aes(xmin = minX, xmax = maxX, ymin = minY-2.5, ymax = -5,
                              fill=as.factor(Krill)),color="black") +
  scale_fill_manual(name = NULL, values = c("1"="#b3ffff","0"="white"),
                    labels = c("Krill Present in Column", "No Krill in Column"),
                    guide = guide_legend(order=3,direction = 'vertical')
  )+

```

```

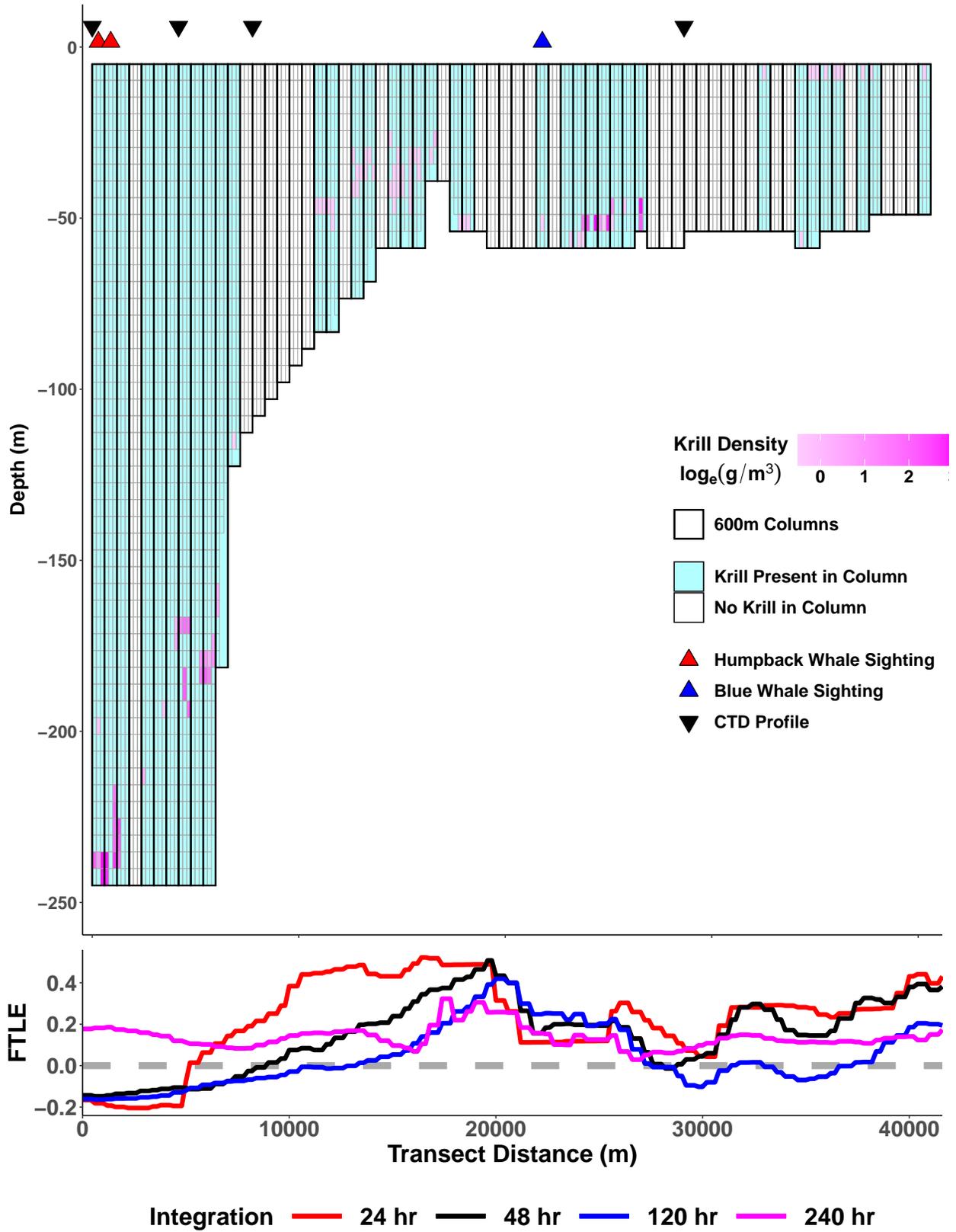
new_scale_fill() +
geom_tile(data=r_df,aes(x=x, y=y+5,fill=log(KrillDensity)),color="darkgrey",
          alpha=0.95) +
scale_fill_gradient(name = expression(atop(bold("Krill Density"),
                                           bold(log[e](g/m^3)) )),
                    low="#FFCCFF",
                    high="magenta",
                    na.value = NA,
                    limits = c(-0.5,3),
                    breaks=seq(0,3,1),
                    oob=scales::squish) +
geom_rect(data=d_sub600, aes(xmin = minX, xmax = maxX, ymin = minY-2.5, ymax = -5,
                             alpha="600m Columns"),color="black",
          fill=NA, size = 0.75) +
scale_alpha_manual(name=NULL, values=c("600m Columns"=0.1)) +
geom_point(data= whaleCTD[whaleCTD$HUWH == 1,],
           aes(y=1.5,x=minX+300, shape = "Humpback Whale Sighting"),
           fill = "red", color="black", size=markerSize)+
geom_point(data= whaleCTD[whaleCTD$BLWH == 1,],
           aes(y=1.5,x=minX+300, shape = "Blue Whale Sighting"),
           fill = "blue", color="black", size=markerSize)+
geom_point(data=cast_SubSum, aes(y=6,x=x, shape = "CTD Profile"),
           fill = "black", color="black", size=markerSize)+
scale_shape_manual(name = NULL, values = c("Humpback Whale Sighting" = 24,
                                           "Blue Whale Sighting" = 24,
                                           "CTD Profile" = 25) )+
scale_y_continuous(limits=c(round(min(r_df$y))-5, 6), expand = c(0.025,0))+
scale_x_continuous(expand = c(0.0115,0.00005),limits=c(-1,(max(r_df$x))-800)
                  )+
xlab("Transect Distance (km)")+ ylab("Depth (m)")+
guides(
  fill = guide_colourbar(order=1,direction = 'horizontal',
                          title.position = "left",
                          title.hjust = .5,label.position = "bottom"),
  alpha = guide_legend(order=2,direction = 'vertical'
                       ),
  shape = guide_legend(order=4,direction = 'vertical',
                       override.aes = list(fill = c("red", "blue", "black")))) +
theme_classic()+
theme(legend.position= c(.85,.375),
      legend.direction = "horizontal",
      legend.title = element_text(face="bold", size=18),
      legend.text = element_text(face="bold", size=16),
      legend.key.size = unit(1, "cm"),
      legend.spacing.y= unit(0.15, "cm"),
      plot.title = element_text(face="bold", size=24,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=22,hjust = 0.5),
      plot.caption = element_text( face="bold", size=18,hjust = 0),
      axis.title = element_text(face="bold", size=18),
      axis.text = element_text(face="bold", size=16),
      axis.title.x = element_blank(),
      axis.text.x = element_blank()
)

```

```

p2<- ggplot()+
  geom_hline(yintercept=0, linetype = 'dashed', size=3,color="darkgrey")+
  geom_line(data = lineSum,aes(x=x,y=ftle24,color="24 hr"),size=2)+
  geom_line(data = lineSum,aes(x=x,y=ftle48,color="48 hr"),size=2)+
  geom_line(data = lineSum,aes(x=x,y=ftle120,color="120 hr"),size=2)+
  geom_line(data = lineSum,aes(x=x,y=ftle240,color="240 hr"),size=2)+
  scale_color_manual(name= "Integration",
                     breaks = c("24 hr", "48 hr", "120 hr","240 hr"),
                     values = c("red", "black", "blue", "magenta"))+
  xlab("Transect Distance (m)")+ ylab("FTLE")+
  scale_x_continuous(expand = c(0,0))+
  theme_classic()+
  theme(legend.position="bottom",
        legend.title = element_text(face="bold", size=22),
        legend.text = element_text(face="bold", size=22),
        legend.key.size = unit(2, "cm"),
        plot.title = element_text(face="bold", size=24,hjust = 0.5),
        plot.subtitle = element_text( face="bold", size=22,hjust = 0.5),
        plot.caption = element_text( face="bold", size=18,hjust = 0),
        axis.title = element_text(face="bold", size=22),
        axis.text = element_text(face="bold", size=20),
        axis.title.y.right = element_text(face="bold", size=18, color="grey"),
        axis.text.y.right = element_text(face="bold", size=10, color="grey"))
tP<-ggarrange(p1,p2,nrow=2,align = c("hv"), heights = c(.75,.25) )
tP

```



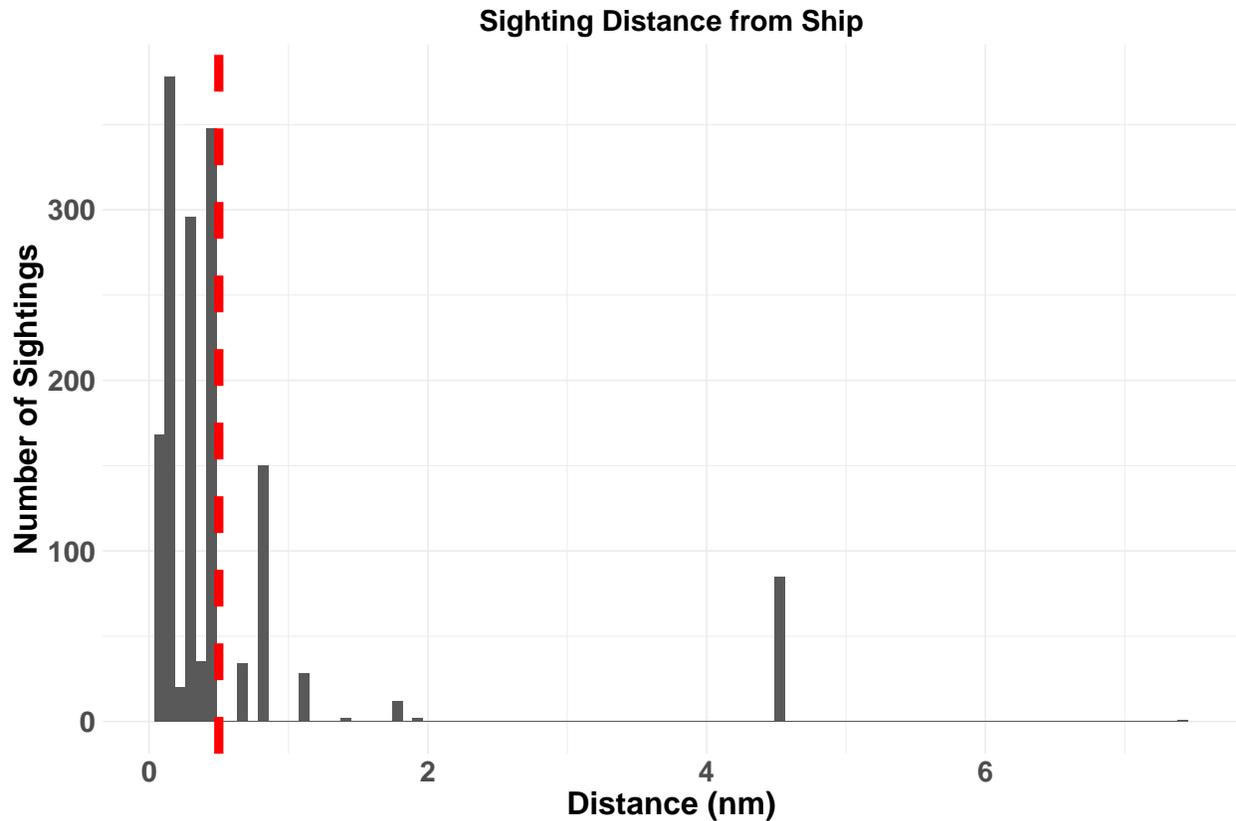
Cetacean Data

In conjunction with the krill transect surveys, cetacean sightings were also recorded. Due to the nature of the transects, sightings were not investigated further (i.e. the ship did not break off the line to pursue groups of whales) resulting in a large number of unidentified whales. For this reason, we excluded outliers in distance from ship to reduce potential bias in location accuracy of sightings (i.e. $> .5$ nm from ship) and excluded sightings where the species could not be verified. Additionally, due to the low sample size of Gray, and Minke whales, they are excluded from our analysis. **80% of sightings were between 0 and 0.5 nm from the ship.**

```
cetacean_DF_FTLE %>%
  dplyr::filter(Type == "ACC") %>%
  mutate(Distance = if_else(DisFrShipNm <= 0.5, "0-0.5 nm", "0.5+ nm")) %>%
  group_by(SpCode, Distance) %>%
  summarise(NumSightings = n())
```

SpCode	Distance	NumSightings
BLWH	0-0.5 nm	140
BLWH	0.5+ nm	37
FIWH	0-0.5 nm	8
GRWH	0-0.5 nm	5
HUWH	0-0.5 nm	750
HUWH	0.5+ nm	139
MIWH	0-0.5 nm	2
UNWH	0-0.5 nm	340
UNWH	0.5+ nm	138

```
ggplot()+
  geom_histogram(data=cetacean_DF_FTLE ,aes(x=DisFrShipNm),bins=100)+
  geom_vline(xintercept = 0.5, color='red',linetype=2,size=3)+
  ggtitle("Sighting Distance from Ship")+
  xlab("Distance (nm)") +
  ylab("Number of Sightings") +
  theme_minimal()+
  theme(axis.title = element_text(face="bold", size=22),
        axis.text = element_text( face="bold", size=20),
        plot.title = element_text(face="bold", size=20,hjust = 0.5),
        plot.subtitle = element_text( face="bold", size=18,hjust = 0.5),
        plot.caption = element_text( face="bold", size=16,hjust = 1),
        legend.text = element_text(face="bold", size=20),
        legend.title = element_text(face="bold", size=24))
```



To examine the relationship between cetacean presence and FTLE, we joined the cetacean sighting data to the 600 meter columns used in the krill analysis (denoting cetacean presence for each species). Columns without cetacean sightings were labeled as absences, and we fitted a logistic regression using the presence/absence for each column as the response.

```

cetacean_Sum <- cetacean_Sub %>%
  group_by(SpCode) %>%
  summarise(NumSightings = n())

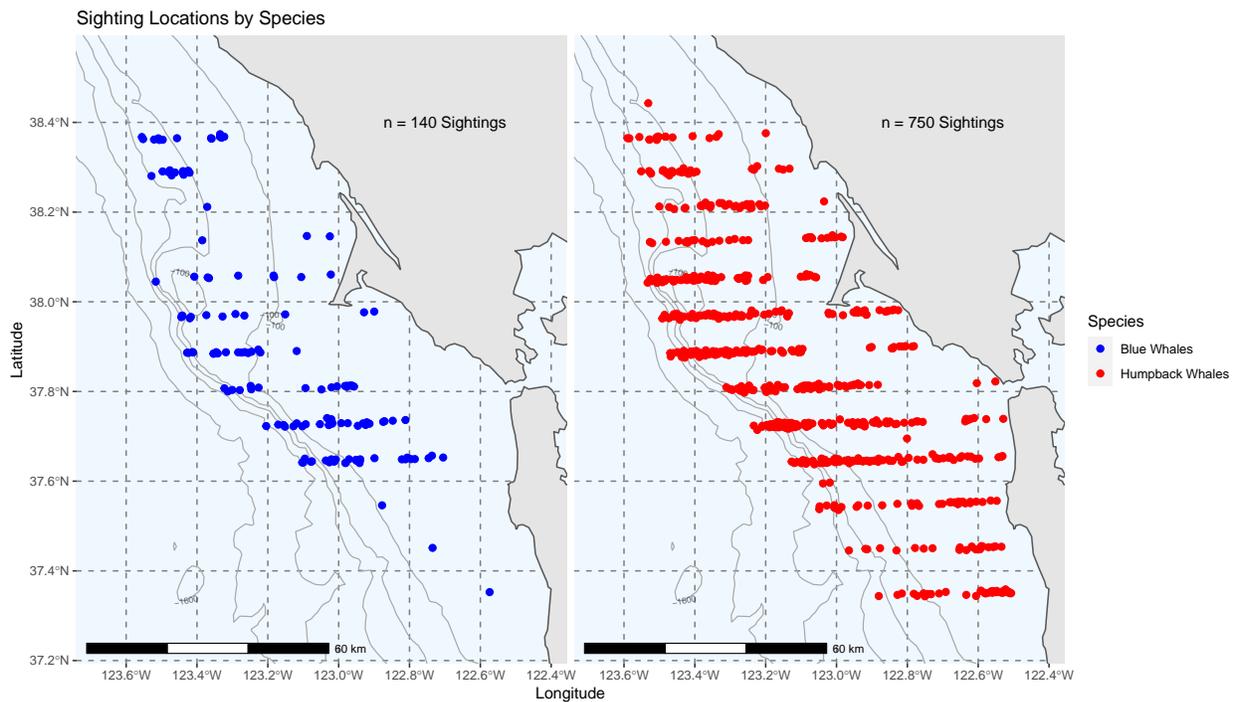
ggplot()+
  geom_sf(data = world) +
  coord_sf(xlim = c(min(cetacean_Sub$SightLong,na.rm = TRUE)-.15,
                      max(cetacean_Sub$SightLong,na.rm = TRUE)+.15),
           ylim = c(min(cetacean_Sub$SightLat,na.rm = TRUE)-.15,
                      max(cetacean_Sub$SightLat,na.rm = TRUE)+.15), expand = FALSE) +
  # add 200m contour
  geom_contour(data = bf,
              aes(x=x, y=y, z=z),
              breaks=c(-100,-200,-400,-800,-1600),
              size=c(0.4),
              colour="darkgrey", show.legend = FALSE) +
  geom_text_contour(data = bf, aes(x=x, y=y,z = z),
                   breaks=c(-100,-200,-400,-800,-1600),
                   show.legend = FALSE, size = 2.2,
                   alpha = .6, nudge_y = -.002) +
  geom_point(data=cetacean_Sub,
            aes(x=SightLong,y=SightLat,color=SpCode, group=SpCode),
            size=2) +

```

