Covariates for Scripps's Murrelet Egg Size model

Amelia J. DuVall & Marcela Todd

4/01/2021

This is v.2021-04-19

Introduction

This document details steps taken to compile covariate data for a linear mixed model on Scripps's Murrelet (*Synthliboramphus scrippsi*) egg size at Santa Barbara Island within Channel Islands National Park from 2009-2017.

We decided to pull relevant covariates from January-June only because murrelets begin to arrive at the colony in January and clutch initiation extends from March to June (Murray et al. 1985).

Potential Covariates

Local oceanographic conditions

Sea Surface Temperature

NOAA buoy station 46025: https://www.ndbc.noaa.gov/station_page.php?station=46025
Historical data here: https://www.ndbc.noaa.gov/station_history.php?station=46025
Data descriptions here: https://www.ndbc.noaa.gov/measdes.shtml

```
## load annual files
wx2009 <- read_csv(here("data", "covariates", "buoy_46025_2009_wx.csv"), na = c("999")) %>%
  slice(-c(1))  %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   `#YY` = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2010 <- read_csv(here("data", "covariates", "buoy_46025_2010_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2011 <- read_csv(here("data", "covariates", "buoy_46025_2011_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2012 <- read_csv(here("data", "covariates", "buoy_46025_2012_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2013 <- read_csv(here("data", "covariates", "buoy_46025_2013_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2014 <- read_csv(here("data", "covariates", "buoy_46025_2014_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2015 <- read_csv(here("data", "covariates", "buoy_46025_2015_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2016 <- read_csv(here("data", "covariates", "buoy_46025_2016_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSP = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
wx2017 <- read_csv(here("data", "covariates", "buoy_46025_2017_wx.csv"), na = c("999")) %>%
  slice(-c(1)) %>%
  dplyr::select(1:4, 15)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   ‘#YY‘ = col_character(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_character(),
##   WSPD = col_character(),
##   GST = col_character(),
##   WVHT = col_character(),
##   DPD = col_character(),
##   APD = col_character(),
##   MWD = col_character(),
##   PRES = col_character(),
##   ATMP = col_character(),
##   WTMP = col_character(),
##   DEWP = col_character(),
##   VIS = col_character(),
##   TIDE = col_character()
## )
```

```
## bind to create one wx file
SSTraw <- rbind(wx2009, wx2010, wx2011, wx2012, wx2013, wx2014, wx2015, wx2016, wx2017) %>%
  rename(Year = "#YY")

## check for data duplicates
## include minute column?

## check NAs per year
table(SSTraw$WTMP, useNA = "always") # 983 NAs
```

```
##
##    0 11.6 12.1 12.2 12.3 12.4 12.5 12.6 12.7 12.8 12.9   13 13.1 13.2 13.3 13.4
##    1    1    1    6   23   48   73   71   74  107  161  224  255  380  437  511
## 13.5 13.6 13.7 13.8 13.9   14 14.1 14.2 14.3 14.4 14.5 14.6 14.7 14.8 14.9   15
##  560  436  690  525  751  889  895 1451 1493 1648 1571 2031 2392 2149 2304 2552
## 15.1 15.2 15.3 15.4 15.5 15.6 15.7 15.8 15.9   16 16.1 16.2 16.3 16.4 16.5 16.6
## 2311 2028 1736 1717 1901 1950 2022 2026 1894 1923 1742 1732 1436 1561 1446 1520
## 16.7 16.8 16.9   17 17.1 17.2 17.3 17.4 17.5 17.6 17.7 17.8 17.9   18 18.1 18.2
## 1663 1528 1333 1436 1475 1586 1764 1659 1508 1453 1442 1451 1418 1640 1485 1378
## 18.3 18.4 18.5 18.6 18.7 18.8 18.9   19 19.1 19.2 19.3 19.4 19.5 19.6 19.7 19.8
## 1436 1389 1561 1554 1629 1698 1514 1576 1395 1263 1367 1355 1526 1439 1289 1290
## 19.9   20 20.1 20.2 20.3 20.4 20.5 20.6 20.7 20.8 20.9   21 21.1 21.2 21.3 21.4
## 1292 1257 1184 1157 1238 1367 1405 1369 1115 1083 1072 1209 1276 1214 1288 1140
## 21.5 21.6 21.7 21.8 21.9   22 22.1 22.2 22.3 22.4 22.5 22.6 22.7 22.8 22.9   23
## 1051 1361 1354 1253 1286 1129  993  700  684  585  567  400  398  345  359  293
## 23.1 23.2 23.3 23.4 23.5 23.6 23.7 23.8 23.9   24 24.1 24.2 24.3 24.4 24.5 24.6
##  238  203  135  121   76   56   23   26   27   12   11   10    8    3    3    3
## 24.7 24.8 24.9   25 <NA>
##    1    2    1    3  983
```

```
sst_NAs <- SSTraw %>%
  group_by(Year) %>%
  summarise(TotalNAs = sum(is.na(WTMP))) # lots of NAs in 2012

## check dataframes links, sampling frequency changes

## where are the NAs in 2012?
# plot(wx2012$MM, wx2012$WTMP) # data missing in spring/summer

## create mean annual sst
SST_full <- SSTraw %>%
  mutate(WTMP = as.numeric(WTMP)) %>%
  group_by(Year) %>%
  summarise(SST = mean(WTMP, na.rm = TRUE)) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning: ‘as.tibble()‘ was deprecated in tibble 2.0.0.
## Please use ‘as_tibble()‘ instead.
## The signature and semantics have changed, see ‘?as_tibble‘.
```

```
## create mean sst for Jan-May
SST <- SSTraw %>%
  mutate(WTMP = as.numeric(WTMP),
         MM = as.numeric(MM)) %>%
  filter(MM == 1:6) %>%
  group_by(Year) %>%
  summarise(SST = mean(WTMP, na.rm = TRUE)) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning in MM == 1:6: longer object length is not a multiple of shorter object
## length
```

