

Technical Documentation, State of the Ecosystem Report

Northeast Fisheries Science Center

27 December 2019

Contents

Introduction	1
1 Data and Code Access	3
2 Aggregate Groups	7
2.1 Methods	7
3 Annual SST Cycles	17
3.1 Methods	17
4 Aquaculture	29
4.1 Methods	29
5 Bennet Indicator	33
5.1 Methods	33
6 Bottom temperatures	41
6.1 Methods	41
7 Catch and Fleet Diversity	47
7.1 Methods	47
8 Chesapeake Bay Water Quality Standards Attainment	57
8.1 Methods	57
9 Chlorophyll <i>a</i> and Primary Production	63
9.1 Methods	63

10 Fishing Community Climate Vulnerability	73
10.1 Methods	73
11 Conceptual Models	81
11.1 Methods	81
12 Fish Condition Indicator	95
12.1 Methods	95
13 Ecological Production Units	103
13.1 Methods	103
14 Gulf Stream Index	109
14.1 Methods	109
15 Harbor Porpoise Bycatch	113
15.1 Methods	113
16 Ichthyoplankton Diversity	117
16.1 Methods	117
17 Commercial Landings Data	125
17.1 Methods	125
18 Long-term Sea Surface Temperature	181
18.1 Methods	181
19 Mid-Atlantic Harmful Algal Bloom Indicator	189
19.1 Methods	189
20 New England Harmful Algal Bloom Indicator	195
20.1 Methods	195
21 Verified Records of Southern Kingfish	201
21.1 Methods	201

22 Habitat Occupancy Models	209
22.1 Methods	209
23 Fish Productivity Indicator	215
23.1 Methods	215
24 Recreational Fishing Indicators	221
24.1 Methods	221
25 Right Whale Abundance	233
25.1 Methods	233
26 Seabird diet and productivity	237
26.1 Methods	237
27 Seasonal SST Anomalies	245
27.1 Methods	245
28 Single Species Status Indicator	249
28.1 Methods	249
29 Slopewater proportions	255
29.1 Methods	255
30 Species Density Estimates	259
30.1 Methods	259
31 Species Distribution Indicators	269
31.1 Methods	269
32 Survey Data	283
32.1 Methods	283
33 Thermal Habitat Projections	297
33.1 Methods	297

34 Trend Analysis	301
34.1 Methods	301
35 Zooplankton	313
35.1 Methods	313
References	329

Introduction

The purpose of this document is to collate the methods used to access, collect, process, and analyze derived data (“indicators”) used to describe the status and trend of social, economical, ecological, and biological conditions in the Northeast Shelf Large Marine Ecosystem (see figure, below). These indicators are further synthesized in State of the Ecosystem Reports produced annually by the Northeast Fisheries Science Center for the New England Fisheries Management Council and the Mid-Atlantic Fisheries Management Council. The metadata for each indicator (in accordance with the Public Access to Research Results (PARR) directive) and the methods used to construct each indicator are described in the subsequent chapters, with each chapter title corresponding to an indicator or analysis present in State of the Ecosystem Reports.

Indicators included in this document were selected to clearly align with management objectives, which is required for integrated ecosystem assessment (Levin et al. 2009), and has been advised many times in the literature (Degnbol and Jarre 2004; Jennings 2005; Rice and Rochet 2005; Link 2005). A difficulty with practical implementation of this in ecosystem reporting can be the lack of clearly specified ecosystem-level management objectives (although some have been suggested (Murawski 2000)). In our case, considerable effort had already been applied to derive both general goals and operational objectives from both US legislation such as the Magnuson-Stevens Fisheries Conservation and Management Act (MSA) and regional sources (DePiper et al. 2017). These objectives are somewhat general and would need refinement together with managers and stakeholders, however, they serve as a useful starting point to structure ecosystem reporting.

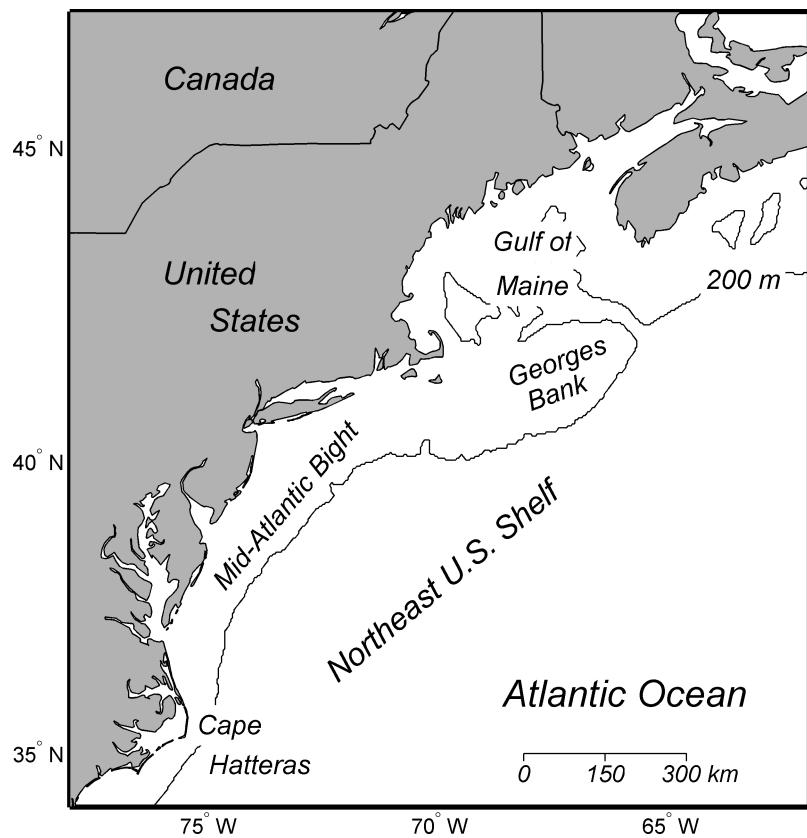


Figure 1: Map of Northeast U.S. Continental Shelf Large Marine Ecosystem from Hare et al. (2016).

Chapter 1

Data and Code Access

1.0.1 About

The Technical Documentation for the State of the Ecosystem (SOE) reports is a bookdown document; hosted on the NOAA Northeast Fisheries Science Center (NEFSC) Ecosystems Dynamics and Assessment Branch Github page, and developed in R. Derived data used to populate figures in this document are queried directly from the ecodata R package or the NEFSC ERDDAP server. ERDDAP queries are made using the R package `erddap`.

1.0.2 Accessing data and build code

In this technical documentation, we hope to shine a light on the processing and analytical steps involved to get from source data to final product. This means that whenever possible, we have included the code involved in source data extraction, processing, and analyses. We have also attempted to thoroughly describe all methods in place of or in supplement to provided code. Example plotting code for each indicator is presented in sections titled “Plotting”, and these code chunks can be used to recreate the figures found in ecosystem reporting documents where each respective indicator was included¹.

Source data for the derived indicators in this document are linked to in the text unless there are privacy concerns involved. In that case, it may be possible to access source data by reaching out to the Point of Contact associated with that data set. Derived data sets make up the majority of the indicators presented in

¹There are multiple R scripts sourced throughout this document in an attempt to keep code concise. These scripts include `BasePlot_source.R`, `GIS_source.R`, and `get_erddap.R`. The scripts `BasePlot_source.R` and `GIS_source.R` refer to deprecated code used prior to the 2019 State of the Ecosystem reports. Indicators that were not included in reports after 2018 make use of this syntax, whereas newer indicators typically use `ggplot2` for plotting.

ecosystem reporting documents, and these data sets are available for download through the `ecodata` R package.

1.0.3 Building the document

Start a local build of the SOE bookdown document by first cloning the project’s associated git repository. Next, if you would like to build a past version of the document, use `git checkout [version_commit_hash]` to revert the project to a past commit of interest, and set `build_latest <- FALSE` in the following code chunk. This will ensure the project builds from a cached data set, and not the most updated versions present on the NEFSC ERDDAP server. Once the `tech-doc.Rproj` file is opened in RStudio, run `bookdown::serve_book()` from the console to build the document.

```
build_latest <- FALSE

if (build_latest) {

  # Relative working directories
  data.dir <- here::here("data")
  r.dir <- here::here("R")

  # Source function for querying ERDDAP server
  source(file.path(r.dir, "get_erddap.R"))

  # Set URL for COMET (server where NEFSC ERDDAP lives)
  comet <- "https://comet.nefsc.noaa.gov/erddap/"

  # List datasets on the NEFSC ERDDAP
  tab_list <- ed_datasets(url = comet)

  # Get updated data set IDs
  erddap_datasets <- tab_list %>% filter(str_detect(Dataset.ID,
    "soe_v")) %>% get_erddap(id = NULL)

  # Save and clean updated IDs for use in rest of report
  save(erddap_datasets, file = file.path(data.dir, "ERDDAP_datasets.Rdata"))

  # Exclude stock assessment status data, which have unique
  # structure
  erddap_datasets <- erddap_datasets %>% dplyr::filter(!str_detect(Dataset.ID,
    "assess"))

  # Create SOE parent data set, filter out NAs. This queries
  # based on data set IDs that were collected above
```

```
SOE.data.erd <- sprintf("http://comet.nefsc.noaa.gov/erddap/tabledap/%s.csv",
  erddap_datasets$Dataset.ID) %>% purrr::map(function(x) {
  readr::read_csv(url(x))
}) %>% do.call(rbind, .) %>% mutate(Value = as.numeric(Value)) %>%
  dplyr::filter(!is.na(Value))

# Convert to data.table
SOE.data <- as.data.table(SOE.data.erd)

# Save data
save(SOE.data, file = file.path(data.dir, "SOE_data_erddap.Rdata"))
}
```

1.0.3.1 A note on data structures

The majority of the derived time series used in State of the Ecosystem reports are in long format. This approach was taken so that all disparate data sets could be “bound” together for ease of use in our base plotting functions.

Chapter 2

Aggregate Groups

Description: Mappings of species into aggregate group categories for different analyses

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Synthesis of published information

Contributor(s): Geret DePiper, Sarah Gaichas, Sean Hardison, Sean Lucey

Data steward: Sean Lucey Sean.Lucey@noaa.gov

Point of contact: Sean Lucey Sean.Lucey@noaa.gov

Public availability statement: Source data is available to the public (see Data Sources).

2.1 Methods

The State of the Ecosystem (SOE) reports are delivered to the New England Fishery Management Council (NEFMC) and Mid-Atlantic Fishery Management Council (MAFMC) to provide ecosystems context. To better understand that broader ecosystem context, many of the indicators are reported at an aggregate level rather than at a single species level. Species were assigned to an aggregate group following the classification scheme of Garrison and Link (2000) and Link et al. (2006). Both works classified species into feeding guilds based on food habits data collected at the Northeast Fisheries Science Center (NEFSC). In 2017, the SOE used seven specific feeding guilds (plus an “other” category; Table 2.1). These seven were the same guilds used in Garrison and Link (2000), which also distinguished ontogenetic shifts in species diets.

Table 2.1: Aggregate groups use in 2017 SOE. Classifications are based on @garrison2000dietary .

Feeding.Guild	Description
Apex Predator	Top of the food chain
Piscivore	Fish eaters
Macrozoo-piscivore	Shrimp and small fish eaters
Macroplanktivore	Amphipod and shrimp eaters
Mesoplanktivore	Zooplankton eaters
Benthivore	Bottom eaters
Benthos	Things that live on the bottom
Other	Things not classified above

Table 2.2: Aggregate groups use in 2018 SOE. Classifications are based on @link2006EMAX.

Feeding.Guild	Description
Apex Predator	Top of the food chain
Piscivore	Fish eaters
Planktivore	Zooplankton eaters
Benthivore	Bottom eaters
Benthos	Things that live on the bottom
Other	Things not classified above

For the purposes of the SOE, species were only assigned to one category based on the most prevalent size available to commercial fisheries. However, several of those categories were confusing to the management councils, so in 2018 those categories were simplified to five (plus “other”; Table 2.2) along the lines of Link et al. (2006). In addition to feeding guilds, species managed by the councils have been identified. This is done to show the breadth of what a given council is responsible for within the broader ecosystem context.

2.1.1 Data sources

In order to match aggregate groups with various data sources, a look-up table was generated which includes species’ common names (COMNAME) along with their scientific names (SCINAME) and several species codes. SVSPP codes are used by the NEFSC Ecosystems Surveys Branch (ESB) in their fishery-independent Survey Database (SVDBS), while NESPP3 codes refer to the codes used by the Commercial Fisheries Database System (CFDBS) for fishery-dependent data. A third species code provided is the ITISSPP, which

refers to species identifiers used by the Integrated Taxonomic Information System (ITIS). Digits within ITIS codes are hierarchical, with different positions in the identifier referring to higher or lower taxonomic levels. More information about the SVDBS, CFDBS, and ITIS species codes are available in the links provided below.

Management responsibilities for different species are listed under the column “Fed.managed” (NEFMC, MAFMC, or JOINT for jointly managed species). More information about these species is available on the FMC websites listed below. Species groupings listed in the “NEIEA” column were developed for presentation on the Northeast Integrated Ecosystem Assessment (NE-IEA) website. These groupings are based on EMAX groupings (Link et al. 2006), but were adjusted based on conceptual models developed for the NE-IEA program that highlight focal components in the Northeast Large Marine Ecosystem (i.e. those components with the largest potential for perturbing ecosystem dynamics). NE-IEA groupings were further simplified to allow for effective communication through the NE-IEA website.

2.1.1.1 Supplemental information

See the following links for more information regarding the NEFSC ESB Bottom Trawl Survey, CFDBS, and ITIS:

- <https://www.itis.gov/>
- <https://inport.nmfs.noaa.gov/inport/item/22561>
- <https://inport.nmfs.noaa.gov/inport/item/22560>
- <https://inport.nmfs.noaa.gov/inport/item/27401>

More information about the NE-IEA program is available here.

More information about the New England Fisheries Management Council is available here.

More information about the Mid-Atlantic Fisheries Management Council is available here.

2.1.2 Data extraction

Species lists are pulled from SVDBS and CFDBS. They are merged using the ITIS code. Classifications from Garrison and Link (Garrison and Link 2000) and Link et al. (Link et al. 2006) are added manually. The R code used in the extraction process is presented below.

```

#Species list
if(Sys.info()['sysname']=="Windows"){
  data.dir <- "L:\\EcoAP\\Data\\survey"
  out.dir <- "L:\\EcoAP\\Data\\survey"
  memory.limit(4000)
}
if(Sys.info()['sysname']=="Linux"){
  data.dir <- "/home/sluey/sluey/EcoAP/Data/survey"
  out.dir <- "/home/sluey/sluey/EcoAP/Data/survey"
  out.dir.2 <- "/home/sluey/sluey/EcoAP/Data/Commercial"
  uid <- 'sluey'
  cat("Oracle Password: ")
  pwd <- scan(stdin(), character(), n = 1)
}

library(RODBC); library(data.table)

if(Sys.info()['sysname']=="Windows"){
  channel <- odbcDriverConnect()
} else {
  channel <- odbcConnect('sole', uid, pwd)
}

#Grab svsp code by itis code
svsp <- as.data.table(sqlQuery(channel, "select SVSPP, ITISSPP, COMNAME, SCINAME
                                         from ITIS_Lookup"))

#Grab cfspp by itis code
cfspp <- as.data.table(sqlQuery(channel, "select NESPP4, SPECIES_ITIS,
                                         COMMON_NAME, SCIENTIFIC_NAME
                                         from CFDBS.Species_itis_ne"))
setnames(cfspp, 'SPECIES_ITIS', 'ITISSPP')
cfspp[, NESPP3 := as.numeric(substr(sprintf('%04d', NESPP4), 1, 3))]
setkey(cfspp, NESPP3)
cfspp <- unique(cfspp)
cfspp[, NESPP4 := NULL]

setkey(cfspp, ITISSPP, NESPP3)
cfspp <- unique(cfspp, by = key(cfspp))

#Merge to master species list
spp <- merge(svsp, cfspp, by = 'ITISSPP', all = T)
spp <- spp[!(is.na(SVSPP) & is.na(NESPP3)), ]

#Fix known issues

```

```

spp <- spp[!SVSPP %in% c(193, 310), ]

spp[ITISSPP == 630979, SVSPP := 193] #Ocean Pout
spp[ITISSPP == 620992, SVSPP := 310] #Deepsea red crab
spp[ITISSPP == 172783, SVSPP := 104] #Fourspot flounder
spp[ITISSPP == 166283, SVSPP := 112] #John Dory
spp[ITISSPP == 161731, SVSPP := 36] #Menhaden
spp[ITISSPP == 98671, SVSPP := 311] #Cancer Crabs unk
spp[ITISSPP == 98455, SVSPP := 317] #Spider crabs
spp[ITISSPP == 159772, NESPP3 := 150] #Hagfish
spp[ITISSPP == 166284, NESPP3 := 188] #John Dory
spp[ITISSPP == 98670, NESPP3 := 714] #Cancer Crabs unk

spp[ITISSPP %in% c(630979, 620992), COMNAME := COMMON_NAME]
spp[ITISSPP %in% c(630979, 620992), SCINAME := SCIENTIFIC_NAME]

#Drop extra columns
spp[is.na(COMNAME), COMNAME := COMMON_NAME]
spp[is.na(SCINAME), SCINAME := SCIENTIFIC_NAME]
spp[, c('COMMON_NAME', 'SCIENTIFIC_NAME') := NULL]
setcolorder(spp, c('ITISSPP', 'SVSPP', 'NESPP3', 'COMNAME', 'SCINAME'))
setkey(spp, SVSPP, NESPP3)

#-----
#Add management authority
mafmc <- c(103, 121, 131, 135, 141, 143, 151, 403, 409, 502, 503, 621)
nefmc <- c(22:28, 32, 69, 72:77, 101, 102, 105:107, 155, 193, 310, 401, 894)
joint <- c(15, 197)
spp[SVSPP %in% mafmc, MANAGE := 'MAFMC']
spp[SVSPP %in% nefmc, MANAGE := 'NEFMC']
spp[SVSPP %in% joint, MANAGE := 'JOINT']

#-----
#Add functional groups
#From NEFMC EBFM PDT work
#See Functional_group_table_Mike.csv in slucey/EcoAP/EBFM_PDT
spp[, EBFM.PDT := factor(NA, levels = c('Apex Predator', 'Benthivore', 'Benthos',
                                         'Macroplanktivore', 'Macrozoo-piscivore',
                                         'Mesoplanktivore', 'Piscivore', 'Other'))]
spp[SVSPP %in% c(11, 700, 704, 747), EBFM.PDT := 'Apex Predator']
spp[SVSPP %in% c(14, 22, 25, 74, 102, 105, 106, 107, 141, 143, 151, 164, 172,
                176, 177, 192, 193, 301, 310, 312, 314, 317, 322, 501),
     EBFM.PDT := 'Benthivore']
spp[SVSPP %in% c(331, 336, 401, 403, 409), EBFM.PDT := 'Benthos']
spp[NESPP3 %in% c(775, 781, 805, 806), EBFM.PDT := 'Benthos']

```

```

spp[SVSPP %in% c(76, 163, 168, 171, 502, 503), EBFM.PDT := 'Macroplanktivore']
spp[SVSPP %in% c(13, 24, 26, 27, 75, 77, 84, 108, 112, 155, 156, 311),
    EBFM.PDT := 'Macrozoo-piscivore']
spp[SVSPP %in% c(32, 33, 34, 35, 36, 121, 131), EBFM.PDT := 'Mesoplanktivore']
spp[SVSPP %in% c(15, 23, 28, 69, 72, 73, 101, 103, 104, 135, 139, 145, 197),
    EBFM.PDT := 'Piscivore']
spp[is.na(EBFM.PDT), EBFM.PDT := 'Other']

#Fix known issues
spp[SVSPP == 160, EBFM.PDT := 'Macroplanktivore']

#Add EMAX
load(file.path(data.dir, 'EMAX_groups.RData'))
emax <- emax[!is.na(SVSPP), ]
spp <- merge(spp, emax[, list(SVSPP, EMAX, Fall.q, Spring.q)], by = 'SVSPP', all.x = T)

#reduce rows
setkey(spp, SVSPP, NESPP3)
species <- unique(spp, by = key(spp))

#Remove EMAX q's
species[, c('Fall.q', 'Spring.q') := NULL]

#Expand EMAX abbreviations
species[EMAX == 'BG', EMAX := 'Benthivore Gadiformes']
species[EMAX == 'BE', EMAX := 'Benthivore Elasmobranchs']
species[EMAX == 'BP', EMAX := 'Benthivore Pleuronectiformes']
species[EMAX == 'BS', EMAX := 'Benthivore Scorpaeniformes']
species[EMAX == 'BO', EMAX := 'Benthivore Others']
species[EMAX == 'BC', EMAX := 'Benthivore Perciformes']
species[EMAX == 'PG', EMAX := 'Piscivore Gadiformes']
species[EMAX == 'PE', EMAX := 'Piscivore Elasmobranchs']
species[EMAX == 'PO', EMAX := 'Piscivore Others']
species[EMAX == 'OE', EMAX := 'Omnivore Elasmobranchs']
species[EMAX == 'OO', EMAX := 'Omnivore Others']
species[EMAX == 'SC', EMAX := 'Southern Perciformes']
species[EMAX == 'SP', EMAX := 'Southern Pleuronectiformes']
species[EMAX == 'SE', EMAX := 'Southern Elasmobranchs']
species[EMAX == 'SG', EMAX := 'Southern Gadiformes']
species[EMAX == 'SO', EMAX := 'Southern Others']

#Rename columns
setnames(species, c('MANAGE', 'EBFM.PDT', 'Feeding.guild'), c('Fed.Managed', 'SOE.17', 'Link.2000'))

#Add classification from Garrison/Link 2000

```

```

size <- data.table(COMNAME = c(rep('SMOOTH DOGFISH', 2),
                                rep('SPINY DOGFISH', 3),
                                rep('WINTER SKATE', 4),
                                rep('LITTLE SKATE', 2),
                                rep('SMOOTH SKATE', 1),
                                rep('THORNY SKATE', 4),
                                rep('ATLANTIC HERRING', 2),
                                rep('ALEWIFE', 1),
                                rep('SILVER HAKE', 3),
                                rep('ATLANTIC COD', 4),
                                rep('HADDOCK', 3),
                                rep('POLLOCK', 4),
                                rep('WHITE HAKE', 3),
                                rep('RED HAKE', 3),
                                rep('SPOTTED HAKE', 2),
                                rep('AMERICAN PLAICE', 3),
                                rep('SUMMER FLOUNDER', 2),
                                rep('FOURSPOT FLOUNDER', 2),
                                rep('YELLOWTAIL FLOUNDER', 3),
                                rep('WINTER FLOUNDER', 3),
                                rep('WITCH FLOUNDER', 2),
                                rep('WINDOWPANE', 2),
                                rep('GULF STREAM FLOUNDER', 1),
                                rep('ATLANTIC MACKEREL', 3),
                                rep('BUTTERFISH', 1),
                                rep('BLUEFISH', 3),
                                rep('ATLANTIC CROAKER', 2),
                                rep('BLACK SEA BASS', 2),
                                rep('SCUP', 1),
                                rep('WEAKFISH', 2),
                                rep('ACADIAN REDFISH', 1),
                                rep('LONGHORN SCULPIN', 2),
                                rep('SEA RAVEN', 2),
                                rep('NORTHERN SAND LANCE', 1),
                                rep('OCEAN POUT', 1),
                                rep('FAWN CUSK-EEL', 1),
                                rep('GOOSEFISH', 1),
                                rep('ATLANTIC SHARPNOSE SHARK', 1),
                                rep('NORTHERN SHORTFIN SQUID', 1),
                                rep('LONGFIN SQUID', 1)),
SizeCat = c('M', 'L', 'S', 'M', 'L', 'S', 'M', 'L', 'XL', 'S',
           'M', 'M', 'S', 'M', 'L', 'XL', 'S', 'M', 'M', 'S',
           'M', 'L', 'S', 'M', 'L', 'XL', 'S', 'M', 'L', 'S',
           'M', 'L', 'XL', 'S', 'M', 'L', 'S', 'M', 'L', 'S',
           'M', 'S', 'M', 'L', 'M', 'L', 'S', 'M', 'S', 'M',
           'M', 'S', 'M', 'L', 'M', 'L', 'S', 'M', 'S', 'M')

```

```

'L', 'S', 'M', 'L', 'M', 'L', 'S', 'M', 'S', 'S',
'M', 'L', 'S', 'S', 'M', 'L', 'S', 'M', 'S', 'M',
'M', 'S', 'M', 'M', 'S', 'M', 'S', 'M', 'M', 'L',
'L', 'L', 'L', 'L', 'L'),
Min.size = c(41, 61, 10, 41, 61, 10, 31, 61, 81, 10, 31, 31,
          10, 31, 61, 81, 10, 21, 21, 10, 21, 41, 10, 21,
          51, 81, 10, 21, 51, 10, 21, 51, 81, 10, 21, 41, 10, 21,
          10, 21, 41, 21, 41, 10, 21, 10, 21, 41, 10, 21,
          41, 21, 41, 10, 21, 10, 10, 21, 36, 10, 10, 31,
          71, 10, 26, 10, 26, 26, 10, 26, 26, 10, 26, 10,
          26, 11, 61, 61, 61, 31, 31),
Garrison.Link = as.character(c(1, 1, 2, 2, 6, 3, 3, 6, 6, 3, 3, 4,
                               6, 2, 2, 2, 4, 4, 6, 3, 3, 6, 6, 5, 5, 5, 5, 4,
                               4, 4, 4, 3, 4, 6, 3, 3, 4, 3, 6, 5, 5, 5, 6,
                               6, 3, 6, 3, 5, 5, 5, 5, 5, 5, 5, 3, 3, 5, 2,
                               2, 2, 2, 6, 6, 6, 5, 5, 1, 1, 5, 6, 6, 4, 3,
                               3, 6, 6, 2, 5, 3, 6, 6, 2, 2))
)

#Give Garrison.link guilds meaniful names
size[Garrison.Link == 1, Garrison.Link := 'Crab Eaters']
size[Garrison.Link == 2, Garrison.Link := 'Planktivores']
size[Garrison.Link == 3, Garrison.Link := 'Amphipod/Shrimp Eaters']
size[Garrison.Link == 4, Garrison.Link := 'Shrimp/Small Fish Eaters']
size[Garrison.Link == 5, Garrison.Link := 'Benthivores']
size[Garrison.Link == 6, Garrison.Link := 'Piscivores']

#Merge Garrison Link into species list
species <- merge(species, size, by = 'COMNAME', all = T)

#Set column order
setcolororder(species, c('ITISSPP', 'SVSPP', 'NESPP3', 'COMNAME', 'SCINAME',
                        'SizeCat', 'Min.size', 'Fed.Managed', 'Garrison.Link',
                        'SOE.17', 'SOE.18', 'EMAX'))

#Add NE.IEA.web category
#get rerddap
devtools::install_github("ropensci/rerddap")
library(rerddap)

#Set URL for COMET (server where NEFSC ERDDAP lives)
comet <- 'https://comet.nefsc.noaa.gov/erddap/'

#List datasets on the NEFSC ERDDAP
tab_list <- ed_datasets(url = comet)

```

```
#download a tabular dataset (input is dataset id; this works on vectors)
ed_spp <- as.data.table(sprintf("https://comet.nefsc.noaa.gov/erddap/tabledap/%s.csv", "species_g

  purrr::map(function(x) {

    readr::read_csv(url(x))

  }))

neiea <- unique(ed_spp[!is.na(NEIEA), list(COMNAME, NEIEA)])

species <- merge(species, neiea, by = 'COMNAME', all = T)

save(species, file = file.path(out.dir, 'SOE_species_list.RData'))
```


Chapter 3

Annual SST Cycles

Description: Annual SST Cycles

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018),
State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull with analysis

Contributor(s): Sean Hardison, Vincent Saba

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Kimberly Bastille, kimberly.bastille@noaa.gov

Public availability statement: Source data are available here.

3.1 Methods

3.1.1 Data sources

Data for annual sea surface tempature (SST) cycles were derived from the NOAA optimum interpolation sea surface temperature (OISST) high resolution dataset (NOAA OISST V2 dataset) provided by NOAA's Earth System Research Laboratory's Physical Sciences Devision, Boulder, CO. The data extend from 1981 to present, and provide a $0.25^\circ \times 0.25^\circ$ global grid of SST measurements (Reynolds et al. 2007). Gridded SST data were masked according to the extent of Ecological Production Units (EPU) in the Northeast Large Marine Ecosystem (NE-LME) (See "EPU_Extended" shapefiles).

3.1.2 Data extraction

Daily mean sea surface temperature data for 2017 and for each year during the period of 1981-2012 were downloaded from the NOAA OI SST V2 site to derive the long-term climatological mean for the period. The use of a 30-year climatological reference period is a standard procedure for meteorological observing (WMO 2017). These reference periods serve as benchmarks for comparing current or recent observations, and for the development of standard anomaly data sets. The reference period of 1982-2012 was chosen to be consistent with previous versions of the State of the Ecosystem report.

R code used in extraction and processing:

```
# libraries
library(ncdf4)
library(dplyr)
library(readr)
library(tidyr)
library(sp)
library(rgdal)
library(raster)
library(stringr)

# get spatial polygons for Ecological Production Units (EPUs)
# that are used to clip SST data.
EPU <- readOGR("Extended_EPU")

map.crs <- CRS("+proj=longlat +lat_1=35 +lat_2=45 +lat_0=40 +lon_0=-77 +x_0=0
+y_0=0 +datum=NAD83 +no_defs +ellps=GRS80 +towgs84=0,0,0")

# find long term daily mean SSTs and 2017 SST anomaly
MAB_sst_daily_mean <- NULL
GB_sst_daily_mean <- NULL
GOM_sst_daily_mean <- NULL

# I split the data into three directories to loop through in
# separate R sessions concurrently.

for (dir. in 1:3) {

  # Loop through directories
  setwd(paste0("c:/users/sean.hardison/documents/sst_data/",
               dir.))
  print(getwd())
```

```

for (f in 1:length(list.files())) {

  if (!str_detect(list.files()[f], ".nc")) {
    print(paste(list.files()[f], "is not a raster")) #Based on file type
    next
  }

  for (j in c("MAB", "GB", "GOM")) {

    sub_region <- EPU[EPU@data$EPU == j, ]
    y <- as.numeric(str_extract(list.files()[f], "[0-9]+")) #get year

    for (i in 1:365) {
      print(paste(j, y, i))
      daily_mean <- raster(paste0(list.files()[f]),
                            band = i) #get band

      # set crs
      daily_mean@crs <- sub_region@proj4string

      # rotate to lon scale from 0-360 to -180-180
      daily_mean <- rotate(daily_mean)

      # mask raster with spatialpolygon
      daily_mean_clipped <- mask(daily_mean, sub_region)

      # add mean value to data.frame
      assign(paste0(j, "_sst_daily_mean"), rbind(get(paste0(j,
      "_sst_daily_mean")), c(mean(daily_mean_clipped@data@values,
      na.rm = T), y, i)))
    }
  }

}

# Put results into data.frames
mab <- data.frame(EPU = "MAB", year = MAB_sst_daily_mean[, 2],
                  day = MAB_sst_daily_mean[, 3], Value = MAB_sst_daily_mean[, 1])

```

```

gb <- data.frame(EPU = "GB", year = GB_sst_daily_mean[, 2], day = GB_sst_daily_mean[, 3], Value = GB_sst_daily_mean[, 1])

gom <- data.frame(EPU = "GOM", year = GOM_sst_daily_mean[, 2], day = GOM_sst_daily_mean[, 3], Value = GOM_sst_daily_mean[, 1])

data3 <- rbind(mab, gb, gom)

# Save as 1 of 3 files (one for each directory containing
# daily mean data)
save(data3, file = "dir3_sst.Rdata")

#-----2017 SSTs-----
MAB_2017 <- NULL
GB_2017 <- NULL
GOM_2017 <- NULL

for (j in c("MAB", "GB", "GOM")) {

  sub_region <- EPU[EPU@data$EPU == j, ]

  for (i in 1:365) {
    print(paste(j, i))
    daily_mean <- raster("sst.day.mean.2017.nc", band = i) #get band

    # set crs
    daily_mean@crs <- sub_region@proj4string

    # rotate to lon scale from 0-360 to -180-180
    daily_mean <- rotate(daily_mean)

    # mask raster with spatialpolygon
    daily_mean_clipped <- mask(daily_mean, sub_region)

    # add mean value to data.frame
    assign(paste0(j, "_2017"), rbind(get(paste0(j, "_2017")),
      c(mean(daily_mean_clipped@data@values, na.rm = T),
      "2017", i)))
  }
}

```

```

}

# Put results into data.frames
mab_2017 <- data.frame(EPU = "MAB", year = MAB_2017[, 2], day = MAB_2017[, 3], Value = MAB_2017[, 1])

gb_2017 <- data.frame(EPU = "GB", year = GB_2017[, 2], day = GB_2017[, 3], Value = GB_2017[, 1])

gom_2017 <- data.frame(EPU = "GOM", year = GOM_2017[, 2], day = GOM_2017[, 3], Value = GOM_2017[, 1])

# Final 2017 daily mean data
sst_2017 <- rbind(mab_2017, gb_2017, gom_2017)
# save(sst_2017, file = 'sst_2017.Rdata')

```

3.1.3 Data analysis

We calculated the long-term mean and standard deviation of SST over the period of 1982-2012 for each EPU, as well as the daily mean for 2017.

```

#-----Load results-----#
load("dir1_sst.Rdata")
load("dir2_sst.Rdata")
load("dir3_sst.Rdata")
load("sst_2017.Rdata")

# Get long term mean and standard deviation
d <- rbind(data1, data2, data3)

ltm <- d %>% group_by(EPU, day) %>% dplyr::summarise(mean = mean(Value),
  sd = sd(Value))

```

3.1.4 Plotting

```

# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

```

```

##-----MAB-----#
par(mfrow = c(1, 3))
doy <- as.numeric(SOE.data[SOE.data$Var == "sst mean 2017 MAB",
  ]$Time)
val_2017 <- SOE.data[SOE.data$Var == "sst mean 2017 MAB", ]$Value
val_LT <- SOE.data[SOE.data$Var == "sst mean long term MAB",
  ]$Value
val_LT_sd <- SOE.data[SOE.data$Var == "sst sd long term MAB",
  ]$Value

# val_2017 <- approx(doy, val_2017, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y val_LT
# <- approx(doy, val_LT, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
# val_LT_sd <- approx(doy, val_LT_sd, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
doy <- seq(doy[1], doy[length(doy)], length.out = 365 * 1)

above_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT[i]) {
    above_mean[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT[i]) {
    above_mean[i] <- NA
  }
}

below_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] <= val_LT[i]) {
    below_mean[i] <- val_2017[i]
  } else if (val_2017[i] > val_LT[i]) {
    below_mean[i] <- NA
  }
}

above_sd <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT_sd[i] + val_LT[i]) {
    above_sd[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT_sd[i] + val_LT[i]) {
    above_sd[i] <- NA
  }
}

```

```

}

below_sd <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] <= val_LT[i] - val_LT_sd[i]) {
    below_sd[i] <- val_2017[i]
  } else if (val_2017[i] > val_LT[i] - val_LT_sd[i]) {
    below_sd[i] <- NA
  }
}

# Lines for polygons
above_sd[is.na(above_sd)] <- val_LT_sd[which(is.na(above_sd))] +
  val_LT[which(is.na(above_sd))]
below_sd[is.na(below_sd)] <- val_LT[which(is.na(below_sd))] -
  val_LT_sd[which(is.na(below_sd))]
above_mean[is.na(above_mean)] <- val_LT[which(is.na(above_mean))]
below_mean[is.na(below_mean)] <- val_LT[which(is.na(below_mean))]

upper <- val_LT_sd + val_LT
lower <- val_LT - val_LT_sd

# Null figure
plot(NULL, xlim = c(doy[1], doy[(length(doy))]), ylim = c(4,
  25), las = 1, ylab = "", yaxt = "n", xaxt = "n", xlab = "")
axis(2, cex.axis = 1.25, las = 1)
axis(1, labels = c("Jan", "Mar", "May", "July", "Sep", "Nov",
  "Jan"), at = c(1, 61, 122, 183, 245, 306, 365), cex.axis = 1.25)
mtext(2, line = 2.3, text = expression(paste("Mean SST (", degree,
  "C)")), cex = 1.1)
mtext(1, line = 2.5, text = "Time", cex = 1.1)
text(15, 25 * 0.95, "A", cex = 1.5)
# +/- 1 sd
polygon(c(doy, rev(doy)), c(upper, rev(lower)), col = "grey85",
  border = NA)

# Fills plot
polygon(c(doy, rev(doy)), c(below_mean + (val_LT - below_mean),
  rev(below_mean)), col = "lightblue", border = NA)
polygon(c(doy, rev(doy)), c(above_mean - (above_mean - val_LT),
  rev(above_mean)), col = "orange", border = NA)
polygon(c(doy, rev(doy)), c(above_sd - (above_sd - (val_LT +
  val_LT_sd)), rev(above_sd)), col = "red", border = NA)
polygon(c(doy, rev(doy)), c(below_sd + (below_sd - (val_LT -
  val_LT_sd)), rev(below_sd)), col = "blue", border = NA)

```

```

points(doy, val_LT, type = "l", lwd = 1, "grey90")

##-----GB-----#
```

```

doy <- as.numeric(SOE.data[SOE.data$Var == "sst mean 2017 GB",
  ]$Time)
val_2017 <- SOE.data[SOE.data$Var == "sst mean 2017 GB", ]$Value
val_LT <- SOE.data[SOE.data$Var == "sst mean long term GB", ]$Value
val_LT_sd <- SOE.data[SOE.data$Var == "sst sd long term GB",
  ]$Value
```

```

# val_2017 <- approx(doy, val_2017, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y val_LT
# <- approx(doy, val_LT, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
# val_LT_sd <- approx(doy, val_LT_sd, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
doy <- seq(doy[1], doy[length(doy)], length.out = 365 * 1)
```

```

above_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT[i]) {
    above_mean[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT[i]) {
    above_mean[i] <- NA
  }
}

below_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] <= val_LT[i]) {
    below_mean[i] <- val_2017[i]
  } else if (val_2017[i] > val_LT[i]) {
    below_mean[i] <- NA
  }
}

above_sd <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT_sd[i] + val_LT[i]) {
    above_sd[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT_sd[i] + val_LT[i]) {
```

```

        above_sd[i] <- NA
    }
}

below_sd <- NULL
for (i in 1:length(val_2017)) {
    if (val_2017[i] <= val_LT[i] - val_LT_sd[i]) {
        below_sd[i] <- val_2017[i]
    } else if (val_2017[i] > val_LT[i] - val_LT_sd[i]) {
        below_sd[i] <- NA
    }
}

# Lines for polygons
above_sd[is.na(above_sd)] <- val_LT_sd[which(is.na(above_sd))] +
    val_LT[which(is.na(above_sd))]
below_sd[is.na(below_sd)] <- val_LT[which(is.na(below_sd))] -
    val_LT_sd[which(is.na(below_sd))]
above_mean[is.na(above_mean)] <- val_LT[which(is.na(above_mean))]
below_mean[is.na(below_mean)] <- val_LT[which(is.na(below_mean))]

upper <- val_LT_sd + val_LT
lower <- val_LT - val_LT_sd

# Null figure
plot(NULL, xlim = c(doy[1], doy[(length(doy))]), ylim = c(4,
    21), las = 1, ylab = "", yaxt = "n", xaxt = "n", xlab = "")
axis(2, cex.axis = 1.25, las = 1)
axis(1, labels = c("Jan", "Mar", "May", "July", "Sep", "Nov",
    "Jan"), at = c(1, 61, 122, 183, 245, 306, 365), cex.axis = 1.25)
# mtext(2, line = 2.5, text = expression(paste('Mean SST
# (' , degree, 'C'))), cex = 1.1)
mtext(1, line = 2.5, text = "Time", cex = 1.1)
text(15, 21 * 0.95, "B", cex = 1.5)
# +/- 1 sd
polygon(c(doy, rev(doy)), c(upper, rev(lower)), col = "grey85",
    border = NA)

# Fills plot
polygon(c(doy, rev(doy)), c(below_mean + (val_LT - below_mean),
    rev(below_mean)), col = "lightblue", border = NA)
polygon(c(doy, rev(doy)), c(above_mean - (above_mean - val_LT),
    rev(above_mean)), col = "orange", border = NA)
polygon(c(doy, rev(doy)), c(above_sd - (above_sd - (val_LT +
    val_LT_sd)), rev(above_sd)), col = "red", border = NA)

```

```

polygon(c(doy, rev(doy)), c(below_sd + (below_sd - (val_LT -
  val_LT_sd)), rev(below_sd)), col = "blue", border = NA)
points(doy, val_LT, type = "l", lwd = 1, "grey90")

#-----#  

## SST GOM

doy <- as.numeric(SOE.data[SOE.data$Var == "sst mean 2017 GOM",
  ]$Time)
val_2017 <- SOE.data[SOE.data$Var == "sst mean 2017 GOM", ]$Value
val_LT <- SOE.data[SOE.data$Var == "sst mean long term GOM",
  ]$Value
val_LT_sd <- SOE.data[SOE.data$Var == "sst sd long term GOM",
  ]$Value

# val_2017 <- approx(doy, val_2017, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
# val_LT <- approx(doy, val_LT, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
# val_LT_sd <- approx(doy, val_LT_sd, xout =
# seq(doy[1], doy[length(doy)], length.out = 365*1))$y
doy <- seq(doy[1], doy[length(doy)], length.out = 365 * 1)

above_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT[i]) {
    above_mean[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT[i]) {
    above_mean[i] <- NA
  }
}

below_mean <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] <= val_LT[i]) {
    below_mean[i] <- val_2017[i]
  } else if (val_2017[i] > val_LT[i]) {
    below_mean[i] <- NA
  }
}

above_sd <- NULL

```

```

for (i in 1:length(val_2017)) {
  if (val_2017[i] >= val_LT_sd[i] + val_LT[i]) {
    above_sd[i] <- val_2017[i]
  } else if (val_2017[i] < val_LT_sd[i] + val_LT[i]) {
    above_sd[i] <- NA
  }
}

below_sd <- NULL
for (i in 1:length(val_2017)) {
  if (val_2017[i] <= val_LT[i] - val_LT_sd[i]) {
    below_sd[i] <- val_2017[i]
  } else if (val_2017[i] > val_LT[i] - val_LT_sd[i]) {
    below_sd[i] <- NA
  }
}

# Lines for polygons
above_sd[is.na(above_sd)] <- val_LT_sd[which(is.na(above_sd))] +
  val_LT[which(is.na(above_sd))]
below_sd[is.na(below_sd)] <- val_LT[which(is.na(below_sd))] -
  val_LT_sd[which(is.na(below_sd))]
above_mean[is.na(above_mean)] <- val_LT[which(is.na(above_mean))]
below_mean[is.na(below_mean)] <- val_LT[which(is.na(below_mean))]

upper <- val_LT_sd + val_LT
lower <- val_LT - val_LT_sd

# Null figure
plot(NULL, xlim = c(doy[1], doy[(length(doy))]), ylim = c(4,
  21), las = 1, ylab = "", yaxt = "n", xaxt = "n", xlab = "")
axis(2, cex.axis = 1.25, las = 1)
axis(1, labels = c("Jan", "Mar", "May", "July", "Sep", "Nov",
  "Jan"), at = c(1, 61, 122, 183, 245, 306, 365), cex.axis = 1.25)
# (2, line = 2.5, text = expression(paste('Mean SST
# (', degree, 'C'))), cex = 1.1)
mtext(1, line = 2.5, text = "Time", cex = 1.1)
text(15, 21 * 0.95, "C", cex = 1.5)
# +/- 1 sd
polygon(c(doy, rev(doy)), c(upper, rev(lower)), col = "grey85",
  border = NA)

# Fills plot
polygon(c(doy, rev(doy)), c(below_mean + (val_LT - below_mean),
  rev(below_mean)), col = "lightblue", border = NA)

```

```

polygon(c(doy, rev(doy)), c(above_mean - (above_mean - val_LT),
  rev(above_mean)), col = "orange", border = NA)
polygon(c(doy, rev(doy)), c(above_sd - (above_sd - (val_LT +
  val_LT_sd)), rev(above_sd)), col = "red", border = NA)
polygon(c(doy, rev(doy)), c(below_sd + (below_sd - (val_LT -
  val_LT_sd)), rev(below_sd)), col = "blue", border = NA)
points(doy, val_LT, type = "l", lwd = 1, "grey90")
box()

```

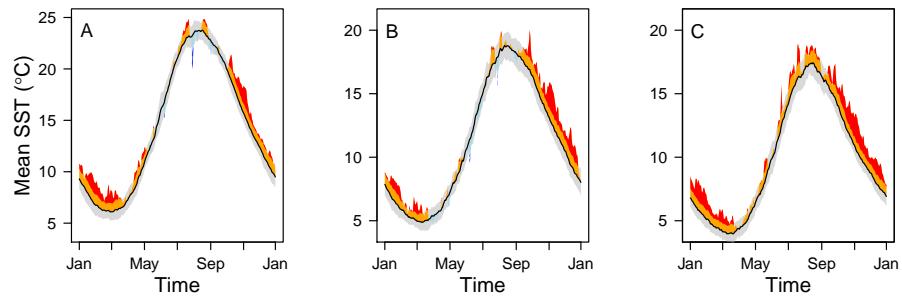


Figure 3.1: Long-term mean SSTs for the Mid-Atlantic Bight (A), Georges Bank (B), and Gulf of Maine (C). Orange and cyan shading show where the 2017 daily SST values were above or below the long-term mean respectively; red and dark blue shades indicate days when the 2017 mean exceeded ± 1 standard deviation from the long-term mean.

Chapter 4

Aquaculture

Description: Aquaculture indicators

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Synthesis of published information

Contributor(s): Sean Hardison, Lisa Calvo, Karl Roscher

Data steward: Kimberly Bastille kimberly.bastille@noaa.gov

Point of contact: Kimberly Bastille kimberly.bastille@noaa.gov

Public availability statement: Source data are publicly available in referenced reports, and are also available for download [here](#).