```

geom_text(data=cetacean_Sum,
          aes( label = paste0("n = ", NumSightings, " Sightings"), group=SpCode ),
          x=-122.7,y=38.4)+
scale_color_manual(name= "Species", values=c("BLWH"="blue", "HUWH"="red"),
                  labels=c("BLWH"="Blue Whales", "HUWH"="Humpback Whales") ) +
annotation_scale(location = "bl", width_hint = 0.5) +
xlab("Longitude") +
ylab("Latitude") +
ggtitle("Sighting Locations by Species") +
theme(panel.grid.major = element_line(color = gray(.5),
                                     linetype = "dashed", size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      strip.background = element_blank(),
      strip.text.x = element_blank() ) +
facet_grid(~SpCode)

```



Oceanographic Data

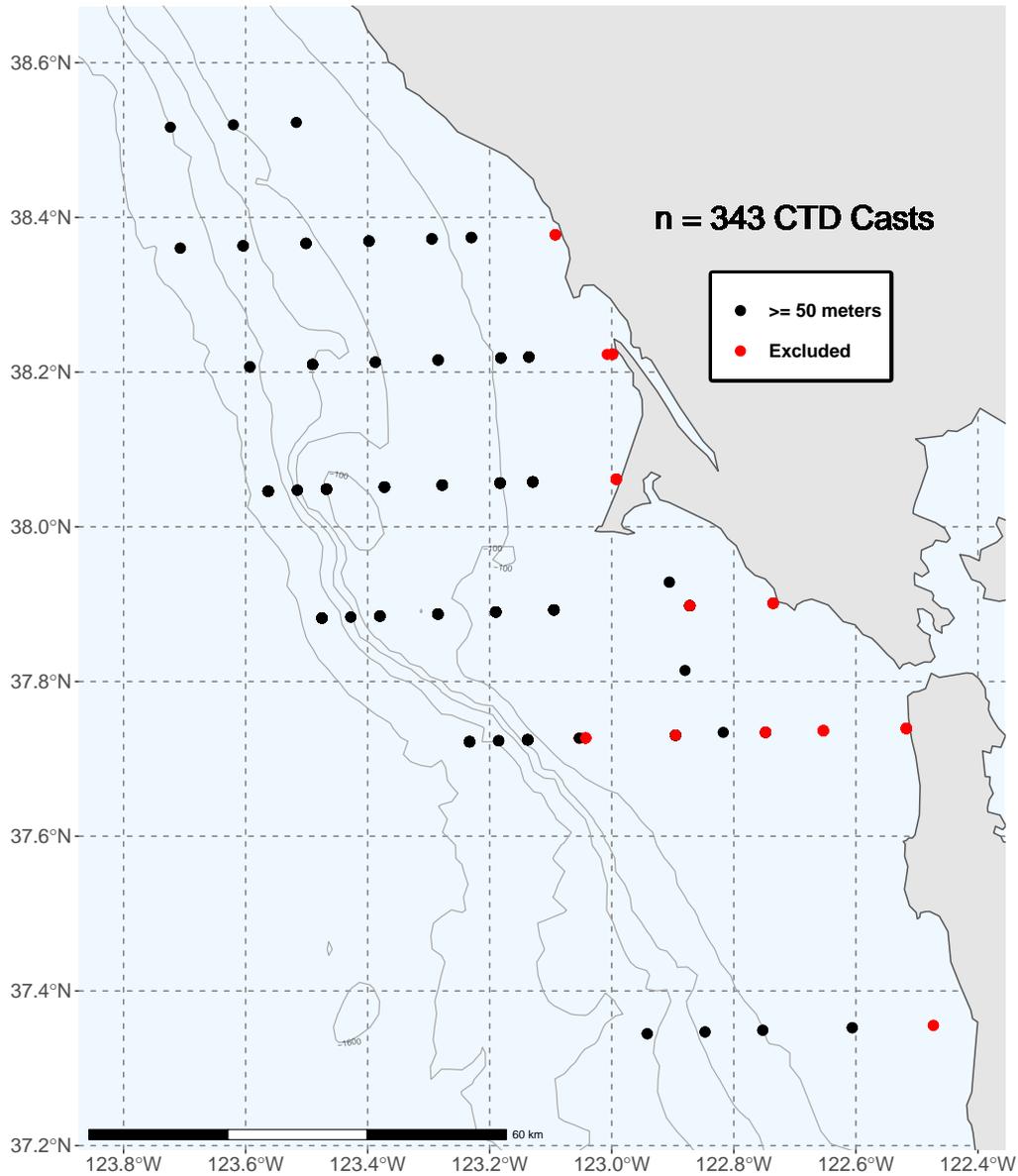
In-situ oceanographic observations were recorded from periodic CTD casts along the krill transect lines. Because the CTD casts were not continuous and were spread out over several nautical miles, we analyzed them separately from the krill data. We extracted the FTLE for the specific time/location of the CTD cast, and calculated several summary metrics from the casts for analysis (Density at 0, 10 20, 30, 40 and 50 meters). To allow for comparability across samples, we excluded CTD casts with a maximum cast depth of less than 50 meters.

```

castSummary_FTLE %>%
mutate(keep = if_else(maxDepth>=50,">= 50 meters","Excluded")) %>%
  ggplot() +
  geom_sf(data = world) +
  coord_sf(xlim = c(min(dataCTD$Long,na.rm = TRUE)-.15,
                    max(dataCTD$Long,na.rm = TRUE)+.25),
          ylim = c(min(dataCTD$Lat,na.rm = TRUE)-.15,
                    max(dataCTD$Lat,na.rm = TRUE)+.15), expand = FALSE) +
# add 200m contour
  geom_contour(data = bf,
              aes(x=x, y=y, z=z),
              breaks=c(-100,-200,-400,-800,-1600),
              size=c(0.4),
              colour="darkgrey", show.legend = FALSE) +
  geom_text_contour(data = bf, aes(x=x, y=y, z = z),
                   breaks=c(-100,-200,-400,-800,-1600),
                   show.legend = FALSE, size = 2.2,
                   alpha = .6, nudge_y = -.002) +
  geom_point(aes(y=Lat,x=Long,color=keep),size=3)+
  geom_text( label = paste0("n = ", length(dataCTD$ID_Date), " CTD Casts"),
            x=-122.7,y=38.4,size=8)+
scale_color_manual(name=NULL,
                  values = c(">= 50 meters" = "black","Excluded" = "red"))+
annotation_scale(location = "bl", width_hint = 0.5) +
xlab("Longitude") +
ylab("Latitude") +
ggtitle("CTD Cast Locations") +
theme(panel.grid.major = element_line(color = gray(.5),
                                       linetype = "dashed", size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      legend.key = element_blank(),
      axis.text = element_text(face="bold", size=14),
      axis.title = element_blank(),
      legend.text = element_text(face="bold", size=13),
      legend.position = c(.78,.72),
      legend.box = 'vertical',
      legend.key.size = unit(1, "cm"),
      legend.title = element_text(face="bold", size=12),
      legend.box.background = element_rect(color="black",
                                           size=2.5, fill="white"),
      plot.margin=margin(t = 0.5, r = 1, b = 0, l = 0.25, "cm"),
      plot.title = element_text(face="bold", size=24, hjust = 0.5),
      plot.subtitle = element_text(face="bold", size=20, hjust = 0.5))

```

CTD Cast Locations



Analysis

Oceanographic Expression of FTLE Features

1. Do surface FTLE features correspond to density differences and if so, to what depth?

GLMM (Seawater Density \sim FTLE) We examined seawater density (Sigma-T) for depths between the surface and 50 meters (at 10 meter increments). Sigma-T (units of kg/m^3) is an estimate of the potential density of seawater, minus $1000 \text{ kg}/\text{m}^3$.

Overall, we found a significant relationship between Seawater Density and FTLE at both the surface and at 10 meters for the 120 hour integration window, and while not significant, the slopes for all other integration windows were positive for the surface and 10 meter depths.

The relationship changes as depth increases beyond 10 meters. For depths of 20-50 meters, the coefficients

for the 24 and 48 hour integration windows are negative. For the 120 and 240 hour integration windows, the coefficients are all positive though not significant, and follow a pattern diminishing slope estimate with increasing depth.

Slopes for all depths followed a similar shape, generally increasing with increasing integration times with a peak at 120-hours and a slight decrease for the 240 hour integration window. Seawater density (Sigma Theta) is comprised of both a temperature and salinity component. These results may indicate colder/denser parcels are found at FTLE features in our study region.

Density at Surface *Results below*

```
## 24
ssSigmaT_GLMM24 <- glmer(SigmaTheta0m ~ ftle24 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle24),
  !is.na(SigmaTheta0m)))

anovassSigmaT_GLMM24 <- car::Anova(ssSigmaT_GLMM24, type=2)

## 48
ssSigmaT_GLMM48 <- glmer(SigmaTheta0m ~ ftle48 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle48),
  !is.na(SigmaTheta0m)))

anovassSigmaT_GLMM48 <- car::Anova(ssSigmaT_GLMM48, type=2)

## 120
ssSigmaT_GLMM120 <- glmer(SigmaTheta0m ~ ftle120 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle120),
  !is.na(SigmaTheta0m)))

anovassSigmaT_GLMM120 <- car::Anova(ssSigmaT_GLMM120, type=2)

## 240
ssSigmaT_GLMM240 <- glmer(SigmaTheta0m ~ ftle240 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle240),
  !is.na(SigmaTheta0m)))

anovassSigmaT_GLMM240 <- car::Anova(ssSigmaT_GLMM240, type=2)

# Results Table
tssSigmaT_GLMM <- data.frame(Model = c(rep("GLMM Sea Surface Sigma Theta",4) ),
  FTLE_Integration = c("24","48","120", "240"),
  Slope = c(round(ssSigmaT_GLMM24@beta[2],4),
  round(ssSigmaT_GLMM48@beta[2],4),
  round(ssSigmaT_GLMM120@beta[2],4),
  round(ssSigmaT_GLMM240@beta[2],4)
  ),
  Intercept = c(round(ssSigmaT_GLMM24@beta[1],4),
  round(ssSigmaT_GLMM48@beta[1],4),
  round(ssSigmaT_GLMM120@beta[1],4),
  round(ssSigmaT_GLMM240@beta[1],4)
  ),
  pValue = c(round(anovassSigmaT_GLMM24$`Pr(>Chisq)`,5),
  round(anovassSigmaT_GLMM48$`Pr(>Chisq)`,5),
```

```

round(anovassSigmaT_GLMM120$`Pr(>Chisq)` ,5),
round(anovassSigmaT_GLMM240$`Pr(>Chisq)` ,5)
)
tssSigmaT_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sea Surface Sigma Theta	24	0.0545	25.1968	0.52495
GLMM Sea Surface Sigma Theta	48	0.1262	25.1935	0.18939
GLMM Sea Surface Sigma Theta	120	0.3704	25.2281	0.00143
GLMM Sea Surface Sigma Theta	240	0.2889	25.2592	0.05069

Density at 10 meters *Results below*

```

## 24
D10_GLMM24 <- glmer(SigmaTheta10m ~ ftle24 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle24),
                                                       !is.na(SigmaTheta10m)))

anovaD10_GLMM24 <- car::Anova(D10_GLMM24, type=2)

## 48
D10_GLMM48 <- glmer(SigmaTheta10m ~ ftle48 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                                       !is.na(SigmaTheta10m)))

anovaD10_GLMM48 <- car::Anova(D10_GLMM48, type=2)

## 120
D10_GLMM120 <- glmer(SigmaTheta10m ~ ftle120 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                                       !is.na(SigmaTheta10m)))

anovaD10_GLMM120 <- car::Anova(D10_GLMM120, type=2)

## 240
D10_GLMM240 <- glmer(SigmaTheta10m ~ ftle240 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                                       !is.na(SigmaTheta10m)))

anovaD10_GLMM240 <- car::Anova(D10_GLMM240, type=2)

# Results Table
tD10_GLMM <- data.frame(Model = c(rep("GLMM Sigma Theta at 10 meters",4) ),
                       FTLE_Integration = c("24", "48", "120", "240"),
                       Slope = c(round(D10_GLMM24@beta[2],4),
                                   round(D10_GLMM48@beta[2],4),
                                   round(D10_GLMM120@beta[2],4),
                                   round(D10_GLMM240@beta[2],4)
                                   ),
                       Intercept = c(round(D10_GLMM24@beta[1],4),
                                      round(D10_GLMM48@beta[1],4),
                                      round(D10_GLMM120@beta[1],4),

```

```

        round(D10_GLMM240@beta[1],4)
    ),
    pValue = c(round(anovaD10_GLMM24$`Pr(>Chisq)` ,5),
              round(anovaD10_GLMM48$`Pr(>Chisq)` ,5),
              round(anovaD10_GLMM120$`Pr(>Chisq)` ,5),
              round(anovaD10_GLMM240$`Pr(>Chisq)` ,5)
    )
)
tD10_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sigma Theta at 10 meters	24	0.0162	25.3555	0.84426
GLMM Sigma Theta at 10 meters	48	0.1176	25.3452	0.19947
GLMM Sigma Theta at 10 meters	120	0.3391	25.3673	0.00154
GLMM Sigma Theta at 10 meters	240	0.2454	25.3987	0.06771

Density at 20 meters *Results below*

```

## 24
D20_GLMM24 <- glmer(SigmaTheta20m ~ ftle24 +
                  (1|Year) + (1|Season) + (1|Line),
                  family = gaussian,
                  data = dataCTD %>% dplyr::filter(!is.na(ftle24),
                                                    !is.na(SigmaTheta20m)))

anovaD20_GLMM24 <- car::Anova(D20_GLMM24, type=2)

## 48
D20_GLMM48 <- glmer(SigmaTheta20m ~ ftle48 +
                  (1|Year) + (1|Season) + (1|Line),
                  family = gaussian,
                  data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                                    !is.na(SigmaTheta20m)))

anovaD20_GLMM48 <- car::Anova(D20_GLMM48, type=2)

## 120
D20_GLMM120 <- glmer(SigmaTheta20m ~ ftle120 +
                  (1|Year) + (1|Season) + (1|Line),
                  family = gaussian,
                  data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                                    !is.na(SigmaTheta20m)))

anovaD20_GLMM120 <- car::Anova(D20_GLMM120, type=2)

## 240
D20_GLMM240 <- glmer(SigmaTheta20m ~ ftle240 +
                  (1|Year) + (1|Season) + (1|Line),
                  family = gaussian,
                  data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                                    !is.na(SigmaTheta20m)))

anovaD20_GLMM240 <- car::Anova(D20_GLMM240, type=2)

# Results Table
tD20_GLMM <- data.frame(Model = c(rep("GLMM Sigma Theta at 20 meters",4) ),
                       FTLE_Integration = c("24", "48", "120", "240"),
                       Slope = c(round(D20_GLMM240@beta[2],4),
                                  round(D20_GLMM48@beta[2],4),
                                  round(D20_GLMM120@beta[2],4),
                                  round(D20_GLMM240@beta[2],4)

```

```

    ),
    Intercept = c(round(D20_GLMM24@beta[1],4),
                  round(D20_GLMM48@beta[1],4),
                  round(D20_GLMM120@beta[1],4),
                  round(D20_GLMM240@beta[1],4)
    ),
    pValue = c(round(anovaD20_GLMM24$`Pr(>Chisq)` ,5),
              round(anovaD20_GLMM48$`Pr(>Chisq)` ,5),
              round(anovaD20_GLMM120$`Pr(>Chisq)` ,5),
              round(anovaD20_GLMM240$`Pr(>Chisq)` ,5)
    )
)
tD20_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sigma Theta at 20 meters	24	-0.1717	25.5993	0.01471
GLMM Sigma Theta at 20 meters	48	-0.0335	25.5727	0.66812
GLMM Sigma Theta at 20 meters	120	0.1635	25.5819	0.06603
GLMM Sigma Theta at 20 meters	240	0.1611	25.6163	0.14203

Density at 30 meters *Results below*

```

## 24
D30_GLMM24 <- glmer(SigmaTheta30m ~ ftle24 +
                   (1|Year) + (1|Season) + (1|Line),
                   family = gaussian,
                   data = dataCTD %>% dplyr::filter(!is.na(ftle24),
                                                    !is.na(SigmaTheta30m)))

anovaD30_GLMM24 <- car::Anova(D30_GLMM24, type=2)

## 48
D30_GLMM48 <- glmer(SigmaTheta30m ~ ftle48 +
                   (1|Year) + (1|Season) + (1|Line),
                   family = gaussian,
                   data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                                    !is.na(SigmaTheta30m)))

anovaD30_GLMM48 <- car::Anova(D30_GLMM48, type=2)

## 120
D30_GLMM120 <- glmer(SigmaTheta30m ~ ftle120 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                                       !is.na(SigmaTheta30m)))

anovaD30_GLMM120 <- car::Anova(D30_GLMM120, type=2)

## 240
D30_GLMM240 <- glmer(SigmaTheta30m ~ ftle240 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                                       !is.na(SigmaTheta30m)))

anovaD30_GLMM240 <- car::Anova(D30_GLMM240, type=2)

# Results Table
tD30_GLMM <- data.frame(Model = c(rep("GLMM Sigma Theta at 30 meters",4) ),
                       FTLE_Integration = c("24","48","120", "240"),
                       Slope = c(round(D30_GLMM24@beta[2],4),