Forage fish

Data from NOAA IEA https://www.integratedecosystemassessment.noaa.gov/regions/california-current/cc-indicator-ecological-integrity
https://swfsc.noaa.gov/textblock.aspx?Division=FED&ParentMenuId=54&id=20615

Adult anchovy - CCC (1990-2018)

Institution: NOAA SWFSC
Dataset summary: Dr. John Field (NOAA; john.field@noaa.gov), from Rockfish Recruitment and Ecosystem Assessment Survey; Samples represent catch (individuals) per standard 15 minute trawl (CPUE). Data are log(CPUE+1) transformed. Geometric means calculated on non-zero data.

```
ANCHAraw <- read.csv(here("data", "covariates", "cciea_EI_FBC_09ec_5398_9325.csv"))

ANCHA <- ANCHAraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(ANCHA = as.numeric(mean_cpue)) %>%
  dplyr::select(Year, ANCHA) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

Larval anchovy - SCC

Dr. Andrew Thompson (NOAA; andrew.thompson@noaa.gov); derived from spring CalCOFI surveys. Larval fish data summed across all stations of the CalCOFI survey in spring (units are in number under 10 sq. m of surface area; ln(abundance+1)). CalCOFI lines 76.7 - 93.3; stations 28.0 - 120/0

Note: No data available for 2017 (check why)

```
ANCHLraw  <- read.csv(here("data", "covariates", "cciea_EI_FBS_a29e_1fd0_409d.csv"), na.strings = "NaN"

ANCHL <- ANCHLraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(ANCHL = as.numeric(relative_abundance)) %>%
  dplyr::select(Year, ANCHL) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

Krill - CCC (1990-2019)

Institution: NOAA SWFSC More information: Dataset summary: Dr. John Field (NOAA; john.field@noaa.gov) from the SWFSC Rockfish Recruitment and Ecosystem Assessment Survey (https://swfsc.noaa.gov/textblock.aspx?Division=FED&ParentMenuId=54&id=20615). Samples represent catch (individuals) per standard 15 minute trawl (CPUE). Data are log(CPUE+1) transformed; Geometric means calculated on non-zero data.

```
KRILLraw <- read.csv(here("data", "covariates", "cciea_EI_FBC_3c3a_8819_d95c.csv"))

KRILL <- KRILLraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(KRILL = as.numeric(mean_cpue)) %>%
  dplyr::select(Year, KRILL) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

Large-scale Oceanographic Indicators

Pacific Decadal Oscillation Index

Summary: UW/JISAO (http://research.jisao.washington.edu/pdo/PDO.latest.txt) The PDO is calculated from an Empirical Orthogonal analysis of sea surface temperature anomalies in the North Pacific. The PDO is the first dominant mode.
Institution: University of Washington, Joint Institute for the Study of the Atmosphere and Ocean (JISAO)
Time span: 1900-01-01T00:00:00Z - 2020-04-01T00:00:00Z