4.1 Methods

Aquaculture data included in the State of the Ecosystem (SOE) report were time series of number of oysters sold in Virginia, Maryland, and New Jersey.

4.1.1 Data sources

Virginia oyster harvest data are collected from mail and internet-based surveys of active oyster aquaculture operations on both sides of the Chesapeake Bay, which are then synthesized in an annual report (Hudson 2017). In Maryland, shellfish aquaculturists are required to report their monthly harvests to the Maryland Department of Natural Resources (MD-DNR). The MD-DNR then aggregates the harvest data for release in the Maryland Aquaculture Coordinating Council Annual Report (ACC 2017), from which data were collected. Similar

to Virginia, New Jersey releases annual reports synthesizing electronic survey results from lease-holding shellfish growers. Data from New Jersey reflects cage reared oysters grown from hatchery seed (Calvo 2017).

4.1.2 Data extraction

Data were collected directly from state aquaculture reports. Oyster harvest data in MD was reported in bushels which were then converted to individual oysters by an estimate of 300 oysters bushel⁻¹. View processing code for this indicator here.

4.1.3 Data analysis

No data analyses occurred for this indicator.

4.1.4 Data processing

Aquaculture data were formatted for inclusion in the ecodata R package using the following R code.

```
# Processing code for oyster harvest data in the Mid-Atlantic

# These data were collected directly from shellfish
# aquaculture surveys performed by state agencies and
# universities.

# NJ:
# http://njseagrant.org/new-jersey-shellfish-aquaculture-situation-outlook-report-new/
# VA:
# http://www.vims.edu/research/units/centerspartners/map/aquaculture/docs_aqua/vims_mr
# MD: Data from MD Aquaculture Coordinating Council meeting
# reports. Newest reports available upon request

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

get_aquaculture <- function(save_clean = F) {
  aquaculture <- read.csv(file.path(raw.dir, "mab_oyster_harvest.csv"))

  if (save_clean) {
    usethis::use_data(aquaculture, overwrite = T)
```

```

} else {
    return(aquaculture)
}

}

```

4.1.5 Plotting

```

aqua <- ecodata::aquaculture %>% group_by(Var) %>% mutate(hline = mean(Value)) %>%
    ungroup() %>% mutate(Var = plyr::mapvalues(Var, from = c("md oyster harvest",
    "nj oyster harvest", "va oyster harvest"), to = c("MD", "NJ",
    "VA"))) %>% dplyr::rename(State = Var)

aqua$State <- factor(aqua$State, levels = c("VA", "MD", "NJ"))

ggplot() + geom_segment(aes(x = 2005, xend = 2017, y = mean(aquaculture[aquaculture$Var ==
    "va oyster harvest", ]$Value), yend = mean(aquaculture[aquaculture$Var ==
    "va oyster harvest", ]$Value)), size = hline.size, alpha = hline.alpha,
    linetype = hline.lty, color = "#1b9e77", inherit.aes = F) +
    geom_segment(aes(x = 2012, xend = 2016, y = mean(aquaculture[aquaculture$Var ==
        "nj oyster harvest", ]$Value), yend = mean(aquaculture[aquaculture$Var ==
        "nj oyster harvest", ]$Value)), size = hline.size, alpha = hline.alpha,
        linetype = hline.lty, color = "#d95f02", inherit.aes = F) +
    geom_segment(aes(x = 2012, xend = 2017, y = mean(aquaculture[aquaculture$Var ==
        "md oyster harvest", ]$Value), yend = mean(aquaculture[aquaculture$Var ==
        "md oyster harvest", ]$Value)), size = hline.size, alpha = hline.alpha,
        linetype = hline.lty, color = "#7570b3", inherit.aes = F) +
    # Highlight last ten years
    geom_line(data = aqua, aes(x = Time, y = Value, color = State),
        size = lwd) + geom_point(data = aqua, aes(x = Time, y = Value,
            color = State), size = pcex) + scale_color_manual(values = c(VA = "#1b9e77",
            MD = "#7570b3", NJ = "#d95f02")) + scale_x_continuous(breaks = seq(2005,
            2018, 3), expand = c(0.01, 0.01)) + scale_y_continuous(labels = function(l) {
        trans = 1/1e+06
}) + ggttitle("Oyster harvest") + ylab(expression("Oysters sold (10^6 * n)")) + xlab("") + theme_ts()

```

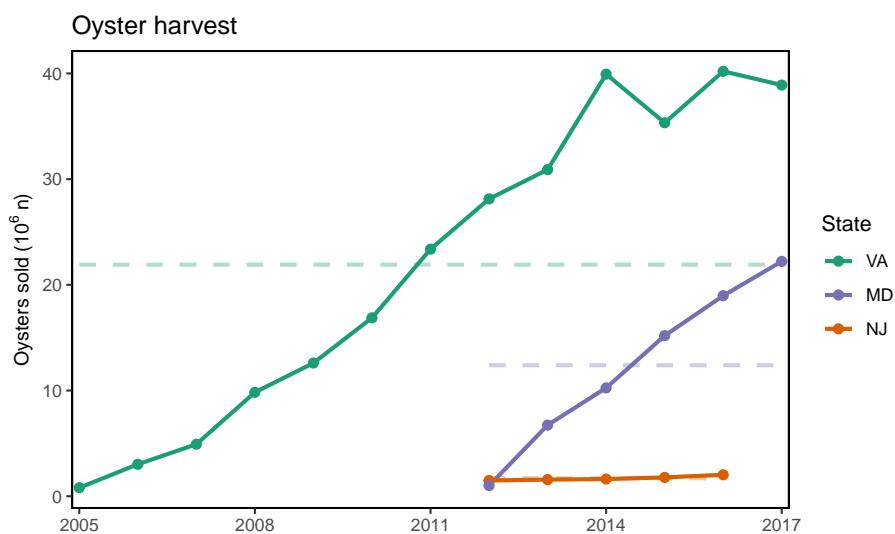


Figure 4.1: Oyster aquaculture production in terms of number of oysters sold from Virginia, Maryland, and New Jersey.

Chapter 5

Bennet Indicator

Description: Bennet Indicator

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Database pull with analysis

Contributor(s): John Walden

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: John Walden, john.walden@noaa.gov

Public availability statement: Derived CFDBS data are available for this analysis (see Comland).

5.1 Methods

5.1.1 Data sources

Data used in the Bennet Indicator were derived from the Comland data set; a processed subset of the Commercial Fisheries Database System (CFDBS). The derived Comland data set is available for download [here](#).

5.1.2 Data extraction

For information regarding processing of CFDBS, please see Comland methods. The Comland dataset containing seafood landings data was subsetted to US landings after 1964 where revenue was ≥ 0 for each Ecological Production Unit (i.e. Mid-Atlantic Bight, Georges Bank, and Gulf of Maine). Each EPU was

run in an individual R script, and the code specific to Georges Bank is shown below.

```
# This code is used to load and process comland data. See
# comland methods for source data (CFDBS) processing methods.

drake::readd(proc_get_mass_inshore_survey.R)

# Packages
PKG <- c("data.table", "plyr", "RColorBrewer", "ggplot2", "cowplot",
        "gridExtra", "grid")
for (p in PKG) {
  if (!require(p, character.only = TRUE)) {
    install.packages(p)
    require(p, character.only = TRUE)
  }
}
# #Setting Save path

# load 'comland' data - These data are unavailble due to PII
# concerns. See aggregated data load below
ecosys2 <- subset(comland, US == "TRUE" & YEAR >= 1964 & SPPVALUE >=
  0)

# Load species and PDT codes
load(file.path(data.dir, "Species_codes.RData"))

# Set EPU
epu <- "GB"

# processing
spp <- subset(spp, NESPP3 > 0)
spp2 <- unique(spp[, c(3, 12)], by = "NESPP3")
spp2 <- spp2[which(!duplicated(spp2$NESPP3)), ]
sp_combine <- merge(ecosys2, spp2, by = "NESPP3", all.x = TRUE)
add.apex <- data.table(NESPP3 = 0, YEAR = 1971, QY = 1, GEAR = "other",
                        SIZE = "small", EPU = epu, UTILCD = 0, SPPLIVMT = 0, SPPVALUE = 0,
                        US = TRUE, Feeding.guild = "Apex Predator")
sp_combine <- rbindlist(list(sp_combine, add.apex))

# Subset data into Georges Bank group
LANDINGS <- subset(sp_combine)
LANDINGS <- LANDINGS[which(!is.na(LANDINGS$Feeding.guild)), ]

# Set Up data Table
```

```

landsum <- data.table(LANDINGS)
# setkey(landsum, 'EPU', 'YEAR', 'Feeding.guild')
setkey(landsum, "EPU", "YEAR", "Feeding.guild")

# Sum by feeding guild
landsum[, lapply(.SD, sum, na.rm = TRUE), by = key(landsum),
       .SDcols = c("SPPLIVMT", "SPPVALUE")]

```

5.1.3 Data analysis

Revenue earned by harvesting resources from a Large Marine Ecosystem (LME) at time t is a function of both the quantity landed of each species and the prices paid for landings. Changes in revenue between any two years depends on both prices and quantities in each year, and both may be changing simultaneously. For example, an increase in the harvest of higher priced species, such as scallops can lead to an overall increase in total revenue from an LME between time periods even if quantities landed of other species decline. Although measurement of revenue change is useful, the ability to see what drives revenue change, whether it is changing harvest levels, the mix of species landed, or price changes provides additional valuable information. Therefore, it is useful to decompose revenue change into two parts, one which is due to changing quantities (or volumes), and a second which is due to changing prices. In an LME, the quantity component will yield useful information about how the species mix of harvests are changing through time.

A Bennet indicator (BI) is used to examine revenue change between 1964 and 2015 for two major LME regions. It is composed of a volume indicator (VI), which measures changes in quantities, and a price indicator (PI) which measures changes in prices. The Bennet (1920) indicator (BI) was first used to show how a change in social welfare could be decomposed into a sum of a price and quantity change indicator (Cross and Färe 2009). It is called an indicator because it is based on differences in value between time periods, rather than ratios, which are referred to as indices. The BI is the indicator equivalent of the more popular Fisher index (Balk 2010), and has been used to examine revenue changes in Swedish pharmacies, productivity change in U.S. railroads (Lim and Lovell 2009), and dividend changes in banking operations (Grifell-Tatjé and Lovell 2004). An attractive feature of the BI is that the overall indicator is equal to the sum of its subcomponents (Balk 2010). This allows one to examine what component of overall revenue is responsible for change between time periods. This allows us to examine whether changing quantities or prices of separate species groups are driving revenue change in each EPU between 1964 and 2015.

Revenue in a given year for any species group is the product of quantity landed times price, and the sum of revenue from all groups is total revenue from the

LME. In any year, both prices and quantities can change from prior years, leading to total revenue change. At time t , revenue (R) is defined as

$$R^t = \sum_{j=1}^J p_j^t y_j^t,$$

where p_j is the price for species group j , and y_j is the quantity landed of species group j . Revenue change between any two time periods, say $t+1$ and t , is then $R^{t+1} - R^t$, which can also be expressed as:

$$\Delta R = \sum_{j=1}^J p_j^{t+1} y_j^{t+1} - \sum_{j=1}^J p_j^t y_j^t.$$

This change can be decomposed further, yielding a VI and PI. The VI is calculated using the following formula (Georgianna, Lee, and Walden 2017):

$$VI = \frac{1}{2} \left(\sum_{j=1}^J p_j^{t+1} y_j^{t+1} - \sum_{j=1}^J p_j^{t+1} y_j^t + \sum_{j=1}^J p_j^t y_j^{t+1} - \sum_{j=1}^J p_j^t y_j^t \right)$$

The price indicator (PI) is calculated as follows:

$$PI = \frac{1}{2} \left(\sum_{j=1}^J y_j^{t+1} p_j^{t+1} - \sum_{j=1}^J y_j^{t+1} p_j^t + \sum_{j=1}^J y_j^t p_j^{t+1} - \sum_{j=1}^J y_j^t p_j^t \right)$$

Total revenue change between time t and $t+1$ is the sum of the VI and PI. Since revenue change is being driven by changes in the individual prices and quantities landed of each species group, changes at the species group level can be examined separately by taking advantage of the additive property of the indicator. For example, if there are five different species groups, the sum of the VI for each group will equal the overall VI, and the sum of the PI for each group will equal the overall PI.

```
# R code to construct Bennet Indicator for Ecosystem Project
# Author: John Walden Date: October 4, 2017 Revised January
# 18, 2018 to calculate the indicator relative to average
# conditions during each time period. Set EPU in
# extraction/processing code chunk above.

# filter by specific EPU
epu = "GB"
value <- subset(landsum, EPU == epu)

# Calculate price
value$PRICE = value$SPPVALUE/value$SPPLIVMT
```

```

value[is.na(value)] <- 0

# Next two lines are to calculate mean values for landings
# and value for the time series by feeding guild

meanval <- as.data.frame(value[, j = list(mean(SPPVALUE, na.rm = TRUE),
    mean(SPPLIVMT, na.rm = TRUE)), by = Feeding.guild])
meanval <- rename(meanval, c(V1 = "BASEV", V2 = "BASEQ"))
meanval$BASEP = meanval$BASEV/meanval$BASEQ

# order by feeding guild

value <- value[order(value$Feeding.guild), ]
meanval <- meanval[order(meanval$Feeding.guild), ]

# Merge Value data frame with Base Year Value Data Frame
value <- merge(value, meanval, by = "Feeding.guild")

# Construct price and Volume Indicators NOTE: ALL values are
# normalized to $1,000,000

value$VI = ((0.5 * (value$BASEP + value$PRICE)) * (value$SPPLIVMT -
    value$BASEQ))/1e+06
value$PI = ((0.5 * (value$BASEQ + value$SPPLIVMT)) * (value$PRICE -
    value$BASEP))/1e+06

value <- value[order(value$YEAR), ]

# The next Data table sets up the yearly aggregate Bennet PI
# and VI

biyear <- data.table(value)
setkey(biyear, "YEAR")
biyear <- biyear[, lapply(.SD, sum), by = key(biyear), .SDcols = c("VI",
    "PI", "BASEV", "SPPVALUE")]
biyear$revchange <- (biyear$VI + biyear$PI)
biyear$BI <- (biyear$VI + biyear$PI)

# The Next Steps restructure the year data frame so the
# yearly Bennet Indicator can be plotted. Negative values are
# difficult in GGPLOT. Since the Bennet indicator can have a
# negative value, separate data frames need to be created.
# First, the data needs to be restructured to use the stacked
# bar function in ggplot. GGPLOT is used because it can graph

```

```
# different data layers on the same graph.

y1 <- biyear[, c(1, 2)]
y1$indicator = "VI"
y2 <- biyear[, c(1, 3)]
y2$indicator = "PI"

colnames(y1)[2] <- "value"
colnames(y2)[2] <- "value"
ytotal <- rbind(y1, y2)
```

5.1.4 Data processing

Bennet indicator time series were formatted for inclusion in the ecodata R package using the following R code.

```
# Process Bennet indicator; price and volume indicators

library(dplyr)
library(tidyr)
library(magrittr)

raw.dir <- here::here("data-raw")

get_bennet <- function(save_clean = F) {

  load(file.path(raw.dir, "Bennet_Index.Rdata"))
  bennet <- bennet %>% rename(EPU = Region)

  if (save_clean) {
    usethis::use_data(bennet, overwrite = T)
  } else {
    return(bennet)
  }

}
get_bennet(save_clean = T)
```

5.1.5 Plotting

```
# Filter data into two dataframes for plotting
indicators <- ecodata::bennet %>% filter(EPU == "MAB", Var %in%
```

```

c("VI EPU aggregate", "PI EPU aggregate")) %>% mutate(Var,
  Var = plyr::mapvalues(Var, from = c("VI EPU aggregate", "PI EPU aggregate"),
    to = c("Volume", "Price")))

revchange <- ecodata::bennet %>% filter(EPU == "MAB", Var %in%
  c("REVCHANGE EPU aggregate"))

# custom bar fill color (color-blind friendly)
ind_fill <- c("#a6cee3", "#b2df8a")

# limits
y.lim <- c(-450, 450)

# plot
ggplot(data = indicators) +
  # Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha, xmin = x.shade.min,
    xmax = x.shade.max, ymin = -Inf, ymax = Inf) +
  geom_bar(aes(x = Time, y = Value, fill = Var), stat = "identity") +
  scale_fill_manual(name = "Indicators", values = ind_fill) +
  geom_line(data = revchange, aes(x = Time, y = Value, colour = "$")) +
  scale_colour_grey(name = "Revenue Change") + ggtitle("Bennet Indicator") +
  labs(y = "Value $1,000,000 ($2015)") + scale_x_continuous(breaks = seq(1985,
  2015, by = 5), expand = c(0.01, 0.01)) + scale_y_continuous(breaks = seq(y.lim[1],
  y.lim[2], by = 100), limits = y.lim, expand = c(0.01, 0.01)) +
  theme_ts() + theme(title = element_text(size = 10))

```

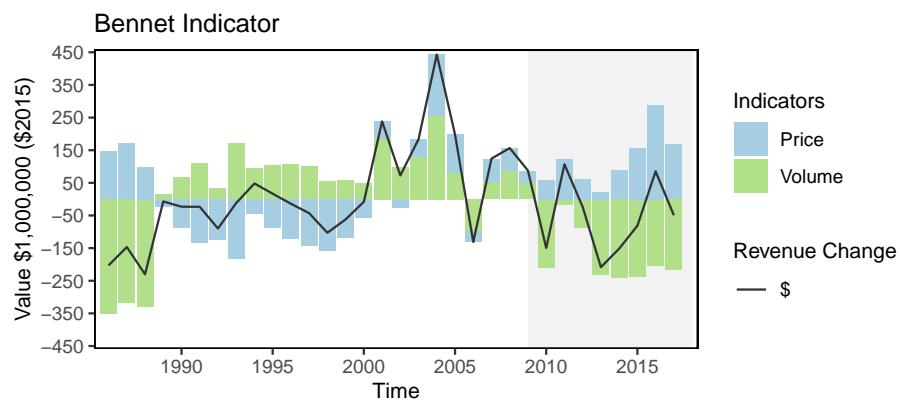


Figure 5.1: Revenue change from the long-term mean in 2015 dollars (black), Price (PI), and Volume Indicators (VI) for commercial landings in the Mid-Atlantic.

Chapter 6

Bottom temperatures

Description: Time series of annual in situ bottom temperatures on the Northeast Continental Shelf.

Indicator category: Extensive analysis; not yet published Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2019); State of the Ecosystem - Mid-Atlantic Bight (2019)

Contributor(s): Paula Fratantoni, paula.fratantoni@noaa.gov

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Paula Fratantoni, paula.fratantoni@noaa.gov

Public availability statement: Source data are publicly available at ftp://ftp.nefsc.noaa.gov/pub/hydro/matlab_files/yearly and in the World Ocean Database housed at <http://www.nodc.noaa.gov/OC5/SELECT/dbsearch/dbsearch.html> under institute code number 258.

6.1 Methods

6.1.1 Data sources

The bottom temperature index incorporates near-bottom temperature measurements collected on Northeast Fisheries Science Center (NEFSC) surveys between 1977-present. Early measurements were made using surface bucket samples, mechanical bathythermographs and expendable bathythermograph probes, but by 1991 the CTD – an acronym for conductivity temperature and depth – became standard equipment on all NEFSC surveys. Near-bottom refers to the deepest observation at each station that falls within 10 m of the reported water depth. Observations encompass the entire continental shelf area extending from

Cape Hatteras, NC to Nova Scotia, Canada, inclusive of the Gulf of Maine and Georges Bank.

6.1.2 Data extraction

While all processed hydrographic data are archived in an Oracle database (OCDBS), we work from Matlab-formatted files stored locally.

6.1.3 Data analysis

Ocean temperature on the Northeast U.S. Shelf varies significantly on seasonal timescales. Any attempt to resolve year-to-year changes requires that this seasonal variability be quantified and removed to avoid bias. This process is complicated by the fact that NEFSC hydrographic surveys conform to a random stratified sampling design meaning that stations are not repeated at fixed locations year after year so that temperature variability cannot be assessed at fixed station locations. Instead, we consider the variation of the average bottom temperature within four Ecological Production Units (EPUs): Middle Atlantic Bight, Georges Bank, Gulf of Maine and Scotian Shelf. Within each EPU, ocean temperature observations are extracted from the collection of measurements made within 10 m of the bottom on each survey and an area-weighted average temperature is calculated. The result of this calculation is a timeseries of regional average near-bottom temperature having a temporal resolution that matches the survey frequency in the database. Anomalies are subsequently calculated relative to a reference annual cycle, estimated using a multiple linear regression model to fit an annual harmonic (365-day period) to historical regional average temperatures from 1981-2010. The curve fitting technique to formulate the reference annual cycle follows the methodologies outlined by Mountain (1991). The reference period was chosen because it is the standard climatological period adopted by the World Meteorological Organization. The resulting anomaly time series represents the difference between the time series of regional mean temperatures and corresponding reference temperatures predicted by a reference annual cycle for the same time of year. Finally, a reference annual average temperature (calculated as the average across the reference annual cycle) is added back into the anomaly timeseries to convert temperature anomalies back to ocean bottom temperature.

6.1.4 Data processing

Derived bottom temperature data were formatted for inclusion in the `ecodata` R package using the following R code.

```

# Process ocean temperature anomaly data

# These data include in situ regional time series of both surface and bottom water temperature and
# on the Northeast Continental Shelf. Raw data is split into four files by EPU (SS, GOM, GB, and
# MAB)

library(dplyr)
library(tidyr)
library(lubridate)

#Get raw
raw.dir <- here::here("data-raw") #input raw

get_oceantemp_institu <- function(save_clean = F){

  ss <- read.csv(file.path(raw.dir,"bot_temp_SS.csv")) %>% mutate(EPU = "SS")
  gom <- read.csv(file.path(raw.dir,"bot_temp_GOM.csv")) %>% mutate(EPU = "GOM")
  gb <- read.csv(file.path(raw.dir,"bot_temp_GB.csv")) %>% mutate(EPU = "GB")
  mab <- read.csv(file.path(raw.dir,"bot_temp_MAB.csv")) %>% mutate(EPU = "MAB")

  oceantemp_institu <- rbind(ss, gom, gb, mab) %>% #bind all
  mutate(Units = "degreesC", Time = as.Date(format(date_decimal(Time), "%Y-%b-%d"), "%Y-%b-%d"))
    Var, Var = plyr::mapvalues(Var, from = c("Tsfc_anom",#Rename variables
                                              "Tsfc_ref",
                                              "Tbot_anom",
                                              "Tbot_ref"),
                                to = c("sst anomaly in situ",
                                       "reference sst in situ (1981-2010)",
                                       "bottom temp anomaly in situ",
                                       "reference bt in situ (1981-2010)))) %>%
  group_by(Time = year(Time), EPU, Var, Units) %>%
  dplyr::summarise(Value = mean(Value)) %>%
  as.data.frame()

  if (save_clean){
    usethis::use_data(oceantemp_institu, overwrite = T)
  } else {
    return(oceantemp_institu)
  }
}

get_oceantemp_institu(save_clean = T)

```

6.1.5 Plotting

```

bot_temp_institu_gom <- oceantemp_institu %>%
  filter(EPU == "GOM",
         Var == "bottom temp anomaly in situ") %>%
  mutate(hline = 0) %>%
  ggplot() + #plot
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min, xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  geom_line(aes(x = Time, y = Value)) +
  geom_gls(aes(x = Time, y = Value)) +
  geom_point(aes(x = Time, y = Value), size = 1) +
  ylab("Temp. Anomaly (°C)") +
  ggtitle("GOM Bottom Temperature Anomaly") +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  geom_hline(aes(yintercept = hline),
             size = hline.size,
             alpha = hline.alpha,
             linetype = hline.lty) +
  theme_ts() +
  theme(strip.text=element_text(hjust=0),
        plot.title = element_text(size = 12))

bot_temp_institu_gb <- oceantemp_institu %>%
  filter(EPU == "GB",
         Var == "bottom temp anomaly in situ") %>%
  mutate(hline = 0) %>%
  ggplot() + #plot
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min, xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  geom_line(aes(x = Time, y = Value)) +
  geom_gls(aes(x = Time, y = Value)) +
  geom_point(aes(x = Time, y = Value), size = 1) +
  ylab("Temp. Anomaly (°C)") +
  ggtitle("GB Bottom Temperature Anomaly") +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  geom_hline(aes(yintercept = hline),
             size = hline.size,
             alpha = hline.alpha,
             linetype = hline.lty) +
  theme_ts() +
  theme(strip.text=element_text(hjust=0),
        plot.title = element_text(size = 12))

```

```
plot.title = element_text(size = 12)

bot_temp_in situ_gom +
bot_temp_in situ_gb + plot_layout(nrow = 1)
```

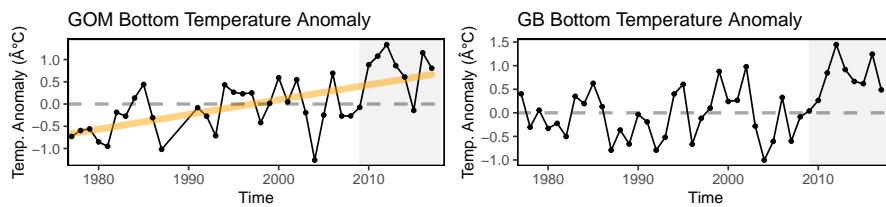


Figure 6.1: GOM and GB annual bottom temperature anomalies.

```
#plot_layout(ncol = 1) &
#theme(plot.margin = unit(c(0, 0, 0, 0), "cm")) #
```


Chapter 7

Catch and Fleet Diversity

Description: Permit-level species diversity and Council-level fleet diversity.

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018),
State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull with analysis; Published methods

Contributor(s): Geret DePiper, Min-Yang Lee

Data steward: Geret DePiper, geret.depiper@noaa.gov

Point of contact: Geret DePiper, geret.depiper@noaa.gov

Public availability statement: Source data is not publicly available due to
PII restrictions. Derived time series are available for download here.

7.1 Methods

Diversity estimates have been developed to understand whether specialization, or alternatively stovepiping, is occurring in fisheries of the Northeastern Large Marine Ecosystem. We use the average effective Shannon indices for species revenue at the permit level, for all permits landing any amount of NEFMC or MAFMC Fishery Management Plan (FMP) species within a year (including both Monkfish and Spiny Dogfish). We also use the effective Shannon index of fleet revenue diversity and count of active fleets to assess the extent to which the distribution of fishing changes across fleet segments.

7.1.1 Data sources

Data for these diversity estimates comes from a variety of sources, including the Commercial Fishery Dealer Database, Vessel Trip Reports, Clam logbooks,

vessel characteristics from Permit database, WPU series producer price index. These data are typically not available to the public.

7.1.2 Data extraction

The following describes both the permit-level species and fleet diversity data generation. Price data was extracted from the Commercial Fishery Dealer database (CFDERS) and linked to Vessel Trip Reports by a hierarchical matching algorithm that matched date and port of landing at its highest resolution. Code used in these analyses is available upon request.

Output data was then matched to vessel characteristics from the VPS VESSEL data set. For the permit-level estimate, species groups are based off of a slightly refined NESPP3 code (Table 7.1), defined in the data as “myspp”, which is further developed in the script to rectify inconsistencies in the data.

Table 7.1: Species grouping

Group	NESPP3	Common Name	Scientific Name
	470	ALBACORE	THUNNUS ALALUNGA
	494	ATLANTIC SHARPNOSE SHARK	RHIZOPRIONODON TERRAENOVAE
	354	BIGEYE THRESHER SHARK	ALOPIAS SUPERCILIOSUS
	469	BIGEYE TUNA	THUNNUS OBESUS
	487	BLACKTIP SHARK	CARCHARHINUS LIMBATUM
	493	BLUE SHARK	PRIONACE GLAUCA
	467	BLUEFIN TUNA	THUNNUS THYNNUS
	468	LITTLE TUNNY	EUTHYNNUS ALLETTERATUS
	358	LONGFIN MAKO	ISURUS PAUCUS
	481	PORBEAGLE SHARK	LAMNA NASUS
	349	SAND TIGER	CARCHARIAS TAURUS
	482	SANDBAR SHARK	CARCHARHINUS PLUMBEUS
	359	SHARK,UNC	CHONDRICHTHYES
	355	SHORTFIN MAKO	ISURUS OXYRINCHUS

Table 7.1: Species grouping (*continued*)

Group	NESPP3	Common Name	Scientific Name
	466	SKIPJACK TUNA	KATSUWONUS PELAMIS
	432	SWORDFISH	XIPHIAS GLADIUS
	353	THRESHER SHARK	ALOPIAS VULPINUS
	491	TIGER SHARK	GALEOCERDO CUVIER
	471	YELLOWFIN TUNA	THUNNUS ALBACARES
	11	GOOSEFISH	LOPHIUS AMERICANUS
Monkfish in Mid-Atlantic Waters	12	GOOSEFISH	LOPHIUS AMERICANUS
Atlantic Scallops	800	SEA SCALLOP	PLACOPECTEN MAGELLANICUS
Shrimp	737	MANTIS SHRIMP UNCL	STOMATOPODA
	737	MANTIS SHRIMPS	STOMATOPODA
	736	NORTHERN SHRIMP	PANDALUS BOREALIS
	738	SHRIMP, ATLANTIC & GULF,BROWN	PANAEIDAE
	735	SHRIMP,UNC (CARIDEA)	CARIDEA
	368	BARNDOR SKATE	DIPTURUS LAEVIS
	372	CLEARNOSE SKATE	RAJA EGLANTERIA
Skates	366	LITTLE SKATE	LEUCORAJA ERINACEA
	365	OCELLATE SKATES	RAJA
	365	SKATES	RAJIDAE
	373	SKATES,LITTLE/WIN MIXED	LEUCORAJA
	369	SMOOTH SKATE	MALACORAJA SENTA
	370	THORNY SKATE	AMBLYRAJA RADIATA
	367	WINTER SKATE	LEUCORAJA OCELLATA
Herring	168	ATLANTIC HERRING	CLUPEA HARENUS

Table 7.1: Species grouping (*continued*)

Group	NESPP3	Common Name	Scientific Name
Ocean Quahog	754	OCEAN QUAHOG	<u>ARCTICA</u> <u>ISLANDICA</u>
Surf Clam	769	ATLANTIC SURFCLAM	SPISULA SOLIDISSIMA
	444	BLUELINE TILEFISH	CAULOLATILUS MICROPS
	445	SAND TILEFISH	MALACANTHUS PLUMIERI
	446	TILEFISH	LOPHOLATILUS CHAMAELEON-TICEPS
Tilefish	447	TILEFISH,UNC	MALACANTHIDAE
	335	BLACK SEA BASS	CENTROPRISTIS STRIATA
Fluke & Black Seabass	121	SUMMER FLOUNDER	PARALICHTHYS DENTATUS
	51	BUTTERFISH	PEPRILUS TRIACANTHUS
Butterfish & Hake	152	RED HAKE	UROPHYCIS CHUSS
	509	SILVER HAKE	MERLUCCIUS BILINEARIS
Bluefish in Mid-Atlantic Waters	23	BLUEFISH	POMATOMUS SALTATRIX
Spiny Dogfish	352	SPINY DOGFISH	SQUALUS ACANTHIAS
Northern Shortfin Squid	802	NORTHERN SHORTFIN SQUID	ILLEX ILLECEBROSUS
American Lobster	727	AMERICAN LOBSTER	HOMARUS AMERICANUS
Longfin Squid	801	LONGFIN SQUID	LOLIGO PEALEII
Menhaden	221	MENHADEN	BREVOORTIA
Offshore Hake	508	OFFSHORE HAKE	MERLUCCIUS ALBIDUS
Scup in Mid-Atlantic Waters	329	SCUP	STENOTOMUS CHRYSOPS
Windowpane Flounder in New England Waters	125	WINDOWPANE	SCOPHTHALMUS AQUOSUS
Ocean Pout in New England Waters	250	OCEAN POUT	ZOARCES AMERICANUS
Wolffish	512	ATLANTIC WOLFFISH	ANARHICHAS LUPUS

Table 7.1: Species grouping (*continued*)

Group	NESPP3	Common Name	Scientific Name
Winter Flounder in Mid-Atlantic Waters	120	WINTER FLOUNDER	<i>PSEUDOPLEURONECTES AMERICANUS</i>
Yellowtail Flounder in Mid-Atlantic Waters	123	YELLOWTAIL FLOUNDER	<i>LIMANDA FERRUGINEA</i>
Unclassified Hake	155	Unclassified Hake	
White Hake in Mid-Atlantic Waters	153	WHITE HAKE	<i>UROPHYCIS TENUIS</i>
	23	BLUEFISH	<i>POMATOMUS SALTATRIX</i>
Bluefish & Scup in New England Waters	329	SCUP	<i>STENOTOMUS CHRYSOPS</i>
Halibut in New England Waters	159	ATLANTIC HALIBUT	<i>HIPPOGLOSSUS HIPPOGLOSSUS</i>
	240	ACADIAN REDFISH	<i>SEBASTES FASCIATUS</i>
	124	AMERICAN PLAICE	<i>HIPPOGLOSSOIDES PLATESSOIDES</i>
	81	ATLANTIC COD	<i>GADUS MORHUA</i>
	11	GOOSEFISH	<i>LOPHIUS AMERICANUS</i>
	12	GOOSEFISH	<i>LOPHIUS AMERICANUS</i>
	147	HADDOCK	<i>MELANOGRAMMUS AEGLEFINUS</i>
Groundfish in New England Waters	269	POLLOCK	<i>POLLACHIUS VIRENS</i>
	153	WHITE HAKE	<i>UROPHYCIS TENUIS</i>
	120	WINTER FLOUNDER	<i>PSEUDOPLEURONECTES AMERICANUS</i>
	122	WITCH FLOUNDER	<i>GLYPTOCEPHALUS CYNOGLOSSUS</i>
	123	YELLOWTAIL FLOUNDER	<i>LIMANDA FERRUGINEA</i>
	240	ACADIAN REDFISH	<i>SEBASTES FASCIATUS</i>
	124	AMERICAN PLAICE	<i>HIPPOGLOSSOIDES PLATESSOIDES</i>
	81	ATLANTIC COD	<i>GADUS MORHUA</i>
	159	ATLANTIC HALIBUT	<i>HIPPOGLOSSUS HIPPOGLOSSUS</i>

Table 7.1: Species grouping (*continued*)

Group	NESPP3	Common Name	Scientific Name
	512	ATLANTIC WOLFFISH	ANARHICHAS LUPUS
	147	HADDOCK	MELANOGRAMMUS AEGLEFINUS
Groundfish in Mid-Atlantic Waters	269	POLLOCK	POLLACHIUS VIRENS
	122	WITCH FLOUNDER	GLYPTOCEPHALUS CYNOGLOSSUS
	155	Unclassified Hake	
	250	OCEAN POUT	ZOARCES AMERICANUS
Windowpane Flounder & Ocean Pout in Mid-Atlantic Waters	125	WINDOWPANE	SCOPHTHALMUS AQUOSUS

For the fleet diversity metric, gears include scallop dredge (gearcodes DRS, DSC, DTC, and DTS), other dredges (gearcodes DRM, DRO, and DRU), gill-net (gearcodes GND, GNT, GNO, GNR, and GNS), hand (gearcode HND), longline (gearcodes LLB and LLP), bottom trawl (gearcodes OTB, OTF, OTO, OTC, OTS, OHS, OTR, OTT, and PTB), midwater trawls (gearcode OTM and PTM), pot (gearcodes PTL, PTW, PTC, PTE, PTF, PTH, PTL, PTO, PTS, and PTX), purse seine (gearcode PUR), and hydraulic clam dredge (gearcode DRC). Vessels were further grouped by length categories of less than 30 feet, 30 to 50 feet, 50 to 75 feet, and 75 feet and above. All revenue was deflated to real dollars using the “WPU0223” Producer Price Index with a base of January 2015. Stata code for data processing is available [here](#).

7.1.3 Data analysis

This permit-level species effective Shannon index is calculated as

$$\exp\left(-\sum_{i=1}^N p_{ijt} \ln(p_{ijt})\right)$$

for all j , with p_{ijt} representing the proportion of revenue generated by species or species group i for permit j in year t , and is a composite of richness (the number of species landed) and abundance (the revenue generated from each species). The annual arithmetic mean value of the effective Shannon index across permits is used as the indicator of permit-level species diversity.

In a similar manner, the fleet diversity metric is estimated as

$$\exp\left(-\sum_{i=1}^N p_{kt} \ln(p_{kt})\right)$$

for all k , where p_{kt} represents the proportion of total revenue generated by fleet segment k (gear and length combination) per year t . The indices each run from 1996 to 2017. A count of the number of fleets active in every year is also provided to assess whether changes in fleet diversity are caused by shifts in abundance (number of fleets), or evenness (concentration of revenue). The work is based off of analysis conducted in Thunberg and Correia (2015) and published in Gaichas et al. (2016).

7.1.4 Data processing

Catch and fleet diversity indicators were formatted for inclusion in the `ecodata` R package using the following R script.

```
# Process commercial diversity data

# More information about these data are available at
# https://noaa-edab.github.io/tech-doc/catch-and-fleet-diversity.html.
# Data are drawn from blend of VTR trip data, CFDBS prices,
# vessel characteristics from PERMIT databases, and major VTR
# gear by permit. Here 'MA' and 'NE' refer to the
# Mid-Atlantic and New England regions respectively, and are
# not derived by EPU.

library(dplyr)
library(tidyverse)
library(stringr)

raw.dir <- here::here("data-raw")

get_commercial_div <- function(save_clean = F) {
  commercial_div <- read.csv(file.path(raw.dir, "Commercial_Diversity.csv")) %>%
    dplyr::select(-X, -Source) %>% dplyr::rename(EPU = Region) %>%
    as.data.frame()

  commercial_div$Var <- str_replace(commercial_div$Var, "diveristy",
                                      "diversity")

  if (save_clean) {
    usethis::use_data(commercial_div, overwrite = T)
  } else {
    return(commercial_div)
  }
}
```

```

}
get_commercial_div(save_clean = T)

```

7.1.5 Plotting

```

# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mfrow = c(2, 1), mar = c(0, 0, 0, 0), oma = c(4,
  6, 2, 6))

soe.plot(SOE.data, "Time", "Mid-Atlantic average fleet diversity",
  stacked = "A", rel.y.num = 0.9, end.start = 2008, tol = 0.15,
  full.trend = F, cex.stacked = 1.5)
soe.stacked.axis("Year", "Fleet diversity", y.line = 2.5, outer = F,
  rel.x.text = 1, rel.y.text = 1)
soe.plot(SOE.data, "Time", "Mid-Atlantic fleet count", stacked = "B",
  rel.y.num = 0.9, end.start = 2008, full.trend = F, cex.stacked = 1.5)

soe.stacked.axis("Year", "Fleet count", y.line = 2.5, outer = F,
  rel.x.text = 1, rel.y.text = 0.95)

```

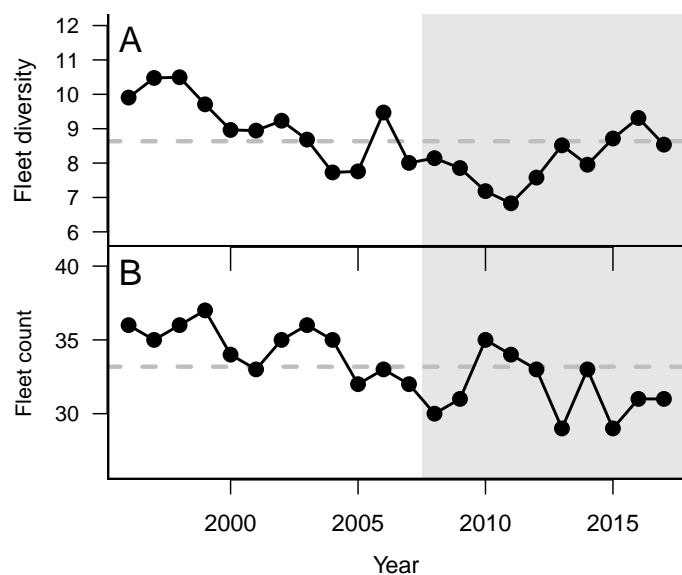


Figure 7.1: Fleet diversity (A) and fleet count (B) in the Mid Atlantic Bight.

Chapter 8

Chesapeake Bay Water Quality Standards Attainment

Description: A multimetric indicator describing the attainment status of Chesapeake Bay with respect to three water quality standards criteria, namely, dissolved oxygen, chlorophyll-a, and water clarity/submerged aquatic vegetation.

Indicator category: Published method; Database pull with analysis

Found in: State of the Ecosystem - Mid-Atlantic (2019)

Contributor(s): Qian Zhang, Rebecca Murphy, Richard Tian, Melinda Forsyth, Emily Trentacoste, Jeni Keisman, and Peter Tango.

Data steward: Qian Zhang, qzhang@chesapeakebay.net

Point of contact: Qian Zhang, qzhang@chesapeakebay.net

Public availability statement: Data are publicly available (see Data Sources below).

8.1 Methods

To protect the aquatic living resources of Chesapeake Bay, the Chesapeake Bay Program (CBP) partnership has developed a guidance framework of ambient water quality criteria with designated uses and assessment procedures for dissolved oxygen, chlorophyll-a, and water clarity/submerged aquatic vegetation (SAV) (USEPA 2003). To achieve consistent assessment over time and between

58CHAPTER 8. CHESAPEAKE BAY WATER QUALITY STANDARDS ATTAINMENT

jurisdictions, a multimetric indicator was proposed by the CBP partnership to provide a means for tracking the progress in all 92 management segments of Chesapeake Bay (USEPA 2017). This indicator has been computed for each three-year assessment period since 1985-1987, providing an integrated measure of Chesapeake Bay's water quality condition over the last three decades.

8.1.1 Data sources

The multimetric indicator required monitoring data on dissolved oxygen (DO) concentrations, chlorophyll-a concentrations, water clarity, SAV acreage, water temperature, and salinity. SAV acreage has been measured by the Virginia Institute of Marine Science in collaboration with the CBP, which is available via <http://web.vims.edu/bio/sav/StateSegmentAreaTable.htm>. Data for all other parameters were obtained from the CBP Water Quality Database. These data have been routinely reported to the CBP by the Maryland Department of Natural Resources, Virginia Department of Environmental Quality, Old Dominion University, Virginia Institute of Marine Science, and citizen/volunteer monitoring initiatives.

8.1.2 Data analysis

Criteria attainment assessment

Monitoring data of DO, chlorophyll-a, and water clarity/SAV were processed and compared with water quality criteria thresholds according to different designated uses (DUs). These DUs are migratory spawning and nursery (MSN), open water (OW), deep water (DW), deep channel (DC), and shallow water (SW), which reflect the seasonal nature of water column structure and the life history needs of living resources. Station-level DO and chlorophyll-a data were spatially interpolated in three dimensions.

Salinity and water temperature data were used to compute the vertical density structure of the water column, which was translated into layers of different DUs. Criteria attainment was determined by comparing violation rates over a 3-year period to a reference cumulative frequency distribution that represents the extent of allowable violation. This approach was implemented using FORTRAN codes, which are provided as a zipped folder. For water clarity/SAV, the single best year in the 3-year assessment period was compared with the segment-specific acreage goal, the water clarity goal, or a combination of both. For more details, refer to the Methods section of Zhang et al. (2018).

Indicator calculation

The multimetric indicator quantifies the fraction of segment-DU-criterion combinations that meet all applicable season-specific thresholds for each 3-year assessment period from 1985-1987 to 2015-2017. For each 3-year assessment

period, all applicable segment-DU-criterion combinations were evaluated in a binomial fashion and scored 1 for “in attainment” and 0 for “nonattainment”. The classified status of each segment-DU-criterion combination was weighted via segments’ surface area and summed to obtain the multimetric index score. This weighting scheme was adopted for two reasons: (1) segments vary in size over four orders of magnitude, and (2) surface area of each segment does not change with time or DUs, unlike seasonally variable habitat volume or bottom water area (USEPA 2017). For more details, refer to the Methods section of Zhang et al. (2018).

The indicator provides an integrated measure of Chesapeake Bay’s water quality condition (Figure 1). In 2015-2017, 42% of all tidal water segment-DU-criterion combinations are estimated to have met or exceeded applicable water quality criteria thresholds, which marks the best 3-year status since 1985-1987. The indicator has a positive and statistically significant trend from 1985 to 2017, which shows that Chesapeake Bay is on a positive trajectory toward recovery. This pattern was statistically linked to total nitrogen reduction, indicating responsiveness of attainment status to management actions implemented to reduce nutrients in the system.

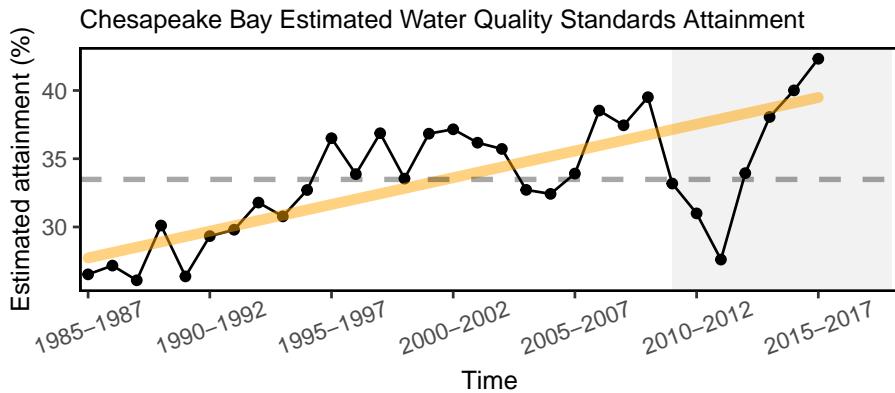


Figure 8.1: Time series of the multimetric indicator score for estimated Chesapeake Bay water quality standards attainment for each 3-year assessment period between 1985-1987 and 2015-2017. A significant positive trend for the time series is shown by the orange line ($p < 0.05$).