```

```

        round(D30_GLMM48@beta[2],4),
        round(D30_GLMM120@beta[2],4),
        round(D30_GLMM240@beta[2],4)
    ),
    Intercept = c(round(D30_GLMM24@beta[1],4),
                  round(D30_GLMM48@beta[1],4),
                  round(D30_GLMM120@beta[1],4),
                  round(D30_GLMM240@beta[1],4)
    ),
    pValue = c(round(anovaD30_GLMM24$`Pr(>Chisq)` ,5),
               round(anovaD30_GLMM48$`Pr(>Chisq)` ,5),
               round(anovaD30_GLMM120$`Pr(>Chisq)` ,5),
               round(anovaD30_GLMM240$`Pr(>Chisq)` ,5)
    )
)
tD30_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sigma Theta at 30 meters	24	-0.1743	25.7775	0.00438
GLMM Sigma Theta at 30 meters	48	-0.0826	25.7596	0.22734
GLMM Sigma Theta at 30 meters	120	0.0776	25.7667	0.31705
GLMM Sigma Theta at 30 meters	240	0.1415	25.7955	0.15451

Density at 40 meters *Results below*

```

## 24
D40_GLMM24 <- glmer(SigmaTheta40m ~ ftle24 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle24),
                                                       !is.na(SigmaTheta40m)))

anovaD40_GLMM24 <- car::Anova(D40_GLMM24, type=2)
## 48
D40_GLMM48 <- glmer(SigmaTheta40m ~ ftle48 +
                    (1|Year) + (1|Season) + (1|Line),
                    family = gaussian,
                    data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                                       !is.na(SigmaTheta40m)))

anovaD40_GLMM48 <- car::Anova(D40_GLMM48, type=2)
## 120
D40_GLMM120 <- glmer(SigmaTheta40m ~ ftle120 +
                     (1|Year) + (1|Season) + (1|Line),
                     family = gaussian,
                     data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                                         !is.na(SigmaTheta40m)))

anovaD40_GLMM120 <- car::Anova(D40_GLMM120, type=2)
## 240
D40_GLMM240 <- glmer(SigmaTheta40m ~ ftle240 +
                     (1|Year) + (1|Season) + (1|Line),
                     family = gaussian,
                     data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                                         !is.na(SigmaTheta40m)))

anovaD40_GLMM240 <- car::Anova(D40_GLMM240, type=2)
# Results Table

```

```

tD40_GLMM <- data.frame(Model = c(rep("GLMM Sigma Theta at 40 meters",4) ),
  FTLE_Integration = c("24","48","120", "240"),
  Slope = c(round(D40_GLMM24@beta[2],4),
    round(D40_GLMM48@beta[2],4),
    round(D40_GLMM120@beta[2],4),
    round(D40_GLMM240@beta[2],4)
  ),
  Intercept = c(round(D40_GLMM24@beta[1],4),
    round(D40_GLMM48@beta[1],4),
    round(D40_GLMM120@beta[1],4),
    round(D40_GLMM240@beta[1],4)
  ),
  pValue = c(round(anovaD40_GLMM24$`Pr(>Chisq)` ,5),
    round(anovaD40_GLMM48$`Pr(>Chisq)` ,5),
    round(anovaD40_GLMM120$`Pr(>Chisq)` ,5),
    round(anovaD40_GLMM240$`Pr(>Chisq)` ,5)
  )
)
tD40_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sigma Theta at 40 meters	24	-0.1408	25.8791	0.00939
GLMM Sigma Theta at 40 meters	48	-0.0418	25.8598	0.49531
GLMM Sigma Theta at 40 meters	120	0.0912	25.8721	0.19005
GLMM Sigma Theta at 40 meters	240	0.1367	25.8948	0.12618

Density at 50 meters *Results below*

```

## 24
D50_GLMM24 <- glmer(SigmaTheta50m ~ ftle24 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle24),
    !is.na(SigmaTheta50m))

anovaD50_GLMM24 <- car::Anova(D50_GLMM24, type=2)
## 48
D50_GLMM48 <- glmer(SigmaTheta50m ~ ftle48 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle48),
    !is.na(SigmaTheta50m))

anovaD50_GLMM48 <- car::Anova(D50_GLMM48, type=2)
## 120
D50_GLMM120 <- glmer(SigmaTheta50m ~ ftle120 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle120),
    !is.na(SigmaTheta50m))

anovaD50_GLMM120 <- car::Anova(D50_GLMM120, type=2)
## 240
D50_GLMM240 <- glmer(SigmaTheta50m ~ ftle240 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle240),

```

```

!is.na(SigmaTheta50m))
anovaD50_GLMM240 <- car::Anova(D50_GLMM240, type=2)
# Results Table
tD50_GLMM <- data.frame(Model = c(rep("GLMM Sigma Theta at 50 meters",4) ),
  FTLE_Integration = c("24","48","120", "240"),
  Slope = c(round(D50_GLMM24@beta[2],4),
    round(D50_GLMM48@beta[2],4),
    round(D50_GLMM120@beta[2],4),
    round(D50_GLMM240@beta[2],4)
  ),
  Intercept = c(round(D50_GLMM24@beta[1],4),
    round(D50_GLMM48@beta[1],4),
    round(D50_GLMM120@beta[1],4),
    round(D50_GLMM240@beta[1],4)
  ),
  pValue = c(round(anovaD50_GLMM24$`Pr(>Chisq)` ,5),
    round(anovaD50_GLMM48$`Pr(>Chisq)` ,5),
    round(anovaD50_GLMM120$`Pr(>Chisq)` ,5),
    round(anovaD50_GLMM240$`Pr(>Chisq)` ,5)
  )
)
tD50_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sigma Theta at 50 meters	24	-0.1295	25.9706	0.00539
GLMM Sigma Theta at 50 meters	48	-0.0282	25.9507	0.59256
GLMM Sigma Theta at 50 meters	120	0.0674	25.9653	0.26360
GLMM Sigma Theta at 50 meters	240	0.0962	25.9782	0.21549

```

p_ctdDensityAllv2 <- plot_models(D50_GLMM24,D50_GLMM48,
  D50_GLMM120,D50_GLMM240,
  D40_GLMM24,D40_GLMM48,
  D40_GLMM120,D40_GLMM240,
  D30_GLMM24,D30_GLMM48,
  D30_GLMM120,D30_GLMM240,
  D20_GLMM24,D20_GLMM48,
  D20_GLMM120,D20_GLMM240,
  D10_GLMM24,D10_GLMM48,
  D10_GLMM120,D10_GLMM240,
  ssSigmaT_GLMM24,ssSigmaT_GLMM48,
  ssSigmaT_GLMM120,ssSigmaT_GLMM240,
  vline.color = "red",show.p=TRUE,
  show.values=TRUE, spacing = 0.75,
  show.legend = TRUE, dot.size = 1.75,
  value.size = 4.5, line.size = 1,
  axis.labels = c("FTLE (240 Hrs)","FTLE (120 Hrs)",
    "FTLE (48 Hrs)","FTLE (24 Hrs)")) +
scale_y_continuous(breaks=c(-0.4,0,0.4,0.8), limits = c(-0.4,1.4)) +
ggtitle("Supplemental Figure 1") +
theme_minimal() +
theme(axis.title = element_text(face="bold", size=16),
  axis.text = element_text( face="bold", size=16),
  plot.title = element_text(face="bold", size=18,hjust = 0.5),

```

```

plot.subtitle = element_text( face="bold", size=16,hjust = 0.5),
plot.caption = element_text( face="bold", size=12,hjust = 0),
legend.position = c(0.785,0.5),
legend.direction = "vertical",
legend.key.size = unit(0.5,"cm"),
legend.background = element_rect(fill="white",color="black"),
legend.text = element_text(face="bold", size=12),
legend.title = element_text(face="bold", size=14, hjust = 0.5))
p_ctdDensityAllv2$guides$colour$title <- "Response Variable"
p_ctdDensityAllv2$data$group <- factor(x = c(rep("Density at 50 meters",4),
      rep("Density at 40 meters",4),
      rep("Density at 30 meters",4),
      rep("Density at 20 meters",4),
      rep("Density at 10 meters",4),
      rep("Density at Surface",4)),
      levels = c("Density at 50 meters",
        "Density at 40 meters",
        "Density at 30 meters",
        "Density at 20 meters",
        "Density at 10 meters",
        "Density at Surface") )

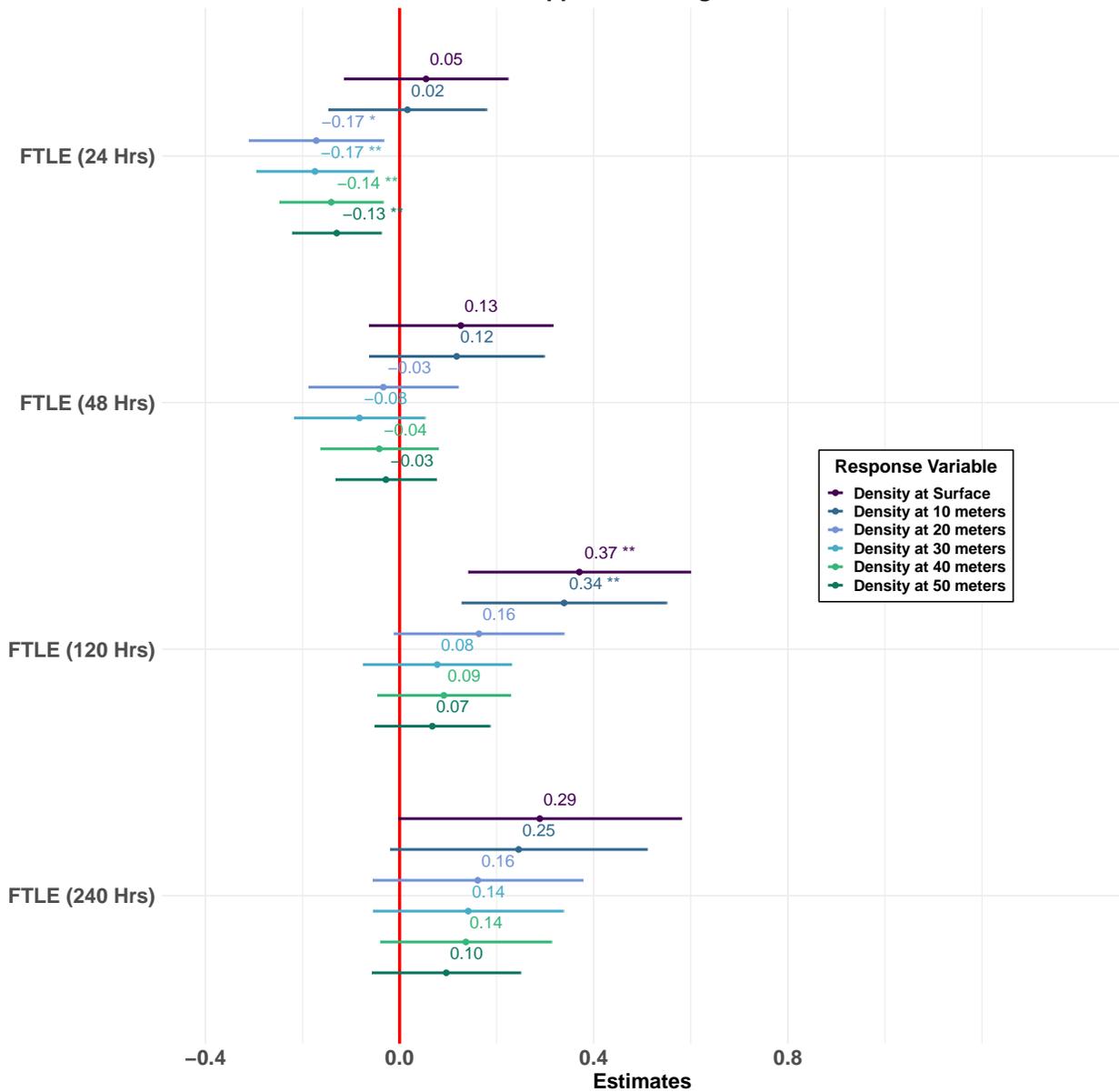
p_ctdDensityAllv2 <- p_ctdDensityAllv2 +
  scale_color_manual(values=c("Density at 50 meters" = "#0B775E",
    "Density at 40 meters" = "#35B779FF",
    "Density at 30 meters" = "#46ACC8",
    "Density at 20 meters" = "#7294D4",
    "Density at 10 meters" = "#31688EFF",
    "Density at Surface" = "#440154FF"

)
)
p_ctdDensityAllv2

```

Supplemental Figure 1

Supplemental Figure 1



```

testSetp_CTD24 <- data.frame(ftle24 = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp_CTD24$fit <- predict(D10_GLMM24,newdata = testSetp_CTD24,
                             type = "response", re.form=-0,se.fit = TRUE)
# Calculate the standard errors of the predicted values
X24 <- model.matrix(~ ftle24, testSetp_CTD24)
#calculate the covariance matrix of the model
V24 <- vcov(D10_GLMM24, full = TRUE)
# take the square root of the diagonal elements to obtain the standard errors
se24 <- sqrt(diag(X24 %*% V24 %*% t(X24)))
# Convert the Log Standard errors into the Response scale
testSetp_CTD24$lwr <- testSetp_CTD24$fit - 1.96 * se24
    
```

```

testSetp_CTD24$upr <- testSetp_CTD24$fit + 1.96 * se24
testSetp_CTD24$model <- "24 hrs"

testSetp_CTD48 <- data.frame(ftle48 = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp_CTD48$fit <- predict(D10_GLMM48,newdata = testSetp_CTD48,
                             type = "response", re.form=-0,se.fit = TRUE)
# Calculate the standard errors of the predicted values
X48 <- model.matrix(~ ftle48, testSetp_CTD48)
#calculate the covariance matrix of the model
V48 <- vcov(D10_GLMM48, full = TRUE)
# take the square root of the diagonal elements to obtain the standard errors
se48 <- sqrt(diag(X48 %*% V48 %*% t(X48)))
# Convert the Log Standard errors into the Response scale
testSetp_CTD48$lwr <- testSetp_CTD48$fit - 1.96 * se48
testSetp_CTD48$upr <- testSetp_CTD48$fit + 1.96 * se48
testSetp_CTD48$model <- "48 hrs"

testSetp_CTD120 <- data.frame(ftle120 = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp_CTD120$fit <- predict(D10_GLMM120, newdata = testSetp_CTD120,
                              type = "response", re.form=-0,se.fit = TRUE)
# Calculate the standard errors of the predicted values
X120 <- model.matrix(~ ftle120, testSetp_CTD120)
#calculate the covariance matrix of the model
V120 <- vcov(D10_GLMM120, full = TRUE)
# take the square root of the diagonal elements to obtain the standard errors
se120 <- sqrt(diag(X120 %*% V120 %*% t(X120)))
# Convert the Log Standard errors into the Response scale
testSetp_CTD120$lwr <- testSetp_CTD120$fit - 1.96 * se120
testSetp_CTD120$upr <- testSetp_CTD120$fit + 1.96 * se120
testSetp_CTD120$model <- "120 hrs"

testSetp_CTD240 <- data.frame(ftle240 = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp_CTD240$fit <- predict(D10_GLMM240,newdata = testSetp_CTD240,
                              type = "response", re.form=-0,se.fit = TRUE)
# Calculate the standard errors of the predicted values
X240 <- model.matrix(~ ftle240, testSetp_CTD240)
#calculate the covariance matrix of the model
V240 <- vcov(D10_GLMM240, full = TRUE)
# take the square root of the diagonal elements to obtain the standard errors
se240 <- sqrt(diag(X240 %*% V240 %*% t(X240)))