```
PDOraw <- read.csv(here("data", "covariates", "cciea_OC_PDO_712b_5843_9069.csv"))

PDO_full <- PDOraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(pdo = as.numeric(PDO)) %>%
  group_by(Year) %>%
  summarise(PDO = mean(pdo, na.rm = TRUE)) %>%
  dplyr::select(Year, PDO) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()

## for January-June only
PDO <- PDOraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time),
         Month = month(time)) %>%
  mutate(pdo = as.numeric(PDO)) %>%
  filter(Month == 1:6) %>%
  group_by(Year) %>%
  summarise(PDO = mean(pdo, na.rm = TRUE)) %>%
  dplyr::select(Year, PDO) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning in Month == 1:6: longer object length is not a multiple of shorter
## object length
```

Oceanic Nino Index

Summary: NOAA/CPC (http://origin.cpc.ncep.noaa.gov/products/analysis_monitoring/ensostuff/ONI_v5.php) The ONI is the 3 month running mean of sea surface temperature anomalies in the Nino 3.4 region
Institution: NOAA, Climate Prediction Center (CPC)
Time span: 1950-01-01T00:00:00Z - 2020-03-01T00:00:00Z

```
ONIraw <- read.csv(here("data", "covariates", "cciea_OC_ONI_712b_5843_9069.csv"))

ONI_full <- ONIraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(oni = as.numeric(ONI)) %>%
  group_by(Year) %>%
  summarise(ONI = mean(oni, na.rm = TRUE)) %>%
  dplyr::select(Year, ONI) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()

## for January-June only
ONI <- ONIraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time),
         Month = month(time)) %>%
  mutate(oni = as.numeric(ONI)) %>%
  filter(Month == 1:6) %>%
  group_by(Year) %>%
  summarise(ONI = mean(oni, na.rm = TRUE)) %>%
  dplyr::select(Year, ONI) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning in Month == 1:6: longer object length is not a multiple of shorter
## object length
```

North Pacific Gyre Oscillation Index

Summary: http://www.o3d.org/npgo/npgo.php The NPGO is calculated from an Empirical Orthogonal Function analysis of sea-surface height in the Northeast Pacific. The NPGO is the second dominant mode.
Institution: Georgia Institute of Technology (GT)
Time span: 1950-01-01T00:00:00Z - 2019-07-01T00:00:00Z

```
NPGOraw <- read.csv(here("data", "covariates", "cciea_OC_NPGO_712b_5843_9069.csv"))

NPGO_full <- NPGOraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(npgo = as.numeric(NPGO)) %>%
  group_by(Year) %>%
  summarise(NPGO = mean(npgo, na.rm = TRUE)) %>%
  dplyr::select(Year, NPGO) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()

## for January-June only
NPGO <- NPGOraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time),
         Month = month(time)) %>%
  mutate(npgo = as.numeric(NPGO)) %>%
  filter(Month == 1:6) %>%
  group_by(Year) %>%
  summarise(NPGO = mean(npgo, na.rm = TRUE)) %>%
  dplyr::select(Year, NPGO) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning in Month == 1:6: longer object length is not a multiple of shorter
## object length
```

Coastal Upwelling Transport Index (39N)

Summary: CUTI is a new upwelling index that leverages state-of-the-art ocean models as well as satellite and in situ data to improve upon historically available upwelling indices for the U.S. west coast. CUTI provides estimates of vertical transport near the coast (i.e., upwelling/downwelling). It was developed as a more accurate alternative to the previously available Bakun Index. See Jacox, M. G., C. A. Edwards, E. L. Hazen, and S. J. Bograd (2018) Coastal upwelling revisited: Ekman, Bakun, and improved upwelling indices for the U.S. west coast. Journal of Geophysical Research, doi:10.1029/2018JC014187./ Institution: NOAA/SWFSC/ERD
Time span: 1988-01-01T00:00:00Z - 2020-04-01T00:00:00Z

```
CUTIraw <- read.csv(here("data", "covariates", "cciea_OC_CUTI_784c_ef9f_af6f.csv"))

CUTI <- CUTIraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(cuti = as.numeric(cuti)) %>%
  group_by(Year) %>%
  summarise(CUTI = mean(cuti, na.rm = TRUE)) %>%
  dplyr::select(Year, CUTI) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