Patterns of attainment of individual DUs are variable (Figure 2). Changes in OW-DO, DC-DO, and water clarity/SAV have shown long-term improvements, which have contributed to overall attainment indicator improvement. By contrast, the MSN-DO attainment experienced a sharp spike in the first few assessment periods but generally degraded after the 1997-1999, which has implications to the survival, growth, and reproduction of the migratory and resident tidal

60CHAPTER 8. CHESAPEAKE BAY WATER QUALITY STANDARDS ATTAINMENT

freshwater fish during spawning and nursery season in the tidal freshwater to low-salinity habitats. The status and trends of tidal segments' attainment may be used to inform siting decisions of aquaculture operations in Chesapeake Bay.

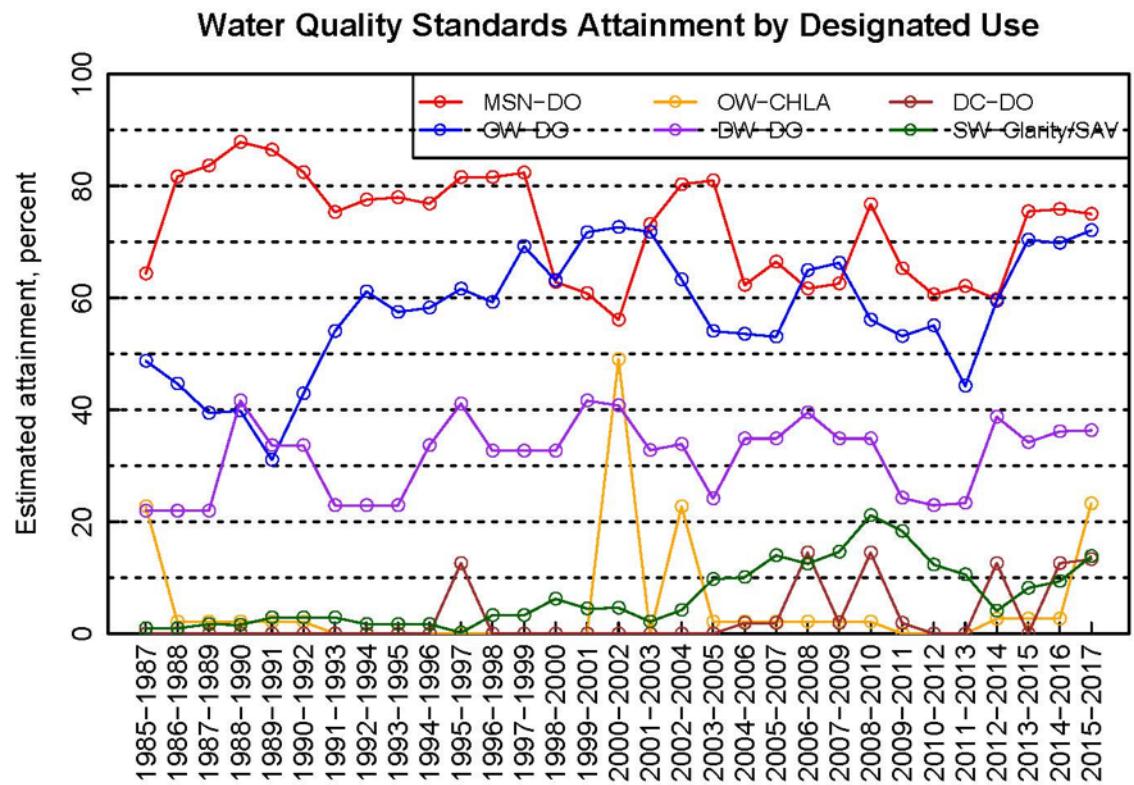


Figure 8.2: Time series of the estimated attainment of water quality standards (i.e., DO: dissolved oxygen; CHLA: chlorophyll-a; Clarity/SAV: water clarity/submerged aquatic vegetation) for five Chesapeake Bay designated uses (MSN: migratory spawning and nursery; OW: open water; DW: deep water; DC: deep channel; SW: shallow water) for each 3-year assessment period between 1985-1987 and 2015-2017.

8.1.3 Data processing

The indicator data set was formatted for inclusion in the ecodata R package using the following R script.

```
# Chesapeake bay water quality attainment indicator

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

get_ches_bay_wq <- function(save_clean = F) {
  ches_bay_wq <- read.csv(file.path(raw.dir, "Attainment_indicator.csv")) %>%
    dplyr::select(Time = Year.1, Value = Total) %>% mutate(Var = "chesapeake bay water quality",
      Units = "estimated attainment, percent", EPU = "MAB")

  if (save_clean) {
    usethis::use_data(ches_bay_wq, overwrite = T)
  } else {
    return(ches_bay_wq)
  }
}
```


Chapter 9

Chlorophyll *a* and Primary Production

Description: Chlorophyll *a* and Primary Production

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Database pull; Database pull with analysis; Published methods

Contributor(s): Kimberly Hyde

Data steward: Kimberly Hyde, kimberly.hyde@noaa.gov

Point of contact: Kimberly Hyde, kimberly.hyde@noaa.gov

Public availability statement: Source data used in these analyses will be made publicly available. Derived data used in State of the Ecosystem Reports can be found here.

9.1 Methods

9.1.1 Data sources

Level 1A ocean color remote sensing data from the Sea-viewing Wide Field-of-view Sensor (SeaWiFS) (NASA Ocean Biology Processing Group 2018) on the OrbView-2 satellite and the Moderate Resolution Imaging Spectroradiometer (MODIS) (NASA Ocean Biology Processing Group 2017) on the Aqua satellite were acquired from the NASA Ocean Biology Processing Group (OBPG). Sea Surface Temperature (SST) data included the 4 km nighttime NOAA Advanced

Very High Resolution Radiometer (AVHRR) Pathfinder (Casey et al. 2010; Saha et al. 2018) and the Group for High Resolution Sea Surface Temperature (GHRSST) Multiscale Ultrahigh Resolution (MUR, version 4.1) Level 4 (Chin, Vazquez-Cuervo, and Armstrong 2017; Project 2015) data.

9.1.2 Data extraction

NA

9.1.3 Data analysis

The SeaWiFS and MODIS L1A files were processed using the NASA Ocean Biology Processing Group SeaDAS software version 7.4. All MODIS files were spatially subset to the U.S. East Coast (SW longitude=-82.5, SW latitude=22.5, NE longitude=-51.5, NE latitude=48.5) using L1AEXTRACT_MODIS. SeaWiFS files were subset using the same coordinates prior to begin downloaded from the Ocean Color Web Browser. SeaDAS's L2GEN program was used to generate Level 2 (L2) files using the default settings and optimal ancillary files, and the L2BIN program spatially and temporally aggregated the L2 files to create daily Level 3 binned (L3B) files. The daily files were binned at 2 km resolution that are stored in a global, nearly equal-area, integerized sinusoidal grids and use the default L2 ocean color flag masks. The global SST data were also subset to the same East Coast region and remapped to the same sinusoidal grid.

The L2 files contain several ocean color products including the default chlorophyll *a*; product (CHL-OCI), photosynthetic available radiation (PAR), remote sensing reflectance ($R_{rs}(\lambda)$), and several inherent optical property products (IOPs). The CHL-OCI product combines two algorithms, the O'Reilly band ratio (OCx) algorithm (O'Reilly et al. 1998) and the Hu color index (CI) algorithm (Hu, Lee, and Franz 2012). The SeaDAS default CHL-OCI algorithm diverges slightly from Hu, Lee, and Franz (2012) in that the transition between CI and OCx occurs at $0.15 < \text{CI} < 0.2 \text{ mg m}^{-3}$ to ensure a smooth transition. The regional chlorophyll *a* algorithm by Pan et al. (2008) was used to create a second chlorophyll product (CHL-PAN). CHL-PAN is an empirical algorithm derived from *in situ* sampling within the Northeast Large Marine Ecosystem (NE-LME) and demonstrated significant improvements from the standard NASA operational algorithm in the NES-LME (Pan et al. 2010). A 3rd-order polynomial function (Equation (9.1)) is used to derive [CHL-PAN] from Rrs band ratios (RBR):

$$\log[\text{CHL-PAN}] = A_0 + A_1X + A_2X^2 + A_3X^3, \quad (9.1)$$

where $X = \log(R_{rs}(\lambda_1)/R_{rs}(\lambda_2))$ and $A_i(i = 0, 1, 2, \text{or } 3)$ are sensor and RBR specific coefficients:

- If SeaWiFS and RBR is $R_{rs}(490)/R_{rs}(555)(R_{3/5})$ then: $A_0 = 0.02534, A_1 = -3.033, A_2 = 2.096, A_3 = -1.607$
- If SeaWiFS and RBR is $R_{rs}(490)/R_{rs}(670)(R_{3/6})$ then: $A_0 = 1.351, A_1 = -2.427, A_2 = 0.9395, A_3 = -0.2432$
- If MODIS and RBR is $R_{rs}(488)/R_{rs}(547)(R_{3/5})$ then: $A_0 = 0.03664, A_1 = -3.451, A_2 = 2.276, A_3 = -1.096$
- If MODIS and RBR is $R_{rs}(488)/R_{rs}(667)(R_{3/6})$ then: $A_0 = 1.351, A_1 = -2.427, A_2 = 0.9395, A_3 = -0.2432$

$C_{3/5}$ and $C_{3/6}$ were calculated for each sensor specific RBR ($R_{3/5}$ and $R_{3/6}$ respectively) and then the following criteria were used to determine to derive CHL-PAN:

If $R_{3/5} > 0.15$ or $R_6 < 0.0001$ then $\text{CHL-PAN} = C_{3/5}$;

Otherwise, $\text{CHL-PAN} = \max(C_{3/5}, C_{3/6})$,

where R_6 is $R_{rs}(670)$ (SeaWiFS) or $R_{rs}(667)$ (Pan et al. 2010).

The Vertically Generalized Production Model (VGPM) estimates net primary production (PP) as a function of chlorophyll *a*, photosynthetically available light and the photosynthetic efficiency (Behrenfeld and Falkowski 1997). In the VGPM-Eppley version, the original temperature-dependent function to estimate the chlorophyll-specific photosynthetic efficiency is replaced with the exponential “Eppley” function (equation PP1) as modified by Morel (1991). The VGPM calculates the daily amount of carbon fixed based on the maximum rate of chlorophyll-specific carbon fixation in the water column, sea surface daily photosynthetically available radiation, the euphotic depth (the depth where light is 1% of that at the surface), chlorophyll *a* concentration, and the number of daylight hours (Equation (9.2)).

$$P_{max}^b(SST) = 4.6 * 1.065^{SST-20^0} \quad (9.2)$$

Where P_{max}^b is the maximum carbon fixation rate and SST is sea surface temperature.

$$PP_{eu} = 0.66125 * P_{max}^b * \frac{I_0}{I_0 + 4.1} * Z_{eu} * \text{CHL} * \text{DL} \quad (9.3)$$

Where PP_{eu} is the daily amount of carbon fixed integrated from the surface to the euphotic depth ($\text{mgC m}^{-2} \text{ day}^{-1}$), P_{max}^b is the maximum carbon fixation rate within the water column ($\text{mgC mgChl}^{-1} \text{ hr}^{-1}$), I_0 is the daily integrated molar photon flux of sea surface PAR (mol quanta $\text{m}^{-2} \text{ day}^{-1}$), Z_{eu} is the euphotic depth (m), CHL is the daily interpolated CHL-OCI (mg m^{-3}), and DL is the photoperiod (hours) calculated for the day of the year and latitude according to Kirk (1994). The light dependent function ($I_0/(I_0 + 4.1)$) describes the relative change in the light saturation fraction of the euphotic zone as a function of

surface PAR (I_0). Z_{eu} is derived from an estimate of the total chlorophyll concentration within the euphotic layer (CHL_{eu}) based on the Case I models of Morel and Berthon (1989):

- For $CHL_{eu} > 10.0$ $Z_{eu} = 568.2 * CHL_{eu}^{-0.746}$
- For $CHL_{eu} \leq 10.0$ $Z_{eu} = 200.0 * CHL_{eu}^{-0.293}$
- For $CHL_0 \leq 1.0$ $CHL_{eu} = 38.0 * CHL_0^{0.425}$
- For $CHL_0 > 1.0$ $CHL_{eu} = 40.2 * CHL_0^{0.507}$

Where CHL_0 is the surface chlorophyll concentration.

Prior to being input into the VGPM-Eppley model, the daily CHL-OCI and AVHRR SST data were temporally interpolated and smoothed (CHL-OCI_{INT} and SST_{INT} respectively) to increase the data coverage and better match data collected from different sensors and different times. The daily PAR data are not affected by cloud cover and MUR SST data is a blended/gap free data product so these products were not interpolated.

Daily data at each pixel location covering the entire date range were extracted to create a pixel time series ($D_{x,y}$). ($D_{x,y}$) are linearly interpolated based on days in the time series using interpX.pro. Prior to interpolation, the CHL data are log-transformed to account for the log-normal distribution of chlorophyll data (Campbell 1995). Interpolating the entire times series requires a large amount of processing time so the series was processed one year at a time. Each yearly series included 60 days from the previous year and 60 days from the following year to improve the interpolation at the beginning and end of the year. Following interpolation, the data are smoothed with a tri-cube filter (width=7) using IDL's CONVOL program. In order to avoid over interpolating data when there were several days of missing data in the time series, the interpolated data were removed and replaced with blank data if the window of interpolation spanned more than 7 days for CHL or 10 days for SST. After all $D_{x,y}$ pixels had been processed, the one-dimensional pixel time series were converted back to two-dimensional daily files.

Statistics, including the arithmetic mean, geometric mean (for CHL and PP), standard deviation, and coefficient of variation were calculated at daily (3 and 8-day running means), weekly, monthly, and annual time steps and for several climatological periods. Annual statistics used the monthly means as inputs to avoid a summer time bias when more data is available due to reduced cloud cover. The daily, weekly, monthly and annual climatological statistics include the entire time series for each specified period. For example, the climatological January uses the monthly mean from each January in the time series and the climatological annual uses the annual mean from each year. The CHL and PP climatological statistics include data from both SeaWiFS (1997-2007) and MODIS (2008-2017).

Weekly, monthly and annual anomalies were calculated for each product by taking the difference between the mean of the input time period (i.e. week,

month, year) and the climatological mean for the same period. Because bio-optical data are typically log-normally distributed (Campbell 1995), the CHL and PP data were first log-transformed prior to taking the difference and then untransformed, resulting in an anomaly ratio.

The ecological production unit (EPU) shapefile that excludes the estuaries was used to spatially extract all data location within an ecoregion from the statistic and anomaly files. The median values, which are equivalent to the geometric mean, were used for the CHL and PP data. For the extended time series, the 1998-2007 data use the SeaWiFS ocean color products and MODIS-Aqua products were used from 2008 to 2017. Prior to June 2002, AVHRR Pathfinder data are used as the SST source and MUR SST in subsequent years.

9.1.4 Data processing

CHL and PPD time series were formatted for inclusion in the `ecodata` R package using the following R code.

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(stringr)

raw.dir <- here::here("data-raw")

ppd <- read.csv(file.path(raw.dir, "SOE_V2019_2-NES_ECOREGIONS-PPD-STATS_ANOMS-SEAWIFS_MODIS.csv"))
  mutate(ALGORITHM = word(str_replace(ALGORITHM, "_", " "))) %>%
  unite(., VARIABLE, c("VARIABLE", "SENSOR", "ALGORITHM"),
    sep = " ") %>% mutate(VARIABLE = ifelse(str_detect(FILENAME,
  "1998_2018"), paste(VARIABLE, "1998_2018"), ifelse(str_detect(FILENAME,
  "1998_2017"), paste(VARIABLE, "1998_2017"), ifelse(str_detect(FILENAME,
  "1997_2018"), paste(VARIABLE, "1997_2018"), ifelse(str_detect(FILENAME,
  "1997_2017"), paste(VARIABLE, "1997_2017"), VARIABLE)))) %>%
  dplyr::select(TIME, UNITS, VARIABLE, VALUE, REGION) %>% dplyr::rename(Time = TIME,
  Units = UNITS, Var = VARIABLE, EPU = REGION, Value = VALUE)

chl <- read.csv(file.path(raw.dir, "SOE_V2019_2-NES_ECOREGIONS-CHLOR_A-STATS_ANOMS-SEAWIFS_MODIS.csv"))
  mutate(ALGORITHM = word(str_replace(ALGORITHM, "_", " "))) %>%
  unite(., VARIABLE, c("VARIABLE", "SENSOR", "ALGORITHM"),
    sep = " ") %>% mutate(VARIABLE = ifelse(str_detect(FILENAME,
  "1998_2018"), paste(VARIABLE, "1998_2018"), ifelse(str_detect(FILENAME,
  "1998_2017"), paste(VARIABLE, "1998_2017"), ifelse(str_detect(FILENAME,
  "1997_2018"), paste(VARIABLE, "1997_2018"), ifelse(str_detect(FILENAME,
```

```

"1997_2017"), paste(VARIABLE, "1997_2017"), VARIABLE)))) %>%
dplyr::select(TIME, UNITS, VARIABLE, VALUE, REGION) %>% dplyr::rename(Time = TIME,
Units = UNITS, Var = VARIABLE, EPU = REGION, Value = VALUE)

chl_pp <- rbind(ppd, chl)

usethis::use_data(chl_pp, overwrite = T)

```

9.1.5 Plotting

The following figures show examples of how Chlorophyll *a* and primary production data have been included into State of the Ecosystem reports. The figure immediately below shows primary production anomaly plotted with the small-large copepod index (see Zooplankton) in the Mid-Atlantic Bight.

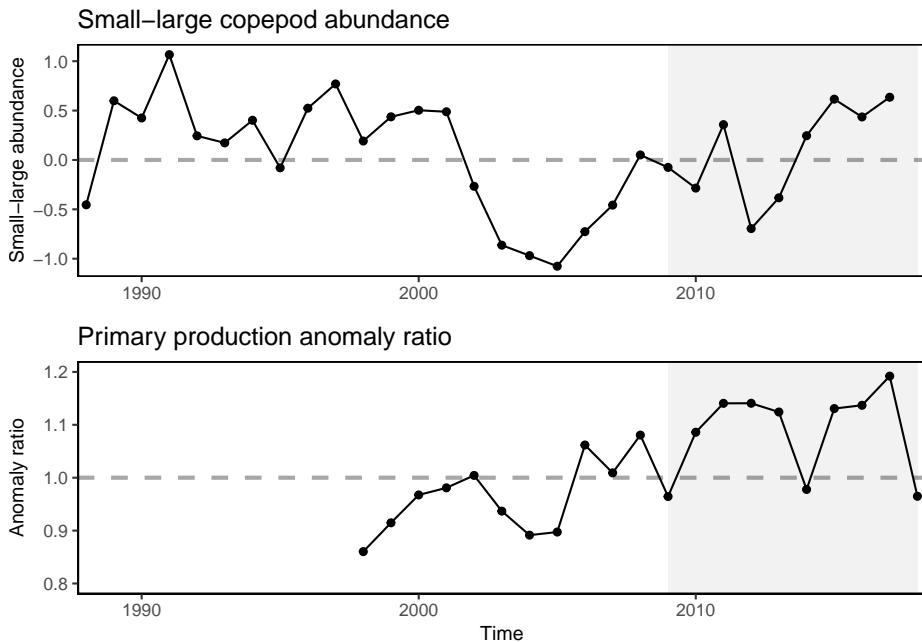


Figure 9.1: MAB small-large zooplankton index and the annual primary production anomaly.

Chl *a* and primary production data were also examined in relation to the long-term means of each series. The figure below shows data specific to the Mid-Atlantic Bight.

```

interp_chl_pp <- function(epu, year = 2018, Variable) {
  out <- ecodata::chl_pp %>% filter(str_detect(Var, Variable),
    EPU == epu) %>% separate(., Time, c("Year", "Week"),
    sep = 4) %>% filter(Year == year) %>% group_by(EPU) %>%
    mutate(Time = 1:length(Year))

  ltm_out <- ecodata::chl_pp %>% filter(str_detect(Var, Variable),
    EPU == epu) %>% separate(., Time, c("Year", "Week"),
    sep = 4) %>% group_by(Week) %>% dplyr::summarise(LTM = mean(Value,
    na.rm = T), SD = sd(Value, na.rm = T)) %>% mutate(Time = 1:length(Week),
    sd.low = LTM - SD, sd.high = LTM + SD) %>% left_join(., out,
    by = c("Time")) %>% mutate(status = ifelse(Value <
    sd.high & Value > sd.low, "near_mean", ifelse(Value >
    sd.high, "high", ifelse(Value < sd.low, "low", NA))),
    group = "PLOT")

  return(ltm_out)
}

MAB_chl <- interp_chl_pp(epu = "MAB", Variable = "WEEKLY_CHLOR_A_MEDIAN MODIS-Aqua PAN")

MAB_chl_weekly <- ggplot(data = MAB_chl) + geom_line(aes(x = Time,
  y = LTM)) + geom_ribbon(aes(x = Time, ymin = sd.low, ymax = sd.high),
  alpha = 0.1, fill = "grey1") + geom_line(aes(x = Time, y = Value),
  size = 1, color = "#33a02c") + ggtitle(expression("Chlorophyll" ~
  italic(a) ~ "")) + guides(color = F) + xlab("") + ylab(expression("CHL (mg m"^-3 * "
  )")) + scale_x_continuous(breaks = seq(1, 52, 10), labels = c("Jan.",
  "Mar.", "May", "July", "Oct.", "Dec."), expand = c(0.01,
  0.01)) + scale_color_manual(values = c("#ef8a62", "#2c7fb8",
  "#a1d99b")) + theme_ts()

MAB_pp <- interp_chl_pp(epu = "MAB", Variable = "WEEKLY_PPD_MEDIAN")

MAB_pp_weekly <- ggplot(data = MAB_pp) + geom_line(aes(x = Time,
  y = LTM)) + geom_ribbon(aes(x = Time, ymin = sd.low, ymax = sd.high),
  alpha = 0.1, fill = "grey1") + geom_line(aes(x = Time, y = Value),
  size = 1, color = "#33a02c") + ggtitle(expression("Primary production")) +
  guides(color = F) + xlab("") + ylab(expression("PP (gC m"^-2 *
  " d"^-1 * ")")) + scale_x_continuous(breaks = seq(1, 52,
  10), labels = c("Jan.", "Mar.", "May", "July", "Oct.", "Dec."),
  expand = c(0.01, 0.01)) + scale_color_manual(values = c("#ef8a62",
  "#2c7fb8", "#a1d99b")) + theme_ts()

# MAB_chl_weekly + MAB_pp_weekly + plot_layout(ncol = 1)

```

MAB_chl_weekly

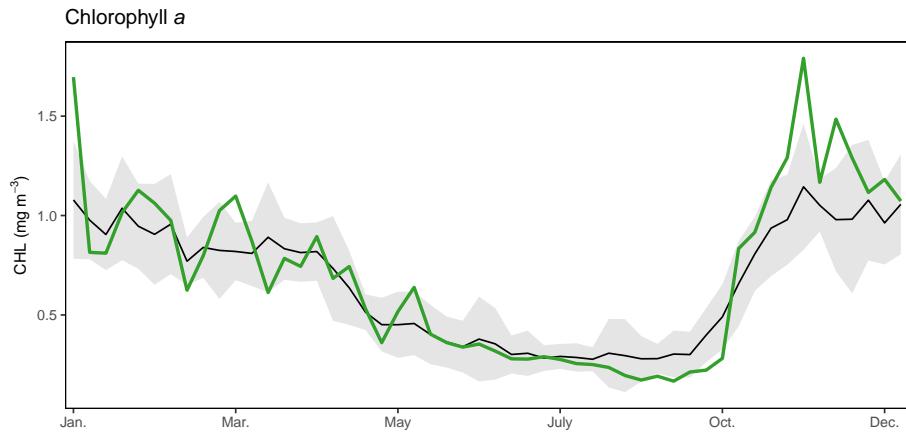


Figure 9.2: Weekly chlorophyll concentrations in the Mid-Atlantic are shown by the colored line for 2018. The long-term mean is shown in black, and shading indicates ± 1 sample SD.

In the figure below, we show monthly primary productivity on an annual time step in the Mid-Atlantic Bight.

```
out_pp <- ecodata::chl_pp %>%
  filter(EPU == "MAB",
         str_detect(Var, "MONTHLY_PPD_MEDIAN")) %>%
  separate(., Time, into = c("Year", "Month"), sep = 4) %>%
  mutate(Month = plyr::mapvalues(Month, from = c("01", "02", "03", "04", "05", "06",
                                                "07", "08", "09", "10", "11", "12"),
                                   to = c(month.abb))) %>%
  group_by(Month) %>%
  mutate(hline = mean(Value))
out_pp$Month <- factor(out_pp$Month, levels = month.abb)

(pp_cci <- ggplot(out_pp) +
  # geom_gls(aes(x = Year, y = Value, group = Month)) +
  geom_point(aes(x = Year, y = Value, group = Month)) +
  geom_line(aes(x = Year, y = Value, group = Month)) +
  scale_x_discrete(name = "Time", breaks = seq(min(out_pp$Year), max(out_pp$Year), 10),
                    geom_hline(aes(yintercept = hline),
                               size = hline.size,
```

```

    alpha = hline.alpha,
    linetype = hline.lty) +
facet_wrap(Month~, ncol = 6) +
ggtitle("Monthly median PPD") +
ylab(expression("PP (gC m"^-2*d"^-1*")")) +
theme_facet() +
theme(axis.text.x = element_text(angle=45, hjust = 1),
      panel.spacing = unit(1, "lines"),
      plot.margin = unit(c(0.1, 0, 0, 0), "cm")))

```

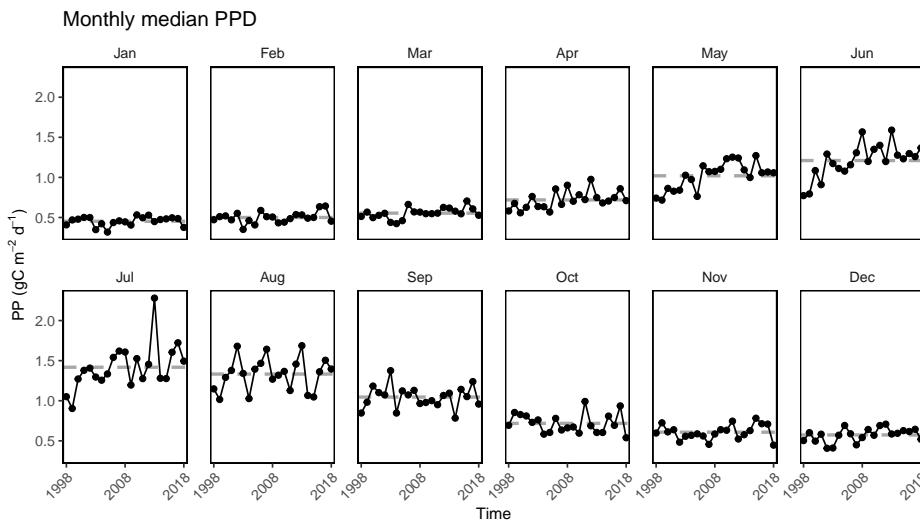


Figure 9.3: Monthly primary production trends show the annual cycle (i.e. the peak during the summer months) and the changes over time for each month.

Chapter 10

Fishing Community Climate Vulnerability

Description: Community climate vulnerability

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018),
State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull with analysis

Contributor(s): Lisa L. Colburn

Data steward: Lisa L. Colburn, lisa.L.colburn@noaa.gov

Point of contact: Lisa L. Colburn, lisa.L.colburn@noaa.gov

Public availability statement: The fisheries data used for this analysis includes confidential information and is not available to the public.

10.1 Methods

10.1.1 Data sources

The data used in community climate vulnerability analyses were derived from the following sources in partnership with the Atlantic Coastal Cooperative Statistics Program's (ACCSP) Standard Atlantic Fisheries Information System (SAFIS).

Database Name	Description
Cfdersyyy	The dealer data are transaction-level pricing at the level of the “market-category.” These data are primarily generated through mandatory reporting by federally-permitted fish dealers. The federal reporting is supplemented with data from non-federally-permitted (state-only) fish dealers. Data are currently reported electronically in partnership with ACCSP through SAFIS.
Cfvessyyy	A related database that contains permit information.

In these databases, the variable “port” contains the port associated with the vessel. The variable “Statenm” refers to the state of the mailing address of the owner.

10.1.2 Data extraction

```

create table cfders2011 as
select *
from connection to oracle
(select port, state, year, dealnum, permit, nespp3, spplndlb, sppvalue from cfders2011
create table cfvess11 as
select *
from connection to oracle
(select permit, homeport, homest from CFDBS.cfvess11 where permit > 0 order by permit)
create table port_name as
select *
from connection to oracle
(select port, portnm from port order by port);
create table st_name as
select *
from connection to oracle
(select state, stateabb from statenm order by state);

Truncated SAS code:
/*CREATE VARIABLES FOR TOTAL LANDINGS WEIGHT AND VALUE (SUM) BY PORT OF LANDING AND BY
data landings_ports1; set landings_ports;
run;
proc sort;
```

```

by port state;
run;
proc means noprint data = landings_ports1; by port state;
var spplndlb sppvalue;
id port state;
output out = landport_totspp sum = L_Totlb L_Totval;
run;
proc sort;
by port;
run;
data landings_ports2; set landings_ports;
run;
proc sort;
by homeport homest;
run;
proc means noprint data = landings_ports2; by homeport homest;
var spplndlb sppvalue;
id homeport homest;
output out = homeport_totspp sum = H_Totlb H_Totval;
run;
proc sort;
by homeport;
run;
/*CREATE SPECIES VARIABLES*/
data landings_ports_NE_spp; set landings_ports;
monklb = 0; monkval = 0; /*monkfish*/
bluelb = 0; blueval = 0; /*bluefish*/
.omitted.
otherlb = 0; otherval = 0; /*other - everything else*/
run;
data landings_ports_NE_spp2; set landings_ports_NE_spp;
if nespp3 = 012 then do; monklb = spplndlb; monkval = sppvalue; end;
...ommitted.
if nespp3 = 406 then do; spotlb = spplndlb; spotval = sppvalue; end;
if nespp3 not in (012, 023, 033, 051, 081, 105, 112, 115, 116, 120, 121, 122, 123, 124, 125, 132,
221, 240, 250, 269, 305, 329, 330, 335, 344, 345, 351, 352, 365, 366, 367, 368, 369, 370, 372, 374,
446, 447, 464, 466, 467, 468, 469, 470, 471, 472, 507, 508, 509, 512, 517, 700, 710, 711, 724, 725,
798, 799, 800, 801, 802, 805, 806, 899, 001, 090, 069, 107, 150, 173, 196, 334, 347, 349, 364, 371);
then do; otherlb = spplndlb; otherval = sppvalue; end;
run;
/*SUM SPECIES LANDINGS BY PORT OF LANDING*/
proc sort; by port; proc means noprint data = landings_ports_NE_spp2; by port state;
. omitted ...
id port state;
output out = spp_porlnd_NE sum = ;

```

```

run;
proc sort;
by port;
run;
/*SUM SPECIES LANDINGS BY HOMEPORt*/
data spp_home; set landings_ports_NE_spp2;
run;
proc sort; by homeport homest; proc means noprint data = spp_home; by homeport homest;
. species are counted..
id homeport homest;
output out = spp_homep_NE sum = ;
run
proc sort;
by homeport; run;
/*MERGE TOTAL PERMITS AND TOTAL DEALERS BY PORT OF LANDING*/
data land_port_totperm2; set land_port_totperm;
run;
proc sort;
by port; run;
data lnd_port_permit; merge spp_porlnd_NE (IN=X) land_port_totperm2 (IN=Y);
by port; if X=1; run;data land_port_totdeal2; set land_port_totdeal;
run;
proc sort;
by port;
run;
data lnd_port_permit_deal; merge lnd_port_permit (IN=x) land_port_totdeal2 (IN=Y);
by port; if X=1; run;
/*MERGE WITH PORT NAME AND STATE ABBREVIATION*/
data lnd_port_permit_deal_nm; merge lnd_port_permit_deal (IN=X) port_name (IN=Y);
by port; if X=1; run;
data lnd_port_permit_deal_nm_st; merge lnd_port_permit_deal_nm (IN=x) st_name (IN=Y);
by port; if X=1; run;
/*MERGE TOTAL PERMITS AND TOTAL DEALERS BY HOMEPORt*/
data home_port_totperm2; set home_port_totperm;
run;
proc sort;
by homeport;
run;
data home_port_permit; merge spp_homep_NE (IN=X) home_port_totperm2 (IN=Y);
by homeport; if X=1; run; data home_port_totdeal2; set home_port_totdeal;
run;
proc sort;
by homeport;
run;
data home_port_permit_deal; merge home_port_permit (IN=x) home_port_totdeal2 (IN=Y);

```

```

by homeport; if X=1; run; proc sort;
by homeport;
run;
/*MERGE TOTAL LANDINGS BY PORT OF LANDING*/
data lnd_port_per_deal_nm_st_tspp; merge lnd_port_permit_deal_nm_st (IN=X) landport_totspp (IN=Y);
by port; if X=1; run;
/*MERGE TOTAL LANDINGS BY HOMEPORT*/
data home_port_per_deal_tspp; merge home_port_permit_deal (IN=X) homeport_totspp (IN=Y);
by homeport; if X=1; run;
data netana.port_landing11; set lnd_port_per_deal_nm_st_tspp;
if state in (22, 32, 24, 42, 7, 35, 33, 8, 23, 49, 36);
run;
proc sort;
by port state;
run;
data netana.homeport11; set home_port_per_deal_tspp;
if homest in ('ME', 'NH', 'MA', 'RI', 'CT', 'NY', 'NJ', 'DE', 'MD', 'VA', 'NC');
run;
proc sort;
by homeport homest;
run;

```

10.1.3 Data analysis

The results described below were developed using the methodology described in Colburn et al. (2016).

1. *Mapping community climate vulnerability* - The map was produced using two variables: total value landed in a community and community species vulnerability, defined below:
 - a. Communities were grouped based on total value of landings into the following categories: 1 (<\$ 200,000), 2 (\$200,000-\$9,999,999), 3 (\$10,000,000-\$49,999,999), and 4 (\$50,000,000 and above). Only communities with a total value landed of \$200,000 or more were selected for the mapping process.
 - b. Community climate vulnerability is determined by the percent contribution of species classified as very high, high, moderate, or low climate vulnerability in a community. The percent contribution of species is calculated as following:
 - % VH & H = value of landing contributed by species classified as having very high or high climate change vulnerability/total value of landings * 100

- % M = value of landing contributed by species classified as having moderate climate change vulnerability/total value of landings * 100
- % L = value of landing contributed by species classified as having low climate change vulnerability/ total value of landings * 100

If a community received a dominant score (50% or more) for any of the above categories, % VH &, %M, or %L, then the community received a respective community species vulnerability ranking of High, Moderate, or Low. For example, if 90% of the total value landed a community is contributed by species classified as having very high or high climate change vulnerability, then this community gets “Very High/High” community species vulnerability. In case of no dominant percentage identified, the community gets a “Mixed” community species vulnerability ranking.

- 2) *Pie charts* - The pie charts were created using the National Marine Fisheries Service (NMFS) landings data pulled from New England Fisheries Science Center (NEFSC) databases in Woods Hole, MA. The percent contribution of each species was calculated by dividing the total value of landings in each port by each species' landed value. Data was calculated and graphed in a pie chart in Excel and given the colors that represent High (red), Moderate (blue), Low (yellow) climate vulnerability. The “other” category consists of species with low landings and/or those that do not have a vulnerability ranking based on Hare et al. (2016). These species were aggregated and given the color gray.

10.1.4 Plotting

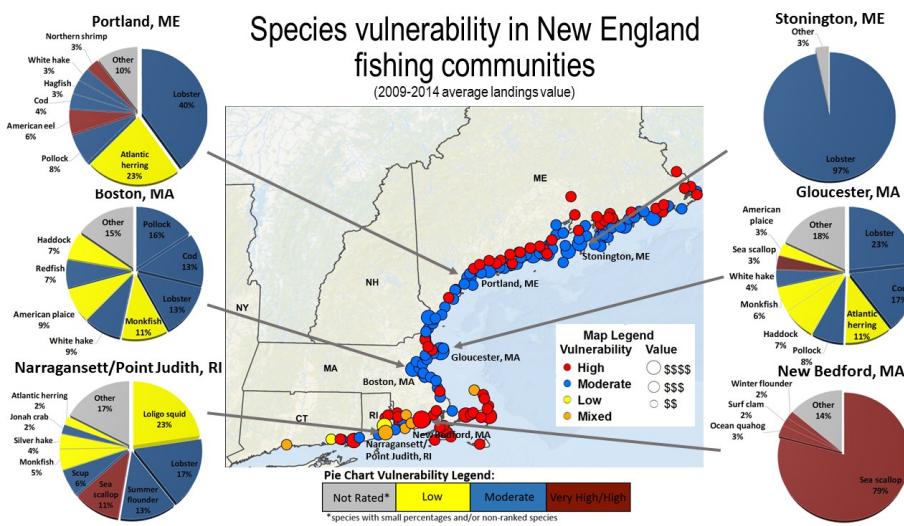


Figure 10.1: Commercial species vulnerability to climate change in in New England fishing communities.

Chapter 11

Conceptual Models

Description: Conceptual models for the New England (Georges Bank and Gulf of Maine) and Mid-Atlantic regions of the Northeast US Large Marine Ecosystem

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018,2019), State of the Ecosystem - Mid-Atlantic (2018,2019)

Indicator category: Synthesis of published information, Extensive analysis; not yet published

Contributor(s): Sarah Gaichas, Patricia Clay, Geret DePiper, Gavin Fay, Michael Fogarty, Paula Fratantoni, Robert Gamble, Sean Lucey, Charles Perretti, Patricia Pinto da Silva, Vincent Saba, Laurel Smith, Jamie Tam, Steve Traynor, Robert Wildermuth

Data steward: Sarah Gaichas, sarah.gaichas@noaa.gov

Point of contact: Sarah Gaichas, sarah.gaichas@noaa.gov

Public availability statement: All source data aside from confidential commercial fisheries data (relevant only to some components of the conceptual models) are available to the public (see Data Sources below).

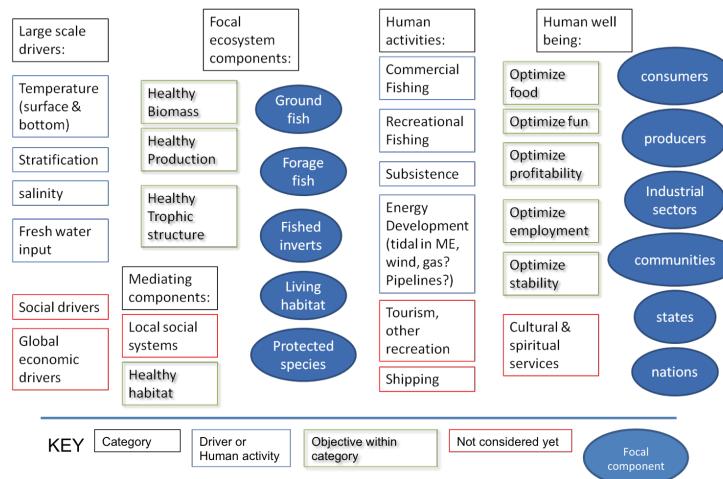
11.1 Methods

Conceptual models were constructed to facilitate multidisciplinary analysis and discussion of the linked social-ecological system for integrated ecosystem assessment. The overall process was to first identify the components of the model (focal groups, human activities, environmental drivers, and objectives), and then to document criteria for including groups and linkages and what the specific links were between the components.

The prototype conceptual model used to design Northeast US conceptual models for each ecosystem production unit (EPU) was designed by the California Current IEA program. The California Current IEA developed an overview conceptual model for the Northern California Current Large Marine Ecosystem (NCC), with models for each focal ecosystem component that detailed the ecological, environmental, and human system linkages. Another set of conceptual models outlined habitat linkages.

An initial conceptual model for Georges Bank and the Gulf of Maine was outlined at the 2015 ICES WGNARS meeting. It specified four categories: Large scale drivers, focal ecosystem components, human activities, and human well being. Strategic management objectives were included in the conceptual model, which had not been done in the NCC. Focal ecosystem components were defined as aggregate species groups that had associated US management objectives (outlined within WGNARS for IEAs, see DePiper et al. (2017)): groundfish, forage fish, fished invertebrates, living habitat, and protected species. These categories roughly align with Fishery Management Plans (FMPs) for the New England Fishery Management Council. The Mid-Atlantic conceptual model was developed along similar lines, but the focal groups included demersals, forage fish, squids, medium pelagics, clams/quahogs, and protected species to better align with the Mid Atlantic Council's FMPs.

First draft IEA Conceptual Model: Georges Bank Gulf of Maine



After the initial draft model was outlined, working groups were formed to develop three submodels following the CCE example: ecological, environmental, and human dimensions. The general approach was to specify what was being included in each group, what relationship was represented by a link between groups, what threshold of the relationship was used to determine whether a relationship was significant enough to be included (we did not want to model everything), the direction and uncertainty of the link, and documentation sup-

porting the link between groups. This information was recorded in a spreadsheet. Submodels were then merged together by common components using the “merge” function in the (currently unavailable) desktop version of Mental Modeler (<http://www.mentalmodeler.org/#home>; Gray et al. (2013)). The process was applied to Georges Bank (GB), the Gulf of Maine (GOM), and the Mid-Atlantic Bight (MAB) Ecological Production Units.

11.1.1 Data sources

11.1.1.1 Ecological submodels

Published food web (EMAX) models for each subregion (J. S. Link et al. 2006; Link et al. 2008), food habits data collected by NEFSC trawl surveys (Smith and Link 2010), and other literature sources (Smith et al. 2015) were consulted. Expert judgement was also used to adjust historical information to current conditions, and to include broad habitat linkages to Focal groups.

11.1.1.2 Environmental submodels

Published literature on the primary environmental drivers (seasonal and inter-annual) in each EPU was consulted. Sources for Georges Bank included Backus and Bourne (1987) and Townsend et al. (2006). Sources for the Gulf of Maine included Smith (1983), Smith et al. (2001), Mupparapu and Brown (2002), Townsend et al. (2006), Smith et al. (2012), and Mountain (2012).

Sources for the Mid Atlantic Bight included Houghton et al. (1982), Beardsley et al. (1985), Lentz (2003), Mountain (2003), Glenn et al. (2004), Sullivan, Cowen, and Steves (2005), Castelao et al. (2008), Shearman and Lentz (2009), Castelao, Glenn, and Schofield (2010), Gong, Kohut, and Glenn (2010), Gawarkiewicz et al. (2012), Forsyth, Andres, and Gawarkiewicz (2015), Fratantoni, Holzwarth-Davis, and Taylor (2015), Zhang and Gawarkiewicz (2015), Miller, Hare, and Alade (2016), and Lentz (2017).

11.1.1.3 Human dimensions submodels

Fishery catch and bycatch information was drawn from multiple regional datasets, including the Greater Atlantic Regional Office Vessel Trip Reports & Commercial Fisheries Dealer databases, Northeast Fishery Observer Program & Northeast At-Sea Monitoring databases, Northeast Fishery Science Center Social Sciences Branch cost survey, and the Marine Recreational Informational Program database. Further synthesis of human welfare derived from fisheries was drawn from Färe, Kirkley, and Walden (2006), Walden et al. (2012), Lee and Thunberg (2013), Lee (2014), and Lee, Steinback, and Wallmo (2017). Bycatch of protected species was taken from Waring et al. (2015), with

additional insights from Bisack and Magnusson (2014). The top 3 linkages were drawn for each node. For example, the top 3 recreational species for the Mid-Atlantic were used to draw linkages between the recreational fishery and species focal groups. A similar approach was used for relevant commercial fisheries in each region.

Habitat-fishery linkages were drawn from unpublished reports, including:

1. Mid-Atlantic Fishery Management Council. 2016. Amendment 16 to the Atlantic Mackerel, Squid, and Butterfish Fishery Management Plan: Measures to protect deep sea corals from Impacts of Fishing Gear. Environmental Assessment, Regulatory Impact Review, and Initial Regulatory Flexibility Analysis. Dover, DE. August, 2016.
2. NOAA. 2016. Deep sea coral research and technology program 2016 Report to Congress. <http://www.habitat.noaa.gov/protection/corals/deepseacorals.html> retrieved February 8, 2017.
3. New England Fishery Management Council. 2016. Habitat Omnibus Deep-Sea Coral Amendment: Draft. <http://www.nefmc.org/library/omnibus-deep-sea-coral-amendment> Retrieved Feb 8, 2017.
4. Bachman et al. 2011. The Swept Area Seabed Impact (SASI) Model: A Tool for Analyzing the Effects of Fishing on Essential Fish Habitat. New England Fisheries Management Council Report. Newburyport, MA.

Tourism and habitat linkages were drawn from unpublished reports, including:

1. <http://neers.org/RESOURCES/Bibliographies.html>
2. Great Bay (GoM) resources <http://greatbay.org/about/publications.htm>
3. Meaney, C.R. and C. Demarest. 2006. Coastal Polution and New England Fisheries. Report for the New England Fisheries Management Council. Newburyport, MA.
4. List of valuation studies, by subregion and/or state, can be found at <http://www.oceanconomics.org/nonmarket/valestim.asp>.

Published literature on human activities in each EPU was consulted.

Sources for protected species and tourism links included Hoagland and Meeks (2000) and Lee (2010).

Sources for links between environmental drivers and human activities included Adams (1973), Matzarakis and Freitas (2001), Scott, McBoyle, and Schwartzen-truber (2004), Hess, Malilay, and Parkinson (2008), Colburn and Jepson (2012), Jepson and Colburn (2013), and Colburn et al. (2016).

Sources for cultural practices and attachments links included Pauly (1997), McGoodwin (2001), St Martin (2001), Norris-Raynbird (2004), Pollnac et al. (2006), Clay and Olson (2007), Clay and Olson (2008), Everett and Aitchison (2008), Donkersloot (2010), Lord (2011), Halpern et al. (2012), Wynveen, Kyle, and Sutton (2012), Cortes-Vazquez and Zedalis (2013), Koehn, Reineeman, and Kittinger (2013), Potschin and Haines-Young (2013), Reed et al. (2013), Urquhart and Acott (2013), Blasiak et al. (2014), Klain, Satterfield, and Chan (2014), Poe, Norman, and Levin (2014), Brown (2015), Donatuto and Poe (2015), Khakzad and Griffith (2016), Oberg et al. (2016), and Seara, Clay, and Colburn (2016).

11.1.2 Data extraction

11.1.2.1 Ecological submodels

“Data” included model estimated quantities to determine whether inclusion thresholds were met for each potential link in the conceptual model. A matrix with diet composition for each modeled group is an input to the food web model. A matrix of mortalities caused by each predator and fishery on each modeled group is a direct output of a food web model (e.g. Ecopath). Food web model biomass flows between species, fisheries, and detritus were summarized using algorithms implemented in visual basic by Kerim Aydin, NOAA NMFS Alaska Fisheries Science Center. Because EMAX model groups were aggregated across species, selected diet compositions for individual species were taken from the NEFSC food habits database using the FEAST program for selected species (example query below). These diet queries were consulted as supplemental information.

Example FEAST sql script for Cod weighted diet on Georges Bank. Queries for different species are standardized by the FEAST application and would differ only in the svsp code.