# Convert the Log Standard errors into the Response scale
testSetp_CTD240$lwr <- testSetp_CTD240$fit - 1.96 * se240
testSetp_CTD240$upr <- testSetp_CTD240$fit + 1.96 * se240
testSetp_CTD240$model <- "240 hrs"

ggplot() +
  geom_ribbon(data=testSetp_CTD240,aes(x=ftle240,ymin = lwr, ymax = upr,
                                     color = NULL,fill=model), alpha = .15) +
  geom_line(data=testSetp_CTD240,aes(x=ftle240,y = fit, color=model),

```

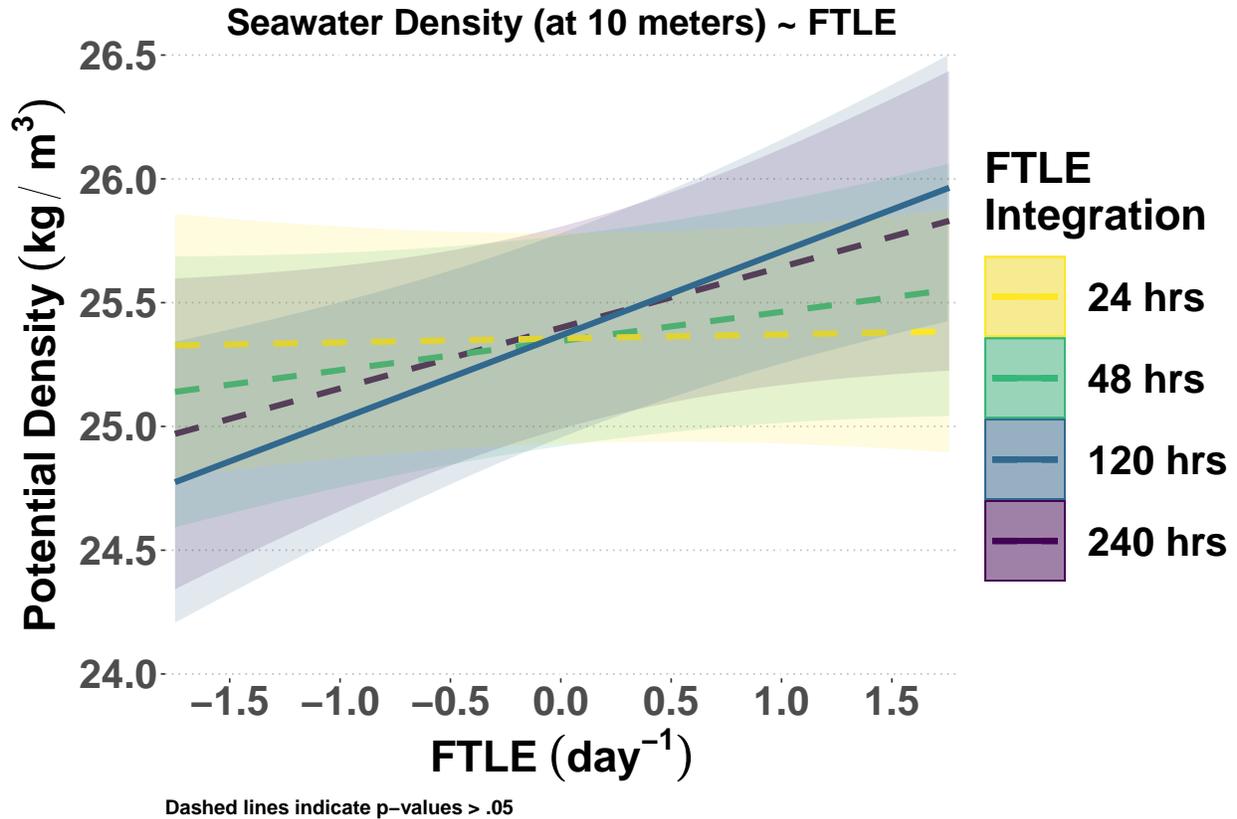
```

      size = 2, linetype=2)+
geom_ribbon(data=testSetp_CTD48,aes(x=ftle48,ymin = lwr, ymax = upr,
      color = NULL,fill=model), alpha = .15) +
geom_line(data=testSetp_CTD48,aes(x=ftle48,y = fit, color=model),
      size = 2, linetype=2)+
geom_ribbon(data=testSetp_CTD24,aes(x=ftle24,ymin = lwr, ymax = upr,
      color = NULL,fill=model), alpha = .15) +
geom_line(data=testSetp_CTD24,aes(x=ftle24,y = fit, color=model),
      size = 2, linetype=2)+
geom_ribbon(data=testSetp_CTD120,aes(x=ftle120,ymin = lwr, ymax = upr,
      color = NULL,fill=model), alpha = .15) +
geom_line(data=testSetp_CTD120,aes(x=ftle120,y = fit, color=model),
      size = 2, linetype=1)+
scale_color_manual(name= "FTLE\nIntegration",
      breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
      values=c( "24 hrs" = "#FDE725FF",
      "48 hrs" = "#35B779FF",
      "120 hrs" = "#31688EFF",
      "240 hrs" = "#440154FF"
      )) +
scale_fill_manual(name= "FTLE\nIntegration",
      breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
      values=c( "24 hrs" = "#FDE725FF",
      "48 hrs" = "#35B779FF",
      "120 hrs" = "#31688EFF",
      "240 hrs" = "#440154FF"
      )) +

scale_x_continuous(breaks=seq(-1.5,2,.5),limits=c(-1.75,1.76),
      expand = c(0.005, 0.025))+
scale_y_continuous(limits=c(24,26.5),
      expand = c(0.005, 0.005)) +
ylab(expression( bold(Potential~Density~(kg/~m^{bold("3")})))) +
xlab(expression( bold(FTLE~(day^{bold({bold("-1")})))) )+
guides(fill = guide_legend(order=1,override.aes = list(alpa=0.05)),
      color = guide_legend(order=1,override.aes = list(alpa=1)))+
labs(caption = "Dashed lines indicate p-values > .05") +
ggtitle("Seawater Density (at 10 meters) ~ FTLE") +
theme_pubclean()+
theme(axis.title = element_text(face="bold", size=30),
      axis.text = element_text( face="bold", size=28),
      plot.title = element_text(face="bold", size=25,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=18,hjust = 0.5),
      plot.caption = element_text( face="bold", size=14,hjust = 0),
      legend.position="right",
      legend.text = element_text(face="bold", size=28),
      legend.title = element_text(face="bold", size=30),
      legend.key.size = unit(2,"cm"),
      strip.text.x = element_text(size = 22, face="bold"))

```

Figure 3A



GLMM (Depth of Sigma Theta ~ FTLE) To further explore the depth of influence of aggregative surface current features, we also fit a linear regression with the depth of potential seawater density (sigma-theta) at 25, 25.5 and 26 kg/m³ as the response variable.

Our analysis of the relationship between FTLE and the depth of specific values of potential seawater density supported the findings above (see Supplemental Figure 2). For example, the depth of the 25.5 kg/m³ isopycnal had a significant negative relationship with FTLE for the 120 and 240-hour integration windows, meaning that there was shoaling in areas of elevated FTLE.

Depth of Sigma Theta 25 *Results below*

```
## 24
Sigma25_GLMM24 <- glmer(Sigma25 ~ ftle24 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle24),
    !is.na(Sigma25)))
anovaSigma25_GLMM24 <- car::Anova(Sigma25_GLMM24, type=2)

## 48
Sigma25_GLMM48 <- glmer(Sigma25 ~ ftle48 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle48),
    !is.na(Sigma25)))
anovaSigma25_GLMM48 <- car::Anova(Sigma25_GLMM48, type=2)
```

```

## 120
Sigma25_GLMM120 <- glmer(Sigma25 ~ ftle120 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle120),
  !is.na(Sigma25)))
anovaSigma25_GLMM120 <- car::Anova(Sigma25_GLMM120, type=2)

## 240
Sigma25_GLMM240 <- glmer(Sigma25 ~ ftle240 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle240),
  !is.na(Sigma25)))
anovaSigma25_GLMM240 <- car::Anova(Sigma25_GLMM240, type=2)
# Results Table
tSigma25_GLMM <- data.frame(Model = c(rep("GLMM Depth of SigmaTheta 25",4) ),
  FTLE_Integration = c("24","48","120", "240"),
  Slope = c(round(Sigma25_GLMM24@beta[2],4),
  round(Sigma25_GLMM48@beta[2],4),
  round(Sigma25_GLMM120@beta[2],4),
  round(Sigma25_GLMM240@beta[2],4)
  ),
  Intercept = c(round(Sigma25_GLMM24@beta[1],4),
  round(Sigma25_GLMM48@beta[1],4),
  round(Sigma25_GLMM120@beta[1],4),
  round(Sigma25_GLMM240@beta[1],4)
  ),
  pValue = c(round(anovaSigma25_GLMM24$`Pr(>Chisq)`^,5),
  round(anovaSigma25_GLMM48$`Pr(>Chisq)`^,5),
  round(anovaSigma25_GLMM120$`Pr(>Chisq)`^,5),
  round(anovaSigma25_GLMM240$`Pr(>Chisq)`^,5)
  )
)
tSigma25_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Depth of SigmaTheta 25	24	0.9432	8.1158	0.59605
GLMM Depth of SigmaTheta 25	48	-0.2272	8.1384	0.90861
GLMM Depth of SigmaTheta 25	120	-1.3907	7.4396	0.49282
GLMM Depth of SigmaTheta 25	240	-3.1431	6.6889	0.19746

Depth of Sigma Theta 25.5 *Results below*

```

## 24
Sigma25_5_GLMM24 <- glmer(Sigma25_5 ~ ftle24 +
  (1|Year) + (1|Season) + (1|Line),
  family = gaussian,
  data = dataCTD %>% dplyr::filter(!is.na(ftle24),
  !is.na(Sigma25_5)))
anovaSigma25_5_GLMM24 <- car::Anova(Sigma25_5_GLMM24, type=2)

## 48
Sigma25_5_GLMM48 <- glmer(Sigma25_5 ~ ftle48 +

```

```

      (1|Year) + (1|Season) + (1|Line),
      family = gaussian,
      data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                       !is.na(Sigma25_5)))
anovaSigma25_5_GLMM48 <- car::Anova(Sigma25_5_GLMM48, type=2)

## 120
Sigma25_5_GLMM120 <- glmer(Sigma25_5 ~ ftle120 +
      (1|Year) + (1|Season) + (1|Line),
      family = gaussian,
      data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                       !is.na(Sigma25_5)))
anovaSigma25_5_GLMM120 <- car::Anova(Sigma25_5_GLMM120, type=2)

## 240
Sigma25_5_GLMM240 <- glmer(Sigma25_5 ~ ftle240 +
      (1|Year) + (1|Season) + (1|Line),
      family = gaussian,
      data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                       !is.na(Sigma25_5)))
anovaSigma25_5_GLMM240 <- car::Anova(Sigma25_5_GLMM240, type=2)
# Results Table
tSigma25_5_GLMM <- data.frame(Model = c(rep("GLMM Depth of SigmaTheta 25.5",4) ),
      FTLE_Integration = c("24","48","120", "240"),
      Slope = c(round(Sigma25_5_GLMM24@beta[2],4),
                round(Sigma25_5_GLMM48@beta[2],4),
                round(Sigma25_5_GLMM120@beta[2],4),
                round(Sigma25_5_GLMM240@beta[2],4)
                ),
      Intercept = c(round(Sigma25_5_GLMM24@beta[1],4),
                    round(Sigma25_5_GLMM48@beta[1],4),
                    round(Sigma25_5_GLMM120@beta[1],4),
                    round(Sigma25_5_GLMM240@beta[1],4)
                    ),
      pValue = c(round(anovaSigma25_5_GLMM24$`Pr(>Chisq)`^,5),
                 round(anovaSigma25_5_GLMM48$`Pr(>Chisq)`^,5),
                 round(anovaSigma25_5_GLMM120$`Pr(>Chisq)`^,5),
                 round(anovaSigma25_5_GLMM240$`Pr(>Chisq)`^,5)
                 )
      )
tSigma25_5_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Depth of SigmaTheta 25.5	24	4.5977	19.4710	0.18006
GLMM Depth of SigmaTheta 25.5	48	-0.6020	20.3434	0.87680
GLMM Depth of SigmaTheta 25.5	120	-10.0798	18.7888	0.02193
GLMM Depth of SigmaTheta 25.5	240	-16.2146	17.6131	0.00341

Depth of Sigma Theta 26 *Results below*

```

## 24
Sigma26_GLMM24 <- glmer(Sigma26 ~ ftle24 +
      (1|Year) + (1|Season) + (1|Line),
      family = gaussian,

```

```

        data = dataCTD %>% dplyr::filter(!is.na(ftle24),
                                          !is.na(Sigma26)))
anovaSigma26_GLMM24 <- car::Anova(Sigma26_GLMM24, type=2)

## 48
Sigma26_GLMM48 <- glmer(Sigma26 ~ ftle48 +
                        (1|Year) + (1|Season) + (1|Line),
                        family = gaussian,
                        data = dataCTD %>% dplyr::filter(!is.na(ftle48),
                                                          !is.na(Sigma26)))
anovaSigma26_GLMM48 <- car::Anova(Sigma26_GLMM48, type=2)

## 120
Sigma26_GLMM120 <- glmer(Sigma26 ~ ftle120 +
                          (1|Year) + (1|Season) + (1|Line),
                          family = gaussian,
                          data = dataCTD %>% dplyr::filter(!is.na(ftle120),
                                                            !is.na(Sigma26)))
anovaSigma26_GLMM120 <- car::Anova(Sigma26_GLMM120, type=2)

## 240
Sigma26_GLMM240 <- glmer(Sigma26 ~ ftle240 +
                          (1|Year) + (1|Season) + (1|Line),
                          family = gaussian,
                          data = dataCTD %>% dplyr::filter(!is.na(ftle240),
                                                            !is.na(Sigma26)))
anovaSigma26_GLMM240 <- car::Anova(Sigma26_GLMM240, type=2)
# Results Table
tSigma26_GLMM <- data.frame(Model = c(rep("GLMM Depth of SigmaTheta 26",4) ),
                             FTLE_Integration = c("24","48","120", "240"),
                             Slope = c(round(Sigma26_GLMM24@beta[2],4),
                                         round(Sigma26_GLMM48@beta[2],4),
                                         round(Sigma26_GLMM120@beta[2],4),
                                         round(Sigma26_GLMM240@beta[2],4)
                                         ),
                             Intercept = c(round(Sigma26_GLMM24@beta[1],4),
                                             round(Sigma26_GLMM48@beta[1],4),
                                             round(Sigma26_GLMM120@beta[1],4),
                                             round(Sigma26_GLMM240@beta[1],4)
                                             ),
                             pValue = c(round(anovaSigma26_GLMM24$`Pr(>Chisq)`^,5),
                                         round(anovaSigma26_GLMM48$`Pr(>Chisq)`^,5),
                                         round(anovaSigma26_GLMM120$`Pr(>Chisq)`^,5),
                                         round(anovaSigma26_GLMM240$`Pr(>Chisq)`^,5)
                                         )
                             )
tSigma26_GLMM

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Depth of SigmaTheta 26	24	14.1729	48.5953	0.00564
GLMM Depth of SigmaTheta 26	48	3.8680	50.5918	0.49933
GLMM Depth of SigmaTheta 26	120	-6.0794	50.3184	0.38792
GLMM Depth of SigmaTheta 26	240	-8.9667	49.7587	0.32645

```

p_ctdSigmathetaAll <- plot_models(Sigma26_GLMM24,Sigma26_GLMM48,
                                Sigma26_GLMM120,Sigma26_GLMM240,
                                Sigma25_5_GLMM24,Sigma25_5_GLMM48,
                                Sigma25_5_GLMM120,Sigma25_5_GLMM240,
                                Sigma25_GLMM24,Sigma25_GLMM48,
                                Sigma25_GLMM120,Sigma25_GLMM240,
                                vline.color = "red",show.p=TRUE,
                                show.values=TRUE, spacing = 0.75,
                                show.legend = TRUE, dot.size = 1.75,
                                value.size = 4.5, line.size = 1,
                                axis.labels = c("FTLE (240 Hrs)","FTLE (120 Hrs)",
                                                "FTLE (48 Hrs)","FTLE (24 Hrs)")) +
scale_y_continuous(breaks=c(-25,0,25), limits = c(-27.5,27.5)) +
ggtitle("Supplemental Figure 2") +
theme_minimal() +
theme(axis.title = element_text(face="bold", size=16),
      axis.text = element_text( face="bold", size=16),
      plot.title = element_text(face="bold", size=18,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=16,hjust = 0.5),
      plot.caption = element_text( face="bold", size=12,hjust = 0),
      legend.position = c(0.785,0.5),
      legend.direction = "vertical",
      legend.key.size = unit(0.5,"cm"),
      legend.background = element_rect(fill="white",color="black"),
      legend.text = element_text(face="bold", size=12),
      legend.title = element_text(face="bold", size=14, hjust = 0.5))

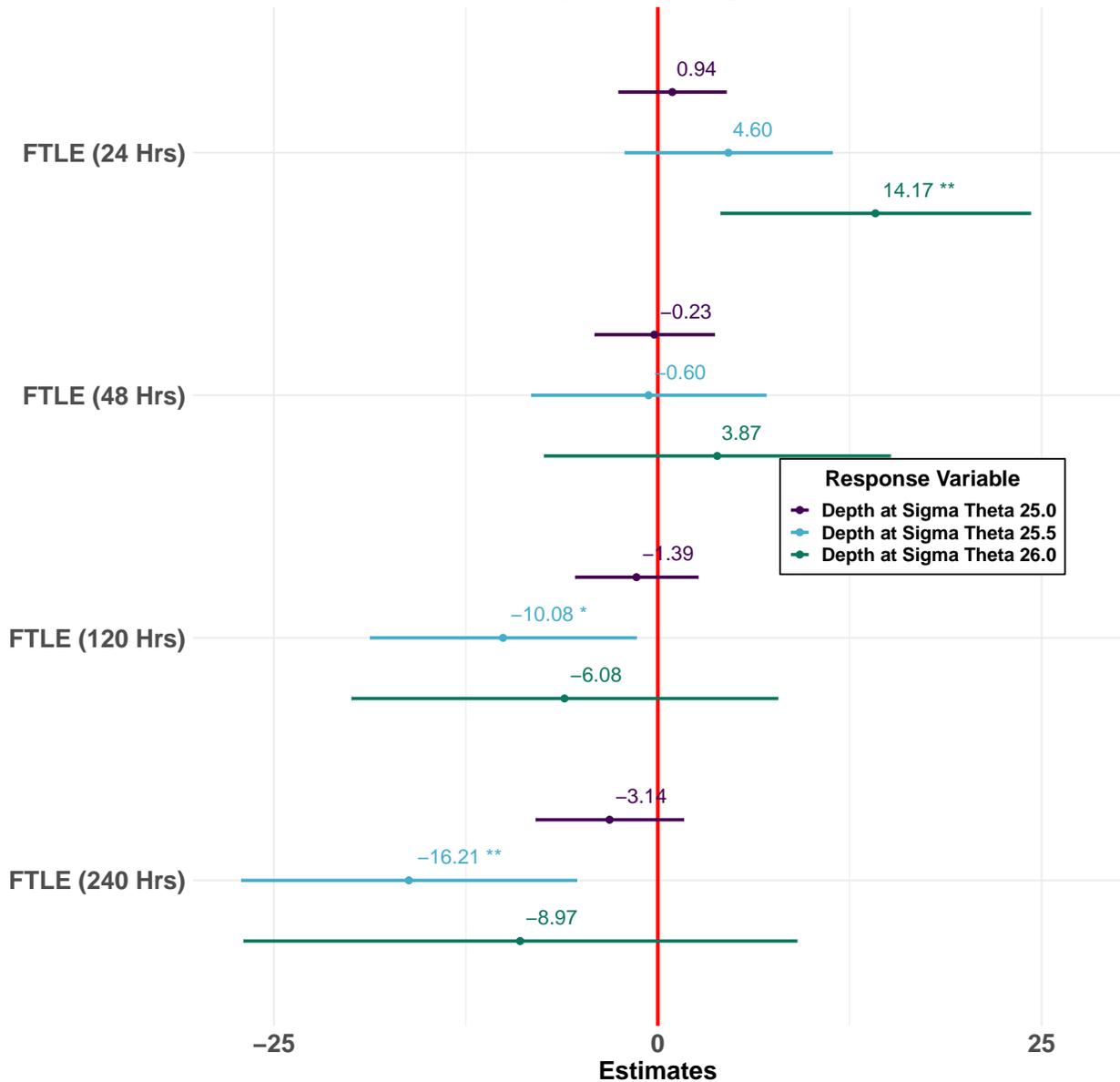
p_ctdSigmathetaAll$guides$colour$title <- "Response Variable"
p_ctdSigmathetaAll$data$group <- factor(x = c(rep("Depth at Sigma Theta 26.0",4),
                                              rep("Depth at Sigma Theta 25.5",4),
                                              rep("Depth at Sigma Theta 25.0",4)),
                                       levels = c("Depth at Sigma Theta 26.0",
                                                "Depth at Sigma Theta 25.5",
                                                "Depth at Sigma Theta 25.0") )

p_ctdSigmathetaAll <- p_ctdSigmathetaAll +
  scale_color_manual(values=c("Depth at Sigma Theta 26.0" = "#0B775E",
                             "Depth at Sigma Theta 25.5" = "#46ACC8",
                             "Depth at Sigma Theta 25.0" = "#440154FF"
                             )
  )
p_ctdSigmathetaAll