Traditional Bakun

Calculation at 15 positions along the west coast of North America from https://oceanview.pfeg.noaa.gov/products/upwelling/dnld Bakun Index Values from NOAA/NMFS/PFEG for: 33N 119W Values are daily average of wind-driven crossshore transports computed from FNMOC six-hourly surface pressure analyses. Indices are in units of cubic meters per second along each 100 meters of coastline. -9999 indicates missing value. Positive numbers indicate offshore transport. Day is based on PST. The Bakun Index uses sea level pressure fields from an atmospheric reanalysis to derive estimated near-surface winds, while CUTI and BEUTI use winds directly from atmospheric reanalyses that assimilate satellite and in situ wind measurements.

```
BAKUNraw <- read.csv(here("data", "covariates", "bakun.csv"), na.strings = "-9999")

BAKUN <- BAKUNraw %>%
  mutate(Date = mdy(Date)) %>%
  mutate(Year = year(Date)) %>%
  mutate(Index = as.numeric(Index)) %>%
  group_by(Year) %>%
  summarise(BAKUN = mean(Index, na.rm = TRUE)) %>%
  dplyr::select(Year, BAKUN) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

Biologically Effective Upwelling Transport Index (39N)

Summary: BEUTI is a new upwelling index that leverages state-of-the-art ocean models as well as satellite and in situ data to improve upon historically available upwelling indices for the U.S. west coast. BEUTI provides estimates of vertical nitrate flux near the coast (i.e., the amount of nitrate upwelled/downwelled), which may be more relevant than upwelling strength when considering some biological responses. See Jacox, M. G., C. A. Edwards, E. L. Hazen, and S. J. Bograd (2018) Coastal upwelling revisited: Ekman, Bakun, and improved upwelling indices for the U.S. west coast. Journal of Geophysical Research, doi:10.1029/2018JC014187.

Institution: NOAA/SWFSC/ERD Time span: 1988-01-01T00:00:00Z - 2020-04-01T00:00:00Z

Note: Check Latitude (39)

```
BEUTIraw <- read.csv(here("data", "covariates", "cciea_OC_BEUTI_784c_ef9f_af6f.csv"))

BEUTI_full <- BEUTIraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time)) %>%
  mutate(beuti = as.numeric(beuti)) %>%
  group_by(Year) %>%
  summarise(BEUTI = mean(beuti, na.rm = TRUE)) %>%
  dplyr::select(Year, BEUTI) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()

## for January-June only
BEUTI <- BEUTIraw %>%
  slice(-1) %>%
  mutate(time = ymd_hms(time)) %>%
  mutate(Year = year(time),
         Month = month(time)) %>%
  mutate(beuti = as.numeric(beuti)) %>%
  filter(Month == 1:6) %>%
  group_by(Year) %>%
  summarise(BEUTI = mean(beuti, na.rm = TRUE)) %>%
  dplyr::select(Year, BEUTI) %>%
  filter(Year >= 2009 & Year <= 2017) %>%
  arrange(Year) %>%
  as.tibble()
```

```
## Warning in Month == 1:6: longer object length is not a multiple of shorter
## object length
```

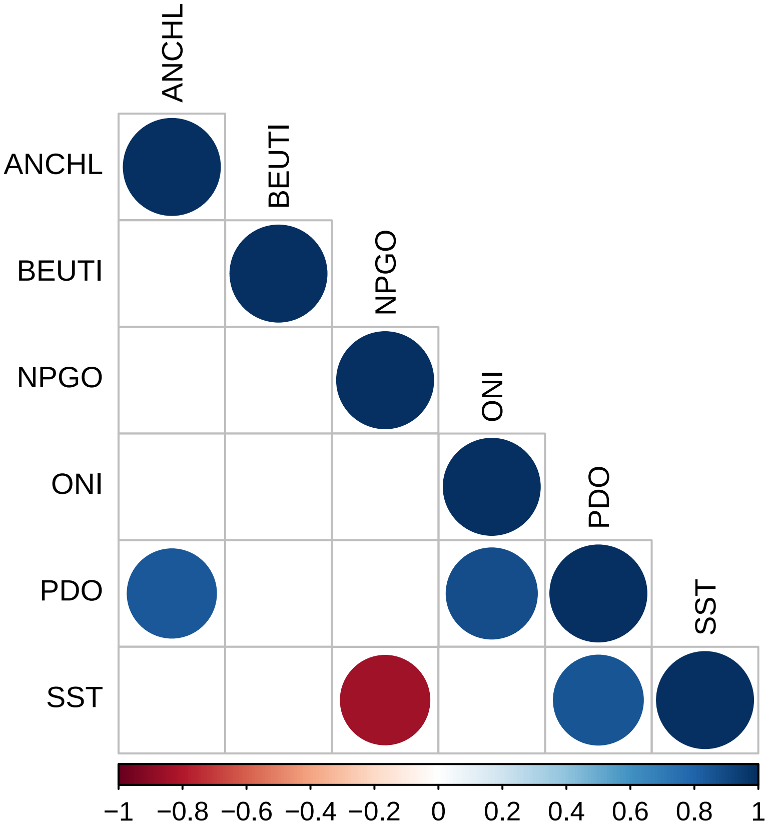
Final Covariates

Based on research and discussion, we selected the final candidate covariates for our model: 1. Larval anchovy - SCC (ANCHL) 2. Biologically Effective Upwelling Transport Index (BEUTI) 3. Sea Surface Temperature (SST) 4. North Pacific Gyre Oscillation Index (NPGO) 5. Pacific Decadal Oscillation Index (PDO) 6. Oceanic Nino Index (ONI)

Correlation

Look at correlation between the environmental covariates. Correlated covariates should not be in the same model together.