```
Select svsp,year,cruise6,stratum,station,catsex,pdid,pdgutw,pflen,pdwgt,perpyw,pyamtw,COLLCAT,nu
Select distinct svsp,year,cruise6,stratum,station from fhds.allfh_feast where pynam <> 'BLOWN'
Select distinct svsp,year,cruise6,stratum,station,catsex,catnum from fhds.allfh_feast where pynam <> 'BLOWN'
Select distinct COLLCAT from fhds.allfh_feast where pynam <> 'BLOWN' and pynam <> 'PRESERVED' and pynam <> 'UNKNOWN'
Select distinct svsp,year,cruise6,stratum,station,catsex,pdid,pflen,pdgutw,pdwgt from fhds.allfh_feast
Select svsp,year,cruise6,stratum,station,catsex,pdid,pflen,COLLCAT,sum(perpyw),sum(pyamtw),sum(pdgutw)
Select svsp,year,cruise6,stratum,station,COLLCAT,sum(pyamtv) sumpvol from fhds.allfh_feast where pynam <> 'BLOWN'
Select svsp,year,cruise6,stratum,station, count(distinct pdid) nstom from fhds.allfh_feast where pynam <> 'BLOWN'
Select svsp,year,cruise6,stratum,station,pflen,numlen,count(distinct pdid) nstom from fhds.allfh_feast
Select svsp,year,cruise6,stratum,station,pflen,COLLCAT,sum(pyamtv) sumpvol from fhds.allfh_feast
Select distinct svsp,year,cruise6,stratum,station,pdid,pflen from fhds.allfh_feast where pynam <> 'BLOWN'
Select distinct year,cruise6,stratum,station,beglat,beglon from fhds.allfh_feast where pynam <> 'BLOWN'
```

11.1.2.2 Environmental submodels

Information was synthesized entirely from published sources and expert knowledge; no additional data extraction was completed for the environmental submodels.

11.1.2.3 Human dimensions submodels

Recreational fisheries data were extracted from the 2010-2014 MRIP datasets. Original data can be found here for each region (New England or Mid-Atlantic as defined by states).

Commercial fishing data was developed as part of the State of the Ecosystem Report, including revenue and food production estimates, with data extraction methodology discussed in the relevant sections of the technical document. In addition, the Northeast Regional Input/Output Model (Steinback and Thunberg 2006) was used as the basis for the strength of the employment linkages.

11.1.3 Data analysis**11.1.3.1 Ecological submodels**

Aggregated diet and mortality information was examined to determine the type of link, direction of link, and which links between which groups should be included in the conceptual models. Two types of ecological links were defined using food web models: prey links and predation/fishing mortality links. Prey links resulted in positive links between the prey group and the focal group, while predation/fishing mortality links resulted in negative links to the focal group to represent energy flows. The intent was to include only the most important linkages between focal groups and with other groups supporting or causing mortality on focal species groups. Therefore, threshold levels of diet and mortality were established (based on those that would select the top 1-3 prey and predators of each focal group): 10% to include a link (or add a linked group) in the model and 20% to include as a strong link. A Primary Production group was included in each model and linked to pelagic habitat to allow environmental effects on habitat to be connected to the ecological submodel. Uncertainty for the inclusion of each link and for the magnitude of each link was qualitatively assessed and noted in the spreadsheet.

Four habitat categories (Pelagic, Seafloor and Demersal, Nearshore, and Freshwater and Estuarine) were included in ecological submodels as placeholders to be developed further along with habitat-specific research. Expert opinion was used to include the strongest links between each habitat type and each Focal group (noting that across species and life stages, members of these aggregate groups likely occupy many if not all of the habitat types). Link direction and

strength were not specified. Environmental drivers were designed to link to habitats, rather than directly to Focal groups, to represent each habitat's important mediation function.

EMAX model groups were aggregated to focal groups for the Georges Bank (GB), Gulf of Maine (GOM) and Mid-Atlantic Bight (MAB) conceptual models according to Table 11.1. “Linked groups” directly support or impact the Focal groups as described above.

Ecological submodels were constructed and visualized in Mental Modeler (Fig. 11.1). Here, we show only the Gulf of Maine submodels as examples.

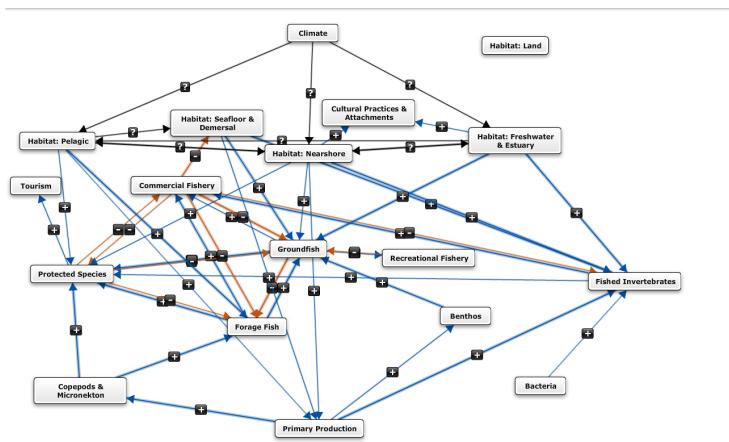


Figure 11.1: Gulf of Maine Ecological submodel

11.1.3.2 Environmental submodels

Environmental submodels were designed to link key oceanographic processes in each ecosystem production unit to the four general habitat categories (Pelagic, Seafloor and Demersal, Nearshore, and Freshwater and Estuarine) with emphasis on the most important physical processes in each ecosystem based on expert knowledge as supported by literature review. The basis of each submodel were environmental variables observable at management-relevant scales as identified by WGNARS: Surface and Bottom Water Temperature and Salinity, Freshwater Input, and Stratification (as well as sea ice timing and cover, which is not relevant to the northeast US shelf). Key drivers changing these observable variables and thus structuring habitat dynamics in each Ecological Production Units were added to the model using expert consensus.

Environmental submodels were initially constructed and visualized in Mental Modeler (Fig. 11.2).

Table 11.1: Relationship between food web model groups and conceptual model focal groups

Group Type	Region	Conceptual model group	EMAX group(s)
Focal	GB	Commercial Fishery	Fishery
Focal	GB	Fished Inverts	Megabenthos filterers
Focal	GB	Forage Fish	Sum of Small pelagics—commercial, other, anadromous,
Focal	GB	Groundfish	Sum of Demersals—omnivores, benthivores, and piscivores
Focal	GB	Protected Species	Sum of Baleen Whales, Odontocetes, and Seabirds
Linked	GB	Benthos	Sum of Macrofauna—polychaetes, crustaceans, molluscs
Linked	GB	Copepods and Microneuston	Sum of Copepods—small and large, and Microneuston
Linked	GB	Detritus and Bacteria	Sum of Bacteria and Detritus-POC
Linked	GB	Gelatinous zooplankton	Gelatinous zooplankton
Linked	GB	Primary Production	Phytoplankton-Primary production
Focal	GOM	Commercial Fishery	Fishery
Focal	GOM	Fished Inverts	Megabenthos filterers
Focal	GOM	Forage Fish	Sum of Small pelagics—commercial, other, anadromous,
Focal	GOM	Groundfish	Sum of Demersals—omnivores, benthivores, and piscivores
Focal	GOM	Protected Species	Sum of Baleen Whales, Odontocetes, Pinnipeds, and Seabirds
Linked	GOM	Benthos	Sum of Macrofauna—polychaetes, crustaceans, molluscs
Linked	GOM	Copepods and Microneuston	Sum of Copepods—small and large, and Microneuston
Linked	GOM	Detritus and Bacteria	Sum of Bacteria and Detritus-POC
Linked	GOM	Gelatinous zooplankton	Gelatinous zooplankton
Linked	GOM	Primary Production	Phytoplankton-Primary production
Focal	MAB	Clams Quahogs	Megabenthos filterers
Focal	MAB	Commercial Fishery	Fishery
Focal	MAB	Demersals	Sum of Demersals—omnivores, benthivores, and piscivores
Focal	MAB	Forage Fish	Sum of Small pelagics—commercial, other, and anadromous
Focal	MAB	Medium Pelagics	Medium pelagics
Focal	MAB	Protected Species	Sum of Baleen whales and Odontocetes
Focal	MAB	Squids	Small pelagics—squids
Linked	MAB	Benthos	Sum of Macrofauna—polychaetes, crustaceans, molluscs
Linked	MAB	Copepods and Microneuston	Sum of Copepods—small and large, and Microneuston
Linked	MAB	Detritus and Bacteria	Sum of Bacteria and Detritus-POC
Linked	MAB	Gelatinous zooplankton	Gelatinous zooplankton
Linked	MAB	Primary Production	Phytoplankton-Primary production
Linked	MAB	Sharks	Sum of Sharks—pelagic and coastal

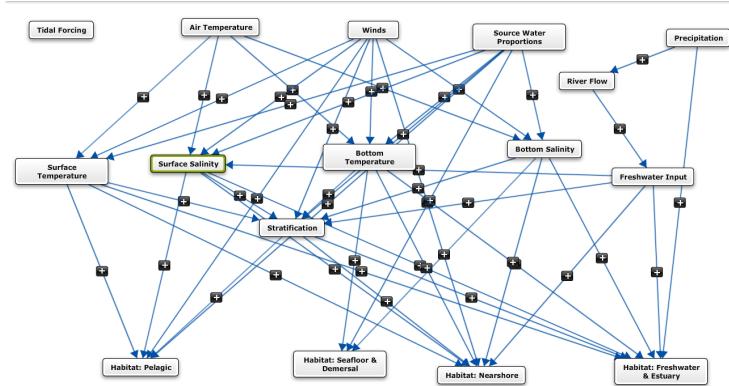


Figure 11.2: Gulf of Maine Environmental submodel

11.1.3.3 Human dimensions submodels

The top 3 species from each mode of recreational fishing (shoreside, private boat, party/charter) were used to assess the potential for missing links between the recreational fishing activity and biological focal components. Given the predominance of Mid-Atlantic groundfish in recreational fishing off New England (summer flounder, bluefish, striped bass), a Mid-Atlantic groundfish focal component was added to the Georges Bank EPU model. The magnitude of benefits generated from recreational fishing was scaled to reflect expert knowledge of target species, coupled with the MRIP data highlighted above. Scales were held consistent across the focal components within recreational fishing.

No additional biological focal components were added to the commercial fishing activity, beyond what was developed in the ecological submodel. Benefits derived from commercial fishing were scaled to be consistent with the State of the Ecosystem revenue estimates, as modulated by expert knowledge and additional data sources. For example, the percentage of landings sold as food was used to map fishing activity to the commercial fishery food production objective, and the Northeast Regional Input/Output Model (Steinback and Thunberg 2006) was used to define the strength of the employment linkages. For profitability, expert knowledge was used to reweight revenue landings, based on ancillary cost data available (Das, Chhandita 2013, 2014). Human activities and objectives for the conceptual sub model are defined in DePiper et al. (2017). As shown in Figure 11.3, human dimensions submodels were also initially constructed and visualized in Mental Modeler.

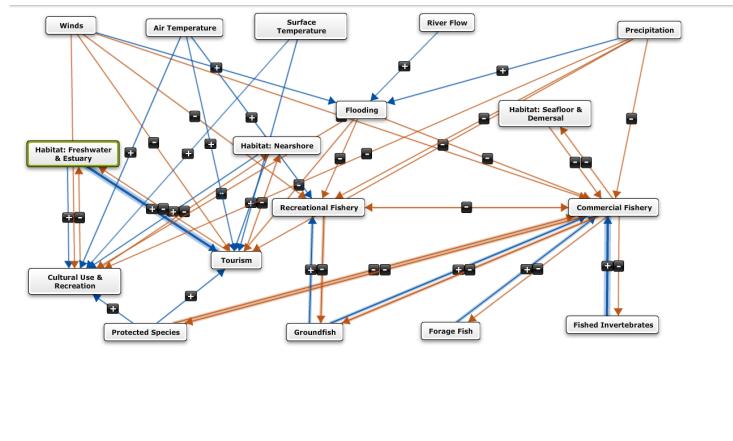


Figure 11.3: Gulf of Maine Human dimensions submodel

11.1.3.4 Merged models

All links and groups from each submodel were preserved in the full merged model for each system. Mental modeler was used to merge the submodels. Full models were then re-drawn in Dia (<http://dia-installer.de/>) with color codes for each model component type for improved readability. Examples for each system are below.

11.1.3.5 Communication tools

The merged models were redrawn for use in communications with the public. These versions lead off the State of the Ecosystem reports for both Fishery Management Councils to provide an overview of linkages between environmental drivers, ecological, and human systems.

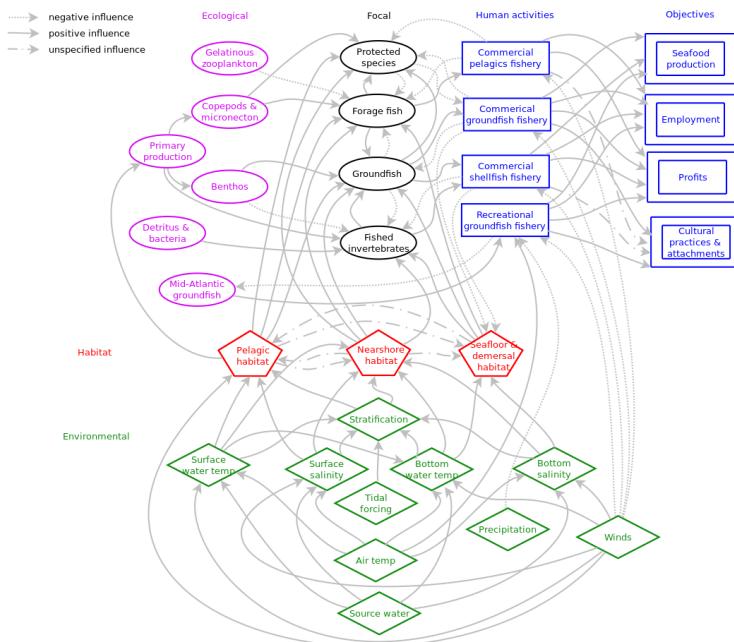


Figure 11.4: Georges Bank conceptual model

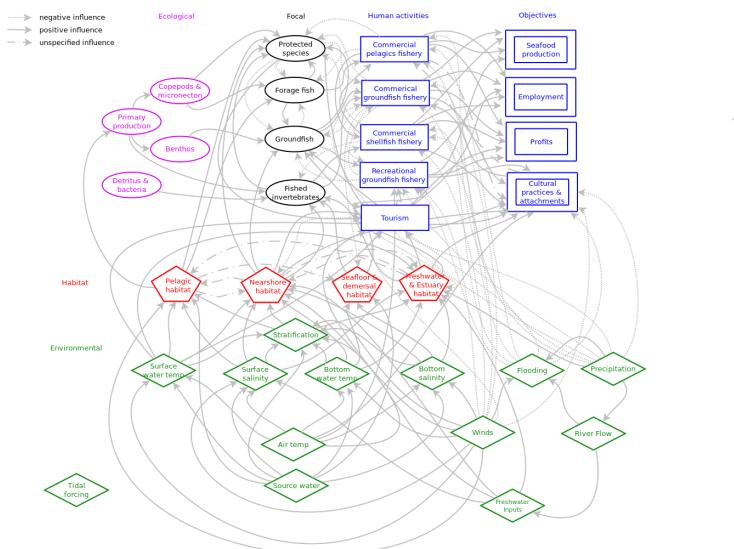


Figure 11.5: Gulf of Maine conceptual model

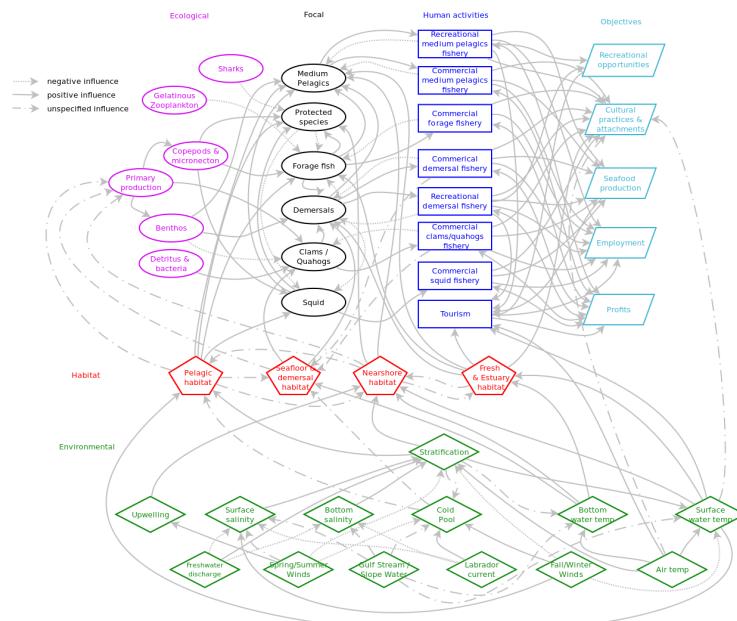


Figure 11.6: Mid-Atlantic Bight conceptual model

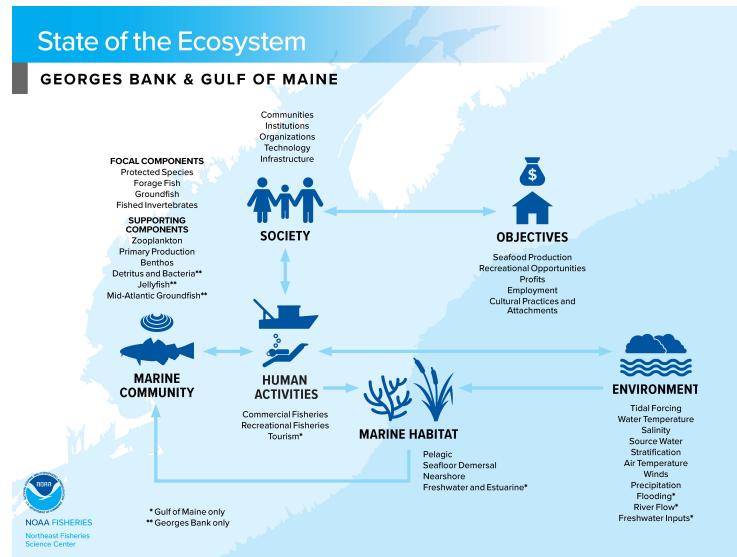


Figure 11.7: New England conceptual model for public communication

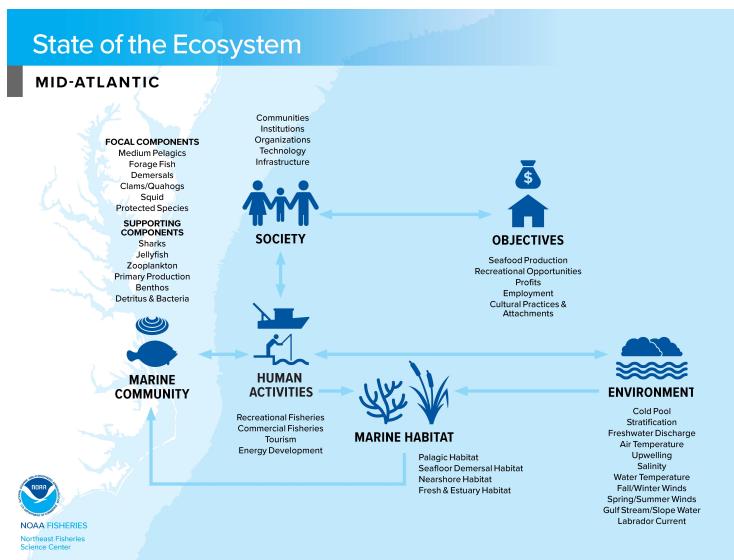


Figure 11.8: Mid-Atlantic conceptual model for public communication

Chapter 12

Fish Condition Indicator

Description: Relative condition

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018,2019),
State of the Ecosystem - Mid-Atlantic (2018,2019)

Indicator category: Database pull with analysis

Contributor(s): Laurel Smith

Data steward: Laurel Smith, laurel.smith@noaa.gov

Point of contact: Laurel Smith, laurel.smith@noaa.gov

Public availability statement: NEFSC survey data used in these analyses
are available upon request (see BTS metadata for access procedures). Derived
condition data are available here.

12.1 Methods

Relative condition (Kn) was introduced by Cren (1951) as a way to remove
the influence of length on condition, and Blackwell, Brown, and Willis (2000)
noted that Kn may be useful in detecting prolonged physical stress on a fish
populations. Relative condition is calculated as

$$Kn = W/W',$$

where W is the weight of an individual fish and W' is the predicted length-
specific mean weight for the fish population in a given region. "Here, relative
condition was calculated for finfish stocks commonly caught on the Northeast
Fisheries Science Center's (NEFSC) autumn and spring bottom trawl surveys,
from 1992-present.

Where data allowed, predicted length-weight parameters were calculated for W' by species, sex and season over the time period 1992-2012. When sample sizes of individual fish weights and lengths were too low, parameters were calculated for aggregated spring and fall survey data over the same time period. Fall survey relative condition was calculated by sex for those species that exhibited differences in growth between sexes and aggregated across sex for those that did not.

12.1.1 Data sources

Individual fish lengths (to the nearest 0.5 cm) and weights (grams) were collected on the NEFSC bottom trawl surveys from 1992-present aboard RVs Albatross IV, Delaware II and the Henry B. Bigelow (see *Survdat*). A small number of outlier values were removed when calculating the length-weight parameters.

12.1.2 Data extraction

Data were extracted from NEFSC's survey database (SVDBS) using SQL.

SQL query:

```
SELECT cruise6,stratum,tow,station,
       year,month,day,time,beglat,beglon,setdepth,
       surftemp,bottemp,
       svsp,sex,length,age,maturity,indid,indwt,stom_volume,stom_wgt, expcatchwt, expcatch
FROM connection TO oracle
(SELECT b.cruise6,b.stratum,b.tow,b.station,
       s.est_year year,est_month month,est_day day,
       substr(est_time,1,2)||substr(est_time,4,2) time,
       round(substr(beglat,1,2) + (substr(beglat,3,7)/60),6) beglat,
       round(((substr(beglon,1,2) + (substr(beglon,3,7)/60)) * -1), 6) beglon,
       setdepth,surftemp, bottemp,
       b.svsp,sex,length,age,maturity,indid,indwt,stom_volume,stom_wgt, expcatchwt, expcatch
FROM union_fscs_svbio b, union_fscs_svcat p, union_fscs_svsta s, svdbs_cruises c
WHERE
       season = &sson AND
       b.svsp IN ('013','015','023','026','028','032','072','073','074','075','076','077',
       (b.cruise6=s.cruise6) AND
       (c.cruise6=b.cruise6) AND
       (p.cruise6=c.cruise6) AND
       (p.stratum=b.stratum) AND
       (b.stratum=s.stratum) AND
       (p.station=b.station) AND
       (b.station=s.station) AND
```

```

(p.svspp=b.svspp) and
(p.tow=b.tow) and
(b.tow=s.tow) ;

%put &sqlxmsg;
%put &sqlxrc;

create view spp as

select comname, svsp
from connection to oracle
(select comname, svsp
from svspes_list);

%put &sqlxmsg;
%put &sqlxrc;

execute (commit) by oracle;

```

12.1.3 Data analysis

The following growth curve was fit through individual fish lengths and weights from the NEFSC bottom trawl survey data from 1992-2012 to produce reference length-weight parameters:

$$\text{Weight} = e^{Fall_{coef}} * \text{Length}^{Fall_{exp}},$$

where length is in cm and weight is in kg. Fall survey data were used where sample sizes allowed for growth curve estimation, otherwise data from spring and fall seasons were combined.

Individual fish lengths from NEFSC fall bottom trawl survey from 1992-2017 were then used to calculate predicted weights using the reference length-weight parameters. Relative condition (Kn) was calculated annually by species and sex (for sexually dimorphic species) by dividing individual fish weights by the predicted weight.

The following R code was used in the analysis:

```

# Length-weight parameter calculation:
function (data, min.n = 25, min.range = 5, data.avail = NA, data.avail.bigelow = NA)
{
  if(is.null(dim(data.avail))) data.avail <- lw.data.availability(data, min.n, min.range)
  data.avail <- data.avail[apply(data.avail[,2:5], 1, any),]
  if(is.null(dim(data.avail.bigelow))) data.avail.bigelow <- lw.data.availability(data[data$dat

```

```

data.avail.bigelow <- data.avail.bigelow[apply(data.avail.bigelow[,2:5], 1, any),]
data.spp <- as.numeric(rownames(data.avail[data.avail$sex.season == TRUE,]))
lw.output <- data.frame(matrix(ncol = 12))
names(lw.output) <- c("species.name", "species.code", "source", "sex", "season", "sp")
for(sp in data.spp){
  sp.data <- lw.data[lw.data$species == sp,]
  sp.name <- unique(as.character(species.names$scientific_name[species.names$svsp])
  print(sp.name)
#All model
  print("Species Level")
  this.data <- bigelow.test(sp.data, data.avail[as.character(sp),], data.avail.b
  ds <- this.data[[2]]
  this.data <- this.data[[1]]
  if(!is.null(dim(this.data))){
    this.model <- lm(weight.log~length.log, data = this.data)
    model.coefs <- coef(this.model)
    model.summary <- coef(summary(this.model))
    length.range <- range(sp.data$length)
    length.log <- log(seq(length.range[1], length.range[2], by=.5))
    species <- rep(sp, length(length.log))
    predict.length <- data.frame(species, length.log)
    check.diffs <- plot.lw(this.data, this.model, "species", sp.name, predict.l
    lw.output <- rbind(lw.output, c(sp.name, sp, ds, NA, NA, model.coefs["leng
  }
#Sex model
  print("Sex Level")
  this.data <- sp.data[sp.data$sex > 0,]
  model.definition <- "weight.log~length.log * factor(sex)"
  this.data <- bigelow.test(this.data, data.avail[as.character(sp),], data.avail.
  ds <- this.data[[2]]
  this.data <- this.data[[1]]
  if(!is.null(dim(this.data))){
    this.model <- lm(weight.log~length.log * factor(sex), data = this.data)
    male.range <- range(this.data$length[this.data$sex == 1])
    female.range <- range(this.data$length[this.data$sex == 2])
    model.summary <- coef(summary(this.model))
    if(any(model.summary[grep("sex", rownames(model.summary)), "Pr(>|t|)"] <=
        model.coefs <- coef(this.model)
        length.log <- rep(log(seq(length.range[1], length.range[2], by=.5)),2)
        sex <- c(rep(1, length(length.log)/2), rep(2, length(length.log)/2))
        predict.length <- data.frame(length.log, sex)
        check.diffs <- plot.lw(this.data, this.model, "sex", sp.name, predict.l
        lw.output <- rbind(lw.output, c(sp.name, sp, ds, 1, NA, model.coefs["l
        lw.output <- rbind(lw.output, c(sp.name, sp, ds, 2, NA, model.coefs["l
  }
}

```

```

    else{
      print(paste("Model parameters not significantly different for", model.definition))
    }
  }

#Season model
model.definition <- "weight.log~length.log * factor(season)"
this.data <- bigelow.test(sp.data, data.avail[as.character(sp),], data.avail.bigelow[as.character(sp),])
ds <- this.data[[2]]
this.data <- this.data[[1]]
if(!is.null(dim(this.data))){
  this.model <- lm(weight.log~length.log * factor(season), data = this.data)
  fall.range <- range(this.data$length[this.data$season == "FALL"])
  spring.range <- range(this.data$length[this.data$season == "SPRING"])
  model.summary <- coef(summary(this.model))
  if(any(model.summary[grep("season", rownames(model.summary)), "Pr(>|t|)"] <= .05)){
    model.coeffs <- coef(this.model)
    length.log <- rep(log(seq(length.range[1], length.range[2], by=.5)),2)
    season <- c(rep("FALL", length(length.log)/2), rep("SPRING", length(length.log)/2))
    predict.length <- data.frame(length.log, season)
    check.diffs <- plot.lw(this.data, this.model, "season", sp.name, predict.length)
    lw.output <- rbind(lw.output, c(sp.name, sp, ds, NA, "FALL", model.coeffs["length.log"]))
    lw.output <- rbind(lw.output, c(sp.name, sp, ds, NA, "SPRING", model.coeffs["length.log"]))
  }
  else{
    print(paste("Model parameters not significantly different for", model.definition))
  }
}

#Sex-Season model
this.data <- sp.data[sp.data$sex > 0,]
model.definition <- "weight.log~length.log * factor(sex) * factor(season)"
this.data <- bigelow.test(this.data, data.avail[as.character(sp),], data.avail.bigelow[as.character(sp),])
ds <- this.data[[2]]
this.data <- this.data[[1]]
if(!is.null(dim(this.data))){
  this.model <- lm(weight.log~length.log * factor(sex) * factor(season), data = this.data)
  model.summary <- coef(summary(this.model))
  male.fall.range <- range(this.data$length[this.data$season == "FALL" & this.data$sex == "M"])
  male.spring.range <- range(this.data$length[this.data$season == "SPRING" & this.data$sex == "M"])
  female.fall.range <- range(this.data$length[this.data$season == "FALL" & this.data$sex == "F"])
  female.spring.range <- range(this.data$length[this.data$season == "SPRING" & this.data$sex == "F"])
  if(any(model.summary[grep("season", rownames(model.summary)), "Pr(>|t|)"] <= .05 | any(model.summary[grep("sex", rownames(model.summary)), "Pr(>|t|)"] <= .05))){
    model.coeffs <- coef(this.model)
    male.fall.int <- model.coeffs["(Intercept)"]
    male.fall.int.p <- model.summary["(Intercept)", "Pr(>|t|)"]
    male.fall.slope <- model.coeffs["length.log"]
  }
}

```

```

male.fall.slope.p <- model.summary["length.log", "Pr(>|t|)"]
male.spring.int <- male.fall.int + model.coefs["factor(season)SPRING"]
male.spring.int.p <- model.summary["factor(season)SPRING", "Pr(>|t|)"]
male.spring.slope <- male.fall.slope + model.coefs["length.log:factor(sea"]
male.spring.slope.p <- model.summary["length.log:factor(season)SPRING"]
female.fall.int <- male.fall.int + model.coefs["factor(sex)2"]
female.fall.int.p <- model.summary["factor(sex)2", "Pr(>|t|)"]
female.fall.slope <- male.fall.slope + model.coefs["length.log:factor(sex)2"]
female.fall.slope.p <- model.summary["length.log:factor(sex)2", "Pr(>|t|)"]
female.spring.int <- male.spring.int + model.coefs["factor(sex)2"] + model.coef
female.spring.int.p <- model.summary["factor(sex)2:factor(season)SPRING"]
female.spring.slope <- male.spring.slope + model.coefs["length.log:factor(sex)2"]
female.spring.slope.p <- model.summary["length.log:factor(sex)2:factor(season)SPRIN
length.log <- rep(log(seq(length.range[1], length.range[2], by=.5)), 4)
sex <- c(rep("1", length(length.log)/2), rep("2", length(length.log)/2))
season <- rep(c(rep("FALL", length(length.log)/4), rep("SPRING", length(length.log)/4))
predict.length <- data.frame(length.log, sex, season)
check.diffs <- plot.lw(this.data, this.model, c("sex", "season"), sp.name)
lw.output <- rbind(lw.output, c(sp.name, sp, ds, 1, "FALL", male.fall.int))
lw.output <- rbind(lw.output, c(sp.name, sp, ds, 1, "SPRING", male.spring.int))
lw.output <- rbind(lw.output, c(sp.name, sp, ds, 2, "FALL", female.fall.int))
lw.output <- rbind(lw.output, c(sp.name, sp, ds, 2, "SPRING", female.spring.int))
}
else{
  print(paste("Model parameters not significantly different for", model.coef))
}
}
lw.output <- lw.output[!is.na(lw.output$species.code),]
lw.output
}
#Relative Condition:
proc import datafile = "lw_parameters.csv"
  out = LWparams
  dbms = csv
  replace;
  getnames = yes;
run;

data LWparams; set LWparams;
  if LW_SVSPP = 13 then sv spp = '013';
  if LW_SVSPP = 15 then sv spp = '015';
  if LW_SVSPP = 23 then sv spp = '023';
  if LW_SVSPP = 26 then sv spp = '026';
  if LW_SVSPP = 28 then sv spp = '028';

```

```

if LW_SVSPP = 32 then svsp = '032';
if LW_SVSPP = 72 then svsp = '072';
if LW_SVSPP = 73 then svsp = '073';
if LW_SVSPP = 74 then svsp = '074';
if LW_SVSPP = 75 then svsp = '075';
if LW_SVSPP = 76 then svsp = '076';
if LW_SVSPP = 77 then svsp = '077';
if LW_SVSPP = 78 then svsp = '078';
if LW_SVSPP ge 100 then svsp = LW_SVSPP;
if sexMF = 'M' then sex = '1';
if sexMF = 'F' then sex = '2';
if sexMF = ' ' then sex = '0';
if EXPONENT_FALL = . then EXPONENT_FALL=SEASONLESS_EXPONENT;
if EXPONENT_SPRING = . then EXPONENT_SPRING=SEASONLESS_EXPONENT;
if COEFFICIENT_FALL = . then COEFFICIENT_FALL=SEASONLESS_COEFFICIENT;
if COEFFICIENT_SPRING = . then COEFFICIENT_SPRING=SEASONLESS_COEFFICIENT;

proc sort data=LWparams;
by svsp sex;

proc sort data=lenwt;
by svsp sex;

data lwdatpar (keep =cruise6 stratum tow station year month day time beglat beglon setdepth
               surftemp bottemp svsp sex length age maturity indid indwt stom_volume stom_wgt expcatchwt exp
               COEFFICIENT_SPRING EXPONENT_SPRING COEFFICIENT_FALL EXPONENT_FALL SEASONLESS_COEFFICIENT
               SEASONLESS_EXPONENT);
merge lenwt (in=d) LWparams (in=p);
by svsp sex;

data sortlw; set lwdatpar;
proc sort; by svsp sex year;

data lwdata; set sortlw;
if indwt = . then delete;
if length = . then delete;
if indwt >0;
svsp1 = svsp*1;
indwtg=indwt*1000.0;
cond=indwtg/(length**3);
if EXPONENT_FALL gt 0 then predwt = (exp(COEFFICIENT_FALL))*length**EXPONENT_FALL;
if EXPONENT_FALL = . then predwt = (exp(SEASONLESS_COEFFICIENT))*length**SEASONLESS_EXPONENT;
  if EXPONENT_FALL gt 0 then predwtPK = exp(COEFFICIENT_FALL+(EXPONENT_FALL*log(length)));
if EXPONENT_FALL = . then predwtPK = exp(SEASONLESS_COEFFICIENT+(SEASONLESS_EXPONENT*log(length)));

```

```
***Relative condition;
RelWt = indwt/predwt*100;

proc sort data=lwdata;
by svsppl sex year;
run;
```

12.1.4 Plotting

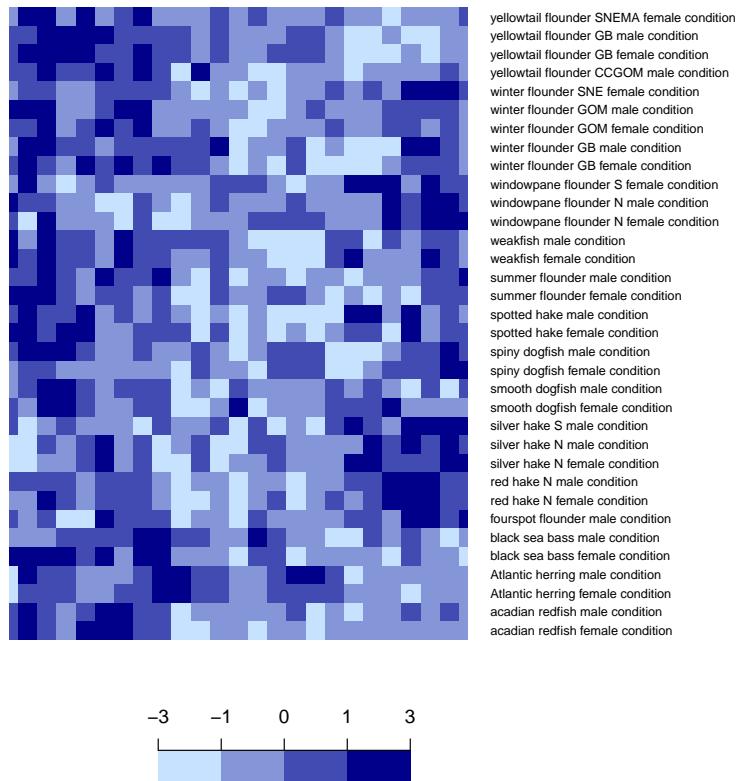


Figure 12.1: Normalized condition factors of managed species in the Northeast Large Marine Ecosystem.

Chapter 13

Ecological Production Units

Description: Ecological Production Units

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Extensive analysis, not yet published

Contributor(s): Robert Gamble

Data steward: NA

Point of contact: Robert Gamble, robert.gamble@noaa.gov

Public availability statement: Ecological production unit (EPU) shapefiles are available here. More information about source data used to derive EPUs can be found [here](#).

13.1 Methods

To define ecological production units (EPUs), we assembled a set of physiographic, oceanographic and biotic variables on the Northeast U.S. Continental Shelf, an area of approximately 264,000 km within the 200 m isobath. The physiographic and hydrographic variables selected have been extensively used in previous analyses of oceanic provinces and regions (e.g Roff and Taylor 2000). Primary production estimates have also been widely employed for this purpose in conjunction with physical variables (Longhurst 2007) to define ecological provinces throughout the world ocean.

We did not include information on zooplankton, benthic invertebrates, fish, protected species, or fishing patterns in our analysis. The biomass and production of the higher trophic level groups in this region has been sharply perturbed

Table 13.1: Variables used in derivation of Ecological Production Units.

Variables	Sampling Method	Units
Surficial Sediments	Benthic Grab	Krumbian Scale
Sea Surface Temperature	Satellite Imagery (4km grid)	°C annual average
Sea Surface Temperature	Satellite Imagery (4km grid)	dimensionless
Sea Surface Temperature	Satellite Imagery (4km grid)	°C annual average
Surface Temperature	Shipboard hydrography (point)	°C (Spring and Fall)
Bottom Temperature	Shipboard hydrography (point)	°C (Spring and Fall)
Surface Salinity	Shipboard hydrography (point)	psu (Spring and Fall)
Bottom Salinity	Shipboard hydrography (point)	psu (Spring and Fall)
Stratification	Shipboard hydrography (point)	Sigma-t units (Spring and Fall)
Chlorophyll-a	Satellite Imagery (1.25 km grid)	mg/C/m^3 (annual average)
Chlorophyll-a gradient	Satellite Imagery (1.25 km grid)	dimensionless
Chlorophyll-a span	Satellite Imagery (1.25 km grid)	mg/C/m^3 (annual average)
Primary Production	Satellite Imagery (1.25 km grid)	gC/m^3/year (cumulative)
Primary Production gradient	Satellite Imagery (1.25 km grid)	dimensionless
Primary Production span	Satellite Imagery (1.25 km grid)	gC/m^3/year (cumulative)

by fishing and other anthropogenic influences. Similarly, fishing patterns are affected by regulatory change, market and economic factors and other external influences.

Because these malleable patterns of change are often unconnected with underlying productivity, we excluded factors directly related to fishing practices. The physiographic variables considered in this analysis are listed in Table 13.1. They include bathymetry and surficial sediments. The physical oceanographic and hydrographic measurements include sea surface temperature, annual temperature span, and temperature gradient water derived from satellite observations for the period 1998 to 2007.

13.1.1 Data sources

Shipboard observations for surface and bottom water temperature and salinity in surveys conducted in spring and fall. Daily sea surface temperature (SST, °C) measurements at 4 km resolution were derived from nighttime scenes composited from the AVHRR sensor on NOAA’s polar-orbiting satellites and from NASA’s MODIS TERRA and MODIS AQUA sensors. We extracted information for the annual mean SST, temperature span, and temperature gradients from these sources. The latter metric provides information on frontal zone locations.

The biotic measurements included satellite-derived estimates of chlorophyll *a* (CHLa) mean concentration, annual span, and CHLa gradients and related

measures of primary production. Daily merged SeaWiFS/MODIS-Aqua CHLa (CHL, mg m⁻³) and SeaWiFS photosynthetically available radiation (PAR, Einsteins m⁻² d⁻¹) scenes at 1.25 km resolution were obtained from NASA Ocean Biology Processing Group.

13.1.2 Data extraction

NA

13.1.3 Data analysis

In all cases, we standardized the data to common spatial units by taking annual means of each observation type within spatial units of 10' latitude by 10' longitude to account for the disparate spatial and temporal scales at which these observations are taken. There are over 1000 spatial cells in this analysis. Shipboard sampling used to obtain direct hydrographic measurements is constrained by a minimum sampling depth of 27 m specified on the basis of prescribed safe operating procedures. As a result nearshore waters are not fully represented in our initial specifications of ecological production units.

The size of the spatial units employed further reflects a compromise between retaining spatial detail and minimizing the need for spatial interpolation of some data sets. For shipboard data sets characterized by relatively coarse spatial resolution, where necessary, we first constructed an interpolated map using an inverse distance weighting function before including it in the analysis. Although alternative interpolation schemes based on geostatistical approaches are possible, we considered the inverse distance weighting function to be both tractable and robust for this application.

We first employed a spatial principal components analysis (PCA; e.g. Pielou 1984; Legendre and Legendre 1998) to examine the multivariate structure of the data and to account for any inter-correlations among the variables to be used in subsequent analysis. The variables included in the analysis exhibited generally skewed distributions and we therefore transformed each to natural logarithms prior to analysis.

The PCA was performed on the correlation matrix of the transformed observations. We selected the eigenvectors associated with eigenvalues of the dispersion matrix with scores greater than 1.0 (the Kaiser-Guttman criterion; Legendre and Legendre 1998) for all subsequent analysis. These eigenvectors represent orthogonal linear combinations of the original variables used in the analysis.

We delineated ecological subunits by applying a disjoint cluster based on Euclidean distances using the K-means procedure (Legendre and Legendre 1998) on the principal component scores. The use of non-independent variables can

strongly influence the results of classification analyses of this type (Pielou 1984), hence the interest in using the PCA results in the cluster.

The eigenvectors were represented as standard normal deviates. We used a Pseudo-F Statistic described by Milligan and Cooper (1985) to objectively define the number of clusters to use in the analysis. The general approach employed is similar to that of Host et al. (1996) for the development of regional ecosystem classifications for terrestrial systems.

After the analyses were done, we next considered options for interpolation of nearshore boundaries resulting from depth-related constraints on shipboard observations. For this, we relied on information from satellite imagery. For the missing nearshore areas in the Gulf of Maine and Mid-Atlantic Bight, the satellite information for chlorophyll concentration and sea surface temperature indicated a direct extension from adjacent observations. For the Nantucket Shoals region south of Cape Cod, similarities in tidal mixing patterns reflected in chlorophyll and temperature observations indicated an affinity with Georges Bank and the boundaries were changed accordingly.

Finally, we next considered consolidation of ecological subareas so that nearshore regions are considered to be special zones nested within the adjacent shelf regions. Similar consideration led to nesting the continental slope regions within adjacent shelf regions in the Mid-Atlantic and Georges Bank regions. This led to four major units: Mid-Atlantic Bight, Georges Bank, Western-Central Gulf of Maine (simply “Gulf of Maine” in the State of the Ecosystem), and Scotian Shelf-Eastern Gulf of Maine. As the State of the Ecosystem reports are specific to FMC managed regions, the Scotian Shelf-Eastern Gulf of Maine EPU is not considered in SOE indicator analyses.

13.1.4 Data processing

Shapefiles were converted to `sf` objects for inclusion in the `ecodata` R package using the following R code.

```
# A function to process and make ecological production unit
# (EPU) shapefiles available in the ecodata package.