```

Supplemental Figure 2

Supplemental Figure 2



FTLE and Krill

Our analysis is centered around the influence of frontal features (areas of elevated FTLE) on krill aggregation (i.e. density). Krill is patchily distributed, which means that there are large, diffuse areas of low density krill, interspersed with areas of increased density. To match the FTLE data resolution, we combined 200 x 5 meter cells into 600 meter columns for analysis.

```
# Only ~ 5% of cells in the full resolution dataset have krill
cat(paste0("Full resolution dataset:\nPercent Cells with Krill = ",
acoustics_DF_FTLE_W600 %>%
  group_by(KrillBiomass>0) %>%
  summarize(Num = n()) %>%
  summarize(PercentWithKrill = round(Num[2]/(Num[1]+Num[2]),3)) %>%
  first()*100,"%" ) )
```

```

# ~44% of cells in the 600 meter dataset have krill
cat(paste0("\n\n600 meter resolution dataset:\nPercent Columns with Krill = ",
dataGamm %>%
  group_by(KrillBiomass>0) %>%
  summarize(Num = n()) %>%
  summarize(PercentWithKrill = round(Num[2]/(Num[1]+Num[2]),3)) %>%
  first()*100,"% " ) )
## Full resolution dataset:
## Percent Cells with Krill = 6.7%
##
## 600 meter resolution dataset:
## Percent Columns with Krill = 50.6%

```

2. Does Krill presence increase with increasing FTLE?

GLMM (Krill Presence ~ FTLE) The relationship between the likelihood of krill presence and FTLE is explored using a linear mixed-effects model. Here we are using a logistic regression to test the probability of krill presence across a range of FTLE values for each of the 4 integration times. We include a correlation structure to address spatial autocorrelation in our data.

- We did not observe a significant positive relationship between FTLE and Krill Presence. However, the trend among the models showed a higher likelihood of krill presence at longer integration times (120 and 240 hour) indicating that krill presence may be influenced by processes at longer time-scales than those included in this study.
- We did observe a significant negative relationship between FTLE and Krill Presence for the 24 hour integration time.

```

# Use previously processed model to save time
if(!exists("krill_glmmPQL_CorExp_24_Hrs")){
  krill_glmmPQL_CorExp_24_Hrs <- glmmPQL(KrillPresence ~ ftle24_L ,
    random = ~ 1 | Year/Season/Line,
    family = binomial(link = "logit"),
    correlation = corExp(form = ~ Long+Lat),
    data = dataGamm %>% dplyr::filter(!is.na(ftle24_L)))

  # Save Model results to a file
  saveRDS(krill_glmmPQL_CorExp_24_Hrs, file = "./Models/krill_glmmPQL_CorExp_24_Hrs.rds")
}

# Use previously processed model to save time
if(!exists("krill_glmmPQL_CorExp_48_Hrs")){
  krill_glmmPQL_CorExp_48_Hrs <- glmmPQL(KrillPresence ~ ftle48_L,
    random = ~ 1 | Year/Season/Line,
    family = binomial(link = "logit"),
    correlation = corExp(form = ~ Long+Lat),
    data = dataGamm %>% dplyr::filter(!is.na(ftle48_L)))

  # Save Model results to a file
  saveRDS(krill_glmmPQL_CorExp_48_Hrs, file = "./Models/krill_glmmPQL_CorExp_48_Hrs.rds")
}

# Use previously processed model to save time
if(!exists("krill_glmmPQL_CorExp_120_Hrs")){
  krill_glmmPQL_CorExp_120_Hrs <- glmmPQL(KrillPresence ~ ftle120_L,
    random = ~ 1 | Year/Season/Line,
    family = binomial(link = "logit"),

```

```

correlation = corExp(form = ~ Long+Lat),
data = dataGamm %>% dplyr::filter(!is.na(ftle120_L))
# Save Model results to a file
saveRDS(krill_glmmPQL_CorExp_120_Hrs, file = "./Models/krill_glmmPQL_CorExp_120_Hrs.rds")
}

# Use previously processed model to save time
if(!exists("krill_glmmPQL_CorExp_240_Hrs")){
krill_glmmPQL_CorExp_240_Hrs <- glmmPQL(KrillPresence ~ ftle240_L,
random = ~ 1 | Year/Season/Line,
family = binomial(link = "logit"),
correlation = corExp(form = ~ Long+Lat),
data = dataGamm %>% dplyr::filter(!is.na(ftle240_L))

# Save Model results to a file
saveRDS(krill_glmmPQL_CorExp_240_Hrs, file = "./Models/krill_glmmPQL_CorExp_240_Hrs.rds")
}

tKrillPres <-
data.frame(
Model = rep("GLMM Krill Presence",4),
FTLE_Integration = c("24","48","120", "240"),
Slope =
c(round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_24_Hrs)))$Value[2],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_48_Hrs)))$Value[2],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_120_Hrs)))$Value[2],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_240_Hrs)))$Value[2],3)
),
Intercept =
c(round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_24_Hrs)))$Value[1],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_48_Hrs)))$Value[1],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_120_Hrs)))$Value[1],3),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_240_Hrs)))$Value[1],3)
),
pValue =
c(round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_24_Hrs)))$'p-value'[2],5),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_48_Hrs)))$'p-value'[2],5),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_120_Hrs)))$'p-value'[2],5),
round(as.data.frame(coef(summary(krill_glmmPQL_CorExp_240_Hrs)))$'p-value'[2],5)
)
)
tKrillPres

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Krill Presence	24	-0.384	0.078	0.01416
GLMM Krill Presence	48	-0.274	0.021	0.13664
GLMM Krill Presence	120	0.263	-0.133	0.29013
GLMM Krill Presence	240	0.285	-0.173	0.36950

```

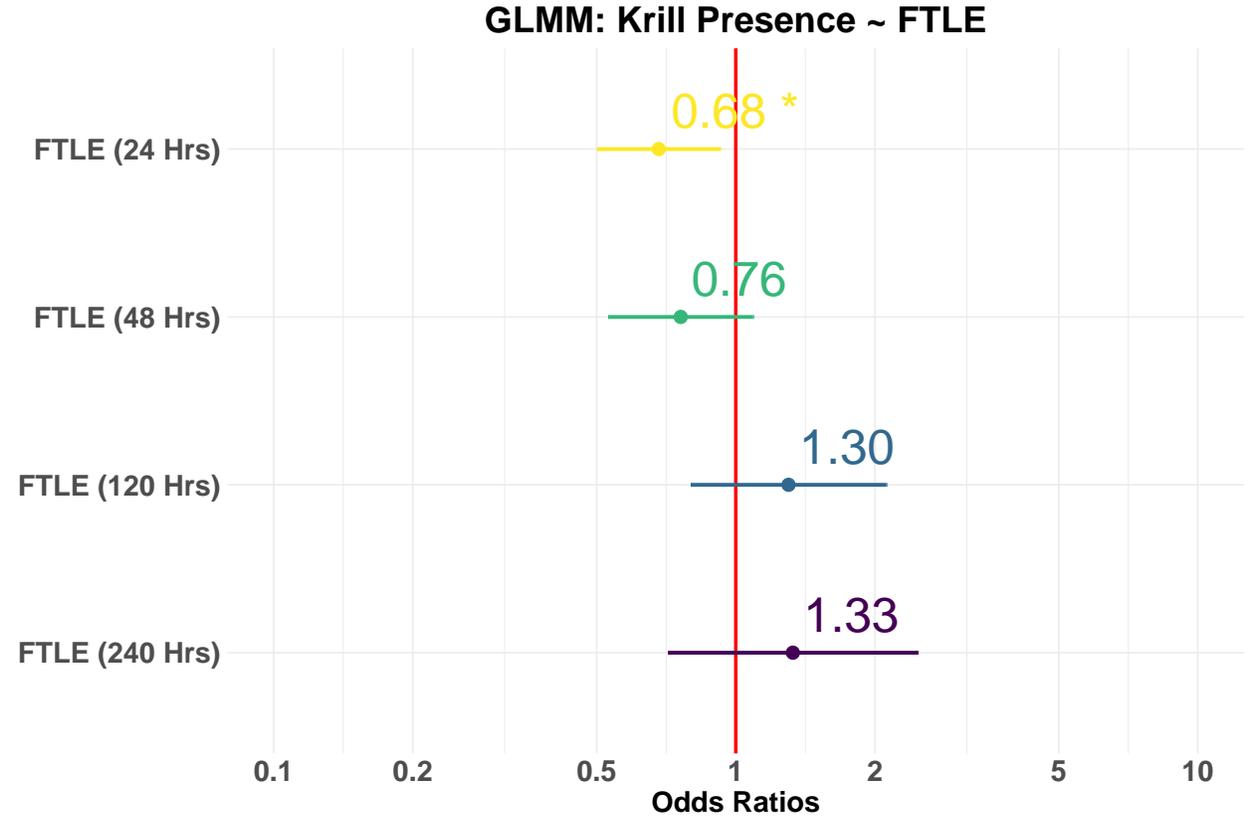
# Plot the models together
plot_models(krill_glmmPQL_CorExp_24_Hrs,krill_glmmPQL_CorExp_48_Hrs,
krill_glmmPQL_CorExp_120_Hrs,krill_glmmPQL_CorExp_240_Hrs,
vline.color = "red",show.p=TRUE,p.shape = FALSE,
show.values=TRUE, show.legend = FALSE,
dot.size = 4, value.size = 12, line.size = 1.25,

```

```

axis.labels = c("FTLE (240 Hrs)", "FTLE (120 Hrs)",
               "FTLE (48 Hrs)", "FTLE (24 Hrs)") +
scale_color_manual(values=c("#440154FF",
                            "#31688EFF",
                            "#35B779FF",
                            "#FDE725FF"
                            )
                  ) +
ggtitle("GLMM: Krill Presence ~ FTLE")+
theme_minimal() +
theme(axis.title = element_text(face="bold", size=20),
      axis.text = element_text( face="bold", size=20),
      plot.title = element_text(face="bold", size=25,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=20,hjust = 0.5),
      plot.caption = element_text( face="bold", size=18,hjust = 0),
      legend.text = element_text(face="bold", size=20),
      legend.title = element_text(face="bold", size=24))

```



3. Does Krill density increase with increasing FTLE?

Here we are modelling the relationship between metrics of Krill Density within a column and FTLE for each of our four Integration lengths (24, 48, 120 and 240 Hours).

Note: for these analyses, we are only testing the influence of FTLE on 600 meter columns that have krill.

GLMM (Krill Density ~ FTLE) For this analysis, we fit generalized linear mixed effects model using the glmmPQL package with FTLE as the independent variable and one of several metrics of Krill Density as the response variable. Krill Density is gamma distributed (positively skewed) in our dataset and we fit the model with a log-link. Our model has Year, Season and Line as random effects to account for the differences we might expect across these variables. We also included a correlation structure to account for spatial autocorrelation in our data.

Geometric Mean Non Zero For every 600 meter column, we calculated the geometric mean krill density for cells within a column containing krill (i.e. For the Geometric mean, Density values were logged prior to taking the mean).

```
## 24
# Use previously processed model to save time
if(!exists("krillDensity_gMNZGLMMPQLLogLink24")){
  krillDensity_gMNZGLMMPQLLogLink24 <-
    glmmPQL(KrillDensity_geoMeanNonZero ~ ftle24_L ,
             random = ~ 1 | Year/Season/Line,
             family = Gamma(link = "log"),
             correlation = corExp(form = ~ Long+Lat),
             data = dataGamm %>% dplyr::filter(!is.na(ftle24_L),
                                               KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_gMNZGLMMPQLLogLink24,
          file = "./Models/krillDensity_gMNZGLMMPQLLogLink24.rds")
}
anovagMNZGLMMPQLLogLink24 <- car::Anova(krillDensity_gMNZGLMMPQLLogLink24, type=2)

## 48
# Use previously processed model to save time
if(!exists("krillDensity_gMNZGLMMPQLLogLink48")){
  krillDensity_gMNZGLMMPQLLogLink48 <-
    glmmPQL(KrillDensity_geoMeanNonZero ~ ftle48_L ,
             random = ~ 1 | Year/Season/Line,
             family = Gamma(link = "log"),
             correlation = corExp(form = ~ Long+Lat),
             data = dataGamm %>% dplyr::filter(!is.na(ftle48_L),
                                               KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_gMNZGLMMPQLLogLink48,
          file = "./Models/krillDensity_gMNZGLMMPQLLogLink48.rds")
}
anovagMNZGLMMPQLLogLink48 <- car::Anova(krillDensity_gMNZGLMMPQLLogLink48, type=2)

## 120
# Use previously processed model to save time
if(!exists("krillDensity_gMNZGLMMPQLLogLink120")){
  krillDensity_gMNZGLMMPQLLogLink120 <-
    glmmPQL(KrillDensity_geoMeanNonZero ~ ftle120_L ,
             random = ~ 1 | Year/Season/Line,
             family = Gamma(link = "log"),
             correlation = corExp(form = ~ Long+Lat),
             data = dataGamm %>% dplyr::filter(!is.na(ftle120_L),
                                               KrillBiomass > 0) )

  # Save Model results to a file
```

```

saveRDS(krillDensity_gMNZGLMMPQLLogLink120,
        file = "./Models/krillDensity_gMNZGLMMPQLLogLink120.rds")
}
anovagMNZGLMMPQLLogLink120 <- car::Anova(krillDensity_gMNZGLMMPQLLogLink120, type=2)

## 240
# Use previously processed model to save time
if(!exists("krillDensity_gMNZGLMMPQLLogLink240")){
  krillDensity_gMNZGLMMPQLLogLink240 <-
    glmpQL(KrillDensity_geoMeanNonZero ~ ftle240_L,
           random = ~ 1 | Year/Season/Line,
           family = Gamma(link = "log"),
           correlation = corGaus(form = ~ Long+Lat),
           data = dataGamm %>% dplyr::filter(!is.na(ftle240_L),
                                           KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_gMNZGLMMPQLLogLink240,
          file = "./Models/krillDensity_gMNZGLMMPQLLogLink240.rds")
}
anovagMNZGLMMPQLLogLink240 <- car::Anova(krillDensity_gMNZGLMMPQLLogLink240, type=2)

tgMNZGLMMPQLLogLink <-
  data.frame(
    Model = c(rep("GLMM Geo-Mean Krill Density",4) ),
    FTLE_Integration = c("24","48","120", "240"),
    Slope = c(round(krillDensity_gMNZGLMMPQLLogLink24$coefficients$fixed[2],4),
              round(krillDensity_gMNZGLMMPQLLogLink48$coefficients$fixed[2],4),
              round(krillDensity_gMNZGLMMPQLLogLink120$coefficients$fixed[2],4),
              round(krillDensity_gMNZGLMMPQLLogLink240$coefficients$fixed[2],4)
              ),
    Intercept = c(round(krillDensity_gMNZGLMMPQLLogLink24$coefficients$fixed[1],4),
                  round(krillDensity_gMNZGLMMPQLLogLink48$coefficients$fixed[1],4),
                  round(krillDensity_gMNZGLMMPQLLogLink120$coefficients$fixed[1],4),
                  round(krillDensity_gMNZGLMMPQLLogLink240$coefficients$fixed[1],4)
                  ),
    pValue = c(round(anovagMNZGLMMPQLLogLink24$`Pr(>Chisq)` ,5),
               round(anovagMNZGLMMPQLLogLink48$`Pr(>Chisq)` ,5),
               round(anovagMNZGLMMPQLLogLink120$`Pr(>Chisq)` ,5),
               round(anovagMNZGLMMPQLLogLink240$`Pr(>Chisq)` ,5)
               )
  )
)
row.names(tgMNZGLMMPQLLogLink) <- NULL
tgMNZGLMMPQLLogLink

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Geo-Mean Krill Density	24	0.2731	0.8225	0.08437
GLMM Geo-Mean Krill Density	48	0.3622	0.8165	0.05616
GLMM Geo-Mean Krill Density	120	0.8218	0.8196	0.00110
GLMM Geo-Mean Krill Density	240	0.9878	0.7580	0.00061