```
## bind covars
covars <- cbind(ANCHL[,2], BEUTI[,2], NPGO[,2], ONI[,2], PDO[,2], SST[,2])
colnames(covars) <- c("ANCHL", "BEUTI", "NPGO", "ONI", "PDO", "SST")

# covars <- cbind(ANCHA[,2], ANCHL[,2], BAKUN[,2], BEUTI[,2], CUTI[,2],
#                 KRILL[,2], NPGO[,2], ONI[,2], PDO[,2], SST[,2])
# colnames(covars) <- c("ANCHA", "ANCHL", "BAKUN", "BEUTI", "CUTI",
#                       "KRILL", "NPGO", "ONI", "PDO", "SST")

## scale covariates
covars <- covars %>%
  mutate_all(., scale) # see ?scale

## check for correlation between predictors (cutoff >0.65)
cp <- as.data.frame((round(cor(covars, use="complete.obs"), 2)))

## this function flattens your data in a particular way, used below
flattenCorrMatrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
    column = rownames(cormat)[col(cormat)[ut]],
    cor  =(cormat)[ut],
    p = pmat[ut]
  )
}

cor <- rcorr(as.matrix(covars))

## creates a new formatted df
cor_vals <- flattenCorrMatrix(cor$r, cor$P) %>% arrange(cor)

## makes a plot
corrplot(cor$r, type = "lower", order = "original", p.mat = cor$P,
         insig = "blank", sig.level = 0.01, tl.col = "black", tl.cex = .9, number.cex = .2)
```



Positive correlations are displayed in blue and negative correlations in red color. Color intensity and the size of the circle are proportional to the correlation coefficients.

We selected +/- 0.65 as our cutoff for correlation (Note: Check with Sarah). The following covariates should not be in the same model: 1. NPGO and SST 2. NPGO and PDO 3. BEUTI and PDO 4. ANCHL and SST 5. ONI and SST 6. ANCHL and ONI 7. ANCHL and PDO 8. PDO and SST 9. ONI and PDO

Note: Why aren't all of the correlations shown in the plot?

Number of Candidate Models

```
## create data frame specifying predictors to include
predictors <- as.data.frame(matrix(c(FALSE, TRUE), 2, 6)) # 6 potential predictors