# Read more about the delineation of EPUs at
# https://noaa-edab.github.io/tech-doc/epu.html

library(sf)
library(rgdal)
library(raster)
library(rnaturalearth)
```

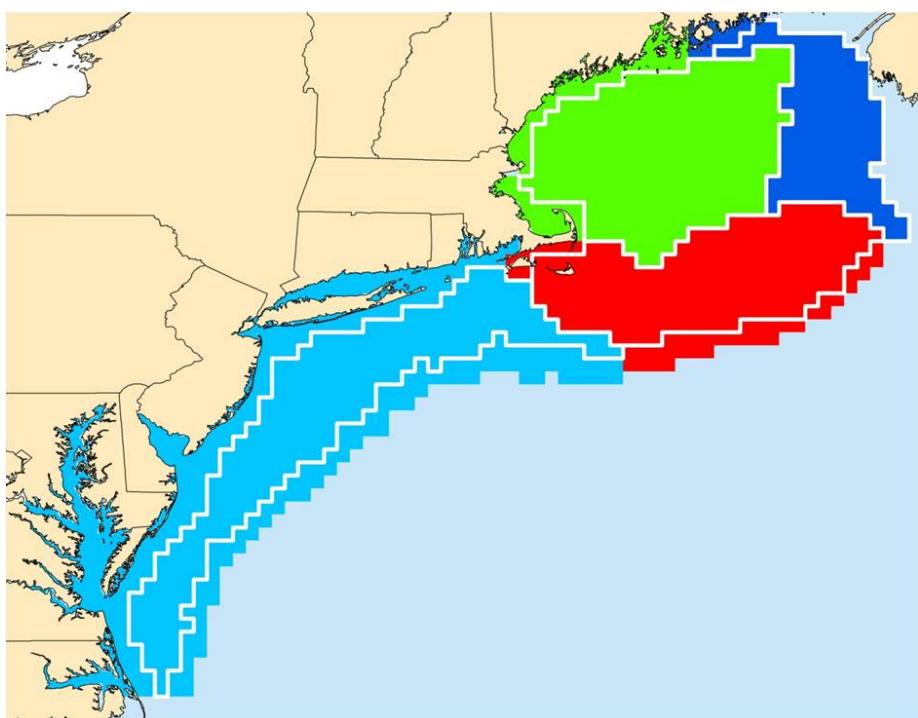


Figure 13.1: Map of the four Ecological Production Units, including the Mid-Atlantic Bight (light blue), Georges Bank (red), Western-Central Gulf of Maine (or Gulf of Maine; green), and Scotian Shelf-Eastern Gulf of Maine (dark blue)

```
gis.dir <- here::here("inst", "extdata", "gis")
crs <- "+proj=longlat +lat_1=35 +lat_2=45 +lat_0=40 +lon_0=-77+x_0=0
+y_0=0 +datum=NAD83 +no_defs +ellps=GRS80 +towgs84=0,0,0"

get_epu_sf <- function(save_clean) {

  epu_shp <- readOGR(file.path(gis.dir, "EPU_Extended.shp"),
    verbose = F)
  crs(epu_shp) <- crs
  epu_sf <- as(epu_shp, "sf")

  if (save_clean) {
    usethis::use_data(epu_sf)
  } else {
    return(epu_sf)
  }
}
```

Chapter 14

Gulf Stream Index

Description: Annual time series of the Gulf Stream index

Indicator category: Published method

Found in: State of the Ecosystem - New England (2019)

Contributor(s): Terry Joyce, Rong Zhang

Data steward: Vincent Saba, vincent.saba@noaa.gov

Point of contact: Vincent Saba, vincent.saba@noaa.gov

Public availability statement: Source data are publicly available

14.1 Methods

Summarized from Joyce et al. (2019), ocean temperature data from NOAA's National Centers for Environmental Information (NCEI) were sorted by latitude, longitude, and time using a resolution of 1° of longitude, latitude, and 3 months of time, respectively, with a Gaussian squared weighting from the selected desired point in a window twice the size of the desired resolution. Editing was used to reject duplicate samples and 3σ outliers from each selected sample point prior to performing the weighting and averaging; the latter was only carried out when there were at least three data points in the selected interval for each sample point. Typically, 50 or more data values were available. The resulting temperature field was therefore smoothed. Data along the Gulf Stream north wall at nine data points were used to assemble a spatial/temporal sampling of the temperature at 200m data along the north wall from 75°W to 55°W. The leading mode of temperature variability of the Gulf Stream is equivalent to a north-south shift of 50–100 km, which is zonally of one sign and amounts to

50% of the seasonal-interannual variance between 75°W and 55°W. The temporal behavior of this mode (PC1) shows the temporal shift of the Gulf Stream path with a dominant approximately 8- to 10-year periodicity over much of the period.

14.1.1 Data sources

Ocean temperatures at 200 m are available at https://www.nodc.noaa.gov/OC5/3M_HEAT_CONTENT/.

14.1.2 Data analysis

For detailed analytical methods, see Joyce et al. (2019).

14.1.3 Data Processing

The Gulf Stream index data set was formatted for inclusion in the `ecodata` R package with the following code.

```
# Processing for Gulf Stream Index data

# GSI = degrees latitude above the average Gulf Stream
# position based on ocean temperature at 200m (15 C) depth
# between 55W to 75W.

library(dplyr)
library(tidyr)
library(lubridate)

raw.dir <- here::here("data-raw")

get_gsi <- function(save_clean = F) {

  gsi <- read.csv(file.path(raw.dir, "GSI.csv")) %>% dplyr::rename(Time = Month,
    Value = GSI) %>% mutate(Var = "gulf stream index", Units = "latitude anomaly",
    EPU = "All")

  if (save_clean) {
    usethis::use_data(gsi, overwrite = T)
  } else {
    return(gsi)
  }
}
```

```
}
```

```
get_gsi(save_clean = T)
```


Chapter 15

Harbor Porpoise Bycatch

Description: Harbor Porpoise Indicator

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Synthesis of published information; Published methods

Contributor(s): Christopher D. Orphandies

Data steward: Chris Orphanides, chris.orphanides@noaa.gov

Point of contact: Chris Orphanides, chris.orphanides@noaa.gov

Public availability statement: Source data are available in public stock assessment reports (2018 report in-press). Derived data as shown in the 2018 SOE reports are available here

15.1 Methods

15.1.1 Data sources

Reported harbor porpoise bycatch estimates and potential biological removal levels can be found in publicly available documents; detailed here. The most recent bycatch estimates for 2016 were taken from the 2018 stock assessment (in-press). More detailed documentation as to the methods employed can be found in NOAA Fisheries Northeast Fisheries Science Center (NEFSC) Center Reference Documents (CRDs) found on the NEFSC publications page.

The document for the 2016 estimates (CRD 19-04) is available [here](#). Additional methodological details are available for previous year's estimates and are documented in numerous published CRDs: CRD 17-18, CRD-16-05, CRD 15-15,

CRD 14-02, CRD 13-13, CRD 11-08, CRD 10-10, CRD 07-20, CRD 06-13, CRD 03-18, CRD 01-15, and CRD 99-17.

15.1.2 Data extraction

Annual gillnet bycatch estimates are documented in a CRD (see sources above). These feed into the Stock Assessment Reports which report both the annual bycatch estimate and the mean 5-year estimate. The 5-year estimate is the one used for management purposes, so that is the one provided for the SOE plot.

15.1.3 Data analysis

Bycatch estimates as found in stock assessment reports were plotted along with confidence intervals. The confidence intervals were calculated from published CVs assuming a normal distribution ($\sigma = \mu CV$; $CI = \bar{x} \pm \sigma * 1.96$).

Data were analyzed and formatted for inclusion in the `ecodata` R package using the following R code.

```
# Process harbor porpoise bycatch estimates

# Time series figure is 5-yr running mean for harbor porpoise
# bycatch estimates for the Northeast US across all
# fisheries.

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

# HP bycatch time series
# estimates-----
get_harborporpoise <- function(save_clean = F) {
  d <- read.csv(file.path(raw.dir, "1994-2017_5yr_hp_est.csv"))

  # Create confidence intervals
  var1nnum <- log(1 + d$CV^2)
  c <- exp(1.96 * sqrt(var1nnum))
  d$up95ci <- d$EST * c
  d$lo95ci <- d$EST/c

  harborporpoise <- d %>% dplyr::rename(Time = YEAR) %>% gather(.,
```

Var, Value, -Time) %>% mutate(Units = "N", EPU = "All",

```

Var, Var = plyr::mapvalues(Var, from = c("EST", "CV",
  "PBR", "up95ci", "lo95ci"), to = c("harbor porpoise bycatch estimate",
  "harbor porpoise bycatch cv", "harbor porpoise bycatch pbr",
  "harbor porpoise bycatch up95ci", "harbor porpoise bycatch lo95ci")) %>%
  as.data.frame()

if (save_clean) {
  usethis::use_data(harborporpoise, overwrite = R)
} else {
  return(harborporpoise)
}
}

```

15.1.4 Plotting

```

# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mar = c(4, 6, 2, 6))

soe.plot(SOE.data, "Time", "Harbor porpoise bycatch estimates",
  rel.y.num = 1.2, end.start = 2007, full.trend = F, point.cex = 1,
  ymax = F, y.upper = 2500, mean_line = F, x.label = "Year",
  y.label = "Bycatch, n", rel.y.text = 1)

legend(2000, 2250, legend = "Potential Biological Removal", col = adjustcolor("red",
  0.5), lwd = 3, bty = "n", cex = 0.9)

# credible intervals and PBI
lw_CI <- SOE.data[Var == "Harbor porpoise bycatch 2.5 CI", list(Time,
  Value)]
up_CI <- SOE.data[Var == "Harbor porpoise bycatch 97.5 CI", list(Time,
  Value)]
pbi <- SOE.data[Var == "Harbor porpoise potential biological removal",
  list(Time, Value)]

```

```
points(pbi, type = "l", lty = 1, col = adjustcolor("red", 0.5),
       lwd = 3)
points(lw_CI, type = "l", lty = 2, col = adjustcolor("darkorange",
          0.9), lwd = 2.5)
points(up_CI, type = "l", lty = 2, col = adjustcolor("darkorange",
          0.9), lwd = 2.5)
```

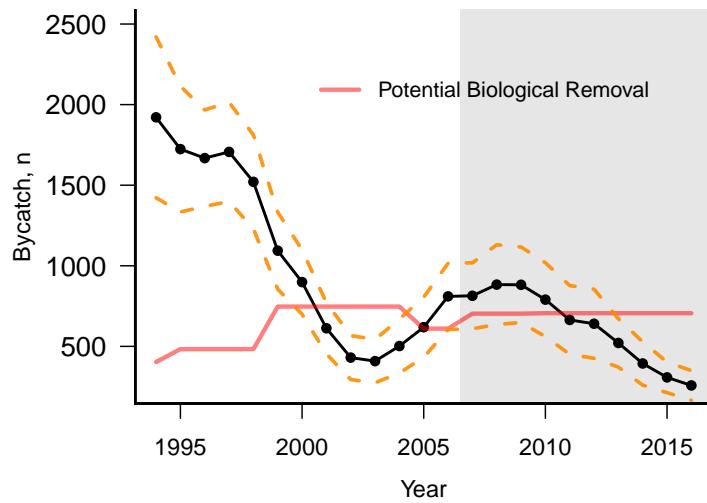


Figure 15.1: Harbor porpoise bycatch estimated shown with Potential Biological Removal (red) and confidence intervals (orange).

Chapter 16

Ichthyoplankton Diversity

Description: NOAA NEFSC Oceans and Climate branch public ichthyoplankton dataset

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Database pull with analysis

Contributor(s): Harvey J. Walsh

Data steward: Harvey Walsh, harvey.walsh@noaa.gov

Point of contact: Harvey Walsh, harvey.walsh@noaa.gov

Public availability statement: Source data are available to the public here. Derived data for this indicator are available here.

16.1 Methods

Data from the NOAA Northeast Fisheries Science Center (NEFSC) Oceans and Climate branch (OCB) public dataset were used to examine changes in diversity of abundance among 45 ichthyoplankton taxa. The 45 taxa were established (Walsh et al. 2015), and include the most abundant taxa from the 1970s to present that represent consistency in the identification of larvae.

16.1.1 Data sources

Multi-species plankton surveys cover the entire Northeast US shelf from Cape Hatteras, North Carolina, to Cape Sable, Nova Scotia, four to six times per year. A random-stratified design based on the NEFSC bottom trawl survey

design (Azarovitz 1981) is used to collect samples from 47 strata. The number of strata is lower than the trawl survey as many of the narrow inshore and shelf-break strata are combined in the EcoMon design. The area encompassed by each stratum determined the number of samples in each stratum. Samples were collected both day and night using a 61 cm bongo net. Net tow speed was 1.5 knots and maximum sample depth was 200 m. Double oblique tows were a minimum of 5 minutes in duration, and fished from the surface to within 5 m of the seabed or to a maximum depth of 200 m. The volume filtered of all collections was measured with mechanical flowmeters mounted across the mouth of each net.

Processing of most samples was conducted at the Morski Instytut Rybacki (MIR) in Szczecin, Poland; the remaining samples were processed at the NEFSC or the Atlantic Reference Center, St Andrews, Canada. Larvae were identified to the lowest possible taxa and enumerated for each sample. Taxon abundance for each station was standardized to number under 10 m^{-2} sea surface.

16.1.2 Data extraction

Data retrieved from NOAA NEFSC Oceans and Climate branch public dataset (Filename: “EcoMon_Plankton_Data_v3_0.xlsx”, File Date: 10/20/2016).

16.1.3 Data analysis

All detailed data processing steps are not currently included in this document, but general steps are outlined. Data were grouped into seasons: spring = February, March, April and fall = September, October, November. Stratified weighted mean abundance was calculated for each taxon for each year and season across all plankton strata ($n = 47$) for 17 years (1999 to 2015). Shannon Diversity Index and count of positive taxon was calculated for each season and year.

MATLAB code used to calculate diversity indices:

```
%%%%%%
% Calculates Shannon Diversity Index of Ichthyoplankton data
%
% Input: excel file of ichthyoplankton data
%
% USES: index_SaW.m
%
% last modified: 03August2018, HJW
%%%%%
%% Data retrieved from NOAA NEFSC Oceans and Climate branch public dataset
% ftp://ftp.nefsc.noaa.gov/pub/hydro/zooplankton_data/
% Filename: EcoMon_Plankton_Data_v3_0.xlsx
```

```
% File Date: 10/20/2016

% Data processing - not included in this file
% Data grouped into seasons: spring = Feb to Apr, fall = Sept to Nov
% Stratified weighted mean abundance was calculated for each taxon for each year and season
% Abundance across all plankton strata (n = 47) for 17 years (1999 to 2015)

%% Import aggregated data from spreadsheet
% Script for importing data from the following spreadsheet:
%
% Workbook: /Users/hwalsh/NEFSC larval samples/CombinedData/SOE_Diversity/NEFSCIchthyoplanktonAbundance.xlsx
% Worksheet: Sheet1
%% Output Data
% SprSW = Spring Shannon Diversity Index
% SprCount = Spring count of positive ichthyoplankton taxa (Max = 45)
% FallSW = Fall Shannon Diversity Index
% FallCount = Fall count of positive ichthyoplankton taxa (Max = 45)

%% Import the data
[~, ~, raw] = xlsread('/Users/hwalsh/NEFSC larval samples/CombinedData/SOE_Diversity/NEFSCIchthyoplanktonAbundance.xlsx');
raw = raw(2:end,7:end);

%% Create output variable
data = reshape([raw{:}],size(raw));

%% Create table
NEFSCIchthyoplanktonAbundance = table;

%% Allocate imported array to column variable names
NEFSCIchthyoplanktonAbundance.Brevoortiatyrrannus = data(:,1);
NEFSCIchthyoplanktonAbundance.Clupeaharengus = data(:,2);
NEFSCIchthyoplanktonAbundance.Cyclothonesp = data(:,3);
NEFSCIchthyoplanktonAbundance.Diaphusspp = data(:,4);
NEFSCIchthyoplanktonAbundance.Ceratoscopelusmaderensis = data(:,5);
NEFSCIchthyoplanktonAbundance.Benthosemaspp = data(:,6);
NEFSCIchthyoplanktonAbundance.Urophycisspp = data(:,7);
NEFSCIchthyoplanktonAbundance.Enchelyopuscimbrius = data(:,8);
NEFSCIchthyoplanktonAbundance.Gadusmorhua = data(:,9);
NEFSCIchthyoplanktonAbundance.Melanogrammusaeaglefinus = data(:,10);
NEFSCIchthyoplanktonAbundance.Pollachiusvirens = data(:,11);
NEFSCIchthyoplanktonAbundance.Merlucciusalbidus = data(:,12);
NEFSCIchthyoplanktonAbundance.Merlucciusbilinearis = data(:,13);
NEFSCIchthyoplanktonAbundance.Centropristesstriata = data(:,14);
NEFSCIchthyoplanktonAbundance.Pomatomussaltatrix = data(:,15);
NEFSCIchthyoplanktonAbundance.Cynoscionregalis = data(:,16);
```

```

NEFSCIchthyoplanktonAbundance.Leiostomusxanthurus = data(:,17);
NEFSCIchthyoplanktonAbundance.Menticirrhuspp = data(:,18);
NEFSCIchthyoplanktonAbundance.Micropogoniasundulatus = data(:,19);
NEFSCIchthyoplanktonAbundance.Tautogolabrusadspersus = data(:,20);
NEFSCIchthyoplanktonAbundance.Tautogaonitis = data(:,21);
NEFSCIchthyoplanktonAbundance.Auxisspp = data(:,22);
NEFSCIchthyoplanktonAbundance.Scomberscombrus = data(:,23);
NEFSCIchthyoplanktonAbundance.Peprilusspp = data(:,24);
NEFSCIchthyoplanktonAbundance.Sebastesspp = data(:,25);
NEFSCIchthyoplanktonAbundance.Prionotusspp = data(:,26);
NEFSCIchthyoplanktonAbundance.Myoxocephalusenaeus = data(:,27);
NEFSCIchthyoplanktonAbundance.Myoxocephalusoctodecemspinosus = data(:,28);
NEFSCIchthyoplanktonAbundance.Ammodytesspp = data(:,29);
NEFSCIchthyoplanktonAbundance.Pholisgunnellus = data(:,30);
NEFSCIchthyoplanktonAbundance.Ulvariasubbifurcata = data(:,31);
NEFSCIchthyoplanktonAbundance.Anarhichasspp = data(:,32);
NEFSCIchthyoplanktonAbundance.Citharichthysarctifrons = data(:,33);
NEFSCIchthyoplanktonAbundance.Etropusspp = data(:,34);
NEFSCIchthyoplanktonAbundance.Syaciumspp = data(:,35);
NEFSCIchthyoplanktonAbundance.Bothusspp = data(:,36);
NEFSCIchthyoplanktonAbundance.Hippoglossinaoblonga = data(:,37);
NEFSCIchthyoplanktonAbundance.Paralichthysdentatus = data(:,38);
NEFSCIchthyoplanktonAbundance.Pseudopleuronectesamericanus = data(:,39);
NEFSCIchthyoplanktonAbundance.Hippoglossoidesplatessoides = data(:,40);
NEFSCIchthyoplanktonAbundance.Limandaferuginea = data(:,41);
NEFSCIchthyoplanktonAbundance.Glyptocephaluscynoglossus = data(:,42);
NEFSCIchthyoplanktonAbundance.Scophtalmusaquosus = data(:,43);
NEFSCIchthyoplanktonAbundance.Symphurusspp = data(:,44);
NEFSCIchthyoplanktonAbundance.Lophiusamericanus = data(:,45);

%% Clear temporary variables
clearvars data raw;
%% Separate Spring (Spr) and Fall data
Spr=table2array(NEFSCIchthyoplanktonAbundance(1:17,:))';
Fall=table2array(NEFSCIchthyoplanktonAbundance(18:34,:))';
%% Shannon-Wiener index
[SprSW]=index_SaW(Spr,exp(1));
[FallSW]=index_SaW(Fall,exp(1));
%% Count of number taxa per year
SprCount=zeros(1,length(SprSW));
for ii=1:length(Spr)
    for yy=1:length(SprCount)
        if Spr(ii,yy)>0
            SprCount(1,yy)=SprCount(1,yy)+1;
        end
    end
end

```

```

    end
end
FallCount=zeros(1,length(FallSW));
for ii=1:length(Fall)
    for yy=1:length(FallCount)
        if Fall(ii,yy)>0
            FallCount(1,yy)=FallCount(1,yy)+1;
        end
    end
end
clear ii yy

```

16.1.4 Data processing

Ichthyoplankton diversity data sets were formatted for inclusion in the `ecodata` R package using the following R code.

```

# Ichthyoplankton diversity

library(dplyr)
library(tidyr)
library(readxl)
library(stringr)

raw.dir <- here::here("data-raw")

get_ichthyo_diversity <- function(save_clean = F) {

    ichthyo_diversity <- read_excel(file.path(raw.dir, "NEFSCIchthyoplankton_v3_6.xlsx")) %>%
        dplyr::select(-Source) %>% dplyr::rename(Time = Year,
        EPU = Region) %>% mutate(Value = as.numeric(Value))

    if (save_clean) {
        usethis::use_data(ichthyo_diversity, overwrite = T)
    } else {
        return(ichthyo_diversity)
    }
}
get_ichthyo_diversity(save_clean = T)

```

16.1.5 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mfrow = c(2, 1), mar = c(0, 0, 0, 0), oma = c(3.5,
  5, 2, 4))

soe.plot(SOE.data, "Time", "Spring_Ich_Shannon Diversity Index",
  stacked = "A", rel.y.num = 1.1, end.start = 2007, full.trend = F,
  cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Fall_Ich_Shannon Diversity Index",
  stacked = "B", rel.y.num = 1.1, end.start = 2007, full.trend = F,
  cex.stack = 1.5)

soe.stacked.axis("Year", "Shannon Index", y.line = 2.5)
```

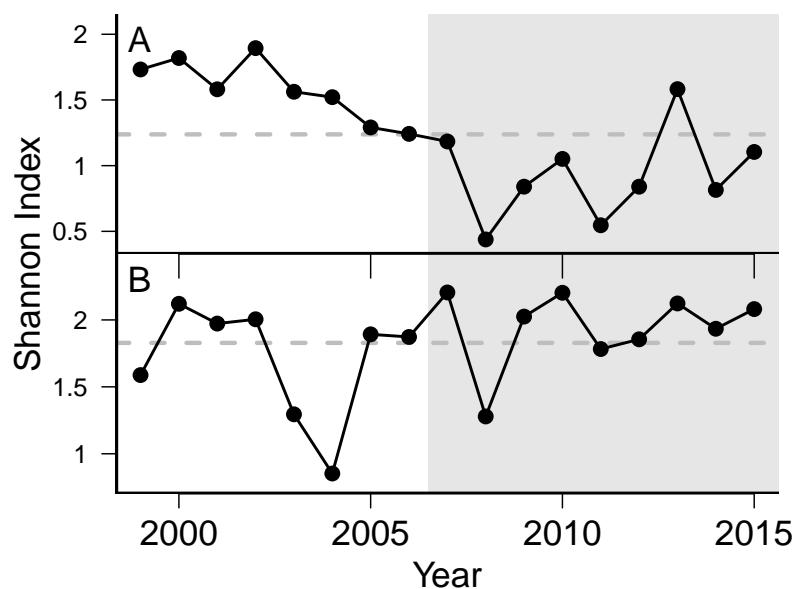


Figure 16.1: Ichthyoplankton Shannon diversity in the spring (A) and fall (B) in the Northeast Large Marine Ecosystem.

Chapter 17

Commercial Landings Data

Description: Commercial landings data pull

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull

Contributor(s): Sean Lucey

Data steward: Sean Lucey, Sean.Lucey@noaa.gov

Point of contact: Sean Lucey, Sean.Lucey@noaa.gov

Public availability statement: Raw data are not publically available due to confidentiality of individual fishery participants. Derived indicator outputs are available here.

17.1 Methods

Fisheries dependent data for the Northeast Shelf extend back several decades. Data from the 1960s on are housed in the Commercial database (CFDBS) of the Northeast Fisheries Science Center which contains the commercial fisheries dealer purchase records (weigh-outs) collected by National Marine Fisheries Service (NMFS) Statistical Reporting Specialists and state agencies from Maine to Virginia. The data format has changed slightly over the time series with three distinct time frames as noted in Table 17.1 below.

Comlands is an R database pull that consolidates the landings records from 1964 on and attempts to associate them with NAFO statistical areas (Figure 17.1). The script is divided into three sections. The first pulls domestic landings data from the yearly landings tables and merges them into a single data source. The

Table 17.1: Data formats

Table	Years
WOLANDS	1964 - 1981
WODETS	1982 - 1993
CFDETS_AA	> 1994

second section applies an algorithm to associate landings that are not allocated to a statistical area using similar characteristics of the trip to trips with known areas. The final section pulls foreign landings from the Northwest Atlantic Fisheries Organization website and rectifies species and gear codes so they can be merged along with domestic landings.

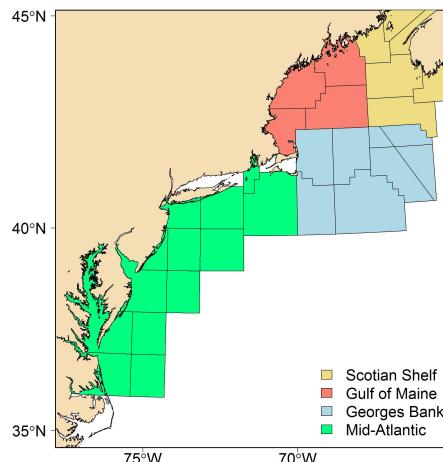


Figure 17.1: Map of the North Atlantic Fisheries Organization (NAFO) Statistical Areas. Colors represent the Ecological Production Unit (EPU) with which the statistical area is associated.

During the first section, the Comlands script pulls the temporal and spatial information as well as vessel and gear characteristics associated with the landings in addition to the weight, value, and utilization code of each species in the landings record. The script includes a toggle to use landed weights as opposed to live weights. For all but shellfish species, live weights are used for the State of the Ecosystem report. Due to the volume of data contained within each yearly landings table, landings are aggregated by species, utilization code, and area as well as by month, gear, and tonnage class. All weights are then converted from pounds to metric tons. Landings values are also adjusted for inflation using the Producer Price Index by Commodity for Processed Foods and Feeds: Unprocessed and Packaged Fish. Inflation is based on January of the terminal year of

Table 17.2: Gear types used in commercial landings

Major gear	
1	Otter Trawls
2	Scallop Dredges
3	Other Dredges
4	Gillnets
5	Longlines
6	Seines
7	Pots/Traps
8	Midwater
9	Other

the data pull ensuring that all values are in current dollar prices.

Several species have additional steps after the data is pulled from CFDBS. Skates are typically landed as a species complex. In order to segregate the catch into species, the ratio of individual skate species in the NEFSC bottom trawl survey is used to disaggregate the landings. A similar algorithm is used to separate silver and offshore hake which can be mistaken for one another. Finally, Atlantic herring landings are pulled from a separate database as the most accurate weights are housed by the State of Maine. Comlands pulls from the State database and replaces the less accurate numbers from the federal database.

The majority of landings data are associated with a NAFO Statistical Area. For those that are not, Comlands attempts to assign them to an area using similar characteristics of trips where the area is known. To simplify this task, landings data are further aggregated into quarter and half year, small and large vessels, and eight major gear categories (Table 17.2). Landings are then proportioned to areas that meet similar characteristics based on the proportion of landings in each area by that temporal/vessel/gear combination. If a given attribute is unknown, the algorithm attempts to assign it one, once again based on matched characteristics of known trips. Statistical areas are then assigned to their respective Ecological Production Unit (Table 17.3).

Table 17.3: Statistical areas making up each EPU

EPU	Stat Areas
Gulf of Maine	500, 510, 512, 513, 514, 515
Georges Bank	521, 522, 523, 524, 525, 526, 551, 552, 561, 562
Mid-Atlantic	537, 539, 600, 612, 613, 614, 615, 616, 621, 622, 625, 626, 631, 632

The final step of Comlands is to pull the foreign landings from the NAFO database. US landings are removed from this extraction so as not to be double counted. NAFO codes and CFDBS codes differ so the script rectifies those codes to ensure that the data is seamlessly merged into the domestic landings. Foreign landings are flagged so that they can be removed if so desired.

17.1.1 Data sources

Comland is a database query of the NEFSC commercial fishery database (CFDBS). More information about the CFDBS is available here.

17.1.2 Data extraction

R code used in the extraction process described above:

```
# Comland.r Version now controlled by git - originally part
# of comcatch.r Grab commercial landings data from US and
# Foreign countries (NAFO) Need to fix menhaden data SML

# Requires the following files:
# data.dir.2\\Comland_skates_hakes.R
# data.dir\\Menhaden.csv data.dir.3\\SS_NAFO_21A.csv
# data.dir.3\\species.txt

# User parameters
if (Sys.info()["sysname"] == "Windows") {
  data.dir <- "L:\\EcoAP\\Data\\Commercial"
  data.dir.2 <- "L:\\Rworkspace\\RCom"
  data.dir.3 <- "L:\\EcoAP\\Data\\NAFO"
  out.dir <- "L:\\EcoAP\\Data\\Commercial"
  memory.limit(4000)
  channel <- odbcDriverConnect()
}

if (Sys.info()["sysname"] == "Linux") {
  data.dir <- "/home/slucey/slucey/EcoAP/Data/Commercial"
  data.dir.2 <- "/home/slucey/slucey/Rworkspace/RCom"
  data.dir.3 <- "/home/slucey/slucey/EcoAP/Data/NAFO"
  out.dir <- "/home/slucey/slucey/EcoAP/Data/Commercial"
  uid <- "slucey"
  cat("Oracle Password: ")
  pwd <- scan(stdin(), character(), n = 1)
}
```

```

landed <- "y" #use landed weight for scallops and clams instead of live weight
foreign <- "y" #Mark foreign landings and keep separate
adjust.ppi <- "y" #Adjust value for inflation
use.existing <- "n" #use raw data from a previous run - saves time
sum.by <- "EPU" #Variable to sum landings by [EPU, stat.area]

# Final year of query
endyear <- 2016
# If adjusting for inflation
refyear <- 2016
refmonth <- 1

#-----
# Required packages
library(RODBC)
library(data.table)
library(rgdal)

#-----
# User created functions Convert NA's to zeros
na.zero <- function(x) {
  for (i in 1:length(x[, ])) {
    if (length(which(is.na(x[, i]))) > 0) {
      x[which(is.na(x[, i])), i] <- 0
    }
  }
  return(x)
}

#-----
# Connect to database
if (Sys.info()["sysname"] == "Windows") channel <- odbcDriverConnect()
if (Sys.info()["sysname"] == "Linux") channel <- odbcConnect("sole",
  uid, pwd)

if (use.existing == "n") {
  # Landings
  tables <- c(paste0("WOLANDS", 64:81), paste0("WODETS", 82:93),
    paste0("CFDETS", 1994:endyear, "AA"))

  # Generate one table
  comland <- c()
  for (i in 1:length(tables)) {
    landings.qry <- paste("select year, month, negear, toncl1, nespp3, nespp4, area,
      spplivlb, spplndlb, sppvalue, utilcd
      from ", tables[i], " where year >= 1994 and year <=", endyear, " and month >= 1 and month <= 12 and
      negear > 0 and toncl1 > 0 and nespp3 > 0 and nespp4 > 0 and spplivlb > 0 and spplndlb > 0 and sppvalue > 0 and utilcd > 0")
    comland <- c(comland, landings.qry)
  }
}

```

```

        from",
tables[i])

comland.yr <- as.data.table(sqlQuery(channel, landings.qry))

setkey(comland.yr, YEAR, MONTH, NEGEAR, TONCL1, NESPP3,
       NESPP4, AREA, UTILCD)

if (landed == "y")
  comland.yr[NESPP3 %in% 743:800, `:=` (SPPLIVLB, SPPLNDLB)]

# Sum landings and value landings
comland.yr[, `:=` (V1, sum(SPPLIVLB)), by = key(comland.yr)]
# value Fix null values
comland.yr[is.na(SPPVALUE), `:=` (SPPVALUE, 0)]
comland.yr[, `:=` (V2, sum(SPPVALUE)), by = key(comland.yr)]

# Remove extra rows/columns
comland.yr <- unique(comland.yr, by = key(comland.yr))
comland.yr[, `:=` (c("SPPLIVLB", "SPPLNDLB", "SPPVALUE"),
                  NULL)]

# Rename summed columns
setnames(comland.yr, c("V1", "V2"), c("SPPLIVLB", "SPPVALUE"))

comland <- rbindlist(list(comland, comland.yr))
}

if (landed == "n")
  save(comland, file = file.path(out.dir, "comland_raw_US.RData"))
# Last run 8/31/16
if (landed == "y")
  save(comland, file = file.path(out.dir, "comland_raw_US_meatwt.RData"))
# Last run 1/25/18
}
if (use.existing == "y") {
  if (landed == "n")
    load(file = file.path(out.dir, "comland_raw_US.RData"))
  if (landed == "y")
    load(file = file.path(out.dir, "comland_raw_US_meatwt.RData"))
}

#-----
# Convert from lbs to metric tons
comland[, `:=` (SPPLIVMT, SPPLIVLB * 0.00045359237)]

```

```

comland[, `:=` (SPPLIVLB, NULL)]

# fix years
comland[YEAR < 100, `:=` (YEAR, YEAR + 1900L)]

if (adjust.ppi == "y") {
  # Adjust SPPVALUE for inflation
  temp <- tempfile()
  download.file("http://download.bls.gov/pub/time.series/wp/wp.data.3.ProcessedFoods",
    temp)
  inflate <- as.data.table(read.delim(temp))
  unlink(temp)

  inflate[, `:=` (series_id, gsub(" ", "", inflate[, series_id]))]
  deflate <- inflate[series_id == "WPU0223", ]
  deflate[, `:=` (MONTH, as.numeric(substr(period, 2, 3)))]
  setnames(deflate, c("year", "value"), c("YEAR", "PPI"))
  deflate <- deflate[, list(YEAR, MONTH, PPI)]

  # Set yearly deflator to 0 instead of 13 to match unknown
  # month designation
  deflate[MONTH == 13, `:=` (MONTH, 0)]
  deflate.base <- deflate[YEAR == refyear & MONTH == refmonth,
    PPI]

  comland <- merge(comland, deflate, by = c("YEAR", "MONTH"),
    all.x = T)
  comland[, `:=` (SPPVALUE, round((SPPVALUE * deflate.base)/PPI))]

  # Remove extra column
  comland[, `:=` (PPI, NULL)]
}

# Remove market categories of parts
comland <- comland[!NESPP4 %in% c(119, 123, 125, 127, 812, 819,
  828, 829, 1731, 2351, 2690, 2699, 3472, as.numeric(paste(348:359,
  8, sep = "")), 3868, as.numeric(paste(469:471, 4, sep = "")),
  as.numeric(paste(480:499, 8, sep = "")), 5018, 5039, 5261,
  5265), ]

# Generate NESPP3 and MKTCAT in comland data
comland[NESPP4 < 100, `:=` (MKTCAT, as.numeric(substring(NESPP4,
  2, 2)))]
comland[NESPP4 > 99 & NESPP4 < 1000, `:=` (MKTCAT, as.numeric(substring(NESPP4,
  3, 3)))]
comland[NESPP4 > 999, `:=` (MKTCAT, as.numeric(substring(NESPP4,

```

```

4, 4)))]]

# drop NESPP4
comland[, `:=` (NESPP4, NULL)]

# Deal with Hakes and
# Skates-----
source(file.path(data.dir.2, "Comland_skates_hakes.R"))

# get little skates and winter skates from skates(ns) - use
# survey in half years Generate Half year variable in comland
comland.skates <- comland[NESPP3 == 365, ]
comland.skates[MONTH %in% 1:6, `:=` (Half, 1)]
comland.skates[MONTH %in% 7:12, `:=` (Half, 2)]

setkey(skate.hake.us, YEAR, Half, AREA)

comland.skates <- merge(comland.skates, skate.hake.us, by = key(skate.hake.us),
all.x = T)

comland.skates[, `:=` (little, little.per * SPPLIVMT)]
comland.skates[, `:=` (little.value, round(little.per * SPPVALUE))]
comland.skates[is.na(little), `:=` (little, 0)]
comland.skates[is.na(little.value), `:=` (little.value, 0)]

comland.skates[, `:=` (winter, winter.per * SPPLIVMT)]
comland.skates[, `:=` (winter.value, round(winter.per * SPPVALUE))]
comland.skates[is.na(winter), `:=` (winter, 0)]
comland.skates[is.na(winter.value), `:=` (winter.value, 0)]

comland.skates[, `:=` (other.skate, SPPLIVMT - (little + winter))]
comland.skates[, `:=` (other.skate.value, SPPVALUE - (little.value +
winter.value))]

# Little (366), winter (367), skates(ns) (365) put skates in
# comland format to merge back
little <- comland.skates[, list(YEAR, Half, AREA, MONTH, NEGEAR,
TONCL1, NESPP3, UTILCD, MKTCAT, little, little.value)]
little[, `:=` (NESPP3, 366)]
setnames(little, c("little", "little.value"), c("SPPLIVMT", "SPPVALUE"))
little <- little[SPPLIVMT > 0, ]

winter <- comland.skates[, list(YEAR, Half, AREA, MONTH, NEGEAR,
TONCL1, NESPP3, UTILCD, MKTCAT, winter, winter.value)]
winter[, `:=` (NESPP3, 367)]

```

```

setnames(winter, c("winter", "winter.value"), c("SPPLIVMT", "SPPVALUE"))
winter <- winter[SPPLIVMT > 0, ]

other <- comland.skates[, list(YEAR, Half, AREA, MONTH, NEGEAR,
    TONCL1, NESPP3, UTILCD, MKTCAT, other.skate, other.skate.value)]
other[, `:=` (NESPP3, 365)]
setnames(other, c("other.skate", "other.skate.value"), c("SPPLIVMT",
    "SPPVALUE"))
other <- other[SPPLIVMT > 0, ]

# merge all three and reformat for comland
skates.add.back <- rbindlist(list(little, winter, other))

skates.add.back[, `:=`(Half, NULL)]
setcolorder(skates.add.back, names(comland))

comland <- rbindlist(list(comland[NESPP3 != 365, ], skates.add.back))

# get silver hake from mixed hakes - use survey in half years
# Generate Half year variable in comland
comland.hakes <- comland[NESPP3 == 507, ]
comland.hakes[MONTH %in% 1:6, `:=`(Half, 1)]
comland.hakes[MONTH %in% 7:12, `:=`(Half, 2)]

comland.hakes <- merge(comland.hakes, skate.hake.us, by = key(skate.hake.us),
    all.x = T)

comland.hakes[, `:=`(silver, silver.per * SPPLIVMT)]
comland.hakes[, `:=`(silver.value, round(silver.per * SPPVALUE))]
comland.hakes[is.na(silver), `:=`(silver, 0)]
comland.hakes[is.na(silver.value), `:=`(silver.value, 0)]

comland.hakes[, `:=`(off.hake, SPPLIVMT - silver)]
comland.hakes[, `:=`(off.hake.value, SPPVALUE - silver.value)]

# Silver hake (509), mix hakes (507) put hakes in comland
# format to merge back
silver <- comland.hakes[, list(YEAR, Half, AREA, MONTH, NEGEAR,
    TONCL1, NESPP3, UTILCD, MKTCAT, silver, silver.value)]
silver[, `:=`(NESPP3, 509)]
setnames(silver, c("silver", "silver.value"), c("SPPLIVMT", "SPPVALUE"))
silver <- silver[SPPLIVMT > 0, ]

offshore <- comland.hakes[, list(YEAR, Half, AREA, MONTH, NEGEAR,
    TONCL1, NESPP3, UTILCD, MKTCAT, off.hake, off.hake.value)]

```

```

offshore[, `:=` (NESPP3, 507)]
setnames(offshore, c("off.hake", "off.hake.value"), c("SPPLIVMT",
  "SPPVALUE"))
offshore <- offshore[SPPLIVMT > 0, ]

# merge both and reformat for comland
hakes.add.back <- rbindlist(list(silver, offshore))

hakes.add.back[, `:=` (Half, NULL)]
setcolorder(hakes.add.back, names(comland))

comland <- rbindlist(list(comland[NESPP3 != 507, ], hakes.add.back))

# Herring-----
# Herring data is housed by the state of Maine.
herr qry <- "select year, month, stock_area, negear, gearname, keptmt, discmt
  from maine_herring_catch"

herr.catch <- as.data.table(sqlQuery(channel, herr qry))
setkey(herr.catch, YEAR, MONTH, STOCK_AREA, NEGEAR)

herring <- herr.catch[, list(sum(KEPTMT), sum(DISCMT)), by = key(herr.catch)]
setnames(herring, c("STOCK_AREA", "V1", "V2"), c("AREA", "SPPLIVMT",
  "DISCMT"))

# Using averages from comland to fill in categories
herring[, `:=` (MKTCAT, 5)]
herring[, `:=` (TONCL1, 2)]
herring[, `:=` (UTILCD, 0)]

# compute price/utilization from CF tables
herring.comland <- comland[NESPP3 == 168, ]
# Price from comland
herring.price <- herring.comland[, (sum(SPPVALUE)/sum(SPPLIVMT)),
  by = c("YEAR", "MONTH")]
setnames(herring.price, "V1", "price")
herring <- merge(herring, herring.price, by = c("YEAR", "MONTH"),
  all.x = T)
# Use 1964 prices for < 1964
herring[YEAR < 1964, `:=` (price, mean(herring[YEAR == 1964, price]))]
# Calculate SPPVALUE from price
herring[, `:=` (SPPVALUE, round(price * SPPLIVMT))]

# Utilization from comland

```

```

herring.util <- herring.comland[, sum(SPPLIVMT), by = c("YEAR",
  "MONTH", "UTILCD")]
setnames(herring.util, "V1", "SPPLIVMT")
herring.util[, `:=` (SPPLIVMT.ALL, sum(SPPLIVMT)), by = c("YEAR",
  "MONTH")]
herring.util[, `:=` (Prop, SPPLIVMT/SPPLIVMT.ALL)]
setorder(herring.util, YEAR, MONTH, Prop)
herring.util[, `:=` (cum.prop, cumsum(Prop)), by = c("YEAR", "MONTH")]

# Apply proportions to Maine data set Not pulled all the time
# - current through 2017
herring[, `:=` (Total, sum(SPPLIVMT)), by = c("YEAR", "MONTH")]
herring[, `:=` (Prop, SPPLIVMT/Total)]
setorder(herring, YEAR, MONTH, Prop)
herring[, `:=` (cum.prop, cumsum(Prop)), by = c("YEAR", "MONTH")]

for (iyear in unique(herring.util[, YEAR])) {
  for (imonth in unique(herring.util[YEAR == iyear, MONTH])) {
    cum.prop.low <- 0
    for (iutil in herring.util[YEAR == iyear & MONTH == imonth,
      UTILCD]) {
      cum.prop.high <- herring.util[YEAR == iyear & MONTH ==
        imonth & UTILCD == iutil, cum.prop]
      herring[YEAR == iyear & MONTH == imonth & cum.prop <=
        cum.prop.high & cum.prop > cum.prop.low, `:=` (UTILCD,
        iutil)]
      cum.prop.low <- cum.prop.high
    }
  }
}

# fix column headings
herring[, `:=` (c("Total", "Prop", "cum.prop", "price", "DISCMT"),
  NULL)]
herring[, `:=` (NESPP3, 168)]
setcolorder(herring, names(comland))

# remove herring from data pull and add in Maine numbers
comland <- rbindlist(list(comland[NESPP3 != 168, ], herring))

# Menhaden-----
# fix menhaden records - data from Tom Miller/ Andre
# Bouchheister menhaden <-
# as.data.table(read.csv(paste(data.dir, 'Menhaden.csv', sep
#   = ''))) menhaden.mab <- menhaden[, MA.Total + CB.Total, by

```

```

# = Year] file metric is 1000s of lbs - convert to mt
# menhaden.mab[, SPPLIVMT := (V1 * 1000) * 0.00045359237]
# menhaden.mab[, V1 := NULL] menhaden.gom <- menhaden[, 
# list(Year, NE.Total)] menhaden.gom[, SPPLIVMT := (NE.Total
# * 1000) * 0.00045359237] menhaden.gom[, NE.Total := NULL]

# save(comland, file = paste(out.dir, 'Comland_unkA.RData',
# sep = ''))

# Deal with
# unknowns-----
comland[NEGEAR == 999, `:=` (NEGEAR, 0)]
comland[is.na(TONCL1), `:=` (TONCL1, 0)]
comland[is.na(AREA), `:=` (AREA, as.factor(0))]
comland[AREA == 999, `:=` (AREA, as.factor(0))]
comland[is.na(MKTCAT), `:=` (MKTCAT, 0)]
comland[is.na(UTILCD), `:=` (UTILCD, 0)]

# 1 - drop unknown species/landings
comland <- comland[NESPP3 != 0 & SPPLIVMT != 0, ]

# Sumarry tables missing area known.area <- comland[AREA !=
# 0, sum(SPPLIVMT), by = NESPP3] unknown.area <- comland[AREA
# == 0, sum(SPPLIVMT), by = NESPP3] setnames(known.area,
# 'V1', 'AREA.MT.known') setnames(unknown.area, 'V1',
# 'AREA.MT.unknown') missing.table <- merge(known.area,
# unknown.area, by = 'NESPP3', all = T)
# missing.table[is.na(AREA.MT.known), AREA.MT.known := 0]
# missing.table[is.na(AREA.MT.unknown), AREA.MT.unknown := 0]
# missing.table[, AREA.Ratio := AREA.MT.unknown /
# AREA.MT.known] missing month known.month <- comland[MONTH
# != 0, sum(SPPLIVMT), by = NESPP3] unknown.month <-
# comland[MONTH == 0, sum(SPPLIVMT), by = NESPP3]
# setnames(known.month, 'V1', 'MONTH.MT.known')
# setnames(unknown.month, 'V1', 'MONTH.MT.unknown')
# missing.table <- merge(missing.table, known.month, by =
# 'NESPP3', all = T) missing.table <- merge(missing.table,
# unknown.month, by = 'NESPP3', all = T)
# missing.table[is.na(MONTH.MT.known), MONTH.MT.known := 0]
# missing.table[is.na(MONTH.MT.unknown), MONTH.MT.unknown :=
# 0] missing.table[, MONTH.Ratio := MONTH.MT.unknown /
# MONTH.MT.known] missing gear known.gear <- comland[NEGEAR
# != 0, sum(SPPLIVMT), by = NESPP3] unknown.gear <-
# comland[NEGEAR == 0, sum(SPPLIVMT), by = NESPP3]
# setnames(known.gear, 'V1', 'GEAR.MT.known')