Mean Non Zero For every 600 meter column, we calculated the mean krill density for cells within a column containing krill (i.e. Mean Non-Zero).

```

## 24
# Use previously processed model to save time
if(!exists("krillDensity_MNZGLMMPQLLogLink24")){
  krillDensity_MNZGLMMPQLLogLink24 <-
    glmmPQL(KrillDensity_MeanNonZero ~ ftle24_L ,
            random = ~ 1 | Year/Season/Line,
            family = Gamma(link = "log"),
            correlation = corExp(form = ~ Long+Lat),
            data = dataGamm %>% dplyr::filter(!is.na(ftle24_L),
                                             KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MNZGLMMPQLLogLink24,
          file = "./Models/krillDensity_MNZGLMMPQLLogLink24.rds")
}
anovaMNZGLMMPQLLogLink24 <- car::Anova(krillDensity_MNZGLMMPQLLogLink24, type=2)

## 48
# Use previously processed model to save time
if(!exists("krillDensity_MNZGLMMPQLLogLink48")){
  krillDensity_MNZGLMMPQLLogLink48 <-
    glmmPQL(KrillDensity_MeanNonZero ~ ftle48_L ,
            random = ~ 1 | Year/Season/Line,
            family = Gamma(link = "log"),
            correlation = corExp(form = ~ Long+Lat),
            data = dataGamm %>% dplyr::filter(!is.na(ftle48_L),
                                             KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MNZGLMMPQLLogLink48,
          file = "./Models/krillDensity_MNZGLMMPQLLogLink48.rds")
}
anovaMNZGLMMPQLLogLink48 <- car::Anova(krillDensity_MNZGLMMPQLLogLink48, type=2)

## 120
# Use previously processed model to save time
if(!exists("krillDensity_MNZGLMMPQLLogLink120")){
  krillDensity_MNZGLMMPQLLogLink120 <-
    glmmPQL(KrillDensity_MeanNonZero ~ ftle120_L ,
            random = ~ 1 | Year/Season/Line,
            family = Gamma(link = "log"),
            correlation = corExp(form = ~ Long+Lat),
            data = dataGamm %>% dplyr::filter(!is.na(ftle120_L),
                                             KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MNZGLMMPQLLogLink120,
          file = "./Models/krillDensity_MNZGLMMPQLLogLink120.rds")
}
anovaMNZGLMMPQLLogLink120 <- car::Anova(krillDensity_MNZGLMMPQLLogLink120, type=2)

## 240
# Use previously processed model to save time
if(!exists("krillDensity_MNZGLMMPQLLogLink240")){
  krillDensity_MNZGLMMPQLLogLink240 <-
    glmmPQL(KrillDensity_MeanNonZero ~ ftle240_L ,

```

```

        random = ~ 1 | Year/Season/Line,
        family = Gamma(link = "log"),
        correlation = corExp(form = ~ Long+Lat),
        data = dataGamm %>% dplyr::filter(!is.na(ftle240_L),
                                          KrillBiomass > 0) )

# Save Model results to a file
saveRDS(krillDensity_MNZGLMMPQLLogLink240,
        file = "./Models/krillDensity_MNZGLMMPQLLogLink240.rds")
}
anovaMNZGLMMPQLLogLink240 <- car::Anova(krillDensity_MNZGLMMPQLLogLink240, type=2)

tMNZGLMMPQLLogLink <-
  data.frame(
    Model = c(rep("GLMM Mean Krill Density",4) ),
    FTLE_Integration = c("24","48","120", "240"),
    Slope = c(round(krillDensity_MNZGLMMPQLLogLink240$coefficients$fixed[2],4),
              round(krillDensity_MNZGLMMPQLLogLink48$coefficients$fixed[2],4),
              round(krillDensity_MNZGLMMPQLLogLink120$coefficients$fixed[2],4),
              round(krillDensity_MNZGLMMPQLLogLink240$coefficients$fixed[2],4)
            ),
    Intercept = c(round(krillDensity_MNZGLMMPQLLogLink240$coefficients$fixed[1],4),
                  round(krillDensity_MNZGLMMPQLLogLink48$coefficients$fixed[1],4),
                  round(krillDensity_MNZGLMMPQLLogLink120$coefficients$fixed[1],4),
                  round(krillDensity_MNZGLMMPQLLogLink240$coefficients$fixed[1],4)
                ),
    pValue = c(round(anovaMNZGLMMPQLLogLink240`Pr(>Chisq)` ,5),
               round(anovaMNZGLMMPQLLogLink48`Pr(>Chisq)` ,5),
               round(anovaMNZGLMMPQLLogLink120`Pr(>Chisq)` ,5),
               round(anovaMNZGLMMPQLLogLink240`Pr(>Chisq)` ,5)
            )
  )
)
row.names(tMNZGLMMPQLLogLink) <- NULL
tMNZGLMMPQLLogLink

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Mean Krill Density	24	0.2064	1.1619	0.18788
GLMM Mean Krill Density	48	0.4448	1.1128	0.01664
GLMM Mean Krill Density	120	0.9310	1.1037	0.00016
GLMM Mean Krill Density	240	0.9675	1.1212	0.00248

Max Density For every 600 meter column, we calculated the maximum krill density for cells containing krill.

```

## 24
# Use previously processed model to save time
if(!exists("krillDensity_MaxGLMMPQLLogLink24")){
  krillDensity_MaxGLMMPQLLogLink24 <-
    glmmpQL(KrillDensity_Max ~ ftle24_L ,
            random = ~ 1 | Year/Season/Line,
            family = Gamma(link = "log"),
            correlation = corExp(form = ~ Long+Lat),
            data = dataGamm %>% dplyr::filter(!is.na(ftle24_L),
                                              KrillBiomass > 0) )

# Save Model results to a file

```

```

saveRDS(krillDensity_MaxGLMMPQLLogLink24,
        file = "./Models/krillDensity_MaxGLMMPQLLogLink24.rds")
}
anovaMaxGLMMPQLLogLink24 <- car::Anova(krillDensity_MaxGLMMPQLLogLink24, type=2)

## 48
# Use previously processed model to save time
if(!exists("krillDensity_MaxGLMMPQLLogLink48")){
  krillDensity_MaxGLMMPQLLogLink48 <-
    glmpQL(KrillDensity_Max ~ ftle48_L ,
           random = ~ 1 | Year/Season/Line,
           family = Gamma(link = "log"),
           correlation = corExp(form = ~ Long+Lat),
           data = dataGamm %>% dplyr::filter(!is.na(ftle48_L),
                                           KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MaxGLMMPQLLogLink48,
          file = "./Models/krillDensity_MaxGLMMPQLLogLink48.rds")
}
anovaMaxGLMMPQLLogLink48 <- car::Anova(krillDensity_MaxGLMMPQLLogLink48, type=2)

## 120
# Use previously processed model to save time
if(!exists("krillDensity_MaxGLMMPQLLogLink120")){
  krillDensity_MaxGLMMPQLLogLink120 <-
    glmpQL(KrillDensity_Max ~ ftle120_L ,
           random = ~ 1 | Year/Season/Line,
           family = Gamma(link = "log"),
           correlation = corExp(form = ~ Long+Lat),
           data = dataGamm %>% dplyr::filter(!is.na(ftle120_L),
                                           KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MaxGLMMPQLLogLink120,
          file = "./Models/krillDensity_MaxGLMMPQLLogLink120.rds")
}
anovaMaxGLMMPQLLogLink120 <- car::Anova(krillDensity_MaxGLMMPQLLogLink120, type=2)

## 240
# Use previously processed model to save time
if(!exists("krillDensity_MaxGLMMPQLLogLink240")){
  krillDensity_MaxGLMMPQLLogLink240 <-
    glmpQL(KrillDensity_Max ~ ftle240_L ,
           random = ~ 1 | Year/Season/Line,
           family = Gamma(link = "log"),
           correlation = corExp(form = ~ Long+Lat),
           data = dataGamm %>% dplyr::filter(!is.na(ftle240_L),
                                           KrillBiomass > 0) )

  # Save Model results to a file
  saveRDS(krillDensity_MaxGLMMPQLLogLink240,
          file = "./Models/krillDensity_MaxGLMMPQLLogLink240.rds")
}
anovaMaxGLMMPQLLogLink240 <- car::Anova(krillDensity_MaxGLMMPQLLogLink240, type=2)

```

```

tMaxGLMMPQLLogLink <-
  data.frame(
    Model = c(rep("GLMM Max Krill Density",4) ),
    FTLE_Integration = c("24","48","120", "240"),
    Slope = c(round(krillDensity_MaxGLMMPQLLogLink24$coefficients$fixed[2],4),
              round(krillDensity_MaxGLMMPQLLogLink48$coefficients$fixed[2],4),
              round(krillDensity_MaxGLMMPQLLogLink120$coefficients$fixed[2],4),
              round(krillDensity_MaxGLMMPQLLogLink240$coefficients$fixed[2],4)
    ),
    Intercept = c(round(krillDensity_MaxGLMMPQLLogLink24$coefficients$fixed[1],4),
                  round(krillDensity_MaxGLMMPQLLogLink48$coefficients$fixed[1],4),
                  round(krillDensity_MaxGLMMPQLLogLink120$coefficients$fixed[1],4),
                  round(krillDensity_MaxGLMMPQLLogLink240$coefficients$fixed[1],4)
    ),
    pValue = c(round(anovaMaxGLMMPQLLogLink24$`Pr(>Chisq)` ,5),
               round(anovaMaxGLMMPQLLogLink48$`Pr(>Chisq)` ,5),
               round(anovaMaxGLMMPQLLogLink120$`Pr(>Chisq)` ,5),
               round(anovaMaxGLMMPQLLogLink240$`Pr(>Chisq)` ,5)
    )
  )
row.names(tMaxGLMMPQLLogLink) <- NULL
tMaxGLMMPQLLogLink

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Max Krill Density	24	0.0676	1.9669	0.68640
GLMM Max Krill Density	48	0.4069	1.8880	0.04000
GLMM Max Krill Density	120	0.8840	1.8287	0.00085
GLMM Max Krill Density	240	0.8341	1.8129	0.01589

```

p_DensityAll <- plot_models(krillDensity_MaxGLMMPQLLogLink24,
  krillDensity_MaxGLMMPQLLogLink48,
  krillDensity_MaxGLMMPQLLogLink120,
  krillDensity_MaxGLMMPQLLogLink240,
  krillDensity_MNZGLMMPQLLogLink24,
  krillDensity_MNZGLMMPQLLogLink48,
  krillDensity_MNZGLMMPQLLogLink120,
  krillDensity_MNZGLMMPQLLogLink240,
  krillDensity_gMNZGLMMPQLLogLink24,
  krillDensity_gMNZGLMMPQLLogLink48,
  krillDensity_gMNZGLMMPQLLogLink120,
  krillDensity_gMNZGLMMPQLLogLink240,
  vline.color = "red",show.p=TRUE,
  show.values=TRUE, spacing = .75,
  #axis.lim = c(0.75,7.5),
  show.legend = TRUE, dot.size = 6,
  value.size = 10, line.size = 2,
  axis.labels = c("FTLE (240 Hrs)","FTLE (120 Hrs)",
                  "FTLE (48 Hrs)","FTLE (24 Hrs)") +
  ggtitle("Metrics of Krill Density ~ FTLE")+
  ylim(0.25,10)+
  theme_minimal() +
  theme(axis.title = element_text(face="bold", size=25),
        axis.text = element_text( face="bold", size=25),

```

```

plot.title = element_text(face="bold", size=25,hjust = 0.5),
plot.subtitle = element_text( face="bold", size=20,hjust = 0.5),
plot.caption = element_text( face="bold", size=12,hjust = 0),
legend.position = c(0.72,0.5),
legend.direction = "vertical",
legend.background = element_rect(fill="white",color="black"),
legend.text = element_text(face="bold", size=22),
legend.title = element_text(face="bold", size=25, hjust = 0.5))

p_DensityAll$guides$colour$title <- "Response Variable"
p_DensityAll$data$group <- factor(x = c(rep("Max of Non-Zero Cells",4),
rep("Mean of Non-Zero Cells",4),
rep("GeoMean of Non-Zero Cells",4)),
levels = c( "Max of Non-Zero Cells",
"Mean of Non-Zero Cells",
"GeoMean of Non-Zero Cells" ) )

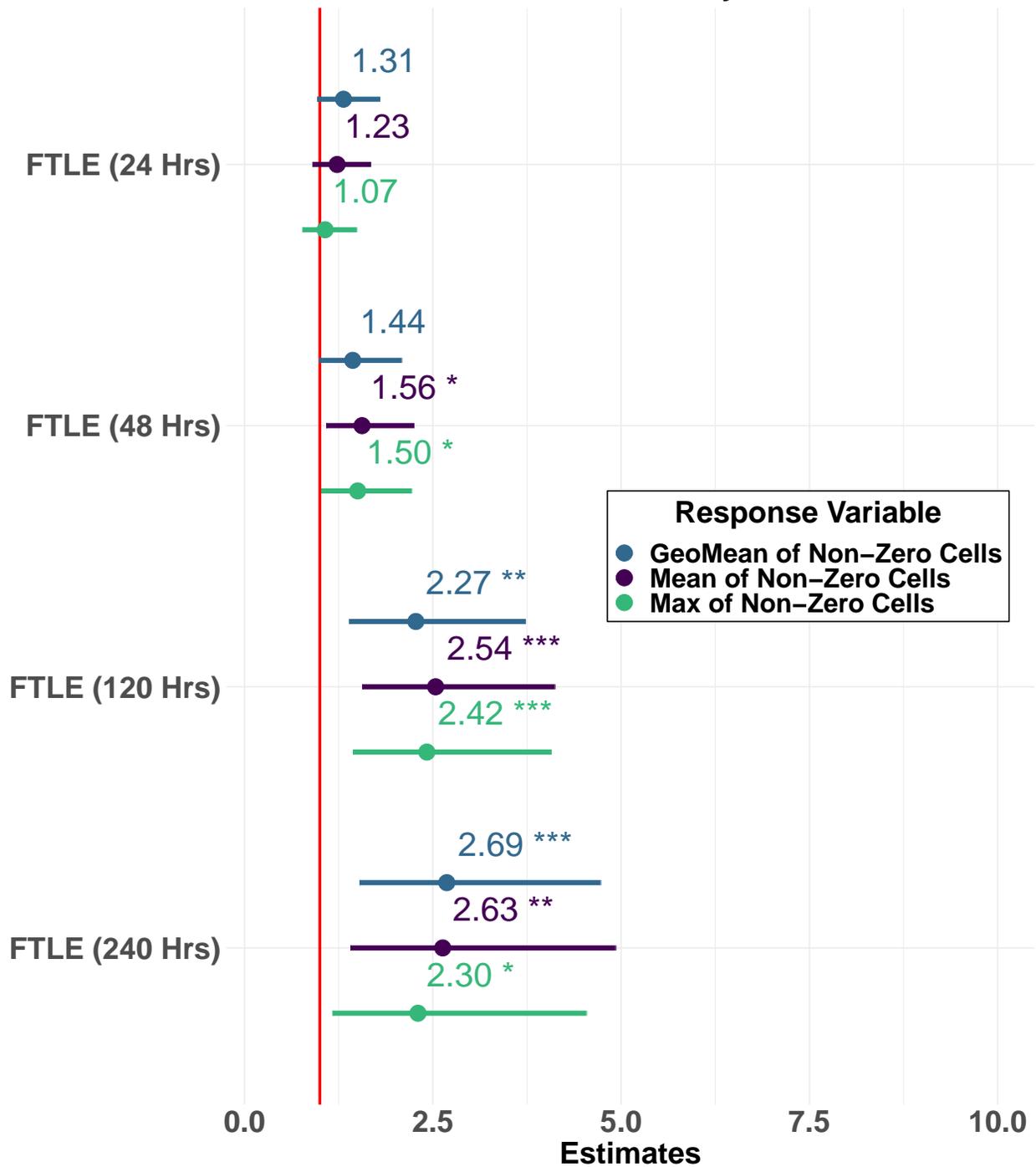
p_DensityAll <- p_DensityAll +
  scale_color_manual(breaks=c("Max of Non-Zero Cells",
"Mean of Non-Zero Cells" ,
"GeoMean of Non-Zero Cells"),
values=c( "Mean of Non-Zero Cells" = "#440154FF",
"GeoMean of Non-Zero Cells" = "#31688EFF",
"Max of Non-Zero Cells" = "#35B779FF"

) )
p_DensityAll