## add column names
```

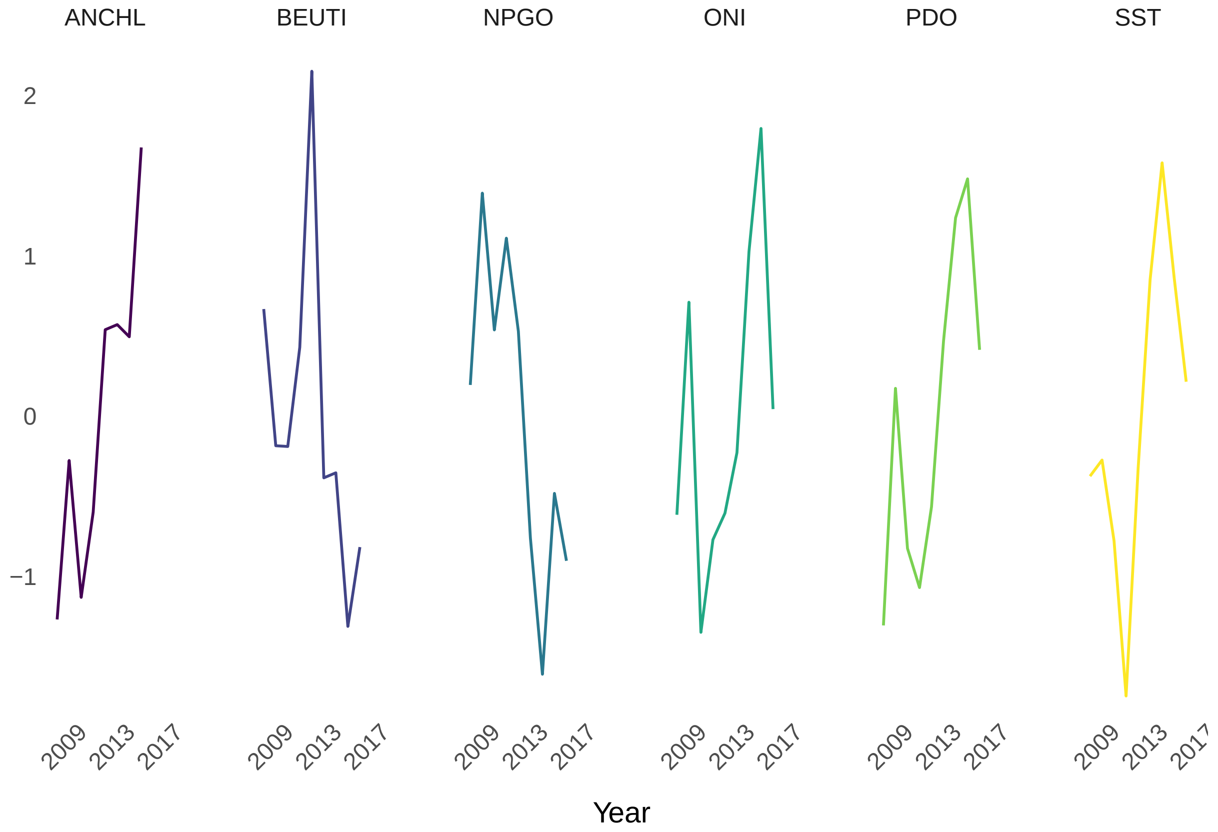
```
cov_names <- colnames(predictors) <- colnames(covars)

## create set of all possible combinations
full_set <- expand.grid(predictors) # 64 combinations

## select models with correlated predictors
ii <- which(full_set$SST + full_set$NPGO == 2 |
              full_set$NPGO + full_set$PDO == 2 |
              full_set$BEUTI + full_set$PDO == 2 |
              full_set$ANCHL + full_set$SST == 2 |
              full_set$ONI + full_set$SST == 2 |
              full_set$ANCHL + full_set$ONI == 2 |
              full_set$ANCHL + full_set$PDO == 2 |
              full_set$PDO + full_set$SST == 2 |
              full_set$ONI + full_set$PDO == 2) # 49 models

## create reduced set of models and convert to a matrix for easier indexing
use_set <- as.matrix(full_set[-ii,]) # 15 models

## number of models in set
(n_mods <- nrow(use_set)) # 15 models out of potential 64
```

```
## [1] 15
```

Final Covariates Plot

```
covarsplot <- covars %>%
  mutate(year = 2009:2017) %>%
  pivot_longer(cols = 1:6, names_to = "covariate", values_to = "value")

ggplot(data = covarsplot, aes(x = year, y = value, color = covariate)) +
  geom_line() +
  facet_wrap(~covariate, nrow = 1) +
  scale_color_viridis(discrete = TRUE) +
  theme_minimal() + xlab("Year") + ylab("") +
  scale_x_discrete(breaks = c(2009, 2013, 2017), limits = c(2009, 2013, 2017)) +
  theme(legend.position = "none",
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_text(angle = 45),
        panel.spacing = unit(2.5, "lines"))
```

```
## Warning: Continuous limits supplied to discrete scale.
## Did you mean ‘limits = factor(...)‘ or ‘scale_*_continuous()‘?

## Warning: Removed 1 row(s) containing missing values (geom_path).
```



Export Out

```
finalcovars <- covars %>%
  mutate(Year = 2009:2017) %>%
  select(Year, everything())

# write.csv(x = finalcovars, file = here("data", "covariates", "covars.csv"))
```