```

```

# setnames(unknown.gear, 'V1', 'GEAR.MT.unknown')
# missing.table <- merge(missing.table, known.gear, by =
# 'NESPP3', all = T) missing.table <- merge(missing.table,
# unknown.gear, by = 'NESPP3', all = T)
# missing.table[is.na(GEAR.MT.known), GEAR.MT.known := 0]
# missing.table[is.na(GEAR.MT.unknown), GEAR.MT.unknown := 0]
# missing.table[, GEAR.Ratio := GEAR.MT.unknown /
# GEAR.MT.known] missing tonnage class known.tc <-
# comland[TONCL1 != 0, sum(SPPLIVMT), by = NESPP3] unknown.tc
# <- comland[TONCL1 == 0, sum(SPPLIVMT), by = NESPP3]
# setnames(known.tc, 'V1', 'TC.MT.known')
# setnames(unknown.tc, 'V1', 'TC.MT.unknown') missing.table
# <- merge(missing.table, known.tc, by = 'NESPP3', all = T)
# missing.table <- merge(missing.table, unknown.tc, by =
# 'NESPP3', all = T) missing.table[is.na(TC.MT.known),
# TC.MT.known := 0] missing.table[is.na(TC.MT.unknown),
# TC.MT.unknown := 0] missing.table[, TC.Ratio :=
# TC.MT.unknown / TC.MT.known] write.csv(missing.table,
# paste(out.dir, '\\Missing_table.csv', sep = ''),
# row.names = F)

# 2 - aggregate by quarter year, half year, major gear, and
# small/large TC
comland[MONTH %in% 1:3, `:=` (QY, 1)]
comland[MONTH %in% 4:6, `:=` (QY, 2)]
comland[MONTH %in% 7:9, `:=` (QY, 3)]
comland[MONTH %in% 10:12, `:=` (QY, 4)]
comland[MONTH == 0, `:=` (QY, 0)]

comland[MONTH %in% 1:6, `:=` (HY, 1)]
comland[MONTH %in% 7:12, `:=` (HY, 2)]
comland[MONTH == 0, `:=` (HY, 0)]

otter <- 50:59
dredge.sc <- 131:132
pot <- c(189:190, 200:219, 300, 301)
longline <- c(10, 40)
seine <- c(70:79, 120:129, 360)
gillnet <- c(100:119, 500, 510, 520)
midwater <- c(170, 370)
dredge.o <- c(281, 282, 380:400)

comland[NEGEAR %in% otter, `:=` (GEAR, "otter")]
comland[NEGEAR %in% dredge.sc, `:=` (GEAR, "dredge.sc")]
comland[NEGEAR %in% pot, `:=` (GEAR, "pot")]

```

```

comland[NEGEAR %in% longline, `:=` (GEAR, "longline")]
comland[NEGEAR %in% seine, `:=` (GEAR, "seine")]
comland[NEGEAR %in% gillnet, `:=` (GEAR, "gillnet")]
comland[NEGEAR %in% midwater, `:=` (GEAR, "midwater")]
comland[NEGEAR %in% dredge.o, `:=` (GEAR, "dredge.o")]
comland[NEGEAR == 0, `:=` (GEAR, "unknown")]
comland[is.na(GEAR), `:=` (GEAR, "other")]
comland[, `:=` (GEAR, as.factor(GEAR))]

comland[TONCL1 %in% 1:3, `:=` (SIZE, "small")]
comland[TONCL1 > 3, `:=` (SIZE, "large")]
comland[TONCL1 == 0, `:=` (SIZE, "unknown")]
comland[, `:=` (SIZE, as.factor(SIZE))]

setkey(comland, YEAR, QY, HY, GEAR, SIZE, AREA, NESPP3, UTILCD)

comland.agg <- comland[, list(sum(SPPLIVMT), sum(SPPVALUE)),
by = key(comland)]

setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

# 3 - Use proportions of known catch to assign unknown catch
# 3.A
# QY/HY-----
unk.month <- comland.agg[QY == 0, ]
k.month <- comland.agg[QY != 0, ]

# 3.A.1 - All match
match.key <- c("YEAR", "NESPP3", "GEAR", "SIZE", "AREA")

unk.month.all <- unk.month[GEAR != "unknown"]
unk.month.all <- unk.month.all[SIZE != "unknown", ]
unk.month.all <- unk.month.all[AREA != 0, ]

k.month.all <- k.month[GEAR != "unknown", ]
k.month.all <- k.month.all[SIZE != "unknown", ]
k.month.all <- k.month.all[AREA != 0, ]

setkeyv(unk.month.all, match.key)
setkeyv(k.month.all, match.key)

month.all <- k.month.all[unk.month.all]

# No match - need to match with larger aggregation
no.match <- month.all[is.na(SPPLIVMT), ]

```

```

no.match[, `:=`(c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match, YEAR, NESPP3, AREA, GEAR)
setkeyv(k.month.all, key(no.match))
month.all.2 <- k.month.all[no.match]
no.match.2 <- month.all.2[is.na(SPPLIVMT), ]
no.match.2[, `:=`(c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match.2, YEAR, NESPP3, AREA)
setkeyv(k.month.all, key(no.match.2))
month.all.3 <- k.month.all[no.match.2]
no.match.3 <- month.all.3[is.na(SPPLIVMT), ]
no.match.3[, `:=`(c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match.3, YEAR, NESPP3)
setkeyv(k.month.all, key(no.match.3))
month.all.4 <- k.month.all[no.match.3]
no.match.4 <- month.all.4[is.na(SPPLIVMT), ]
no.match.4[, `:=`(c("AREA", "GEAR", "SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.AREA", "i.GEAR", "i.SIZE", "i.QY",
  "i.HY", "iUTILCD", "i.SPPLIVMT", "i.SPPVALUE"), c("AREA",
  "GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.4[, `:=`(c("QY", "HY"), 1)]

# Merge all together and proportion catch to known months
month.all <- month.all[!is.na(SPPLIVMT), ]
month.all.2 <- month.all.2[!is.na(SPPLIVMT), ]
month.all.2[, `:=`(SIZE, i.SIZE)]
month.all.2[, `:=`(i.SIZE, NULL)]
setcolororder(month.all.2, names(month.all))
month.all.3 <- month.all.3[!is.na(SPPLIVMT), ]
month.all.3[, `:=`(GEAR, i.GEAR)]

```

```

month.all.3[, `:=` (SIZE, i.SIZE)]
month.all.3[, `:=` (i.GEAR, NULL)]
month.all.3[, `:=` (i.SIZE, NULL)]
setcolororder(month.all.3, names(month.all))
month.all.4 <- month.all.4[!is.na(SPPLIVMT), ]
month.all.4[, `:=` (AREA, i.AREA)]
month.all.4[, `:=` (GEAR, i.GEAR)]
month.all.4[, `:=` (SIZE, i.SIZE)]
month.all.4[, `:=` (i.AREA, NULL)]
month.all.4[, `:=` (i.GEAR, NULL)]
month.all.4[, `:=` (i.SIZE, NULL)]
setcolororder(month.all.4, names(month.all))

month.all <- rbindlist(list(month.all, month.all.2, month.all.3,
    month.all.4))

month.all[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.all[, `:=` (unk, i.SPPLIVMT * prop)]
month.all[, `:=` (unk2, i.SPPVALUE * prop)]
month.all[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
    "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.all, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))

setcolororder(no.match.4, names(month.all))
month.solved <- rbindlist(list(month.all, no.match.4))
rm(list = c(ls(pattern = "month.all"), ls(pattern = "no.match")))

# 3.A.2 - GEAR/SIZE
match.key <- c("YEAR", "NESPP3", "GEAR", "SIZE")

unk.month.g.s <- unk.month[GEAR != "unknown"]
unk.month.g.s <- unk.month.g.s[SIZE != "unknown", ]
unk.month.g.s <- unk.month.g.s[AREA == 0, ]
unk.month.g.s <- unk.month.g.s[, list(sum(SPPLIVMT), sum(SPPVALUE)),
    by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.g.s, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.g.s <- k.month[GEAR != "unknown", ]
k.month.g.s <- k.month.g.s[SIZE != "unknown", ]
k.month.g.s <- k.month.g.s[, list(sum(SPPLIVMT), sum(SPPVALUE)),
    by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.g.s, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyvv(unk.month.g.s, match.key)
setkeyvv(k.month.g.s, match.key)

```

```

month.g.s <- k.month.g.s[unk.month.g.s]

# No match - need to match with larger aggregation
no.match <- month.g.s[is.na(SPPLIVMT), ]
no.match[, `:=`(c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match, YEAR, NESPP3, GEAR)
setkeyv(k.month.g.s, key(no.match))
month.g.s.2 <- k.month.g.s[no.match]
no.match.2 <- month.g.s.2[is.na(SPPLIVMT), ]
no.match.2[, `:=`(c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match.2, YEAR, NESPP3)
setkeyv(k.month.g.s, key(no.match.2))
month.g.s.3 <- k.month.g.s[no.match.2]
no.match.3 <- month.g.s.3[is.na(SPPLIVMT), ]
no.match.3[, `:=`(c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.3[, `:=`(c("QY", "HY"), 1)]
no.match.3[, `:=`(AREA, 0)]

# Merge all together and proportion catch to known months
month.g.s <- month.g.s[!is.na(SPPLIVMT), ]
month.g.s.2 <- month.g.s.2[!is.na(SPPLIVMT), ]
if (nrow(month.g.s.2) > 0) {
  month.g.s.2[, `:=`(SIZE, i.SIZE)]
  month.g.s.2[, `:=`(i.SIZE, NULL)]
  setcolororder(month.g.s.2, names(month.g.s))
  month.g.s <- rbindlist(list(month.g.s, month.g.s.2))
}
month.g.s.3 <- month.g.s.3[!is.na(SPPLIVMT), ]
if (nrow(month.g.s.3) > 0) {
  month.g.s.3[, `:=`(GEAR, i.GEAR)]
  month.g.s.3[, `:=`(SIZE, i.SIZE)]
}

```

```

month.g.s.3[, `:=` (i.GEAR, NULL)]
month.g.s.3[, `:=` (i.SIZE, NULL)]
setcolorder(month.g.s.3, names(month.g.s))
month.g.s <- rbindlist(list(month.g.s, month.g.s.3))
}

month.g.s[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.g.s[, `:=` (unk, i.SPPLIVMT * prop)]
month.g.s[, `:=` (unk2, i.SPPVALUE * prop)]
month.g.s[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.g.s, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.g.s[, `:=` (AREA, 0)]

setcolorder(month.g.s, names(month.solved))
setcolorder(no.match.3, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.g.s, no.match.3))
rm(list = c(ls(pattern = "month.g.s"), ls(pattern = "no.match")))

# 3.A.3 - AREA/GEAR
match.key <- c("YEAR", "NESPP3", "GEAR", "AREA")

unk.month.a.g <- unk.month[GEAR != "unknown"]
unk.month.a.g <- unk.month.a.g[SIZE == "unknown", ]
unk.month.a.g <- unk.month.a.g[AREA != 0, ]
unk.month.a.g <- unk.month.a.g[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.a.g, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.a.g <- k.month[GEAR != "unknown", ]
k.month.a.g <- k.month.a.g[AREA != 0, ]
k.month.a.g <- k.month.a.g[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.a.g, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.month.a.g, match.key)
setkeyv(k.month.a.g, match.key)

month.a.g <- k.month.a.g[unk.month.a.g]

# No match - need to match with larger aggregation
no.match <- month.a.g[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",

```

```

  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match, YEAR, NESPP3, AREA)
setkeyv(k.month.a.g, key(no.match))
month.a.g.2 <- k.month.a.g[no.match]
no.match.2 <- month.a.g.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("GEAR", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.GEAR", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))

# Drop AREA
setkey(no.match.2, YEAR, NESPP3)
setkeyv(k.month.a.g, key(no.match.2))
month.a.g.3 <- k.month.a.g[no.match.2]
no.match.3 <- month.a.g.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("AREA", "GEAR", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.AREA", "i.GEAR", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "GEAR", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.3[, `:=` (c("QY", "HY"), 1)]
no.match.3[, `:=` (SIZE, factor("unknown", levels = c("large",
  "small", "unknown")))]

# Merge all together and proportion catch to known months
month.a.g <- month.a.g[!is.na(SPPLIVMT), ]
month.a.g.2 <- month.a.g.2[!is.na(SPPLIVMT), ]
if (nrow(month.a.g.2) > 0) {
  month.a.g.2[, `:=` (GEAR, i.GEAR)]
  month.a.g.2[, `:=` (i.GEAR, NULL)]
  setcolororder(month.a.g.2, names(month.a.g))
  month.a.g <- rbindlist(list(month.a.g, month.a.g.2))
}
month.a.g.3 <- month.a.g.3[!is.na(SPPLIVMT), ]
if (nrow(month.a.g.3) > 0) {
  month.a.g.3[, `:=` (AREA, i.AREA)]
  month.a.g.3[, `:=` (GEAR, i.GEAR)]
  month.a.g.3[, `:=` (i.AREA, NULL)]
  month.a.g.3[, `:=` (i.GEAR, NULL)]
  setcolororder(month.a.g.3, names(month.a.g))
  month.a.g <- rbindlist(list(month.a.g, month.a.g.3))
}

```

```

month.a.g[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.a.g[, `:=` (unk, i.SPPLIVMT * prop)]
month.a.g[, `:=` (unk2, i.SPPVALUE * prop)]
month.a.g[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.a.g, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.a.g[, `:=` (SIZE, factor("unknown", levels = c("large",
  "small", "unknown")))]

setcolororder(month.a.g, names(month.solved))
setcolororder(no.match.3, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.a.g, no.match.3))
rm(list = c(ls(pattern = "month.a.g"), ls(pattern = "no.match")))

# 3.4.4 - AREA/TC
match.key <- c("YEAR", "NESPP3", "SIZE", "AREA")

unk.month.a.s <- unk.month[GEAR == "unknown"]
unk.month.a.s <- unk.month.a.s[SIZE != "unknown", ]
unk.month.a.s <- unk.month.a.s[AREA != 0, ]
unk.month.a.s <- unk.month.a.s[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.a.s, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.a.s <- k.month[SIZE != "unknown", ]
k.month.a.s <- k.month.a.s[AREA != 0, ]
k.month.a.s <- k.month.a.s[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.a.s, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.month.a.s, match.key)
setkeyv(k.month.a.s, match.key)

month.a.s <- k.month.a.s[unk.month.a.s]

# No match - need to match with larger aggregation
no.match <- month.a.s[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE",
  NULL)]]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match, YEAR, NESPP3, AREA)
setkeyv(k.month.a.s, key(no.match))
month.a.s.2 <- k.month.a.s[no.match]

```

```

no.match.2 <- month.a.s.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match.2, YEAR, NESPP3)
setkeyv(k.month.a.s, key(no.match.2))
month.a.s.3 <- k.month.a.s[no.match.2]
no.match.3 <- month.a.s.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("AREA", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.AREA", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.3[, `:=` (c("QY", "HY"), 1)]
no.match.3[, `:=` (GEAR, factor("unknown", levels = levels(k.month[, GEAR])))]

# Merge all together and proportion catch to known months
month.a.s <- month.a.s[!is.na(SPPLIVMT), ]
month.a.s.2 <- month.a.s.2[!is.na(SPPLIVMT), ]
if (nrow(month.a.s.2) > 0) {
  month.a.s.2[, `:=` (SIZE, i.SIZE)]
  month.a.s.2[, `:=` (i.SIZE, NULL)]
  setcolororder(month.a.s.2, names(month.a.s))
  month.a.s <- rbindlist(list(month.a.s, month.a.s.2))
}
month.a.s.3 <- month.a.s.3[!is.na(SPPLIVMT), ]
if (nrow(month.a.s.3) > 0) {
  month.a.s.3[, `:=` (AREA, i.AREA)]
  month.a.s.3[, `:=` (SIZE, i.SIZE)]
  month.a.s.3[, `:=` (i.AREA, NULL)]
  month.a.s.3[, `:=` (i.SIZE, NULL)]
  setcolororder(month.a.s.3, names(month.a.s))
  month.a.s <- rbindlist(list(month.a.s, month.a.s.3))
}

month.a.s[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.a.s[, `:=` (unk, i.SPPLIVMT * prop)]
month.a.s[, `:=` (unk2, i.SPPVALUE * prop)]
month.a.s[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]

```

```

setnames(month.a.s, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.a.s[, `:=` (GEAR, factor("unknown", levels = levels(k.month[, GEAR])))]

setcolororder(month.a.s, names(month.solved))
setcolororder(no.match.3, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.a.s, no.match.3))
rm(list = c(ls(pattern = "month.a.s"), ls(pattern = "no.match")))

# 3.4.5 - SIZE
match.key <- c("YEAR", "NESPP3", "SIZE")

unk.month.si <- unk.month[GEAR == "unknown"]
unk.month.si <- unk.month.si[SIZE != "unknown", ]
unk.month.si <- unk.month.si[AREA == 0, ]
unk.month.si <- unk.month.si[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.si, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.si <- k.month[SIZE != "unknown", ]
k.month.si <- k.month.si[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.si, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.month.si, match.key)
setkeyv(k.month.si, match.key)

month.si <- k.month.si[unk.month.si]

# No match - need to match with larger aggregation
no.match <- month.si[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match, YEAR, NESPP3)
setkeyv(k.month.si, key(no.match))
month.si.2 <- k.month.si[no.match]
no.match.2 <- month.si.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))

```

```

# Still no match - assign to first QY/HY
no.match.2[, `:=` (c("QY", "HY"), 1)]
no.match.2[, `:=` (AREA, 0)]
no.match.2[, `:=` (GEAR, factor("unknown", levels = levels(k.month[, GEAR])))]

# Merge all together and proportion catch to known months
month.si <- month.si[!is.na(SPPLIVMT), ]
month.si.2 <- month.si.2[!is.na(SPPLIVMT), ]
if (nrow(month.si.2) > 0) {
  month.si.2[, `:=` (SIZE, i.SIZE)]
  month.si.2[, `:=` (i.SIZE, NULL)]
  setcolororder(month.si.2, names(month.si))
  month.si <- rbindlist(list(month.si, month.si.2))
}

month.si[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.si[, `:=` (unk, i.SPPLIVMT * prop)]
month.si[, `:=` (unk2, i.SPPVALUE * prop)]
month.si[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.si, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.si[, `:=` (AREA, 0)]
month.si[, `:=` (GEAR, factor("unknown", levels = levels(k.month[, GEAR])))]

setcolororder(month.si, names(month.solved))
setcolororder(no.match.2, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.si, no.match.2))
rm(list = c(ls(pattern = "month.si"), ls(pattern = "no.match")))

# 3.A.6 - GEAR
match.key <- c("YEAR", "NESPP3", "GEAR")

unk.month.g <- unk.month[GEAR != "unknown"]
unk.month.g <- unk.month.g[SIZE == "unknown", ]
unk.month.g <- unk.month.g[AREA == 0, ]
unk.month.g <- unk.month.g[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.g, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.g <- k.month[GEAR != "unknown", ]
k.month.g <- k.month.g[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.g, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

```

```

setkeyv(unk.month.g, match.key)
setkeyv(k.month.g, match.key)

month.g <- k.month.g[unk.month.g]

# No match - need to match with larger aggregation
no.match <- month.g[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match, YEAR, NESPP3)
setkeyv(k.month.g, key(no.match))
month.g.2 <- k.month.g[no.match]
no.match.2 <- month.g.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("GEAR", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.GEAR", "i.QY", "i.HY", "iUTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "QY", "HY", "UTILCD",
    "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.2[, `:=` (c("QY", "HY"), 1)]
no.match.2[, `:=` (SIZE, factor("unknown", levels = c("large",
    "small", "unknown")))]
no.match.2[, `:=` (AREA, 0)]

# Merge all together and proportion catch to known months
month.g <- month.g[!is.na(SPPLIVMT), ]
month.g.2 <- month.g.2[!is.na(SPPLIVMT), ]
if (nrow(month.g.2) > 0) {
    month.g.2[, `:=` (GEAR, i.GEAR)]
    month.g.2[, `:=` (i.GEAR, NULL)]
    setcolorder(month.g.2, names(month.g))
    month.g <- rbindlist(list(month.g, month.g.2))
}

month.g[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.g[, `:=` (unk, i.SPPLIVMT * prop)]
month.g[, `:=` (unk2, i.SPPVALUE * prop)]
month.g[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
    "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.g, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.g[, `:=` (SIZE, factor("unknown", levels = c("large", "small",
    "unknown")))]

```

```

month.g[, `:=` (AREA, 0)]

setcolororder(month.g, names(month.solved))
setcolororder(no.match.2, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.g, no.match.2))
rm(list = c(ls(pattern = "month.g"), ls(pattern = "no.match")))

# 3.A.7 - AREA
match.key <- c("YEAR", "NESPP3", "AREA")

unk.month.a <- unk.month[GEAR == "unknown"]
unk.month.a <- unk.month.a[SIZE == "unknown", ]
unk.month.a <- unk.month.a[AREA != 0, ]
unk.month.a <- unk.month.a[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.a, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.a <- k.month[AREA != 0, ]
k.month.a <- k.month.a[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(k.month.a, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.month.a, match.key)
setkeyv(k.month.a, match.key)

month.a <- k.month.a[unk.month.a]

# No match - need to match with larger aggregation
no.match <- month.a[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.QY", "i.HY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match, YEAR, NESPP3)
setkeyv(k.month.a, key(no.match))
month.a.2 <- k.month.a[no.match]
no.match.2 <- month.a.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("AREA", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.2, c("i.AREA", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to first QY/HY
no.match.2[, `:=` (c("QY", "HY"), 1)]

```

```

no.match.2[, `:=` (SIZE, factor("unknown", levels = c("large",
  "small", "unknown")))]
no.match.2[, `:=` (GEAR, factor("unknown", levels = levels(k.month[,
  GEAR])))]

# Merge all together and proportion catch to known months
month.a <- month.a[!is.na(SPPLIVMT), ]
month.a.2 <- month.a.2[!is.na(SPPLIVMT), ]
if (nrow(month.a.2) > 0) {
  month.a.2[, `:=` (AREA, i.AREA)]
  month.a.2[, `:=` (i.AREA, NULL)]
  setcolororder(month.a.2, names(month.a))
  month.a <- rbindlist(list(month.a, month.a.2))
}

month.a[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.a[, `:=` (unk, i.SPPLIVMT * prop)]
month.a[, `:=` (unk2, i.SPPVALUE * prop)]
month.a[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.HY", "i.QY", "iUTILCD", "prop"), NULL)]
setnames(month.a, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.a[, `:=` (SIZE, factor("unknown", levels = c("large", "small",
  "unknown")))]
month.a[, `:=` (GEAR, factor("unknown", levels = levels(k.month[,
  GEAR])))]

setcolororder(month.a, names(month.solved))
setcolororder(no.match.2, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.a, no.match.2))
rm(list = c(ls(pattern = "month.a"), ls(pattern = "no.match")))

# 3.4.8 - Species only - no other match
match.key <- c("YEAR", "NESPP3")

unk.month.sp <- unk.month[GEAR == "unknown"]
unk.month.sp <- unk.month.sp[SIZE == "unknown", ]
unk.month.sp <- unk.month.sp[AREA == 0, ]
unk.month.sp <- unk.month.sp[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = c(match.key, "QY", "HY", "UTILCD")]
setnames(unk.month.sp, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

k.month.sp <- k.month[, list(sum(SPPLIVMT), sum(SPPVALUE)), by = c(match.key,
  "QY", "HY", "UTILCD")]
setnames(k.month.sp, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

```

```

setkeyv(unk.month.sp, match.key)
setkeyv(k.month.sp, match.key)

month.sp <- k.month.sp[unk.month.sp]

# No match - assign to first QY/HY
no.match <- month.sp[is.na(SPPLIVMT), ]
no.match[, `:=` (c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match, c("i.QY", "i.HY", "i.UTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
no.match[, `:=` (c("QY", "HY"), 1)]
no.match[, `:=` (AREA, 0)]
no.match[, `:=` (GEAR, factor("unknown", levels = levels(k.month[,
    GEAR])))]
no.match[, `:=` (SIZE, factor("unknown", levels = c("large", "small",
    "unknown")))]

# proportion catch to known months
month.sp <- month.sp[!is.na(SPPLIVMT), ]

month.sp[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
month.sp[, `:=` (unk, i.SPPLIVMT * prop)]
month.sp[, `:=` (unk2, i.SPPVALUE * prop)]
month.sp[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
    "i.HY", "i.QY", "i.UTILCD", "prop"), NULL)]
setnames(month.sp, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
month.sp[, `:=` (AREA, 0)]
month.sp[, `:=` (GEAR, factor("unknown", levels = levels(k.month[,
    GEAR])))]
month.sp[, `:=` (SIZE, factor("unknown", levels = c("large", "small",
    "unknown")))]

setcolorder(month.sp, names(month.solved))
setcolorder(no.match, names(month.solved))
month.solved <- rbindlist(list(month.solved, month.sp, no.match))
rm(list = c(ls(pattern = "month.sp"), ls(pattern = "no.match")))

# Merge back month.solved
setcolorder(month.solved, names(comland.agg))
comland.agg <- rbindlist(list(k.month, month.solved))
setkey(comland.agg, YEAR, QY, HY, SIZE, GEAR, AREA, NESPP3, UTILCD)
comland.agg <- comland.agg[, list(sum(SPPLIVMT), sum(SPPVALUE)),
    by = key(comland.agg)]
setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

```

```

# 3.B
# SIZE-----
unk.size <- comland.agg[SIZE == "unknown", ]
k.size <- comland.agg[SIZE != "unknown", ]

# 3.B.1 - All match
match.key <- c("YEAR", "NESPP3", "QY", "HY", "GEAR", "AREA")

unk.size.all <- unk.size[GEAR != "unknown"]
unk.size.all <- unk.size.all[AREA != 0, ]

k.size.all <- k.size[GEAR != "unknown", ]
k.size.all <- k.size.all[AREA != 0, ]

setkeyv(unk.size.all, match.key)
setkeyv(k.size.all, match.key)

size.all <- k.size.all[unk.size.all]

# No match - need to match with larger aggregation
no.match <- size.all[is.na(SPPLIVMT), ]
no.match[, `:=` (c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match, c("i.SIZE", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
    c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY, GEAR, AREA)
setkeyv(k.size.all, key(no.match))
size.all.2 <- k.size.all[no.match]
no.match.2 <- size.all.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.UTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY
setkey(no.match.2, YEAR, NESPP3, GEAR, AREA)
setkeyv(k.size.all, key(no.match.2))
size.all.3 <- k.size.all[no.match.2]
no.match.3 <- size.all.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.SIZE", "i.QY", "i.HY", "i.UTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
    "SPPLIVMT", "SPPVALUE"))
# Drop GEAR

```

```

setkey(no.match.3, YEAR, NESPP3, AREA)
setkeyv(k.size.all, key(no.match.3))
size.all.4 <- k.size.all[no.match.3]
no.match.4 <- size.all.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match.4, YEAR, NESPP3)
setkeyv(k.size.all, key(no.match.4))
size.all.5 <- k.size.all[no.match.4]
no.match.5 <- size.all.5[is.na(SPPLIVMT), ]
no.match.5[, `:=` (c("AREA", "GEAR", "SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"), NULL)]
setnames(no.match.5, c("i.AREA", "i.GEAR", "i.SIZE", "i.QY",
  "i.HY", "iUTILCD", "i.SPPLIVMT", "i.SPPVALUE"), c("AREA",
  "GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to SIZE to small
no.match.5[, `:=` (SIZE, factor("small", levels = c("large", "small",
  "unknown")))]

# Merge all together and proportion catch to known sizes
size.all <- size.all[!is.na(SPPLIVMT), ]
size.all.2 <- size.all.2[!is.na(SPPLIVMT), ]
size.all.2[, `:=` (QY, i.QY)]
size.all.2[, `:=` (i.QY, NULL)]
setcolorder(size.all.2, names(size.all))
size.all.3 <- size.all.3[!is.na(SPPLIVMT), ]
size.all.3[, `:=` (QY, i.QY)]
size.all.3[, `:=` (HY, i.HY)]
size.all.3[, `:=` (i.QY, NULL)]
size.all.3[, `:=` (i.HY, NULL)]
setcolorder(size.all.3, names(size.all))
size.all.4 <- size.all.4[!is.na(SPPLIVMT), ]
size.all.4[, `:=` (QY, i.QY)]
size.all.4[, `:=` (HY, i.HY)]
size.all.4[, `:=` (GEAR, i.GEAR)]
size.all.4[, `:=` (i.QY, NULL)]
size.all.4[, `:=` (i.HY, NULL)]
size.all.4[, `:=` (i.GEAR, NULL)]
setcolorder(size.all.4, names(size.all))
size.all.5 <- size.all.5[!is.na(SPPLIVMT), ]
size.all.5[, `:=` (QY, i.QY)]

```

```

size.all.5[, `:=` (HY, i.HY)]
size.all.5[, `:=` (GEAR, i.GEAR)]
size.all.5[, `:=` (AREA, i.AREA)]
size.all.5[, `:=` (i.QY, NULL)]
size.all.5[, `:=` (i.HY, NULL)]
size.all.5[, `:=` (i.GEAR, NULL)]
size.all.5[, `:=` (i.AREA, NULL)]
setcolorder(size.all.5, names(size.all))

size.all <- rbindlist(list(size.all, size.all.2, size.all.3,
                           size.all.4, size.all.5))

size.all[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
size.all[, `:=` (unk, i.SPPLIVMT * prop)]
size.all[, `:=` (unk2, i.SPPVALUE * prop)]
size.all[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
                 "i.SIZE", "i.UTILCD", "prop"), NULL)]
setnames(size.all, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))

setcolorder(no.match.5, names(size.all))
size.solved <- rbindlist(list(size.all, no.match.5))
rm(list = c(ls(pattern = "size.all"), ls(pattern = "no.match")))

# 3.B.2 - GEAR
match.key <- c("YEAR", "NESPP3", "QY", "HY", "GEAR")

unk.size.g <- unk.size[GEAR != "unknown"]
unk.size.g <- unk.size.g[AREA == 0, ]
unk.size.g[, `:=` (AREA, NULL)]

k.size.g <- k.size[GEAR != "unknown", ]
k.size.g <- k.size.g[, list(sum(SPPLIVMT), sum(SPPVALUE)), by = c(match.key,
                     "SIZE", "UTILCD")]
setnames(k.size.g, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.size.g, match.key)
setkeyv(k.size.g, match.key)

size.g <- k.size.g[unk.size.g]

# No match - need to match with larger aggregation
no.match <- size.g[is.na(SPPLIVMT), ]
no.match[, `:=` (c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"),
                NULL)]
setnames(no.match, c("i.SIZE", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE")),

```

```

  c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY, GEAR)
setkeyv(k.size.g, key(no.match))
size.g.2 <- k.size.g[no.match]
no.match.2 <- size.g.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "iUTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY
setkey(no.match.2, YEAR, NESPP3, GEAR)
setkeyv(k.size.g, key(no.match.2))
size.g.3 <- k.size.g[no.match.2]
no.match.3 <- size.g.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match.3, YEAR, NESPP3)
setkeyv(k.size.g, key(no.match.3))
size.g.4 <- k.size.g[no.match.3]
no.match.4 <- size.g.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to SIZE to small
no.match.4[, `:=` (SIZE, factor("small", levels = c("large", "small",
  "unknown")))]
no.match.4[, `:=` (AREA, 0)]

# Merge all together and proportion catch to known sizes
size.g <- size.g[!is.na(SPPLIVMT), ]
size.g.2 <- size.g.2[!is.na(SPPLIVMT), ]
size.g.2[, `:=` (QY, i.QY)]
size.g.2[, `:=` (i.QY, NULL)]
setcolororder(size.g.2, names(size.g))
size.g.3 <- size.g.3[!is.na(SPPLIVMT), ]
size.g.3[, `:=` (QY, i.QY)]
size.g.3[, `:=` (HY, i.HY)]
size.g.3[, `:=` (i.QY, NULL)]

```

```

size.g.3[, `:=`(i.HY, NULL)]
setcolororder(size.g.3, names(size.g))
size.g.4 <- size.g.4[!is.na(SPPLIVMT), ]
size.g.4[, `:=`(QY, i.QY)]
size.g.4[, `:=`(HY, i.HY)]
size.g.4[, `:=`(GEAR, i.GEAR)]
size.g.4[, `:=`(i.QY, NULL)]
size.g.4[, `:=`(i.HY, NULL)]
size.g.4[, `:=`(i.GEAR, NULL)]
setcolororder(size.g.4, names(size.g))

size.g <- rbindlist(list(size.g, size.g.2, size.g.3, size.g.4))

size.g[, `:=`(prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
size.g[, `:=`(unk, i.SPPLIVMT * prop)]
size.g[, `:=`(unk2, i.SPPVALUE * prop)]
size.g[, `:=`(c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.SIZE", "iUTILCD", "prop"), NULL)]
setnames(size.g, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
size.g[, `:=`(AREA, 0)]

setcolororder(size.g, names(size.solved))
setcolororder(no.match.4, names(size.g))
size.solved <- rbindlist(list(size.solved, size.g, no.match.4))
rm(list = c(ls(pattern = "size.g"), ls(pattern = "no.match")))

# 3.B.3 - AREA
match.key <- c("YEAR", "NESPP3", "QY", "HY", "AREA")

unk.size.a <- unk.size[GEAR == "unknown"]
unk.size.a <- unk.size.a[AREA != 0, ]
unk.size.a[, `:=`(GEAR, NULL)]

k.size.a <- k.size[AREA != 0, ]
k.size.a <- k.size.a[, list(sum(SPPLIVMT), sum(SPPVALUE)), by = c(match.key,
  "SIZE", "UTILCD")]
setnames(k.size.a, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.size.a, match.key)
setkeyv(k.size.a, match.key)

size.a <- k.size.a[unk.size.a]

# No match - need to match with larger aggregation
no.match <- size.a[is.na(SPPLIVMT), ]

```

```

no.match[, `:=` (c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match, c("i.SIZE", "iUTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
    c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY, AREA)
setkeyv(k.size.a, key(no.match))
size.a.2 <- k.size.a[no.match]
no.match.2 <- size.a.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "iUTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY
setkey(no.match.2, YEAR, NESPP3, AREA)
setkeyv(k.size.a, key(no.match.2))
size.a.3 <- k.size.a[no.match.2]
no.match.3 <- size.a.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.SIZE", "i.QY", "i.HY", "iUTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
    "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match.3, YEAR, NESPP3)
setkeyv(k.size.a, key(no.match.3))
size.a.4 <- k.size.a[no.match.3]
no.match.4 <- size.a.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("AREA", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.AREA", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "SIZE", "QY", "HY",
    "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to SIZE to small
no.match.4[, `:=` (SIZE, factor("small", levels = c("large", "small",
    "unknown")))]
no.match.4[, `:=` (GEAR, factor("unknown", levels = levels(k.size[,
    GEAR])))]

# Merge all together and proportion catch to known sizes
size.a <- size.a[!is.na(SPPLIVMT), ]
size.a.2 <- size.a.2[!is.na(SPPLIVMT), ]
size.a.2[, `:=` (QY, i.QY)]
size.a.2[, `:=` (i.QY, NULL)]
setcolorder(size.a.2, names(size.a))

```

```

size.a.3 <- size.a.3[!is.na(SPPLIVMT), ]
size.a.3[, `:=`(QY, i.QY)]
size.a.3[, `:=`(HY, i.HY)]
size.a.3[, `:=`(i.QY, NULL)]
size.a.3[, `:=`(i.HY, NULL)]
setcolorder(size.a.3, names(size.a))
size.a.4 <- size.a.4[!is.na(SPPLIVMT), ]
size.a.4[, `:=`((QY, i.QY)]
size.a.4[, `:=`((HY, i.HY)]
size.a.4[, `:=`((AREA, i.AREA)]
size.a.4[, `:=`((i.QY, NULL)]
size.a.4[, `:=`((i.HY, NULL)]
size.a.4[, `:=`((i.AREA, NULL)]
setcolorder(size.a.4, names(size.a))

size.a <- rbindlist(list(size.a, size.a.2, size.a.3, size.a.4))

size.a[, `:=`((prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
size.a[, `:=`((unk, i.SPPLIVMT * prop)]
size.a[, `:=`((unk2, i.SPPVALUE * prop)]
size.a[, `:=`((c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.SIZE", "iUTILCD", "prop"), NULL)]
setnames(size.a, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
size.a[, `:=`((GEAR, factor("unknown", levels = levels(k.size[,
  GEAR])))]

setcolorder(size.a, names(size.solved))
setcolorder(no.match.4, names(size.a))
size.solved <- rbindlist(list(size.solved, size.a, no.match.4))
rm(list = c(ls(pattern = "size.a"), ls(pattern = "no.match")))

# 3.B.4 - Species only - no other match
match.key <- c("YEAR", "NESPP3", "QY", "HY")

unk.size.sp <- unk.size[GEAR == "unknown"]
unk.size.sp <- unk.size.sp[SIZE == "unknown", ]
unk.size.sp <- unk.size.sp[AREA == 0, ]
unk.size.sp[, `:=`((c("GEAR", "AREA"), NULL)]

k.size.sp <- k.size[, list(sum(SPPLIVMT), sum(SPPVALUE)), by = c(match.key,
  "SIZE", "UTILCD")]
setnames(k.size.sp, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.size.sp, match.key)
setkeyv(k.size.sp, match.key)

```

```

size.sp <- k.size.sp[unk.size.sp]

# No match - need to match with larger aggregation
no.match <- size.sp[is.na(SPPLIVMT), ]
no.match[, `:=`(c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.SIZE", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
  c("SIZE", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY)
setkeyv(k.size.sp, key(no.match))
size.sp.2 <- k.size.sp[no.match]
no.match.2 <- size.sp.2[is.na(SPPLIVMT), ]
no.match.2[, `:=`(c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match.2, c("i.SIZE", "i.QY", "i.UTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("SIZE", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY
setkey(no.match.2, YEAR, NESPP3)
setkeyv(k.size.sp, key(no.match.2))
size.sp.3 <- k.size.sp[no.match.2]
no.match.3 <- size.sp.3[is.na(SPPLIVMT), ]
no.match.3[, `:=`(c("SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.SIZE", "i.QY", "i.HY", "i.UTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to SIZE to small
no.match.3[, `:=`(SIZE, factor("small", levels = c("large", "small",
  "unknown")))]
no.match.3[, `:=`(GEAR, factor("unknown", levels = levels(k.size[,,
  GEAR])))]
no.match.3[, `:=`(AREA, 0)]

# Merge together and proportion catch to known sizes
size.sp <- size.sp[!is.na(SPPLIVMT), ]
size.sp.2 <- size.sp.2[!is.na(SPPLIVMT), ]
size.sp.2[, `:=`(QY, i.QY)]
size.sp.2[, `:=`(i.QY, NULL)]
setcolorder(size.sp.2, names(size.sp))
size.sp.3 <- size.sp.3[!is.na(SPPLIVMT), ]
size.sp.3[, `:=`(QY, i.QY)]
size.sp.3[, `:=`(HY, i.HY)]
size.sp.3[, `:=`(i.QY, NULL)]
size.sp.3[, `:=`(i.HY, NULL)]

```

```

setcolororder(size.sp.3, names(size.sp))

size.sp <- rbindlist(list(size.sp, size.sp.2, size.sp.3))

size.sp[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
size.sp[, `:=` (unk, i.SPPLIVMT * prop)]
size.sp[, `:=` (unk2, i.SPPVALUE * prop)]
size.sp[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.SIZE", "iUTILCD", "prop"), NULL)]
setnames(size.sp, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
size.sp[, `:=` (AREA, 0)]
size.sp[, `:=` (GEAR, factor("unknown", levels = levels(k.size[,
  GEAR])))]

setcolororder(size.sp, names(size.solved))
setcolororder(no.match.3, names(size.solved))
size.solved <- rbindlist(list(size.solved, size.sp, no.match.3))
rm(list = c(ls(pattern = "size.sp"), ls(pattern = "no.match")))

# Merge back size.solved
setcolororder(size.solved, names(comland.agg))
comland.agg <- rbindlist(list(k.size, size.solved))
setkey(comland.agg, YEAR, QY, HY, SIZE, GEAR, AREA, NESPP3, UTILCD)
comland.agg <- comland.agg[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = key(comland.agg)]
setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

# 3.C
# GEAR-----
unk.gear <- comland.agg[GEAR == "unknown", ]
k.gear <- comland.agg[GEAR != "unknown", ]

# 3.C.1 - All match
match.key <- c("YEAR", "NESPP3", "QY", "HY", "SIZE", "AREA")

unk.gear.all <- unk.gear[AREA != 0, ]

k.gear.all <- k.gear[AREA != 0, ]

setkeyv(unk.gear.all, match.key)
setkeyv(k.gear.all, match.key)

gear.all <- k.gear.all[unk.gear.all]

# No match - need to match with larger aggregation

```

```

no.match <- gear.all[is.na(SPPLIVMT), ]
no.match[, `:=` (c("GEAR", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match, c("i.GEAR", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
    c("GEAR", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY, SIZE, AREA)
setkeyv(k.gear.all, key(no.match))
gear.all.2 <- k.gear.all[no.match]
no.match.2 <- gear.all.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("GEAR", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match.2, c("i.GEAR", "i.QY", "i.UTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("GEAR", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY
setkey(no.match.2, YEAR, NESPP3, SIZE, AREA)
setkeyv(k.gear.all, key(no.match.2))
gear.all.3 <- k.gear.all[no.match.2]
no.match.3 <- gear.all.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("GEAR", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.GEAR", "i.QY", "i.HY", "i.UTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "QY", "HY", "UTILCD",
    "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match.3, YEAR, NESPP3, AREA)
setkeyv(k.gear.all, key(no.match.3))
gear.all.4 <- k.gear.all[no.match.3]
no.match.4 <- gear.all.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
    "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "i.UTILCD",
    "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
    "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop AREA
setkey(no.match.4, YEAR, NESPP3)
setkeyv(k.gear.all, key(no.match.4))
gear.all.5 <- k.gear.all[no.match.4]
no.match.5 <- gear.all.5[is.na(SPPLIVMT), ]
no.match.5[, `:=` (c("AREA", "GEAR", "SIZE", "QY", "HY", "UTILCD",
    "SPPLIVMT", "SPPVALUE"), NULL)]
setnames(no.match.5, c("i.AREA", "i.GEAR", "i.SIZE", "i.QY",
    "i.HY", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"), c("AREA",
    "GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to GEAR to other

```

```

no.match.5[, `:=` (GEAR, factor("other", levels = levels(k.gear[, GEAR])))]

# Merge all together and proportion catch to known gears
gear.all <- gear.all[!is.na(SPPLIVMT), ]
gear.all.2 <- gear.all.2[!is.na(SPPLIVMT), ]
gear.all.2[, `:=` (QY, i.QY)]
gear.all.2[, `:=` (i.QY, NULL)]
setcolororder(gear.all.2, names(gear.all))
gear.all.3 <- gear.all.3[!is.na(SPPLIVMT), ]
gear.all.3[, `:=` (QY, i.QY)]
gear.all.3[, `:=` (HY, i.HY)]
gear.all.3[, `:=` (i.QY, NULL)]
gear.all.3[, `:=` (i.HY, NULL)]
setcolororder(gear.all.3, names(gear.all))
gear.all.4 <- gear.all.4[!is.na(SPPLIVMT), ]
gear.all.4[, `:=` (QY, i.QY)]
gear.all.4[, `:=` (HY, i.HY)]
gear.all.4[, `:=` (SIZE, i.SIZE)]
gear.all.4[, `:=` (i.QY, NULL)]
gear.all.4[, `:=` (i.HY, NULL)]
gear.all.4[, `:=` (i.SIZE, NULL)]
setcolororder(gear.all.4, names(gear.all))
gear.all.5 <- gear.all.5[!is.na(SPPLIVMT), ]
gear.all.5[, `:=` (QY, i.QY)]
gear.all.5[, `:=` (HY, i.HY)]
gear.all.5[, `:=` (SIZE, i.SIZE)]
gear.all.5[, `:=` (AREA, i.AREA)]
gear.all.5[, `:=` (i.QY, NULL)]
gear.all.5[, `:=` (i.HY, NULL)]
gear.all.5[, `:=` (i.SIZE, NULL)]
gear.all.5[, `:=` (i.AREA, NULL)]
setcolororder(gear.all.5, names(gear.all))

gear.all <- rbindlist(list(gear.all, gear.all.2, gear.all.3,
                           gear.all.4, gear.all.5))

gear.all[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
gear.all[, `:=` (unk, i.SPPLIVMT * prop)]
gear.all[, `:=` (unk2, i.SPPVALUE * prop)]
gear.all[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
                 "i.GEAR", "i.UTILCD", "prop"), NULL)]
setnames(gear.all, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))

setcolororder(no.match.5, names(gear.all))

```

```

gear.solved <- rbindlist(list(gear.all, no.match.5))
rm(list = c(ls(pattern = "gear.all"), ls(pattern = "no.match")))

# 3.C.2 - Species only - no other match
match.key <- c("YEAR", "NESPP3", "QY", "HY", "SIZE")

unk.gear.sp <- unk.gear[AREA == 0, ]
unk.gear.sp[, `:=`("AREA", NULL)]

k.gear.sp <- k.gear[, list(sum(SPPLIVMT), sum(SPPVALUE)), by = c(match.key,
  "GEAR", "UTILCD")]
setnames(k.gear.sp, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

setkeyv(unk.gear.sp, match.key)
setkeyv(k.gear.sp, match.key)

gear.sp <- k.gear.sp[unk.gear.sp]

# No match - need to match with larger aggregation
no.match <- gear.sp[is.na(SPPLIVMT), ]
no.match[, `:=`(c("GEAR", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.GEAR", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
  c("GEAR", "UTILCD", "SPPLIVMT", "SPPVALUE"))

# Drop QY
setkey(no.match, YEAR, NESPP3, HY, SIZE)
setkeyv(k.gear.sp, key(no.match))
gear.sp.2 <- k.gear.sp[no.match]
no.match.2 <- gear.sp.2[is.na(SPPLIVMT), ]
no.match.2[, `:=`(c("GEAR", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match.2, c("i.GEAR", "i.QY", "i.UTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("GEAR", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))

# Drop HY
setkey(no.match.2, YEAR, NESPP3, SIZE)
setkeyv(k.gear.sp, key(no.match.2))
gear.sp.3 <- k.gear.sp[no.match.2]
no.match.3 <- gear.sp.3[is.na(SPPLIVMT), ]
no.match.3[, `:=`(c("GEAR", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.GEAR", "i.QY", "i.HY", "i.UTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))

# Drop SIZE
setkey(no.match.3, YEAR, NESPP3)