```

Supplemental Figure 3

Metrics of Krill Density ~ FTLE



```
testSetp24 <- data.frame(ftle24_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp24$fit <- predict(krillDensity_gMNZGLMMPQLLogLink24, testSetp24,
                          type = "response", level=0)
# This function outputs the predicted SE values in Log scale
SE_p24 <- predictSE(krillDensity_gMNZGLMMPQLLogLink24, testSetp24,
```

```

        type = "response", level = 0, se.fit = TRUE)
# Convert the Log Standard errors into the Response scale
testSetp24$lwr <- exp(SE_p24$fit-SE_p24$se.fit)
testSetp24$upr <- exp(SE_p24$fit+SE_p24$se.fit)
testSetp24$model <- "24 hrs"

testSetp48 <- data.frame(ftle48_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp48$fit <- predict(krillDensity_gMNZGLMMPQLLogLink48, testSetp48,
                        type = "response", level=0)
# This function outputs the predicted SE values in Log scale
SE_p48 <- predictSE(krillDensity_gMNZGLMMPQLLogLink48, testSetp48,
                   type = "response", level = 0, se.fit = TRUE)
# Convert the Log Standard errors into the Response scale
testSetp48$lwr <- exp(SE_p48$fit-SE_p48$se.fit)
testSetp48$upr <- exp(SE_p48$fit+SE_p48$se.fit)
testSetp48$model <- "48 hrs"

testSetp120 <- data.frame(ftle120_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp120$fit <- predict(krillDensity_gMNZGLMMPQLLogLink120, testSetp120,
                          type = "response", level=0)
# This function outputs the predicted SE values in Log scale
SE_p120 <- predictSE(krillDensity_gMNZGLMMPQLLogLink120, testSetp120,
                    type = "response", level = 0, se.fit = TRUE)
# Convert the Log Standard errors into the Response scale
testSetp120$lwr <- exp(SE_p120$fit-SE_p120$se.fit)
testSetp120$upr <- exp(SE_p120$fit+SE_p120$se.fit)
testSetp120$model <- "120 hrs"

testSetp240 <- data.frame(ftle240_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in Response scale
testSetp240$fit <- predict(krillDensity_gMNZGLMMPQLLogLink240, testSetp240,
                          type = "response", level=0)
# This function outputs the predicted SE values in Log scale
SE_p240 <- predictSE(krillDensity_gMNZGLMMPQLLogLink240, testSetp240,
                    type = "response", level = 0, se.fit = TRUE)
# Convert the Log Standard errors into the Response scale
testSetp240$lwr <- exp(SE_p240$fit-SE_p240$se.fit)
testSetp240$upr <- exp(SE_p240$fit+SE_p240$se.fit)
testSetp240$model <- "240 hrs"

ggplot() +
  geom_ribbon(data=testSetp240, aes(x=ftle240_L, ymin = lwr, ymax = upr,
                                color = NULL, fill=model), alpha = .15) +
  geom_line(data=testSetp240, aes(x=ftle240_L, y = fit, color=model),
           size = 2, linetype=1)+
  geom_ribbon(data=testSetp48, aes(x=ftle48_L, ymin = lwr, ymax = upr,
                                color = NULL, fill=model), alpha = .15) +
  geom_line(data=testSetp48, aes(x=ftle48_L, y = fit, color=model),
           size = 2, linetype=2)+
  geom_ribbon(data=testSetp24, aes(x=ftle24_L, ymin = lwr, ymax = upr,
                                color = NULL, fill=model), alpha = .15) +

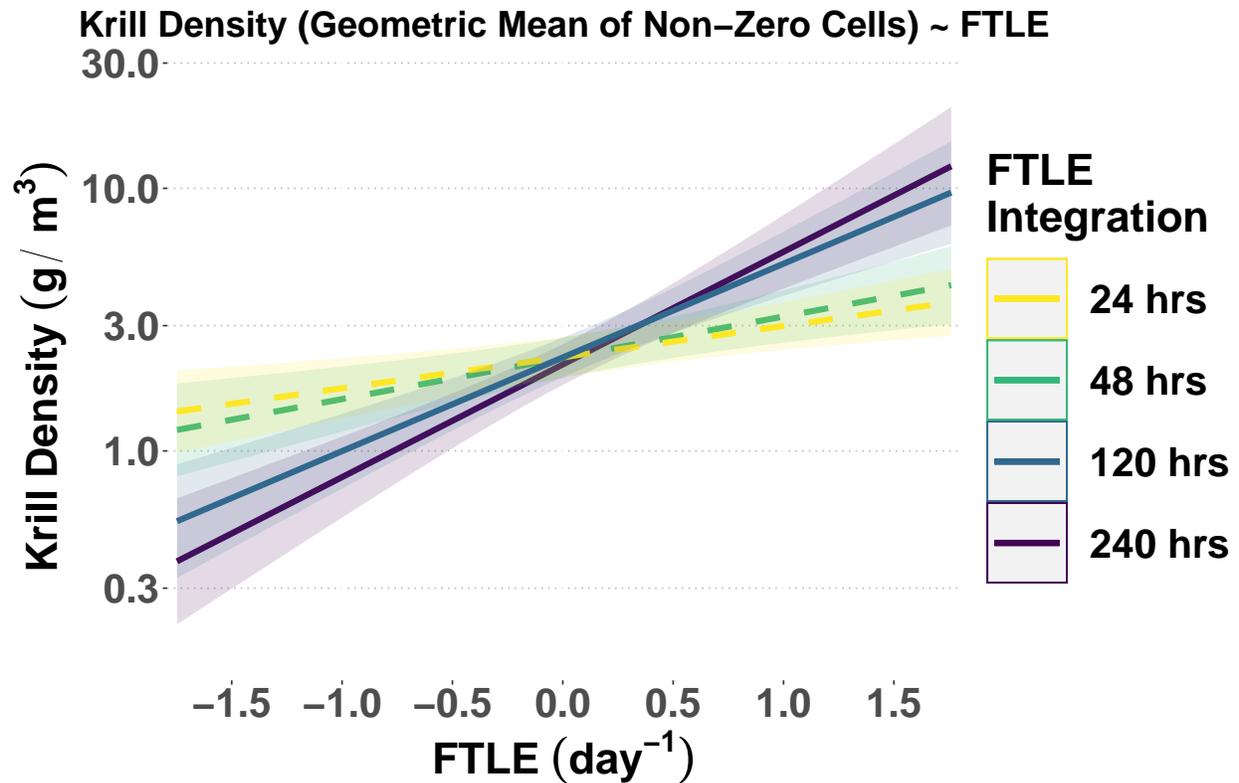
```

```

geom_line(data=testSetp24,aes(x=ftle24_L,y = fit, color=model),
          size = 2, linetype=2)+
geom_ribbon(data=testSetp120,aes(x=ftle120_L,ymin = lwr, ymax = upr,
                                color = NULL,fill=model), alpha = .15) +
geom_line(data=testSetp120,aes(x=ftle120_L,y = fit, color=model),
          size = 2, linetype=1)+
scale_color_manual(name= "FTLE\nIntegration",
                   breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
                   values=c( "24 hrs" = "#FDE725FF",
                              "48 hrs" = "#35B779FF",
                              "120 hrs" = "#31688EFF",
                              "240 hrs" = "#440154FF"
                              )) +
scale_fill_manual(name= "FTLE\nIntegration",
                  breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
                  values=c( "24 hrs" = "#FDE725FF",
                             "48 hrs" = "#35B779FF",
                             "120 hrs" = "#31688EFF",
                             "240 hrs" = "#440154FF"
                             )) +
scale_x_continuous(breaks=seq(-1.5,2,.5),limits=c(-1.75,1.76),
                  expand = c(0.005, 0.025))+
scale_y_log10(expand = c(0.0000000001,0.075)
)+
ylab(expression( bold(Krill~Density~(g/~m^{bold("3")})))) +
xlab(expression( bold(FTLE~(day^{bold({"-1"})})))) )+
guides(fill = guide_legend(order=1,override.aes =
                           list(alpha=0.25, linetype=1)),
       color = guide_legend(order=1,override.aes =
                             list(alpha=1, linetype=1,fill=NA))
       )+
labs(caption = "Dashed lines indicate p-values > .05") +
ggtitle("Krill Density (Geometric Mean of Non-Zero Cells) ~ FTLE") +
theme_pubclean()+
theme(axis.title = element_text(face="bold", size=30),
      axis.text = element_text( face="bold", size=28),
      plot.title = element_text(face="bold", size=25,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=18,hjust = 0.5),
      plot.caption = element_text( face="bold", size=14,hjust = 0),
      legend.position="right",
      legend.text = element_text(face="bold", size=28),
      legend.title = element_text(face="bold", size=30),
      legend.key.size = unit(2,"cm"),
      strip.text.x = element_text(size = 22, face="bold"))

```

Figure 3B



FTLE and Cetacean Sightings

4. Are Cetacean sightings more likely in areas with higher FTLE?

GLMM (Whale Presence ~ FTLE)

Blue Whale Sightings The relationship between the likelihood of blue whale presence and FTLE is explored using a linear mixed-effects model. Here we are using a logistic regression to test the probability of blue whale presence across a range of FTLE values for each of the 4 integration times.

```
# Use previously processed model to save time
if(!exists("bw_glmmPQL_24_Hrs")){
  bw_glmmPQL_24_Hrs <- glmmPQL(BLWH ~ ftle24_L ,
                              random = ~ 1 | Year/Season/Line,
                              family = binomial(link = "logit"),
                              data = dataGamm %>% dplyr::filter(!is.na(ftle24_L)))
}

# Use previously processed model to save time
if(!exists("bw_glmmPQL_48_Hrs")){
  bw_glmmPQL_48_Hrs <- glmmPQL(BLWH ~ ftle48_L,
                              random = ~ 1 | Year/Season/Line,
                              family = binomial(link = "logit"),
                              data = dataGamm %>% dplyr::filter(!is.na(ftle48_L)))
}

# Use previously processed model to save time
if(!exists("bw_glmmPQL_120_Hrs")){
```

```

bw_glmmPQL_120_Hrs <- glmmPQL(BLWH ~ ftle120_L,
                             random = ~ 1 | Year/Season/Line,
                             family = binomial(link = "logit"),
                             data = dataGamm %>% dplyr::filter(!is.na(ftle120_L)))
}

# Use previously processed model to save time
if(!exists("bw_glmmPQL_240_Hrs")){
  bw_glmmPQL_240_Hrs <- glmmPQL(BLWH ~ ftle240_L,
                                random = ~ 1 | Year/Season/Line,
                                family = binomial(link = "logit"),
                                data = dataGamm %>% dplyr::filter(!is.na(ftle240_L)))
}

tBW <-
data.frame(
  Model = rep("GLMM BLWH Presence",4),
  FTLE_Integration = c("24","48","120", "240"),
  Slope =
    c(round(as.data.frame(coef(summary(bw_glmmPQL_24_Hrs)))$Value[2],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_48_Hrs)))$Value[2],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_120_Hrs)))$Value[2],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_240_Hrs)))$Value[2],3)
    ),
  Intercept =
    c(round(as.data.frame(coef(summary(bw_glmmPQL_24_Hrs)))$Value[1],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_48_Hrs)))$Value[1],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_120_Hrs)))$Value[1],3),
      round(as.data.frame(coef(summary(bw_glmmPQL_240_Hrs)))$Value[1],3)
    ),
  pValue =
    c(round(as.data.frame(coef(summary(bw_glmmPQL_24_Hrs)))$'p-value'[2],5),
      round(as.data.frame(coef(summary(bw_glmmPQL_48_Hrs)))$'p-value'[2],5),
      round(as.data.frame(coef(summary(bw_glmmPQL_120_Hrs)))$'p-value'[2],5),
      round(as.data.frame(coef(summary(bw_glmmPQL_240_Hrs)))$'p-value'[2],5)
    )
)
tBW

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM BLWH Presence	24	0.786	-5.920	0.00686
GLMM BLWH Presence	48	1.339	-5.991	0.00007
GLMM BLWH Presence	120	1.200	-5.828	0.01009
GLMM BLWH Presence	240	2.116	-5.975	0.00077

```

testSet_bw_p24 <- data.frame(ftle24_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in probability space
testSet_bw_p24$fit <- predict(bw_glmmPQL_24_Hrs,testSet_bw_p24,
                             type = "response", level=0)

# This function outputs the predicted values in Logit space
SE_p24 <- predictSE(bw_glmmPQL_24_Hrs,testSet_bw_p24,
                   type = "response", level = 0,se.fit = TRUE)

```

```

# Convert the Logit Standard errors into Probabilities
testSet_bw_p24$lwr <- logit2prob(SE_p24$fit-SE_p24$se.fit)
testSet_bw_p24$upr <- logit2prob(SE_p24$fit+SE_p24$se.fit)
testSet_bw_p24$model <- "24 hrs"

testSet_bw_p48 <- data.frame(ftle48_L = seq(-2, 2,length.out = 1000))
# This function outputs the predicted values in probability space
testSet_bw_p48$fit <- predict(bw_glmPQL_48_Hrs,testSet_bw_p48,
                             type = "response", level=0)
# This function outputs the predicted values in Logit space
SE_p48 <- predictSE(bw_glmPQL_48_Hrs,testSet_bw_p48,
                    type = "response", level = 0,se.fit = TRUE)
# Convert the Logit Standard errors into Probabilities
testSet_bw_p48$lwr <- logit2prob(SE_p48$fit-SE_p48$se.fit)
testSet_bw_p48$upr <- logit2prob(SE_p48$fit+SE_p48$se.fit)
testSet_bw_p48$model <- "48 hrs"

testSet_bw_p120 <- data.frame(ftle120_L = seq(-2, 2,length.out = 1000))
# This function outputs the predicted values in probability space
testSet_bw_p120$fit <- predict(bw_glmPQL_120_Hrs,testSet_bw_p120,
                              type = "response", level=0)
# This function outputs the predicted values in Logit space
SE_p120 <- predictSE(bw_glmPQL_120_Hrs,testSet_bw_p120,
                     type = "response", level = 0,se.fit = TRUE)
# Convert the Logit Standard errors into Probabilities
testSet_bw_p120$lwr <- logit2prob(SE_p120$fit-SE_p120$se.fit)
testSet_bw_p120$upr <- logit2prob(SE_p120$fit+SE_p120$se.fit)
testSet_bw_p120$model <- "120 hrs"

testSet_bw_p240 <- data.frame(ftle240_L = seq(-2, 2, length.out = 1000))
# This function outputs the predicted values in probability space
testSet_bw_p240$fit <- predict(bw_glmPQL_240_Hrs,testSet_bw_p240,
                              type = "response", level=0)
# This function outputs the predicted values in Logit space
SE_p240 <- predictSE(bw_glmPQL_240_Hrs,testSet_bw_p240,
                     type = "response", level = 0,se.fit = TRUE)
# Convert the Logit Standard errors into Probabilities
testSet_bw_p240$lwr <- logit2prob(SE_p240$fit-SE_p240$se.fit)
testSet_bw_p240$upr <- logit2prob(SE_p240$fit+SE_p240$se.fit)
testSet_bw_p240$model <- "240 hrs"

ggplot() +
  geom_ribbon(data=testSet_bw_p24,aes(x=ftle24_L,ymin = lwr, ymax = upr,
                                     color = NULL,fill=model), alpha = .15) +
  geom_line(data=testSet_bw_p24,aes(x=ftle24_L,y = fit, color=model),
            size = 2, linetype=1)+
  geom_ribbon(data=testSet_bw_p240,aes(x=ftle240_L,ymin = lwr, ymax = upr,
                                       color = NULL,fill=model), alpha = .15) +
  geom_line(data=testSet_bw_p240,aes(x=ftle240_L,y = fit, color=model),
            size = 2, linetype=1)+
  geom_ribbon(data=testSet_bw_p48,aes(x=ftle48_L,ymin = lwr, ymax = upr,
                                      color = NULL,fill=model), alpha = .15) +
  geom_line(data=testSet_bw_p48,aes(x=ftle48_L,y = fit, color=model),

```

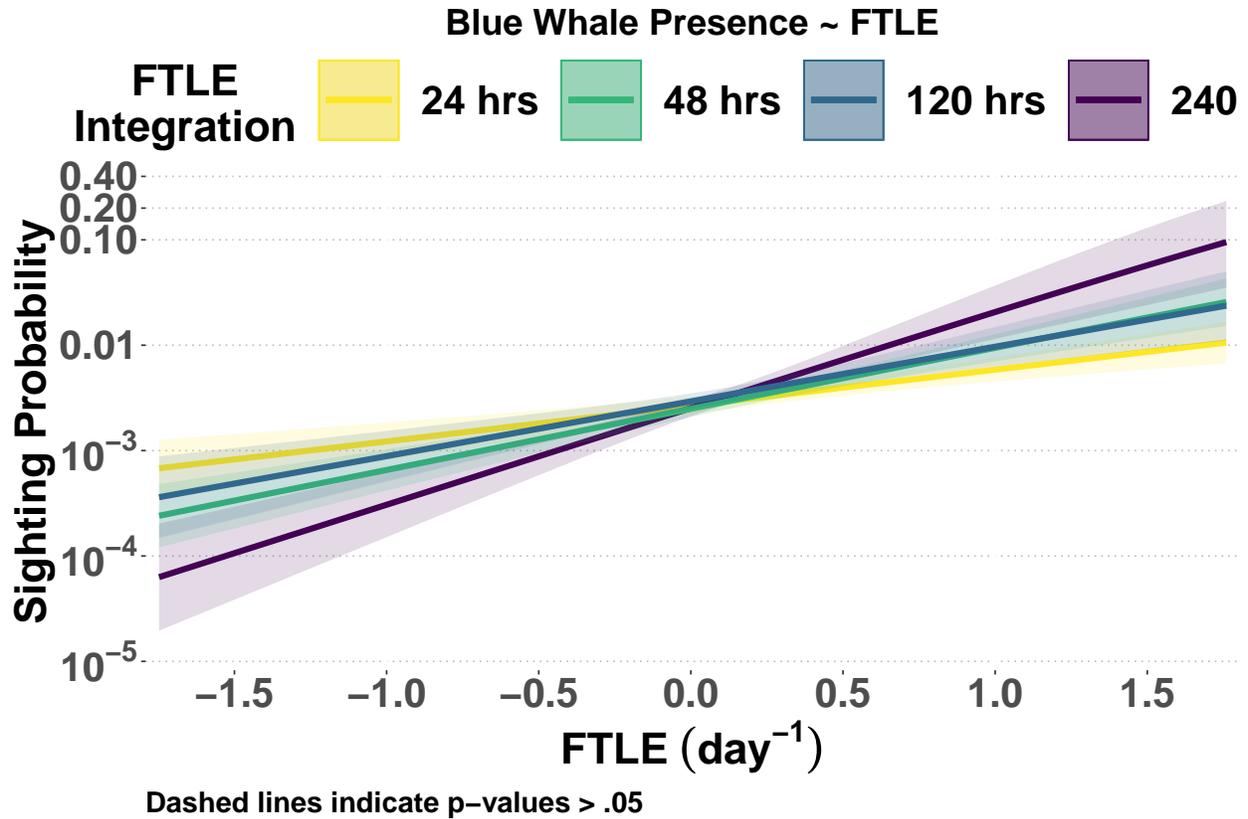
```

    size = 2, linetype=1)+

geom_ribbon(data=testSet_bw_p120,aes(x=ftle120_L,ymin = lwr, ymax = upr,
    color = NULL,fill=model), alpha = .15) +
geom_line(data=testSet_bw_p120,aes(x=ftle120_L,y = fit, color=model),
    size = 2, linetype=1)+
scale_color_manual(name= "FTLE\nIntegration",
    breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
    values=c( "24 hrs" = "#FDE725FF",
              "48 hrs" = "#35B779FF",
              "120 hrs" = "#31688EFF",
              "240 hrs" = "#440154FF"
            )) +
scale_fill_manual(name= "FTLE\nIntegration",
    breaks=c("24 hrs","48 hrs","120 hrs","240 hrs"),
    values=c( "24 hrs" = "#FDE725FF",
              "48 hrs" = "#35B779FF",
              "120 hrs" = "#31688EFF",
              "240 hrs" = "#440154FF"
            )) +
scale_x_continuous(breaks=seq(-1.5,2,.5),limits=c(-1.75,1.76),
    expand = c(0.005, 0.025))+
scale_y_log10(name="Sighting Probability",limits=c(-5,-0.4),
    breaks = c(0.00001,0.0001,0.001, 0.01, 0.1, 0.2, 0.4),
    labels = c(expression(bold("10"^(bold("-5")))),
              expression(bold("10"^(bold("-4")))),
              expression(bold("10"^(bold("-3")))),
              expression(bold("0.01")),
              expression(bold("0.10")),
              expression(bold("0.20")),
              expression(bold("0.40"))),
    expand = c(0.000000001,0.075)
  )+
xlab(expression( bold(FTLE~(day^(bold({bold("-1")})))))) )+
guides(fill = guide_legend(order=1,override.aes = list(alpa=0.05)),
    color = guide_legend(order=1,override.aes = list(alpa=1)))+
labs(caption = "Dashed lines indicate p-values > .05") +
ggtitle("Blue Whale Presence ~ FTLE") +
theme_pubclean()+
theme(axis.title = element_text(face="bold", size=30),
    axis.text = element_text( face="bold", size=28),
    plot.title = element_text(face="bold", size=25,hjust = 0.5),
    plot.subtitle = element_text( face="bold", size=18,hjust = 0.5),
    plot.caption = element_text( face="bold", size=20,hjust = 0),
    legend.position= "top",
    legend.direction = "horizontal",
    legend.text = element_text(face="bold", size=28),
    legend.title = element_text(face="bold", size=30,hjust=0.5),
    legend.key.size = unit(2,"cm"),
    strip.text.x = element_text(size = 22, face="bold"))

```

Figure 3C



Humpback Whale Sightings The relationship between the likelihood of humpback whale presence and FTLE is explored using a linear mixed-effects model. Here we are using a logistic regression to test the probability of humpback whale presence across a range of FTLE values for each of the 4 integration times.