```

```

setkeyv(k.gear.sp, key(no.match.3))
gear.sp.4 <- k.gear.sp[no.match.3]
no.match.4 <- gear.sp.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.GEAR", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("GEAR", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - assign to GEAR to other
no.match.4[, `:=` (GEAR, factor("other", levels = levels(k.gear[, 
  GEAR])))]
no.match.4[, `:=` (AREA, 0)]

# Merge all together and proportion catch to known gears
gear.sp <- gear.sp[!is.na(SPPLIVMT), ]
gear.sp.2 <- gear.sp.2[!is.na(SPPLIVMT), ]
gear.sp.2[, `:=` (QY, i.QY)]
gear.sp.2[, `:=` (i.QY, NULL)]
setcolororder(gear.sp.2, names(gear.sp))
gear.sp.3 <- gear.sp.3[!is.na(SPPLIVMT), ]
gear.sp.3[, `:=` (QY, i.QY)]
gear.sp.3[, `:=` (HY, i.HY)]
gear.sp.3[, `:=` (i.QY, NULL)]
gear.sp.3[, `:=` (i.HY, NULL)]
setcolororder(gear.sp.3, names(gear.sp))
gear.sp.4 <- gear.sp.4[!is.na(SPPLIVMT), ]
gear.sp.4[, `:=` (QY, i.QY)]
gear.sp.4[, `:=` (HY, i.HY)]
gear.sp.4[, `:=` (SIZE, i.SIZE)]
gear.sp.4[, `:=` (i.QY, NULL)]
gear.sp.4[, `:=` (i.HY, NULL)]
gear.sp.4[, `:=` (i.SIZE, NULL)]
setcolororder(gear.sp.4, names(gear.sp))

gear.sp <- rbindlist(list(gear.sp, gear.sp.2, gear.sp.3, gear.sp.4))

gear.sp[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
gear.sp[, `:=` (unk, i.SPPLIVMT * prop)]
gear.sp[, `:=` (unk2, i.SPPVALUE * prop)]
gear.sp[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
  "i.GEAR", "iUTILCD", "prop"), NULL)]
setnames(gear.sp, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))
gear.sp[, `:=` (AREA, 0)]

setcolororder(gear.sp, names(gear.solved))

```

```

setcolorder(no.match.4, names(gear.solved))
gear.solved <- rbindlist(list(gear.solved, gear.sp, no.match.4))
rm(list = c(ls(pattern = "gear.sp"), ls(pattern = "no.match")))

# Merge back gear.solved
setcolorder(gear.solved, names(comland.agg))
comland.agg <- rbindlist(list(k.gear, gear.solved))
setkey(comland.agg, YEAR, QY, HY, SIZE, GEAR, AREA, NESPP3, UTILCD)
comland.agg <- comland.agg[, list(sum(SPPLIVMT), sum(SPPVALUE)),
  by = key(comland.agg)]
setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

# 3.D
# AREA-----
unk.area <- comland.agg[AREA == 0, ]
k.area <- comland.agg[AREA != 0, ]

# 3.C.1 - All match
match.key <- c("YEAR", "NESPP3", "QY", "HY", "SIZE", "GEAR")

unk.area.all <- unk.area

k.area.all <- k.area

setkeyv(unk.area.all, match.key)
setkeyv(k.area.all, match.key)

area.all <- k.area.all[unk.area.all]

# No match - need to match with larger aggregation
no.match <- area.all[is.na(SPPLIVMT), ]
no.match[, `:=` (c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match, c("i.AREA", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
  c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop QY
setkey(no.match, YEAR, NESPP3, HY, SIZE, GEAR)
setkeyv(k.area.all, key(no.match))
area.all.2 <- k.area.all[no.match]
no.match.2 <- area.all.2[is.na(SPPLIVMT), ]
no.match.2[, `:=` (c("AREA", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"),
  NULL)]
setnames(no.match.2, c("i.AREA", "i.QY", "i.UTILCD", "i.SPPLIVMT",
  "i.SPPVALUE"), c("AREA", "QY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop HY

```

```

setkey(no.match.2, YEAR, NESPP3, SIZE, GEAR)
setkeyv(k.area.all, key(no.match.2))
area.all.3 <- k.area.all[no.match.2]
no.match.3 <- area.all.3[is.na(SPPLIVMT), ]
no.match.3[, `:=` (c("AREA", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.3, c("i.AREA", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"))
# Drop SIZE
setkey(no.match.3, YEAR, NESPP3, GEAR)
setkeyv(k.area.all, key(no.match.3))
area.all.4 <- k.area.all[no.match.3]
no.match.4 <- area.all.4[is.na(SPPLIVMT), ]
no.match.4[, `:=` (c("AREA", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT",
  "SPPVALUE"), NULL)]
setnames(no.match.4, c("i.AREA", "i.SIZE", "i.QY", "i.HY", "iUTILCD",
  "i.SPPLIVMT", "i.SPPVALUE"), c("AREA", "SIZE", "QY", "HY",
  "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Drop GEAR
setkey(no.match.4, YEAR, NESPP3)
setkeyv(k.area.all, key(no.match.4))
area.all.5 <- k.area.all[no.match.4]
no.match.5 <- area.all.5[is.na(SPPLIVMT), ]
no.match.5[, `:=` (c("AREA", "GEAR", "SIZE", "QY", "HY", "UTILCD",
  "SPPLIVMT", "SPPVALUE"), NULL)]
setnames(no.match.5, c("i.AREA", "i.GEAR", "i.SIZE", "i.QY",
  "i.HY", "iUTILCD", "i.SPPLIVMT", "i.SPPVALUE"), c("AREA",
  "GEAR", "SIZE", "QY", "HY", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# Still no match - use 3 or 5 year window then drop year
years <- unique(no.match.5[, YEAR], by = key(no.match.5))
no.match.6 <- c()
area.all.6 <- c()
for (i in 1:length(years)) {
  # 3 year window
  k.area.3y <- comland.agg[AREA != 0 & YEAR %in% (years[i] -
    1):(years[i] + 1), ]
  setkey(k.area.3y, NESPP3, AREA)
  k.area.3y <- k.area.3y[, list(sum(SPPLIVMT), sum(SPPVALUE)),
    by = c(key(k.area.3y), "UTILCD")]
  setnames(k.area.3y, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

  unk.area.3y <- no.match.5[YEAR == years[i], ]

  setkey(unk.area.3y, NESPP3)
}

```

```

setkey(k.area.3y, NESPP3)
area.3y <- k.area.3y[unk.area.3y]

no.match.3y <- area.3y[is.na(SPPLIVMT), ]
no.match.3y[, `:=` (c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"),
    NULL)]
setnames(no.match.3y, c("i.AREA", "i.UTILCD", "i.SPPLIVMT",
    "i.SPPVALUE"), c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"))
no.match.6 <- rbindlist(list(no.match.6, no.match.3y))
area.all.6 <- rbindlist(list(area.all.6, area.3y))
}

years <- unique(no.match.6[, YEAR], by = key(no.match.6))
no.match.7 <- c()
area.all.7 <- c()
for (i in 1:length(years)) {
    # 5 year window
    k.area.5y <- comland.agg[AREA != 0 & YEAR %in% (years[i] -
        2):(years[i] + 2), ]
    setkey(k.area.5y, NESPP3, AREA)
    k.area.5y <- k.area.5y[, list(sum(SPPLIVMT), sum(SPPVALUE)),
        by = c(key(k.area.5y), "UTILCD")]
    setnames(k.area.5y, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

    unk.area.5y <- no.match.6[YEAR == years[i], ]

    setkey(unk.area.5y, NESPP3)
    setkey(k.area.5y, NESPP3)
    area.5y <- k.area.5y[unk.area.5y]

    no.match.5y <- area.5y[is.na(SPPLIVMT), ]
    no.match.5y[, `:=` (c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"),
        NULL)]
    setnames(no.match.5y, c("i.AREA", "i.UTILCD", "i.SPPLIVMT",
        "i.SPPVALUE"), c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"))

    no.match.7 <- rbindlist(list(no.match.7, no.match.5y))
    area.all.7 <- rbindlist(list(area.all.7, area.5y))
}
# Drop year
setkey(no.match.7, NESPP3)
k.area.all <- k.area.all[, list(sum(SPPLIVMT), sum(SPPVALUE)),
    by = c("NESPP3", "AREA", "UTILCD")]
setnames(k.area.all, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))
setkey(k.area.all, NESPP3)

```

```

area.all.8 <- k.area.all[no.match.7]
no.match.8 <- area.all.8[is.na(SPPLIVMT), ]
no.match.8[, `:=` (c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"),
NULL)]
setnames(no.match.8, c("i.AREA", "i.UTILCD", "i.SPPLIVMT", "i.SPPVALUE"),
c("AREA", "UTILCD", "SPPLIVMT", "SPPVALUE"))
# If still no match - leave as unknown

# Merge all together and proportion catch to known areas
area.all <- area.all[!is.na(SPPLIVMT), ]
area.all.2 <- area.all.2[!is.na(SPPLIVMT), ]
area.all.2[, `:=` (QY, i.QY)]
area.all.2[, `:=` (i.QY, NULL)]
setcolororder(area.all.2, names(area.all))
area.all.3 <- area.all.3[!is.na(SPPLIVMT), ]
area.all.3[, `:=` (QY, i.QY)]
area.all.3[, `:=` (HY, i.HY)]
area.all.3[, `:=` (i.QY, NULL)]
area.all.3[, `:=` (i.HY, NULL)]
setcolororder(area.all.3, names(area.all))
area.all.4 <- area.all.4[!is.na(SPPLIVMT), ]
area.all.4[, `:=` (QY, i.QY)]
area.all.4[, `:=` (HY, i.HY)]
area.all.4[, `:=` (SIZE, i.SIZE)]
area.all.4[, `:=` (i.QY, NULL)]
area.all.4[, `:=` (i.HY, NULL)]
area.all.4[, `:=` (i.SIZE, NULL)]
setcolororder(area.all.4, names(area.all))
area.all.5 <- area.all.5[!is.na(SPPLIVMT), ]
area.all.5[, `:=` (QY, i.QY)]
area.all.5[, `:=` (HY, i.HY)]
area.all.5[, `:=` (SIZE, i.SIZE)]
area.all.5[, `:=` (GEAR, i.GEAR)]
area.all.5[, `:=` (i.QY, NULL)]
area.all.5[, `:=` (i.HY, NULL)]
area.all.5[, `:=` (i.SIZE, NULL)]
area.all.5[, `:=` (i.GEAR, NULL)]
setcolororder(area.all.5, names(area.all))
area.all.6 <- area.all.6[!is.na(SPPLIVMT), ]
setcolororder(area.all.6, names(area.all))
area.all.7 <- area.all.7[!is.na(SPPLIVMT), ]
setcolororder(area.all.7, names(area.all))
area.all.8 <- area.all.8[!is.na(SPPLIVMT), ]
setcolororder(area.all.8, names(area.all))

```

```

area.all <- rbindlist(list(area.all, area.all.2, area.all.3,
                           area.all.4, area.all.5, area.all.6, area.all.7, area.all.8))

area.all[, `:=` (prop, SPPLIVMT/sum(SPPLIVMT)), by = match.key]
area.all[, `:=` (unk, i.SPPLIVMT * prop)]
area.all[, `:=` (unk2, i.SPPVALUE * prop)]
area.all[, `:=` (c("SPPLIVMT", "SPPVALUE", "i.SPPLIVMT", "i.SPPVALUE",
                 "i.AREA", "iUTILCD", "prop"), NULL)]
setnames(area.all, c("unk", "unk2"), c("SPPLIVMT", "SPPVALUE"))

setcolorder(no.match.8, names(area.all))
area.solved <- rbindlist(list(area.all, no.match.8))
rm(list = ls(pattern = "area.all"), ls(pattern = "no.match"))

# Merge back area.solved
setcolorder(area.solved, names(comland.agg))
comland.agg <- rbindlist(list(k.area, area.solved))
setkey(comland.agg, YEAR, QY, HY, SIZE, GEAR, AREA, NESPP3, UTILCD)
comland.agg <- comland.agg[, list(sum(SPPLIVMT), sum(SPPVALUE)),
                           by = key(comland.agg)]
setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))

#-----
if (sum.by == "EPU") {
  # Assign EPU based on statarea
  gom <- c(500, 510, 512:515)
  gb <- c(521:526, 551, 552, 561, 562)
  mab <- c(537, 539, 600, 612:616, 621, 622, 625, 626, 631,
           632)
  ss <- c(463:467, 511)

  comland.agg[AREA %in% gom, `:=` (EPU, "GOM")]
  comland.agg[AREA %in% gb, `:=` (EPU, "GB")]
  comland.agg[AREA %in% mab, `:=` (EPU, "MAB")]
  comland.agg[AREA %in% ss, `:=` (EPU, "SS")]
  comland.agg[is.na(EPU), `:=` (EPU, "OTHER")]
  comland.agg[, `:=` (EPU, factor(EPU, levels = c("GOM", "GB",
                                                "MAB", "SS", "OTHER")))]

  setkey(comland.agg, YEAR, NESPP3, QY, GEAR, SIZE, EPU, UTILCD)

  comland.agg <- comland.agg[, list(sum(SPPLIVMT), sum(SPPVALUE)),
                             by = key(comland.agg)]

  setnames(comland.agg, c("V1", "V2"), c("SPPLIVMT", "SPPVALUE"))
}

```

```

# Note - NAFO landings by division only so not available in
# sum.by = 'stat.area' Add NAFO foreign landings - Data from
# http://www.nafo.int/data/frames/data.html
temp <- tempfile()
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-60-69.zip?ver=2016-08"
              temp)
nafo.60 <- as.data.table(read.csv(unz(temp, "NAFO21B-60-69.txt")))
unlink(temp)
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-70-79.zip?ver=2016-08"
              temp)
nafo.70 <- as.data.table(read.csv(unz(temp, "NAFO21B-70-79.txt")))
unlink(temp)
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-80-89.zip?ver=2016-08"
              temp)
nafo.80 <- as.data.table(read.csv(unz(temp, "NAFO21B-80-89.txt")))
unlink(temp)
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-90-99.zip?ver=2016-08"
              temp)
nafo.90 <- as.data.table(read.csv(unz(temp, "NAFO21B-90-99.txt")))
unlink(temp)
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-2000-09.zip?ver=2016-08"
              temp)
nafo.00 <- as.data.table(read.csv(unz(temp, "NAFO21B-2000-09.txt")))
unlink(temp)
download.file("https://www.nafo.int/Portals/0/Stats/nafo-21b-2010-15.zip?ver=2017-08"
              temp)
nafo.10 <- as.data.table(read.csv(unz(temp, "nafo-21b-2010-15/NAFO21B-2010-15.csv")))
unlink(temp)

# 2010 + data have different column headers
setnames(nafo.10, c("Gear", "AreaCode", "SpeciesEffort",
                  "GearCode", "Divcode", "Code"))

nafo <- rbindlist(list(nafo.60, nafo.70, nafo.80, nafo.90,
                       nafo.00, nafo.10))

# Remove US landings (Country code 22), extra divisions (use
# only 47, 51:56, 61:63), and effort codes (1:3)
nafo <- nafo[Country != 22 & Divcode %in% c(47, 51:56, 61:63) &
            Code > 3, ]

# Deal with unknown monthly catch?????

# Get nafo code in a similar format to comland
nafoland <- nafo[, list(Year, GearCode, Tonnage, Divcode,

```

```

    Code, Catches)]
nafoland[, `:=` (MONTH, 0)]
setnames(nafoland, "Catches", "SPPLIVMT")

month <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",
         "Aug", "Sep", "Oct", "Nov", "Dec")
for (i in 1:12) {
  nafoland.month <- nafo[, list(Year, GearCode, Tonnage,
    Divcode, Code, get(month[i]))]
  nafoland.month[, `:=` (MONTH, i)]
  setnames(nafoland.month, names(nafoland.month)[6], "SPPLIVMT")
  nafoland <- rbindlist(list(nafoland, nafoland.month))
}

nafoland <- nafoland[SPPLIVMT != 0, ]

nafoland[, `:=` (EPU, factor(NA, levels = c("GOM", "GB", "MAB",
                                             "SS", "OTHER")))]
nafoland[Divcode == 47, `:=` (EPU, "SS")]
nafoland[Divcode == 51, `:=` (EPU, "GOM")]
nafoland[Divcode %in% c(52, 54:56), `:=` (EPU, "GB")]
nafoland[Divcode %in% c(53, 61:63), `:=` (EPU, "MAB")]
nafoland[is.na(EPU), `:=` (EPU, "OTHER")]

nafoland[, `:=` (Divcode, NULL)]

## Fix missing Scotian Shelf data from 21B
SS.nafo <- as.data.table(read.csv(file.path(data.dir.3, "SS_NAFO_21A.csv"),
                                    skip = 8))

# Add NAFOSPP code to SS.nafo
nafo.spp <- as.data.table(read.csv(file.path(data.dir.3,
                                             "species.txt")))
setnames(nafo.spp, "Abbreviation", "Species ASFIS")
nafo.spp <- nafo.spp[, list(Code, Species ASFIS)]

SS.nafo <- merge(SS.nafo, nafo.spp, by = "Species ASFIS",
                  all.x = T)

# Only grab missing data
SS.nafo <- SS.nafo[Year %in% c(2003, 2008, 2009), ]

setkey(SS.nafo, Year, Code)

SS.land <- SS.nafo[, sum(Catch...000.Kg.), by = key(SS.nafo)]

```

```

setnames(SS.land, "V1", "SPPLIVMT")

# Add GearCode, Tonnage, Month, and EPU
SS.land[, `:=` (GearCode, 99)]
SS.land[, `:=` (Tonnage, 0)]
SS.land[, `:=` (MONTH, 0)]
SS.land[, `:=` (EPU, "SS")]

setcolorder(SS.land, names(nafoland))

nafoland <- rbindlist(list(nafoland, SS.land))

# Rectify NAFO codes with US codes Species
setnames(nafoland, c("Year", "GearCode", "Tonnage", "Code"),
          c("YEAR", "NAFOGEAR", "TONCL1", "NAFOSPP"))

spp <- as.data.table(sqlQuery(channel, "select NAFOSPP, NESPP3 from CFSPP"))

# Fix missing NAFO codes
missing.spp <- data.table(NAFOSPP = c(110, 141, 189, 480,
                                         484, 487, 488, 489), NESPP3 = c(240, 509, 512, 366, 368,
                                         367, 370, 369))
spp <- rbindlist(list(spp, missing.spp))

setkey(spp, NAFOSPP)
spp <- unique(spp, by = key(spp))

# Fix many to one relationships
spp[NAFOSPP == 199, `:=` (NESPP3, 524)]
spp[NAFOSPP == 299, `:=` (NESPP3, 525)]
spp[NAFOSPP == 469, `:=` (NESPP3, 359)]
spp[NAFOSPP == 499, `:=` (NESPP3, 526)]
spp[NAFOSPP == 529, `:=` (NESPP3, 764)]
spp[NAFOSPP == 699, `:=` (NESPP3, 899)]

nafoland <- merge(nafoland, spp, by = "NAFOSPP", all.x = T)

# fix codes
nafoland[NAFOSPP == 309, `:=` (NESPP3, 150L)]
nafoland[NAFOSPP == 462, `:=` (NESPP3, 481L)]
nafoland[NAFOSPP == 464, `:=` (NESPP3, 355L)]
nafoland[NAFOSPP == 468, `:=` (NESPP3, 493L)]
nafoland[NAFOSPP == 704, `:=` (NESPP3, 817L)]

# remove species without a match

```

```

nafoland <- nafoland[!is.na(NESPP3), ]

# Remove herring catch - already included from Maine Data
# earlier
nafoland <- nafoland[NESPP3 != 168, ]

# Gearcodes
gear <- as.data.table(sqlQuery(channel, "select NEGEAR, NAFOGEAR from Gear"))
gear <- unique(gear, by = "NAFOGEAR")

nafoland <- merge(nafoland, gear, by = "NAFOGEAR", all.x = T)

# fix codes
nafoland[NAFOGEAR == 8, `:=` (NEGEAR, 50L)]
nafoland[NAFOGEAR == 9, `:=` (NEGEAR, 370L)]
nafoland[NAFOGEAR == 19, `:=` (NEGEAR, 58L)]
nafoland[NAFOGEAR == 49, `:=` (NEGEAR, 60L)]
nafoland[NAFOGEAR == 56, `:=` (NEGEAR, 21L)]

# Tonnage
nafoland[TONCL1 == 7, `:=` (TONCL1, 6L)]

# Drop NAFO codes
nafoland[, `:=` (c("NAFOGEAR", "NAFOSPP"), NULL)]

# Fix skates get little skates and winter skates from
# skates(ns) - use survey in half years Generate Half year
# variable in comland
nafoland.skates <- nafoland[NESPP3 == 365, ]
nafoland.skates[MONTH %in% 1:6, `:=` (Half, 1)]
nafoland.skates[MONTH %in% 7:12, `:=` (Half, 2)]

setkey(skate.hake.nafo, YEAR, Half, EPU)

nafoland.skates <- merge(nafoland.skates, skate.hake.nafo,
    by = key(skate.hake.nafo), all.x = T)

nafoland.skates[NESPP3 == 365, `:=` (little, little.per *
    SPPLIVMT)]
nafoland.skates[is.na(little), `:=` (little, 0)]

nafoland.skates[NESPP3 == 365, `:=` (winter, winter.per *
    SPPLIVMT)]
nafoland.skates[is.na(winter), `:=` (winter, 0)]

```

```

nafoland.skates[NESPP3 == 365, `:=` (other.skate, SPPLIVMT -
(little + winter))]

# Little (366), winter (367), skates(ns) (365) put skates in
# nafoland format to merge back
little <- nafoland.skates[, list(YEAR, Half, EPU, TONCL1,
MONTH, NESPP3, NEGEAR, little)]
little[, `:=`(NESPP3, 366L)]
setnames(little, "little", "SPPLIVMT")
little <- little[SPPLIVMT > 0, ]

winter <- nafoland.skates[, list(YEAR, Half, EPU, TONCL1,
MONTH, NESPP3, NEGEAR, winter)]
winter[, `:=`(NESPP3, 367L)]
setnames(winter, "winter", "SPPLIVMT")
winter <- winter[SPPLIVMT > 0, ]

other <- nafoland.skates[, list(YEAR, Half, EPU, TONCL1,
MONTH, NESPP3, NEGEAR, other.skate)]
other[, `:=`(NESPP3, 365L)]
setnames(other, "other.skate", "SPPLIVMT")
other <- other[SPPLIVMT > 0, ]

# merge all three and reformat for nafoland
skates.add.back <- rbindlist(list(little, winter, other))

skates.add.back[, `:=`(Half, NULL)]
setcolorder(skates.add.back, names(nafoland))

nafoland <- rbindlist(list(nafoland[NESPP3 != 365, ], skates.add.back))

# aggregate nafo landings 2 - aggregate by quarter year, half
# year, major gear, and small/large TC
nafoland[MONTH %in% 1:3, `:=`(QY, 1)]
nafoland[MONTH %in% 4:6, `:=`(QY, 2)]
nafoland[MONTH %in% 7:9, `:=`(QY, 3)]
nafoland[MONTH %in% 10:12, `:=`(QY, 4)]
nafoland[MONTH == 0, `:=`(QY, 1)]

nafoland[NEGEAR %in% otter, `:=`(GEAR, "otter")]
nafoland[NEGEAR %in% dredge.sc, `:=`(GEAR, "dredge.sc")]
nafoland[NEGEAR %in% pot, `:=`(GEAR, "pot")]
nafoland[NEGEAR %in% longline, `:=`(GEAR, "longline")]
nafoland[NEGEAR %in% seine, `:=`(GEAR, "seine")]
nafoland[NEGEAR %in% gillnet, `:=`(GEAR, "gillnet")]

```

```

nafoland[NEGEAR %in% midwater, `:=` (GEAR, "midwater")]
nafoland[NEGEAR %in% dredge.o, `:=` (GEAR, "dredge.o")]
nafoland[NEGEAR == 99, `:=` (GEAR, "unknown")]
nafoland[is.na(GEAR), `:=` (GEAR, "other")]
nafoland[, `:=` (GEAR, as.factor(GEAR))]

nafoland[TONCL1 %in% 1:3, `:=` (SIZE, "small")]
nafoland[TONCL1 > 3, `:=` (SIZE, "large")]
nafoland[TONCL1 == 0, `:=` (SIZE, "unknown")]
nafoland[, `:=` (SIZE, as.factor(SIZE))]

setkey(nafoland, YEAR, QY, GEAR, SIZE, EPU, NESPP3)

nafoland.agg <- nafoland[, sum(SPPLIVMT), by = key(nafoland)]

setnames(nafoland.agg, "V1", "SPPLIVMT")

# Create dummy variable for value
nafoland.agg[, `:=` (SPPVALUE, 0)]
nafoland.agg[, `:=` (UTILCD, 0)]

# Merge comland and nafoland
setcolororder(nafoland.agg, names(comland.agg))

if (foreign == "y") {
  comland.agg[, `:=` (US, T)]
  nafoland.agg[, `:=` (US, F)]
}

comland.nafo <- rbindlist(list(comland.agg, nafoland.agg))

# Remove Menhaden data save(comland.nafo, file =
# paste(out.dir, 'comland_Menhaden.RData', sep = ''))
comland <- comland.nafo[NESPP3 != 221, ]
}

if (sum.by == "stat.area") comland <- comland.agg

# Output file
if (landed == "n") file.landed <- "" else file.landed <- "_meatwt"
if (adjust.ppi == "n") file.adjust <- "" else file.adjust <- "_deflated"
if (sum.by == "EPU") file.by <- "" else file.by <- "_stat_areas"
file.name <- paste0("comland", file.landed, file.adjust, file.by,
  ".RData")

```

```
save(comland, file = file.path(out.dir, file.name))
```

17.1.2.1 Data Processing

The landings data were formatted for inclusion in the ecodata R package with the following R code.

```
# Process commerical data

# These data were derived from the Commercial Fisheries
# Database Biological Sample. More information about these
# data are available at
# https://noaa-edab.github.io/tech-doc/comdat.html

library(dplyr)

raw.dir <- here::here("data-raw")

get_comdat <- function(save_clean = F) {

  load(file.path(raw.dir, "Commercial_data_pull.RData"))

  comdat <- commercial %>% dplyr::rename(EPU = Region) %>%
    dplyr::select(-Source)

  if (save_clean) {
    usethis::use_data(comdat, overwrite = T)
  } else {
    return(comdat)
  }

}
get_comdat(save_clean = T)
```

17.1.3 Data analysis

Fisheries dependent data from Comlands is used in several indicators for the State of the Ecosystem report; the more complicated analyses are detailed in their own sections. The most straightforward use of this data are the aggregate landings indicators. These are calculated by first assigning the various species into aggregate groups. Species are also marked by which management body

manages them. Landings are then summed by year, EPU, aggregate group, and whether they are managed or not. Both managed and unmanaged totals are added together to get the final amount of total landings for that aggregate group within its respective region. Both the total and those landings managed by the management body receiving the report are reported. Proportions of managed landings to total landings are also reported in tabular form.

17.1.4 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

## Ecosystem-wide and managed species total landings
opar <- par(mfrow = c(5, 1), mar = c(0, 0, 0, 0), oma = c(4,
  6.5, 2, 6))

soe.plot(SOE.data, "Time", "Apex Predator Landings GB", stacked = "A",
  status = F, endshade = T, rel.y.num = 1.1, full.trend = T,
  scale.axis = 10^3, end.start = 2007, x.start = 1986, ymin = FALSE,
  y.lower = 0, cex.stacked = 1.5)

soe.plot(SOE.data, "Time", "Piscivore Landings GB", stacked = "B",
  status = F, endshade = T, rel.y.num = 1.1, full.trend = T,
  scale.axis = 10^3, end.start = 2007, x.start = 1986, ymin = FALSE,
  y.lower = 0, extra = TRUE, x.var2 = "Time", y.var2 = "Piscivore NEFMC managed species sea foo
  cex.stacked = 1.5)

soe.plot(SOE.data, "Time", "Planktivore Landings GB", stacked = "C",
  status = F, endshade = T, rel.y.num = 1.1, full.trend = T,
  scale.axis = 10^3, end.start = 2007, x.start = 1986, ymin = FALSE,
  y.lower = 0, extra = TRUE, x.var2 = "Time", y.var2 = "Planktivore NEFMC managed species sea f
  cex.stacked = 1.5)

soe.plot(SOE.data, "Time", "Benthivore Landings GB", stacked = "D",
  status = F, endshade = T, rel.y.num = 1.1, full.trend = T,
  scale.axis = 10^3, end.start = 2007, x.start = 1986, ymin = FALSE,
  y.lower = 0, extra = TRUE, x.var2 = "Time", y.var2 = "Benthivore NEFMC managed species sea fo
```

```
cex.stack = 1.5)

soe.plot(SOE.data, "Time", "Benthos Landings GB", stacked = "E",
status = F, endshade = T, rel.y.num = 1.1, full.trend = T,
scale.axis = 10^3, end.start = 2007, x.start = 1986, ymin = FALSE,
y.lower = 0, extra = TRUE, x.var2 = "Time", y.var2 = "Benthos NEFMC managed species",
cex.stack = 1.5)

soe.stack.axis("Year", expression("Landings, 10^3 * metric tons"),
x.line = 2.7)
```

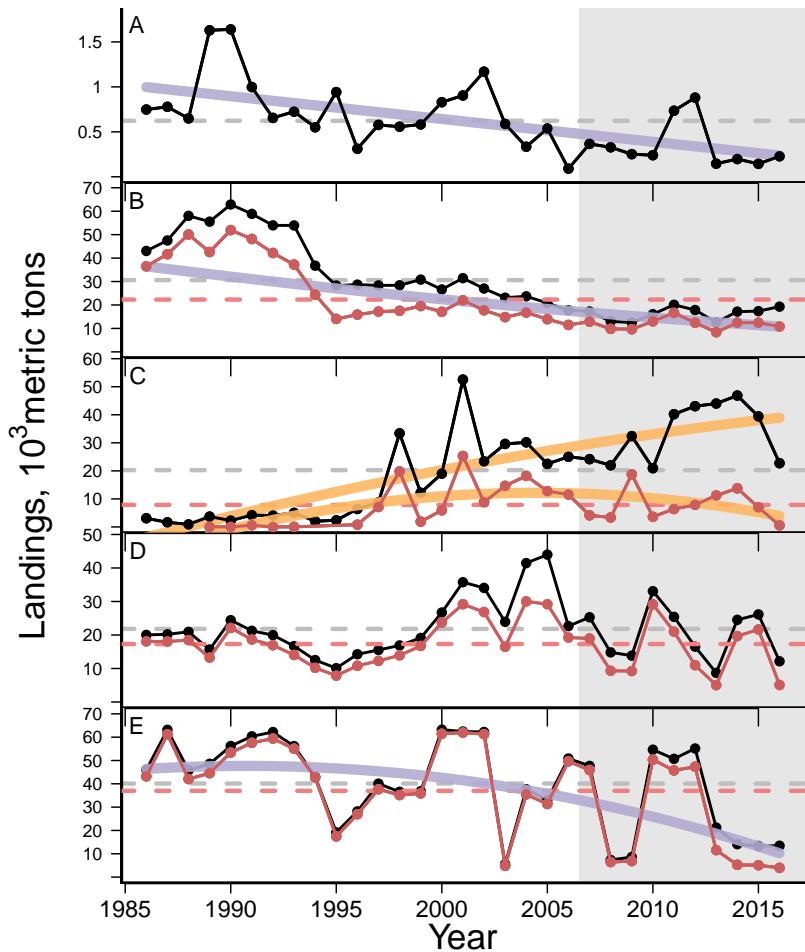


Figure 17.2: NEFMC seafood specific landings (red) and total commercial landings (black) in Georges Bank (A: Apex predators, B: Piscivore, C: Planktivore, D: Benthivore, E: Benthos).

Chapter 18

Long-term Sea Surface Temperature

Description: Long-term sea-surface temperatures

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull

Contributor(s): Kevin Friedland

Data steward: Kevin Friedland, kevin.friedland@noaa.gov

Point of contact: Kevin Friedland, kevin.friedland@noaa.gov

Public availability statement: Source data are available here.

18.1 Methods

Data for long-term sea-surface temperatures were derived from the National Oceanographic and Atmospheric Administration (NOAA) extended reconstructed sea surface temperature data set (ERSST V5). The ERSST V5 dataset is parsed into $2^\circ \times 2^\circ$ gridded bins between 1854-present with monthly temporal resolution. Data were interpolated in regions with limited spatial coverage, and heavily damped during the period between 1854-1880 when collection was inconsistent (Huang, Thorne, et al. 2017a, 2017b). For this analysis, 19 bins were selected that encompassed the Northeast US Continental Shelf region (see Friedland and Hare 2007).

Table 18.1: Coordinates used in NOAA ERSST V5 data extraction.

Longitude	Latitude
-74	40
-74	38
-72	40
-70	44
-70	42
-70	40
-68	44
-68	42

18.1.1 Data sources

This indicator is derived from the NOAA ERSST V5 dataset (Huang, Thorne, et al. 2017a).

18.1.2 Data extraction

R code used in extracting time series of long-term SST data.

```
# Include R code here all year
years = "ersst"

# a year years='2016'

# which data
setwd("C:/2_ersst/datafiles_v4")
setwd("C:/2_ersst/datafiles_v5")

# NES standard bounded by 34-46N and 78-62W
minlon = -78
maxlon = -62
minlat = 34
maxlat = 46
dataoutfile = "C:/2_ersst/nes_std_area_v5.csv"

# DELETE ONLY FILE WILL APPEND AND DOUBLE DATA
file.remove(dataoutfile)

##### END SET
```

```

# LABRADOR SEA minlon= -66; #maxlon= -44; #minlat= 50;
# #maxlat= 70 dataoutfile='C:/2_ersst/lab_sea.csv'

# New eel minlon= -80; maxlon= -40; minlat= 20; maxlat= 40
# dataoutfile='C:/2_ersst/new eel.csv'

# G of Mex minlon= -98; maxlon= -82; minlat= 18; maxlat= 30
# dataoutfile='C:/2_ersst/gomex_area.csv'

# USSE INCLUDING 28-36N and 80-76W not bounding minlon= -80;
# #maxlon= -76; #minlat= 28; #maxlat= 36
# dataoutfile='C:/2_ersst/usse_area.csv'

# GSL standard bounded by 34-46N and 78-62W minlon= -68;
# maxlon= -60; minlat= 46; maxlat= 48
# dataoutfile='C:/2_ersst/gsl.csv'

# PACIFIC area bounded by 10-30N and 166-146W minlon= -166;
# #maxlon= -146; #minlat= 10; #maxlat= 30
# dataoutfile='C:/1_analyses/ersst/pac_islands_area.csv'

# Baltic area bounded by 52-66N and 14-28E minlon= 14;
# #maxlon= 28; #minlat= 52; #maxlat= 66
# dataoutfile='C:/2_ersst/baltic_area.csv'

# North Atlantic Area bounded by 30-70N and 80W-20E minlon=
# -80; #maxlon= -2; #minlat= 30; #maxlat= 70
# dataoutfile='C:/2_ersst/na area1.csv' and ... minlon= 0;
# #maxlon= 20; #minlat= 30; #maxlat= 70
# dataoutfile='C:/1_analyses/ersst/na area2.csv'

# NorthEAST Atlantic Area bounded by 55-70N and 10W-20E
# minlon= -10; #maxlon= -2; #minlat= 56; #maxlat= 70
# dataoutfile='C:/1_analyses/ersst/ne atl 1.csv' and ...
# minlon= 0; #maxlon= 10; #minlat= 56; #maxlat= 70
# dataoutfile='C:/1_analyses/ersst/ne atl 2.csv'

# Pacific steelhead minlon= -160; #maxlon= -122; #minlat= 40;
# #maxlat= 62 dataoutfile='C:/1_analyses/ersst/pac sh.csv'

# North Pacific in two parts 1 minlon= -180; #maxlon= -120;
# #minlat= 30; #maxlat= 72
# dataoutfile='C:/1_analyses/ersst/n_pac_1.csv' 2 minlon=
# 120; #maxlon= 178; #minlat= 30; #maxlat= 72
# dataoutfile='C:/1_analyses/ersst/n_pac_2.csv'

```

```

# North Atlantic Area bounded by 20-70N and 100W-30E minlon=
# -100; #maxlon= -2; #minlat= 20; #maxlat= 70
# dataoutfile='C:/1_analyses/ersst/na area1.csv' and ...
# minlon= 0; #maxlon= 30; #minlat= 20; #maxlat= 70
# dataoutfile='C:/1_analyses/ersst/na area2.csv'

# constants for area
R <- 6371 # Earth mean radius [km]
dheight = 222

# library(ncdf)
library(ncdf4)

# ERSST data lon goes from 0E to 358E with lon at center of
# box lat goes from 88S to 88N with lat at center of box

# start with lon based on degrees lonew array 1 2 ... 90 91
# 92 ... 180 lon 0 2 ... 178 180 182 ... 358 lonew 0 2 ...
# 178 -180 -178 ... -2 star with lat + deg N, - deg S array
# 1 ... 45 ... 89 lon -88 ... 0 ... 88

# -> -> -> TO KEEP THINGS SIMPLE, RETRIEVALS CAN'T BE
# CONTINUOUS FROM - LONS TO + LONs have to extract from -180W
# to -2W separately from 0E to 180E

# -> -> -> OUTPUT APPENDS SO NEED TO DELETE FILE IF ALREADY
# EXISTS

# -> -> -> USE APPROPRIATE lon lat and outfile block:

# set lon limits in array units
if (minlon < 0) {

```

```
arrayminlon = (minlon + 360)/2 + 1
} else {
    arrayminlon = minlon/2 + 1
}

if (maxlon < 0) {
    arraymaxlon = (maxlon + 360)/2 + 1
} else {
    arraymaxlon = maxlon/2 + 1
}

# set lat limits in array units
arrayminlat = minlat/2 + 45
arraymaxlat = maxlat/2 + 45

filelist = list.files(pattern = years)

numfiles = length(filelist)

for (filenum in 1:numfiles) {

    # ersst = open.ncdf(filelist[filenum])
    ersst = nc_open(filelist[filenum])
    print(filelist[filenum])

    # sst = get.var.ncdf( ersst, 'sst')
    sst <- ncvar_get(ersst, "sst")

    year = as.numeric(substr(filelist[filenum], 10, 13))
    month = as.numeric(substr(filelist[filenum], 14, 15))

    for (arrlons in arrayminlon:arraymaxlon) {
        for (arrlats in arrayminlat:arraymaxlat) {

            if (arrlons < 91) {
                regenlon = (arrlons - 1) * 2
            } else {
                regenlon = (arrlons - 1) * 2 - 360
            }

            regenlat = (arrlats - 45) * 2
```

```

long1 = regenlon - 1 * pi/180
lat1 = regenlat - 1 * pi/180
long2 = regenlon + 1 * pi/180
lat2 = regenlat - 1 * pi/180
dwidth1 <- acos(sin(lat1) * sin(lat2) + cos(lat1) *
    cos(lat2) * cos(long2 - long1)) * R
long1 = regenlon - 1 * pi/180
lat1 = regenlat + 1 * pi/180
long2 = regenlon + 1 * pi/180
lat2 = regenlat + 1 * pi/180
dwidth2 <- acos(sin(lat1) * sin(lat2) + cos(lat1) *
    cos(lat2) * cos(long2 - long1)) * R
area = ((dwidth1 + dwidth2)/2) * dheight

dataline <- matrix(c(year, month, regenlon, regenlat,
    round(sst[arrlons, arrlats], digits = 2), area),
    1, 6)

if (is.finite(sst[arrlons, arrlats])) {
    write.table(dataline, file = dataoutfile, sep = ",",
        row.names = F, col.names = F, append = TRUE)
}

}

}

# close.ncdf(ersst)
nc_close(ersst)

}

```

18.1.3 Data Processing

Data were formatted for inclusion in the `ecodata` R package with the following R code.

```

# process ERSST long-term SST data

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

```

```
get_long_term_sst <- function(save_clean = F) {
  long_term_sst <- read.csv(file.path(raw.dir, "ersst annual mean.csv")) %>%
    dplyr::rename(Time = Year, Value = Mean) %>% mutate(Var = "long-term sst",
    EPU = "All", Units = "degreesC")

  if (save_clean) {
    usethis::use_data(long_term_sst, overwrite = T)
  } else {
    return(long_term_sst)
  }
}
```

18.1.4 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mar = c(4, 6, 2, 6))
soe.plot(SOE.data, "Time", "long term sst", end.start = 2007,
  line.forward = TRUE, point.cex = 0.8, rel.y.text = 1.1, x.line = 2,
  y.line = 2.3, x.label = "Year", rel.y.num = 1.1, y.label = expression(paste("Mean SST (",
  degree, "C"))))
```

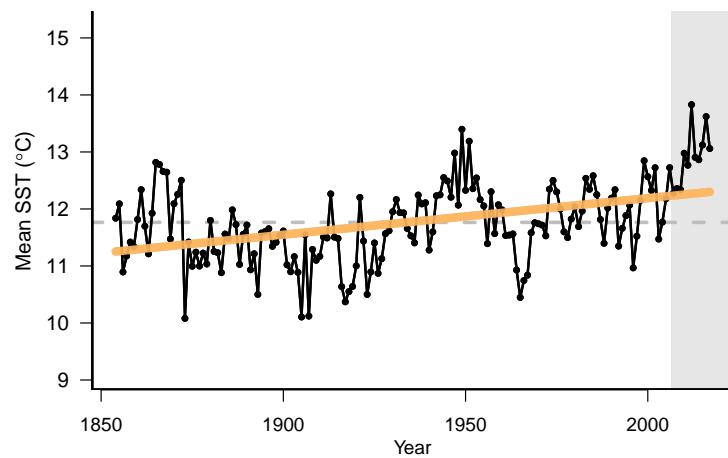


Figure 18.1: Long-term sea surface temperatures on the Northeast Continental Shelf.

Chapter 19

Mid-Atlantic Harmful Algal Bloom Indicator

Description: An aggregation of reported algal bloom data in Chesapeake Bay between 2007-2017.

Found in: State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull

Contributor(s): Sean Hardison, Virginia Department of Health

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Kimberly Bastille, kimberly.bastille@noaa.gov

Public availability statement: Source data for this indicator are available here. Processed time series can be found [here](#).

19.1 Methods

We presented two indicator time series for reports of algal blooms in the southern portion of Chesapeake Bay between 2007-2017. The first indicator was observations of algal blooms above 5000 cell ml⁻¹. This threshold was developed by the Virginia Department of Health (VDH) for *Microcystis* spp. algal blooms based on World Health Organization guidelines (Organization 2003; Health 2011). VDH also uses this same threshold for other algal species blooms in Virginia waters. When cell concentrations are above 5000 cell ml⁻¹, VDH recommends initiation of biweekly water sampling and that relevant local agencies be notified of the elevated cell concentrations.

The second indicator we reported, blooms of *Cochlodinium polykrikoides* at cell concentrations >300 cell ml^{-1} , was chosen due to reports of high ichthyotoxicity seen at these levels. Tang and Gobler (2009) showed that fish exposed to cultured *C. polykrikoides* at densities as low 330 cells ml^{-1} saw 100% mortality within 1 hour, which is often far less than *C. polykrikoides* cell concentrations seen in the field. Algal bloom data were not available for 2015 nor 2010. The algal bloom information presented here are a synthesis of reported events, and has been updated to include data not presented in the 2018 State of the Ecosystem Report.

19.1.1 Data sources

Source data were obtained from VDH. Sampling, identification, and bloom characterization was completed by the VDH, Phytoplankton Analysis Laboratory at Old Dominion University (ODU), Reece Lab at the Virginia Institute of Marine Science (VIMS), and Virginia Department of Environmental Quality. Problem algal species were targeted for identification via light microscopy followed by standard or quantitative PCR assays and/or enzyme-linked immunosorbent assay (ELISA). Reports specifying full methodologies from ODU, VIMS, and VDH source data are available upon request.