```

# Use previously processed model to save time
if(!exists("mn_glmmPQL_24_Hrs")){
  mn_glmmPQL_24_Hrs <- glmmPQL(HUWH ~ ftle24_L ,
                              random = ~ 1 | Year/Season/Line,
                              family = binomial(link = "logit"),
                              data = dataGamm %>% dplyr::filter(!is.na(ftle24_L)))
}

# Use previously processed model to save time
if(!exists("mn_glmmPQL_48_Hrs")){
  mn_glmmPQL_48_Hrs <- glmmPQL(HUWH ~ ftle48_L,
                              random = ~ 1 | Year/Season/Line,
                              family = binomial(link = "logit"),
                              data = dataGamm %>% dplyr::filter(!is.na(ftle48_L)))
}

# Use previously processed model to save time
if(!exists("mn_glmmPQL_120_Hrs")){
  mn_glmmPQL_120_Hrs <- glmmPQL(HUWH ~ ftle120_L,
                              random = ~ 1 | Year/Season/Line,
                              family = binomial(link="logit"),
                              data = dataGamm %>% dplyr::filter(!is.na(ftle120_L)))
}

```

```

# Use previously processed model to save time
if(!exists("mn_glmmPQL_240_Hrs")){
  mn_glmmPQL_240_Hrs <- glmmPQL(HUWH ~ ftle240_L,
                                random = ~ 1 | Year/Season/Line,
                                family = binomial(link = "logit"),
                                data = dataGamm %>% dplyr::filter(!is.na(ftle240_L)))
}

tHW <-
  data.frame(
    Model = rep("GLMM HUWH Presence",4),
    FTLE_Integration = c("24","48","120", "240"),
    Slope =
      c(round(as.data.frame(coef(summary(mn_glmmPQL_24_Hrs)))$Value[2],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_48_Hrs)))$Value[2],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_120_Hrs)))$Value[2],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_240_Hrs)))$Value[2],3)
      ),
    Intercept =
      c(round(as.data.frame(coef(summary(mn_glmmPQL_24_Hrs)))$Value[1],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_48_Hrs)))$Value[1],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_120_Hrs)))$Value[1],3),
        round(as.data.frame(coef(summary(mn_glmmPQL_240_Hrs)))$Value[1],3)
      ),
    pValue =
      c(round(as.data.frame(coef(summary(mn_glmmPQL_24_Hrs)))$'p-value'[2],5),
        round(as.data.frame(coef(summary(mn_glmmPQL_48_Hrs)))$'p-value'[2],5),
        round(as.data.frame(coef(summary(mn_glmmPQL_120_Hrs)))$'p-value'[2],5),
        round(as.data.frame(coef(summary(mn_glmmPQL_240_Hrs)))$'p-value'[2],5)
      )
  )
)
tHW

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM HUWH Presence	24	0.361	-3.626	0.03571
GLMM HUWH Presence	48	0.449	-3.605	0.02978
GLMM HUWH Presence	120	-0.432	-3.477	0.13831
GLMM HUWH Presence	240	0.629	-3.604	0.10102

```

# Plot the models together
p_WhaleAll <- plot_models(bw_glmmPQL_24_Hrs,bw_glmmPQL_48_Hrs,
                          bw_glmmPQL_120_Hrs,bw_glmmPQL_240_Hrs,
                          mn_glmmPQL_24_Hrs,mn_glmmPQL_48_Hrs,
                          mn_glmmPQL_120_Hrs,mn_glmmPQL_240_Hrs,
                          vline.color = "red",show.p=TRUE,
                          show.values=TRUE, spacing = .75,
                          show.legend = TRUE, dot.size = 6,
                          value.size = 10, line.size = 2,
                          axis.labels = c("FTLE (240 Hrs)","FTLE (120 Hrs)",
                                           "FTLE (48 Hrs)","FTLE (24 Hrs)") +
                          scale_y_log10( limits = c(-100,200))+
                          theme_minimal() +

```

```

theme(axis.title = element_text(face="bold", size=30),
      axis.text = element_text( face="bold", size=30),
      plot.title = element_text(face="bold", size=25,hjust = 0.5),
      plot.subtitle = element_text( face="bold", size=20,hjust = 0.5),
      plot.caption = element_text( face="bold", size=12,hjust = 0),
      legend.position = c(0.8,0.5),
      legend.direction = "vertical",
      legend.background = element_rect(fill="white",color="black"),
      legend.text = element_text(face="bold", size=26),
      legend.title = element_text(face="bold", size=28, hjust = 0.5))

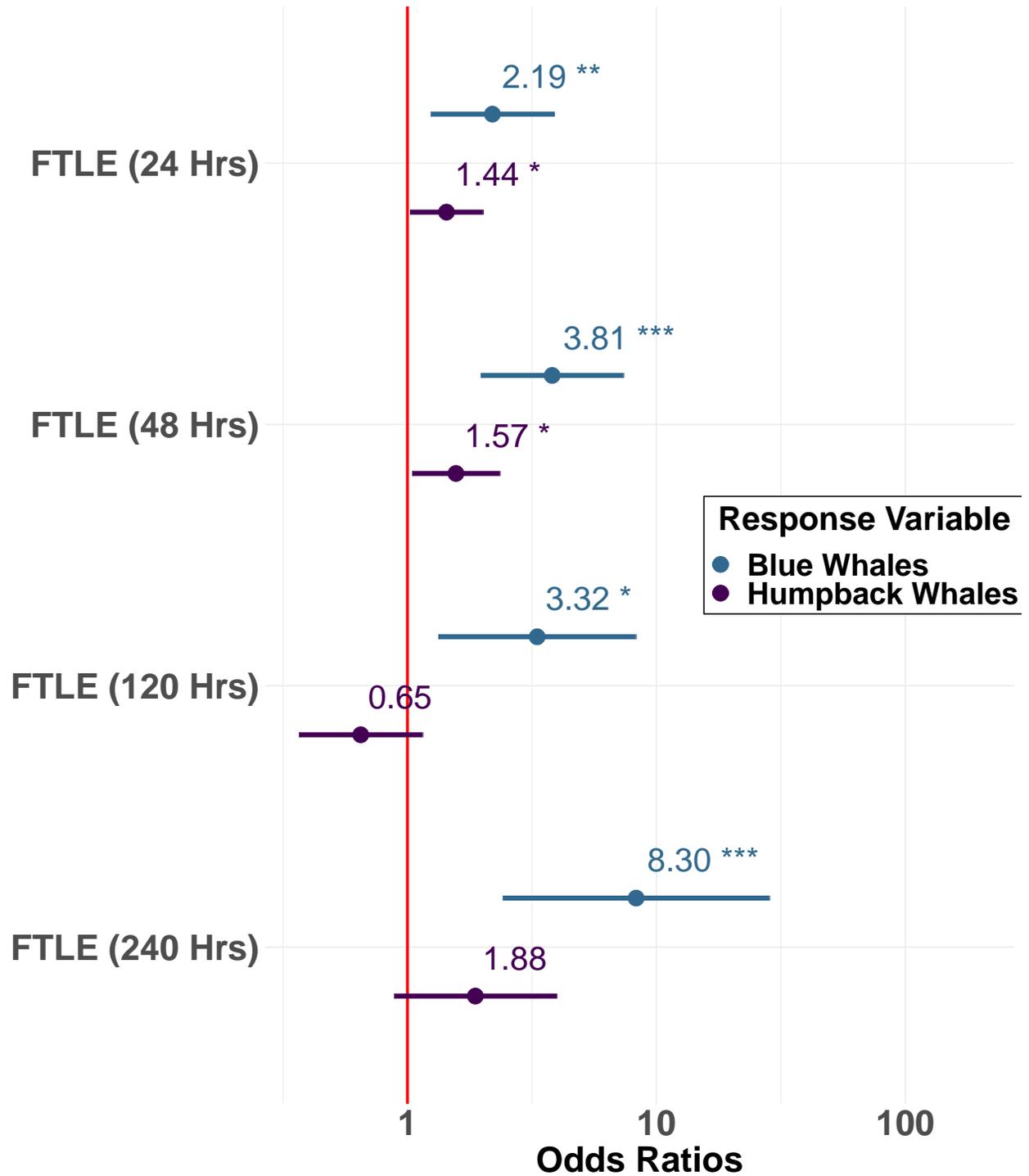
p_WhaleAll$guides$colour$title <- "Response Variable"

p_WhaleAll$data$group <- factor(x = c(rep("Blue Whales",4),
                                     rep("Humpback Whales",4)),
                              levels = c("Humpback Whales" , "Blue Whales") )

p_WhaleAll <- p_WhaleAll +
  scale_color_manual(values=c( "Humpback Whales" = "#440154FF",
                              "Blue Whales" = "#31688EFF"
                              ) )
p_WhaleAll

```

Supplemental Figure 4



Results Table

```

resultsT <- rbind(tssSigmaT_GLMM, tD10_GLMM, tD20_GLMM,tD30_GLMM,tD40_GLMM, tD50_GLMM,
  tSigma25_GLMM,tSigma25_5_GLMM,tSigma26_GLMM,
  tKrillPres,tgMNZGLMMPQLLogLink,tMNZGLMMPQLLogLink,tMaxGLMMPQLLogLink,
  tBW,tHW)
resultsT_wide <- resultsT %>%

```

```

pivot_wider(names_from = FTLE_Integration, values_from = Slope:pValue) %>%
dplyr::select(1,2,6,10,3,7,11,4,8,12,5,9,13)
write_csv(resultsT_wide, "./Output/Table2-Results.csv" )
kable(resultsT, row.names = FALSE, longtable=T) %>%
kable_styling(latex_options=c("striped", "repeat_header", "scale_down"))
## Warning in styling_latex_scale_down(out, table_info): Longtable cannot be resized.

```

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Sea Surface Sigma Theta	24	0.0545	25.1968	0.52495
GLMM Sea Surface Sigma Theta	48	0.1262	25.1935	0.18939
GLMM Sea Surface Sigma Theta	120	0.3704	25.2281	0.00143
GLMM Sea Surface Sigma Theta	240	0.2889	25.2592	0.05069
GLMM Sigma Theta at 10 meters	24	0.0162	25.3555	0.84426
GLMM Sigma Theta at 10 meters	48	0.1176	25.3452	0.19947
GLMM Sigma Theta at 10 meters	120	0.3391	25.3673	0.00154
GLMM Sigma Theta at 10 meters	240	0.2454	25.3987	0.06771
GLMM Sigma Theta at 20 meters	24	-0.1717	25.5993	0.01471
GLMM Sigma Theta at 20 meters	48	-0.0335	25.5727	0.66812
GLMM Sigma Theta at 20 meters	120	0.1635	25.5819	0.06603
GLMM Sigma Theta at 20 meters	240	0.1611	25.6163	0.14203
GLMM Sigma Theta at 30 meters	24	-0.1743	25.7775	0.00438
GLMM Sigma Theta at 30 meters	48	-0.0826	25.7596	0.22734
GLMM Sigma Theta at 30 meters	120	0.0776	25.7667	0.31705
GLMM Sigma Theta at 30 meters	240	0.1415	25.7955	0.15451
GLMM Sigma Theta at 40 meters	24	-0.1408	25.8791	0.00939
GLMM Sigma Theta at 40 meters	48	-0.0418	25.8598	0.49531
GLMM Sigma Theta at 40 meters	120	0.0912	25.8721	0.19005
GLMM Sigma Theta at 40 meters	240	0.1367	25.8948	0.12618
GLMM Sigma Theta at 50 meters	24	-0.1295	25.9706	0.00539
GLMM Sigma Theta at 50 meters	48	-0.0282	25.9507	0.59256
GLMM Sigma Theta at 50 meters	120	0.0674	25.9653	0.26360
GLMM Sigma Theta at 50 meters	240	0.0962	25.9782	0.21549
GLMM Depth of SigmaTheta 25	24	0.9432	8.1158	0.59605
GLMM Depth of SigmaTheta 25	48	-0.2272	8.1384	0.90861
GLMM Depth of SigmaTheta 25	120	-1.3907	7.4396	0.49282
GLMM Depth of SigmaTheta 25	240	-3.1431	6.6889	0.19746
GLMM Depth of SigmaTheta 25.5	24	4.5977	19.4710	0.18006
GLMM Depth of SigmaTheta 25.5	48	-0.6020	20.3434	0.87680
GLMM Depth of SigmaTheta 25.5	120	-10.0798	18.7888	0.02193
GLMM Depth of SigmaTheta 25.5	240	-16.2146	17.6131	0.00341
GLMM Depth of SigmaTheta 26	24	14.1729	48.5953	0.00564
GLMM Depth of SigmaTheta 26	48	3.8680	50.5918	0.49933
GLMM Depth of SigmaTheta 26	120	-6.0794	50.3184	0.38792
GLMM Depth of SigmaTheta 26	240	-8.9667	49.7587	0.32645
GLMM Krill Presence	24	-0.3840	0.0780	0.01416
GLMM Krill Presence	48	-0.2740	0.0210	0.13664
GLMM Krill Presence	120	0.2630	-0.1330	0.29013
GLMM Krill Presence	240	0.2850	-0.1730	0.36950
GLMM Geo-Mean Krill Density	24	0.2731	0.8225	0.08437
GLMM Geo-Mean Krill Density	48	0.3622	0.8165	0.05616
GLMM Geo-Mean Krill Density	120	0.8218	0.8196	0.00110
GLMM Geo-Mean Krill Density	240	0.9878	0.7580	0.00061

(continued)

Model	FTLE_Integration	Slope	Intercept	pValue
GLMM Mean Krill Density	24	0.2064	1.1619	0.18788
GLMM Mean Krill Density	48	0.4448	1.1128	0.01664
GLMM Mean Krill Density	120	0.9310	1.1037	0.00016
GLMM Mean Krill Density	240	0.9675	1.1212	0.00248
GLMM Max Krill Density	24	0.0676	1.9669	0.68640
GLMM Max Krill Density	48	0.4069	1.8880	0.04000
GLMM Max Krill Density	120	0.8840	1.8287	0.00085
GLMM Max Krill Density	240	0.8341	1.8129	0.01589
GLMM BLWH Presence	24	0.7860	-5.9200	0.00686
GLMM BLWH Presence	48	1.3390	-5.9910	0.00007
GLMM BLWH Presence	120	1.2000	-5.8280	0.01009
GLMM BLWH Presence	240	2.1160	-5.9750	0.00077
GLMM HUWH Presence	24	0.3610	-3.6260	0.03571
GLMM HUWH Presence	48	0.4490	-3.6050	0.02978
GLMM HUWH Presence	120	-0.4320	-3.4770	0.13831
GLMM HUWH Presence	240	0.6290	-3.6040	0.10102

References

- Matt K Gough, Ad Reniers, M. Josefina Olascoaga, Brian K Haus, Jamie MacMahan, Jeff Paduan, Chris Halle, (2016). *Lagrangian Coherent Structures in a coastal upwelling environment*. Continental Shelf Research, <https://doi.org/10.1016/j.csr.2016.09.007>
- Fahlbusch, J. A., Czapanskiy, M. F., Calambokidis, J., Cade, D. E., Abrahms, B., Hazen, E. L., & Goldbogen, J. A. (2022). *Blue whales increase feeding rates at fine-scale ocean features*. Proceedings of the Royal Society B, <https://royalsocietypublishing.org/doi/full/10.1098/rspb.2022.1180>