19.1.2 Data extraction

Data were extracted from a series of spreadsheets provided by the VDH. We quantified the number of algal blooms in each year reaching target cell density thresholds in the southern Chesapeake Bay.

```
# SOE R packages

library(readxl)
library(dplyr)
library(tidyr)
library(stringr)

data.dir <- "data/CB_HAB"

# Function to process data - cpm specifies cells per ml
# filter
fixer <- function(cpm) {
  hab_2007_2012 <- read_excel(file.path(data.dir, "Query_2007-2012.xlsx"))
  hab_2013_odu <- read_excel(file.path(data.dir, "2013 ODU Data.xlsx"),
    skip = 4)
  hab_2013_vims <- read_excel(file.path(data.dir, "vims_2013.xlsx"),
```

```

    skip = 6)
hab_2014_odu <- read_excel(file.path(data.dir, "2014 ODU data.xlsx"))
hab_2014_vims <- read_excel(file.path(data.dir, "FINALforVDH_22Dec14final.xlsx"))
hab_2016 <- read_excel(file.path(data.dir, "HAB_MAP_Data_2016.xlsx"))
hab_2017 <- read_excel(file.path(data.dir, "HAB_MAP_Data_2017.xlsx"),
                       sheet = 2)

# 2012-----
HAB_2007_2012 <- hab_2007_2012 %>% filter(!is.na(cells_per_ml)) %>%
  filter(!is.na(date)) %>% mutate(year = format(as.POSIXct(date),
  "%Y")) %>% filter(cells_per_ml >= cpm) %>% group_by(year,
  species) %>% dplyr::summarise(Events = n()) %>% as.data.frame()

# 2013-----
# ODU
odu_2013 <- gather(hab_2013_odu, species, cells_per_ml, `Pfiesteria like dinoflagellate`:`A.
  filter(!is.na(cells_per_ml)) %>% filter(cells_per_ml >=
  cpm) %>% mutate(year = 2013) %>% group_by(year, species) %>%
  dplyr::summarise(Events = n()) %>% as.data.frame()

# VIMS
vims_2013 <- hab_2013_vims %>% filter(!is.na(cells_per_ml)) %>%
  mutate(year = "2013") %>% filter(cells_per_ml >= cpm) %>%
  group_by(year, species) %>% dplyr::summarise(Events = n()) %>%
  as.data.frame()

HAB_2013 <- rbind(vims_2013, odu_2013)

# 2014-----
# ODU
long <- gather(hab_2014_odu, species, cells_per_ml, `Karlodinium veneficum`:`Cyanobacteria bl
  factor_key = TRUE)
hab_2014_odu <- long %>% filter(cells_per_ml != 0)
hab_2014_odu$species <- sub("[.]", " ", hab_2014_odu$species)
hab_2014_odu$cells_per_ml <- gsub("[A-Za-z+//]", "", hab_2014_odu$cells_per_ml)
hab_2014_odu$cells_per_ml <- as.numeric(hab_2014_odu$cells_per_ml)

hab_2014_odu <- hab_2014_odu %>% mutate(year = "2014") %>%
  filter(cells_per_ml >= cpm) %>% group_by(year, species) %>%
  dplyr::summarise(Events = n()) %>% as.data.frame()

# VIMS
long <- gather(hab_2014_vims, species, cells_per_ml, `A. monilatum`:`C. subsalsa`)
hab_2014_vims <- long %>% mutate(year = "2014") %>% filter(!is.na(cells_per_ml)) %>%
  mutate(cells_per_ml = as.numeric(cells_per_ml)) %>% filter(cells_per_ml >=

```

```

cpm) %>% group_by(year, species) %>% dplyr::summarise(Events = n()) %>%
  as.data.frame()
HAB_2014 <- rbind(hab_2014_odu, hab_2014_vims)

# 2015-----
# No data

# 2016-----
HAB_2016 <- hab_2016 %>% mutate(species = plyr::mapvalues(species,
  from = c("Eugelna sanguinea", "Microcystin aeruginosa",
    "Microcystis aeruginosa", "Alexandrium monilatum-likely",
    "Alexandrium monilatum"), to = c("Eugelena spp.",
    "Microcystis spp.", "Microcystis spp.", "Alexandrium spp.",
    "Alexandrium spp.")))
HAB_2016$cells_per_ml <- gsub("[a-zA-Z+<>]", "", HAB_2016$cells_per_ml)
HAB_2016 <- HAB_2016 %>% filter(!is.na(cells_per_ml)) %>%
  mutate(year = 2016, cells_per_ml = as.numeric(cells_per_ml)) %>%
  filter(cells_per_ml >= cpm) %>% group_by(year, species) %>%
  dplyr::summarise(Events = n()) %>% as.data.frame()

# 2017-----
hab_2017$species = str_trim(hab_2017$species)
HAB_2017 <- hab_2017 %>% mutate(species = plyr::mapvalues(species,
  c("A. monilatum", "Anabaena sp", "Anabaena sp.", "Anabaena spp",
    "none", "NO HABs", "C. polykrikoides", "Microcystis aeruginosa",
    "Cylindrospermopsis sp"), c("Alexandrium monilatum",
    "Anabaena spp.", "Anabaena spp.", "Anabaena spp.",
    "NA", "NA", "Cochlodinium polykrikoides", "Microcystis aeruginosa",
    "Cylindrospermopsis sp.")))
HAB_2017$cells_per_ml <- gsub("[a-zA-Z+/]", "", HAB_2017$cells_per_ml)
HAB_2017$cells_per_ml <- str_trim(HAB_2017$cells_per_ml)
HAB_2017$cells_per_ml <- as.numeric(HAB_2017$cells_per_ml)
HAB_2017 <- HAB_2017 %>% filter(!is.na(cells_per_ml)) %>%
  mutate(year = "2017") %>% filter(cells_per_ml >= cpm) %>%
  group_by(year, species) %>% dplyr::summarise(Events = n()) %>%
  as.data.frame()

# Aggregate-----
ts <- rbind(HAB_2007_2012, HAB_2013, HAB_2014, HAB_2016,
  HAB_2017)

return(ts)
}

# All blooms > 5000 cells ml^-1

```

```

full <- fixer(cpm = 5000)
full <- full %>% group_by(year) %>% dplyr::summarise(total = sum(Events))
plot(full$year, full$total, type = "o", ylim = c(0, 90), pch = 20,
     ylab = "Bloom Events", las = 1, xlab = "Time", lwd = 2)

# cochlodinium > 300 cells ml-1
cochllo <- fixer(cpm = 300)
cochllo[cochllo$species == "C. polykrikoides" | cochllo$species ==
    "C.polykrikoides", ]$species <- "Cochlodinium polykrikoides"
cochllo <- cochllo[cochllo$species == "Cochlodinium polykrikoides",
    ]
cochllo <- cochllo %>% group_by(year) %>% dplyr::summarise(total = sum(Events))
points(cochllo$year, cochllo$total, type = "o", pch = 20, col = "indianred",
       lwd = 2)
legend(x = 2007, y = 80, legend = c(expression(paste("All reports >5000 cells ml-1")),
                                         expression(paste(italic("C. polykrikoides "),"reports >300 cells ml-1"))),
       col = c("black", "indianred"), lwd = 2, bty = "n")

```

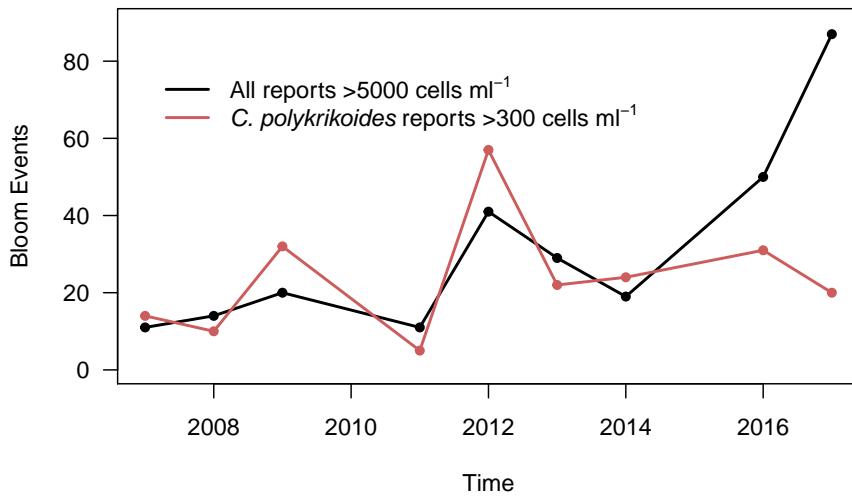


Figure 19.1: All reported algal blooms >5000 cells ml^{-1} (black), and reports of *C. polykrikoides* blooms >300 cells ml^{-1} (red) between 2007-2017.

19.1.3 Data analysis

No data analysis steps took place for this indicator.

Chapter 20

New England Harmful Algal Bloom Indicator

Description: Regional incidence of shellfish bed closures due to presence of toxins associated with harmful algae.

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018)

Indicator Category: Synthesis of published information

Contributor(s): Dave Kulis, Donald M Anderson, Sean Hardison

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Kimberly Bastille, kimberly.bastille@noaa.gov

Public availability statement: Data are publicly available (see Data Sources below).

20.1 Methods

The New England Harmful Algal Bloom (HAB) indicator is a synthesis of shellfish bed closures related to the presence of HAB-associated toxins above threshold levels from 2007-2016 (Figure 20.1). Standard detection methods were used to identify the presence of toxins associated with Amnesic Shellfish Poisoning (ASP), Paralytic Shellfish Poisoning (PSP), and Diarrhetic Shellfish Poisoning (DSP) by state and federal laboratories.

20.1.0.1 Paralytic Shellfish Poisoning

The most common cause of shellfish bed closures in New England is the presence of paralytic shellfish toxins (PSTs) produced by the dinoflagellate *Alexandrium*

catenella. All New England states except Maine relied on the Association of Official Analytical Chemists (AOAC) approved mouse bioassay method to detect PSTs in shellfish during the 2007-2016 period reported here (International 2005).

In Maine, PST detection methods were updated in May 2014 when the state adopted the hydrophilic interaction liquid chromatography (HILIC) UPLC-MS/MS protocol (Boundy et al. 2015) in concordance with National Shellfish Sanitation Program (NSSP) requirements. Prior to this, the primary method used to detect PST in Maine was with the mouse bioassay.

20.1.0.2 Amnesic Shellfish Poisoning

Amnesic shellfish poisoning (ASP) is caused by the toxin domoic acid (DA), which is produced by several phytoplankton species belonging to the genus *Pseudo-nitzchia*. In New England, a UV-HPLC method (Quilliam, Xie, and Hardstaff 1995), which specifies a HPLC-UV protocol, is used for ASP detection.

20.1.0.3 Diarrhetic Shellfish Poisoning

Diarrhetic Shellfish Poisoning (DSP) is rare in New England waters, but the presence of the DSP-associated okadaic acid (OA) in mussels was confirmed in Massachusetts in 2015 (J. Deeds, personal communication, July 7, 2018). Preliminary testing for OA in Massachusetts utilized the commercially available Protein Phosphatase Inhibition Assay (PPIA) and these results are confirmed through LC-MS/MS when necessary (Smienk et al. 2012; Stutts and Deeds 2017).

20.1.1 Data sources

Data used in this indicator were drawn from the 2017 Report on the ICES-IOC Working Group on Harmful Algal Bloom Dynamics (WGHABD). The report and data are available [here](#).

Closure information was collated from information provided by the following organizations:

20.1.2 Data extraction

Data were extracted from the original report visually and accuracy confirmed with report authors.

Table 20.1: Shellfish closure information providers.

State	Source Organization
Maine	Maine Department of Marine Resources
New Hampshire	New Hampshire Department of Environmental Services
Massachusetts	Massachusetts Division of Marine Fisheries
Rhode Island	Rhode Island Department of Environmental Management
Connecticut	Connecticut Department of Agriculture

20.1.3 Data analysis

No data analysis steps took place for this indicator.

The script used to develop the figure in the SOE report is below.

```
# get map data and set constants

# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")
gis.dir <- here::here("gis")

# Source GIS script
source(file.path(r.dir, "GIS_source.R"))

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# projection
map.crs <- CRS("+proj=longlat +lat_1=35 +lat_2=45 +lat_0=40 +lon_0=-77 +x_0=0
+y_0=0 +datum=NAD83 +no_defs +ellps=GRS80 +towgs84=0,0,0")

# coastline
coast <- readOGR(gis.dir, "NES_LME_coast", verbose = F)
coast <- spTransform(coast, map.crs)

# define extents for cropping
e1 <- extent(-78.5, -64, 41, 45)

# crop
coast1 <- crop(coast, e1)

# Get data from SOE dataset
```

```

events <- SOE.data[grep1("SP occurrence", SOE.data$Var), ]
lon <- SOE.data %>% dplyr::filter(Var == "NE HAB Regional center Lon") %>%
  pull(Value)
lat <- SOE.data %>% dplyr::filter(Var == "NE HAB Regional center Lat") %>%
  pull(Value)

events_df <- data.frame(lon = lon, lat = lat, val = events$Value,
                        var = events$Var)

g1 <- events_df %>% filter(val == 8)
# g2 <- events_df %>% filter(val == 3.5)
g3 <- events_df %>% filter(val == 1)

# data.frame to sp object
coordinates(g1) <- ~lon + lat
g1@proj4string <- map.crs
g1 <- spTransform(g1, map.crs)

coordinates(g3) <- ~lon + lat
g3@proj4string <- map.crs
g3 <- spTransform(g3, map.crs)

# plot map and dots of different size based on category
par(mar = c(0, 2.8, 0, 0.2))
plot(coast1, xlim = c(-71.5, -64.5), ylim = c(41, 45), col = "grey",
     yaxs = "i")
plot(g1, add = T, cex = 6, pch = 16, col = "darkorange")
# plot(g2, add = T, cex = 4, pch = 16, col = 'purple')
plot(g3, add = T, cex = 2, pch = 16, col = c("purple", "#56B4E9"))

axis(1, at = c(-71, -69, -67, -65), labels = paste(c(-71, -69,
  -67, -65) * -1, "W"), col = NA, col.ticks = 1, pos = 41)
axis(2, at = c(45, 44, 43, 42, 41.05), labels = paste(c(45, 44,
  43, 42, 41), "N"), las = T, pos = -71.78, col = NA, col.ticks = 1)
legend(-66.2, 42.5, c("PSP", "ASP", "DSP"), col = c("darkorange",
  "purple", "#56B4E9"), pch = 16, cex = 1.1, bty = "n", pt.cex = 2)
text(-65.65, 42.495, "Category")
arrows(-71.765, 41, -71.765, 45, angle = 90, lwd = 2)
arrows(-64.22, 41, -64.22, 45, angle = 90, lwd = 2)
abline(h = 41, lwd = 2)
abline(h = 45, lwd = 2)
abline(v = -64, lwd = 2)
legend(-67.9, 42.5, c("6-10", "2-5", "1"), col = c("black"),

```

```
pch = 16, cex = 1.1, pt.cex = c(6, 4, 2), x.intersp = 1.45,  
y.intersp = 1.75, bty = "n")  
text(-67.7, 42.525, "2007–2016 Detections")
```

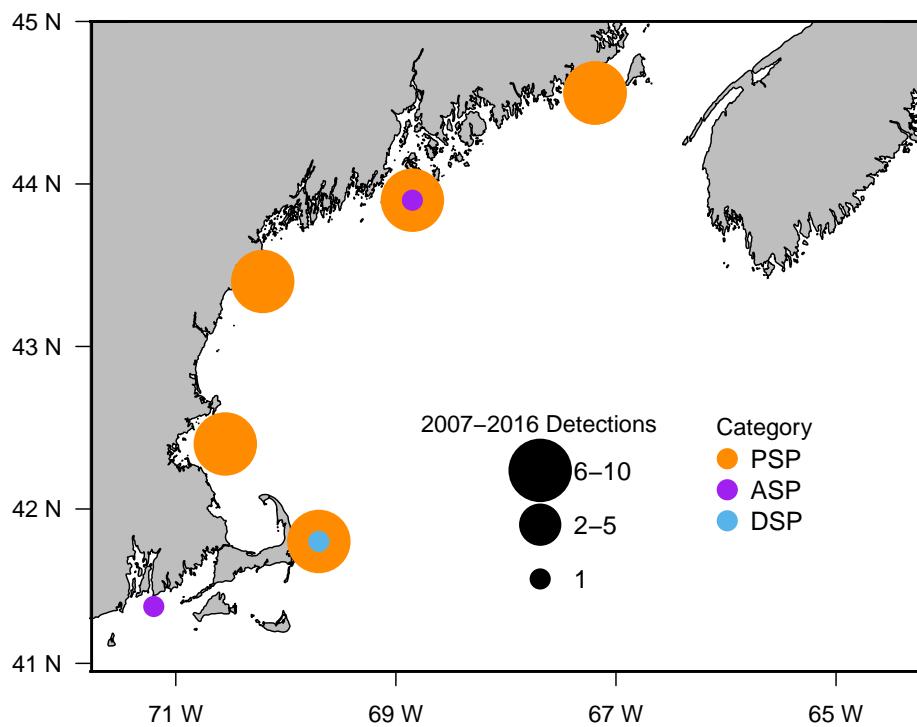


Figure 20.1: Regional HAB related shellfish bed closures in New England between 2007 and 2016.

Chapter 21

Verified Records of Southern Kingfish

Description: Fisheries Observer Data – Verified Records of Southern Kingfish

Found in: State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull

Contributor(s): Debra Duarte, Loren Kellogg

Data steward: Gina Shield, gina.shield@noaa.gov

Point of contact: Gina Shield, gina.shield@noaa.gov

Public availability statement: Due to PII concerns data for this indicator are not publicly available.

21.1 Methods

21.1.1 Data sources

The Fisheries Sampling Branch deploys observers on commercial fisheries trips from Maine to North Carolina. On observed tows, observers must fully document all kept and discarded species encountered. Observers must comply with a Species Verification Program (SVP), which requires photo or sample submissions of high priority species at least once per quarter. Photos and samples submitted for verification are identified independently by at least two reviewers.

The derived data presented in the Mid-Atlantic State of the Ecosystem report for southern kingfish include records verified by the SVP program only. The occurrence of southern kingfish in SVP records were chosen for inclusion in

the report due to the recent increases of the species in SVP observer records since 2010. These data are not a complete list from the New England Fisheries Observer Program (NEFOP). Southern Kingfish are less common than Northern Kingfish in observer data and are possibly misidentified so we have initially included records here only when a specimen record was submitted to and verified through the SVP (see Data extraction).

21.1.2 Data extraction

SQL query:

```

/* training trips, 900 trips, incidental takes, most duplicate records (unless different)
/* this is a general script to pull the data on any species from the SVP*/
/* specimens submitted as photos and samples included separately */
/* nefop_ims.species_verification query that approximates is: select * from nefop_ims... */

SELECT ss.year YEAR, MONTH, ss.obid as OBSID, ss.tripid TRIPID, NEGEAR, DATELAND, SS.HAUL,
       HAUL, TO_CHAR(t.dateland, 'DD-MON-YY') dateland, DATELAND,
       substr(c.tripid, 1,3) obid, c.tripid, c.year, HAUL, TO_CHAR(t.dateland, 'DD-MON-YY') dateland
FROM
       (SELECT substr(c.tripid, 1,3) obid, c.tripid, c.year, HAUL, TO_CHAR(t.dateland, 'DD-MON-YY') dateland,
              idmethod, idcomments, obs_contacted, filelocation
       from nefop_ims.species_verification a, obdbs.obspec b
       where a.species=b.nespp4) c
       left join obdbs.obspec d
       on c.incorrectspp=d.nespp4
       join OBSCON.OBSCON_TRIPS_FSB t on c.year||c.tripid=t.year||substr(t.LINK1,10,6)
       where c.YEAR like '%%' and c.TRIPID like '%%' and c.inc is null and DATELAND between t
UNION
SELECT substr(g.tripid,1,3) obid, g.tripid, g.year, HAUL, TO_CHAR(s.dateland, 'DD-MON-YY') dateland, DATELAND,
       idmethod, idcomments
FROM ((select id_num, e.tripid, e.year, lpad(CAST(haulnum as varchar2(4 byte)),4,'0') haul,
           idmethod, idcomments, obs_contacted, filelocation
       from nefop_ims.species_verification e, obdbs.obspec f
       where e.species=f.nespp4) g
       left join obdbs.obspec h
       on g.incorrectspp=h.nespp4
       join obdbs.asmtrp_entry s on g.year||g.tripid=s.yearland||s.tripid)

```

```

where g.YEAR like '%%' and g.TRIPID like '%%' and g.inc is null and DATELAND between to_date('01-01-1900 00:00:00') and to_date('31-12-1900 23:59:59')

UNION

SELECT substr(r.tripid, 1,3) obid, r.tripid, r.year, HAUL, TO_CHAR(s.dateland, 'DD-MON-YY') as DATELAND,
       idmethod, idcomments
  FROM ((select id_num, a.tripid, a.year, lpad(CAST(haulnum as varchar2(4 byte)),4,'0') haul, inc,
             idmethod, idcomments, obs_contacted, filelocation
        from nefop_ims.species_verification a, obdbs.obspec b
       where a.species=b.nespp4) r
  left join obdbs.obspec d
  on r.incorrectspp=d.nespp4
 join OBPRELIM.TRP_BASE s on r.year||r.tripid=s.yearland||S.TRIPID)
 where r.YEAR like '%%' and r.TRIPID like '%%' and r.inc is null and DATELAND between to_date('01-01-1900 00:00:00') and to_date('31-12-1900 23:59:59')

left join

/* OBDBS OBHAU */

select
      substr(link3,1,15)  LINK1 ,
      LINK3,
      substr(link3,1,3)  PROGRAM ,
      substr(link3,4,4)  YEAR ,
      substr(link3,8,2)  MONTH ,
      substr(link3,10,6) TRIPID ,
      substr(link3,16,4) HAULNUM ,
      decode(substr(link3,8,2), '01', 1, '02', 1, '03', 1, '04', 2,
            '05', 2, '06', 2, '07', 3, '08', 3, '09', 3, '10', 4, '11', 4, '12', 4, null) QTR,
      NEGEAR,
      OBSRFLAG,
      CATEXIST,
      case when yearhbeg is not null and monthhbeg is not null and dayhbeg is not null then
            to_date(ltrim(rtrim(YEARHBEG))||ltrim(rtrim(MONTHBEG))||ltrim(rtrim(DAYHBEG))||ltrim(rtrim(TIMEHBEG)))
      else null
      end datehbeg,
      case when yearhend is not null and monthend is not null and dayhend is not null then
            to_date(ltrim(rtrim(YEARHEND))||ltrim(rtrim(MONTHEND))||ltrim(rtrim(DAYHEND))||ltrim(rtrim(TIMEHEND)))
      else null
      end datehend,
      LATHBEG,
      LONHBEG,
      LATEND,
      LONHEND,
      cast( substr(latsbeg,1,2) + substr(latsbeg,3,2) / 60 + substr(latsbeg,5,2) / 3600 as number(8,6))

```

```

-1 * cast(substr(lonsbeg,1,2) + substr(lonsbeg,3,2) / 60 + substr(lonsbeg,5,2) / 3600 as number)
cast(substr(latsend,1,2) + substr(latsend,3,2) / 60 + substr(latsend,5,2) / 3600 as number)
-1 * cast(substr(lonsend,1,2) + substr(lonsend,3,2) / 60 + substr(lonsend,5,2) / 3600 as number)
cast(substr(lathbeg,1,2) + substr(lathbeg,3,2) / 60 + substr(lathbeg,5,2) / 3600 as number)
-1 * cast(substr(lonhbeg,1,2) + substr(lonhbeg,3,2) / 60 + substr(lonhbeg,5,2) / 3600 as number)
cast(substr(lathend,1,2) + substr(lathend,3,2) / 60 + substr(lathend,5,2) / 3600 as number)
-1 * cast(substr(lonhend,1,2) + substr(lonhend,3,2) / 60 + substr(lonhend,5,2) / 3600 as number)
AREA,
HAUCOMMENTS
from obdbs.obhau_base

UNION
select
substr(link3,1,15)  LINK1 ,
LINK3,
substr(link3,1,3)  PROGRAM ,
substr(link3,4,4)  YEAR ,
substr(link3,8,2)  MONTH ,
substr(link3,10,6) TRIPID ,
substr(link3,16,4) HAULNUM ,
decode(substr(link3,8,2), '01', 1, '02', 1, '03', 1, '04', 2,
      '05', 2, '06', 2, '07', 3, '08', 3, '09', 3, '10', 4, '11', 4, '12', 4, null) QTR,

NEGEAR,
OBSRFLAG,
CATEXIST,
case when yearhbeg is not null and monthhbeg is not null and dayhbeg is not null then
      to_date(ltrim(rtrim(YEARHBEG))||ltrim(rtrim(MONTHHBEG))||ltrim(rtrim(DAYHBEG))||ltrim(HAULNUM))
else null
end datehbeg,
case when yearhend is not null and monthend is not null and dayhend is not null then
      to_date(ltrim(rtrim(YEARHEND))||ltrim(rtrim(MONTHEND))||ltrim(rtrim(DAYHEND))||ltrim(HAULNUM))
else null
end datehend,
LATHBEG,
LONHBEG,
LATHEND,
LONHEND,
cast( substr(latsbeg,1,2) + substr(latsbeg,3,2) / 60 + substr(latsbeg,5,2) / 3600 as number)
-1 * cast(substr(lonsbeg,1,2) + substr(lonsbeg,3,2) / 60 + substr(lonsbeg,5,2) / 3600 as number)
cast(substr(latsend,1,2) + substr(latsend,3,2) / 60 + substr(latsend,5,2) / 3600 as number)
-1 * cast(substr(lonsend,1,2) + substr(lonsend,3,2) / 60 + substr(lonsend,5,2) / 3600 as number)
cast(substr(lathbeg,1,2) + substr(lathbeg,3,2) / 60 + substr(lathbeg,5,2) / 3600 as number)
-1 * cast(substr(lonhbeg,1,2) + substr(lonhbeg,3,2) / 60 + substr(lonhbeg,5,2) / 3600 as number)

```

```

cast(substr(lathend,1,2) + substr(lathend,3,2) / 60 + substr(lathend,5,2) / 3600 as number(8,6))
-1 * cast(substr(lonhend,1,2) + substr(lonhend,3,2) / 60 + substr(lonhend,5,2) / 3600 as number(8,6))
AREA,
HAUCOMMENTS
from obv10.obhau_base

UNION

/* ASM HAU */

select
    substr(link3,1,15) LINK1 ,
    LINK3,
    substr(link3,1,3) PROGRAM ,
    substr(link3,4,4) YEAR ,
    substr(link3,8,2) MONTH ,
    substr(link3,10,6) TRIPID ,
    substr(link3,16,4) HAULNUM ,

    decode(substr(link3,8,2), '01', 1, '02', 1, '03', 1, '04', 2,
           '05', 2, '06', 2, '07', 3, '08', 3, '09', 3, '10', 4, '11', 4, '12', 4, null) QTR,
    NEGEAR,
    OBSRFLAG,
    cast(null as varchar2(1)) CATEXIST,
    to_date(ltrim(rtrim(YEARHBEG))||ltrim(rtrim(MONTHBEG))||ltrim(rtrim(DAYHBEG))||ltrim(rtrim(TIM
    to_date(ltrim(rtrim(YEARHEND))||ltrim(rtrim(MONTHEND))||ltrim(rtrim(DAYHEND))||ltrim(rtrim(TIM
LATHBEG,
LONHBEG,
LATHEND,
LONHEND,
cast(null as number(8,6)) GIS_LATSBE ,  

cast(null as number) GIS_LONSBE ,  

cast(null as number(8,6)) GIS_LATSE ,  

cast(null as number) GIS_LONSE ,  

cast(substr(lathbeg,1,2) + substr(lathbeg,3,2) / 60 + substr(lathbeg,5,2) / 3600 as number(8,6))
-1 * cast(substr(lonhbeg,1,2) + substr(lonhbeg,3,2) / 60 + substr(lonhbeg,5,2) / 3600 as number(8,6))
cast(substr(lathend,1,2) + substr(lathend,3,2) / 60 + substr(lathend,5,2) / 3600 as number(8,6))
-1 * cast(substr(lonhend,1,2) + substr(lonhend,3,2) / 60 + substr(lonhend,5,2) / 3600 as number(8,6))
AREA,
HAUCOMMENTS

from asmhau_prelim p
left outer join obtriptrack t
    on substr(p.link3,1,15) = t.link1

```

```

where (substr(link3,4,6) between '201005' and '201012' and substr(link3,1,3) between
      or (substr(link3,4,6) > '201012' and substr(link3,1,3) <> '900')
      and substr(link3,1,15) not in (select link1 from obv10.obtrp_base) ) rr

on ss.YEAR||ss.TRIPID||ss.HAUL = rr.YEAR||rr.TRIPID||ltrim(rr.HAULNUM)

where ss.OBID like '%%' and ss.TRIPID like '%%' and (CODE like '%6617%') and HAUL like
ORDER BY SPECIES, YEAR, TRIPID, HAUL
;

```

21.1.3 Data analysis

Time series were summed by year and plotted, and mapped data for individual records were plotted according to the location where gear was hauled. As coordinate data were not always available for each record, the map does not include all occurrences of southern kingfish, but was included for spatial context.

R code use to aggregate data:

```

library(dplyr)

total_by_year<- source %>%
  filter(SPECIES == "KINGFISH, SOUTHERN") %>% #Filter species
  group_by(YEAR) %>%
  dplyr::summarise(n = n()) #Sum observations per year

```

21.1.4 Plotting

```

# For map and plot. Latitude and longitude data for this
# figure are not publicly available.
sk.dat <- SOE.data[grep("Lat", SOE.data$Units) & grep("Southern Kingfish",
  SOE.data$Var), ]
lon <- sk.dat[as.numeric(sk.dat$value) < 0, ]$Value
lat <- sk.dat[as.numeric(sk.dat$value) > 0, ]$Value

# create data.frame
df <- data.frame(year = sk.dat$time, lon = lon, lat = lat)

# set color palette
colors1 <- adjustcolor(matlab.like2(8), 0.5)
colors2 <- adjustcolor(matlab.like2(8), 0.5)

```

```

colors3 <- adjustcolor(matlab.like2(8), 0.5)
colors4 <- adjustcolor(matlab.like2(8), 1)
colors <- c(colors1[1:2], colors2[3:4], colors3[5:6], colors4[7:8])

# map values to colors
df <- df %>% arrange(year) %>% mutate(colors = plyr::mapvalues(year,
  from = c("2010", "2011", "2012", "2013", "2014", "2015",
  "2016", "2017"), to = c(colors)))
colors <- df$colors

# projection
map.crs <- CRS("+proj=longlat +lat_1=35 +lat_2=45 +lat_0=40 +lon_0=-77 +x_0=0
  +y_0=0 +datum=NAD83 +no_defs +ellps=GRS80 +towgs84=0,0,0")

# data.frame to sp object
coordinates(df) <- ~lon + lat
df@proj4string <- map.crs
df <- spTransform(df, map.crs)

# plot
par(fig = c(0, 1, 0, 1))
plot(coast, xlim = c(-76, -73), ylim = c(35, 40.5), col = "grey")
plot(df, pch = 16, col = colors, add = T, cex = 2.5)

occur <- SOE.data[SOE.data$Var == "Southern Kingfish observer sightings",
  ]$Value
time <- SOE.data[SOE.data$Var == "Southern Kingfish observer sightings",
  ]$Time

ts <- zoo(occur, time)
par(fig = c(0.5, 1, 0.1, 0.5), new = T, bty = "l", mar = c(5,
  6, 3, 1))
barplot(occur, time, col = matlab.like2(8), xlab = c("Time"),
  ylab = "S. Kingfish Occurrence, n", cex.lab = 1, las = 1,
  cex.
is = 1)

is(1, at = seq(1250, 18500, length.out = 8), labels = c("2010",
  "2011", "2012", "2013", "2014", "2015", "2016", "2017"),
  cex.
is = 1)

```

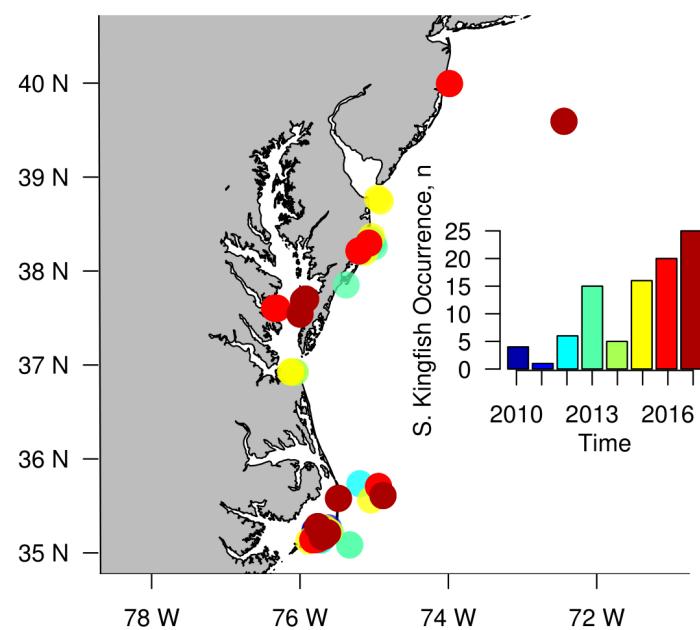


Figure 21.1: Verified records of Southern Kingfish occurrence in the Mid-Atlantic.

Chapter 22

Habitat Occupancy Models

Description: Habitat Occupancy

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018), State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Database pull with analysis; Extensive analysis; not yet published; Published methods

Contributor(s): Kevin Friedland

Data steward: Kevin Friedland, kevin.friedland@noaa.gov

Point of contact: Kevin Friedland, kevin.friedland@noaa.gov

Public availability statement: Source data are available upon request (see Survdat, CHL/PP, and Data Sources below for more information). Model-derived time series are available here.

22.1 Methods

Habitat area with a probability of occupancy greater than 0.5 was modeled for many species throughout the Northeast Large Marine Ecosystem (NE-LME) using random forest decision tree models.

22.1.1 Data sources

Models were parameterized using a suite of static and dynamic predictor variables, with occurrence and catch per unit effort (CPUE) of species from spring and fall Northeast Fisheries Science Center (NEFSC) bottom trawl surveys (BTS) serving as response variables. Sources of variables used in the analyses are described below.

Table 22.1: Habitat descriptors used in model parameterization.

Variables	Notes
Namera_vrm	Vector Ruggedness Measure (VRM) measures terrain ruggedness
Prcurv (2 km, 10 km, and 20 km)	Benthic profile curvature at 2km, 10km and 20 km spatial scales
Rugosity	A measure of small-scale variations of amplitude in the height of
seabedforms	Seabed topography as measured by a combination of seabed posi-
Slp (2 km, 10 km, and 20 km)	Benthic slope at 2km, 10km and 20km spatial scales.
Slpslp (2 km, 10 km, and 20 km)	Benthic slope of slope at 2km, 10km and 20km spatial scales
soft_sed	Soft-sediments is based on grain size distribution from the USGS
Vort (fall - fa; spring - sp; summer - su; winter - wi)	Benthic current vorticity at a 1/6 degree (approx. 19 km) spatial

22.1.1.1 Station depth

The NEFSC BTS data included depth observations made concurrently with trawls at each station. Station depth was a static variable for these analyses.

22.1.1.2 Ocean temperature and salinity

Sea surface and bottom water temperature and salinity measurements were included as dynamic predictor variables in the model, and were collected using Conductivity/Temperature/Depth (CTD) instruments. Ocean temperature and salinity measurements had the highest temporal coverage during the spring (February-April) and fall (September-November) months. Station salinity data were available between 1992-2016.

22.1.1.3 Habitat descriptors

A variety of benthic habitat descriptors were incorporated as predictor variables in occupancy models (Table 22.1). The majority of these parameters are based on depth (e.g. *BPI*, *VRM*, *Prcury*, *rugosity*, *seabedforms*, *slp*, and *slpslp*). The vorticity variable is based on current estimates, and the variable *soft_sed* based on sediment grain size.

22.1.1.4 Zooplankton

Zooplankton data are acquired through the NEFSC Ecosystem Monitoring Program (“EcoMon”). For more information regarding the collection process for these data, see Kane (2007), Kane (2011), and Morse et al. (2017). The bio-volume of the 18 most abundant zooplankton taxa were considered as potential predictor variables.

22.1.1.5 Remote sensing data

Both chlorophyll concentration and sea surface temperature (SST) from remote sensing sources were incorporated as static predictor variables in the model. During the period of 1997-2016, chlorophyll concentrations were derived from observations made by the Sea-viewing Wide Field of View Sensor (SeaWiFS), Moderate Resolution Imaging Spectroradiometer (MODIS-Aqua), Medium Resolution Imaging Spectrometer (MERIS), and Visible and Infrared Imaging/Radiometer Suite (VIIRS).

22.1.2 Data processing

22.1.2.1 Zooplankton

Missing values in the EcoMon time series were addressed by summing data over five-year time steps for each seasonal time frame and interpolating a complete field using ordinary kriging. Missing values necessitated interpolation for spring data in 1989, 1990, 1991, and 1994. The same was true of the fall data for 1989, 1990, and 1992.

22.1.2.2 Remote sensing data

An overlapping time series of observations from the four sensors listed above was created using a bio-optical model inversion algorithm (Maritorena et al. 2010). Monthly SST data were derived from MODIS-Terra sensor data (available here).

22.1.2.3 Ocean temperature and salinity

Date of collection corrections for ocean temperature data were developed using linear regressions for the spring and fall time frames; standardizing to collection dates of April 3 and October 11 for spring and fall. No correction was performed for salinity data. Annual data for ocean temperature and salinity were combined with climatology by season through an optimal interpolation approach. Specifically, mean bottom temperature or salinity was calculated by year and season on a 0.5° grid across the ecosystem, and data from grid cells with >80% temporal coverage were used to calculate a final seasonal mean. Annual seasonal means were then used to calculate combined anomalies for seasonal surface and bottom climatologies.

An annual field was then estimated using raw data observations for a year, season, and depth using universal kriging (Hiemstra et al. 2008), with depth included as a covariate (on a standard 0.1° grid). This field was then combined with the climatology anomaly field and adjusted by the annual mean using the variance field from kriging as the basis for a weighted mean between the

two. The variance field was divided into quartiles with the lowest quartile assigned a weighting of 4:1 between the annual and climatology values. The optimally interpolated field at these locations was therefore skewed towards the annual data, reflecting their proximity to actual data locations and associated low kriging variance. The highest kriging variance quartile (1:1) reflected less information from the annual field and more from the climatology.

22.1.3 Data analysis

22.1.3.1 Occupancy models

Prior to fitting the occupancy models, predictor variables were tested for multicollinearity and removed if found to be correlated. The final model variables were then chosen utilizing a model selection process as shown by Murphy, Evans, and Storfer (2010) and implemented with the R package `rfUtilities` (Evans and Murphy 2018). Occupancy models were then fit as two-factor classification models (absence as 0 and presence as 1) using the `randomForest` R package (Liaw and Wiener 2002).

22.1.3.2 Selection criteria and variable importance

The `irr` R package (Gamer, Lemon, and Singh 2012) was used to calculate Area Under the ROC Curve (AUC) and Cohen's Kappa for assessing accuracy of occupancy habitat models. Variable importance was assessed by plotting the occurrence of a variable as a root variable versus the mean minimum node depth for the variable (Paluszynska and Biecek 2017), as well as by plotting the Gini index decrease versus accuracy decrease.

22.1.4 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mfrow = c(2, 1), mar = c(0, 0, 0, 0), oma = c(3.5,
```

```

5, 2, 4))

soe.plot(SOE.data, "Time", "sumflo spring habitat occupancy",
          stacked = "A", rel.y.num = 1.1, scale.axis = 10^3, end.start = 2007,
          full.trend = F, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "sumflo fall habitat occupancy", stacked = "B",
          rel.y.num = 1.1, scale.axis = 10^3, end.start = 2007, full.trend = F,
          cex.stack = 1.5)

soe.stack.axis("Year", expression("Habitat Area, 10^3 * km^2"),
               y.line = 2.5)

```

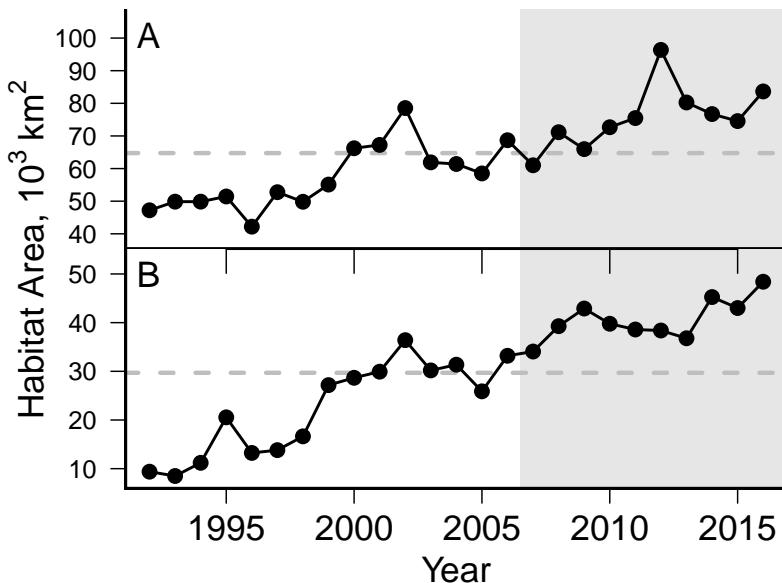


Figure 22.1: Summer flounder spring (A) and fall (B) occupancy habitat area in the Northeast Large Marine Ecosystem.

Chapter 23

Fish Productivity Indicator

Description: Groundfish productivity estimated as the ratio of small fish to large fish

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull with analysis; Published methods

Contributor(s): Charles Perretti

Data steward: Charles Perretti, charles.perretti@noaa.gov

Point of contact: Charles Perretti, charles.perretti@noaa.gov

Public availability statement: Source data are available upon request.

23.1 Methods

23.1.1 Data sources

Survey data from the Northeast Fisheries Science Center (NEFSC) trawl database. These data in their derived form are available through Survdat.

23.1.2 Data extraction

Data were extracted from Survdat.

23.1.3 Data analysis

We defined size thresholds separating small and large fish for each species based on the 20th percentile of the length distribution across all years. This threshold was then used to calculate a small and large fish index (numbers below and above the threshold, respectively) each year. Although the length percentile corresponding to age-1 fish will vary with species, we use the 20th percentile as an approximation. Biomass was calculated using length-weight relationships directly from the survey data. Following Wigley, McBride, and McHugh (2003), the length-weight relationship was modeled as

$$\ln W = \ln a + b \ln L$$

where W is weight (kg), L is length (cm), and a and b are parameters fit via linear regression. The ratio of small fish numbers of the following year to larger fish biomass in the current year was used as the index of recruitment success. The fall and spring recruitment success anomalies were averaged to provide an annual index of recruitment success.

Further details of methods described in Perretti et al. (2017a).

23.1.4 Data processing

Productivity data were formatted for inclusion in the `ecodata` R package using the following R code.

```
# processing small fish per large fish biomass indicator

library(dplyr)
library(tidyr)
library(ggplot2)

raw.dir <- here::here("data-raw")

load(file.path(raw.dir, "dat_spec_rec_forSOE.Rdata"))
load(file.path(raw.dir, "dat_spec_rec_epu_forSOE.Rdata"))

# Select and rename
epu_rec_anom <- dat_spec_rec_epu_forSOE %>% dplyr::select(Time,
  EPU = Region, Value, Units, -Source, Var)

# Select, rename, and bind
productivity_anomaly <- dat_spec_rec_forSOE %>% dplyr::select(-Source) %>%
  mutate(EPU = "All", Var = paste("NE LME", Var)) %>% rbind(.,
  epu_rec_anom) %>% as.data.frame()
```

```
usethis::use_data(productivity_anomaly, overwrite = TRUE)
```

23.1.5 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

##### Functions for plotting #####
library(ggplot2)
library(rpart)
library(dplyr)
library(stringr)

all_epu <- SOE.data %>% filter(str_detect(Var, "All EPU productivity"))

all_epu$Var <- str_trim(str_replace(all_epu$Var, "All EPU productivity",
                                     ""))

# Adjust plot properties
adjustAxes <- ggplot2::theme(axis.title = element_text(size = 18),
                             axis.text = element_text(size = 15), plot.title = element_text(size = 20))

ggplot <- function(...) {
  ggplot2::ggplot(...) + ggplot2::theme_bw() + adjustAxes
}

# Plot stacked bar with cpts for single var
plot_stackbarcpts_single <- function(YEAR, var2bar, x, xlab,
                                       ylab, titl, file_suffix, leg_font_size = 10, remove_leg = FALSE,
                                       leg_ncol = 1, wcpts = TRUE, wdashed = TRUE, height = 5.5,
                                       width = 8) {

  dat2bar <- data.frame(YEAR, var2bar, x)

  dat2plot <- dat2bar %>% tidyverse::gather(variable, value, -YEAR,
                                              -var2bar) %>% dplyr::mutate(var2bar = gsub(pattern = "_",
                                                replacement = " ", x = var2bar), var2bar = gsub(pattern = "Atl.",
```

```

replacement = "", x = var2bar), var2bar = gsub(pattern = "Atl",
replacement = "", x = var2bar), var2bar = gsub(pattern = "NS and combined",
replacement = "", x = var2bar), var2bar = gsub(pattern = "YT",
replacement = "Yellowtail", x = var2bar), var2bar = gsub(pattern = " GoM",
replacement = " GOM", x = var2bar))

p <- ggplot(dat2plot, aes(x = YEAR)) + geom_bar(data = dat2plot %>%
  filter(value > 0), aes(y = value, fill = var2bar), stat = "identity") +
  geom_bar(data = dat2plot %>% filter(value < 0), aes(y = value,
  fill = var2bar), stat = "identity") + geom_hline(size = 0.3,
aes(yintercept = 0)) + xlab(xlab) + ylab(ylab) + ggtitle(titl) +
guides(fill = guide_legend(ncol = leg_ncoll)) + theme(axis.title = element_text(
axis.text = element_text(size = 15), plot.title = element_text(size = 20),
legend.text = element_text(size = leg_font_size), legend.title = element_blank))

if (remove_leg)
  p <- p + theme(legend.position = "none")

print(p)

# ggsave(plot = p, filename = './productivity_all.eps', width
# = width, height = height)
}

# Plot stacked bars
plot_stackbarcpts <- function(YEAR, var2bar, top, mid, bot, top_lab,
mid_lab, bot_lab, xlab = "", ylab = "", titl = "") {

  dat2bar <- data.frame(YEAR, var2bar, top, mid, bot)

  dat2plot <- dat2bar %>% tidyverse::gather(variable, value, -YEAR,
  -var2bar) %>% dplyr::mutate(variable = ifelse(variable ==
  "top", top_lab, ifelse(variable == "mid", mid_lab, bot_lab)))

  dat2plot$variable <- factor(dat2plot$variable, levels = c(top_lab,
  mid_lab, bot_lab))

  p <- ggplot(dat2plot, aes(x = YEAR)) + geom_bar(data = dat2plot %>%
    filter(value > 0), aes(y = value, fill = var2bar), stat = "identity") +
    geom_bar(data = dat2plot %>% filter(value < 0), aes(y = value,
    fill = var2bar), stat = "identity") + facet_wrap(~variable,
    ncol = 1) + geom_hline(size = 0.3, aes(yintercept = 0)) +
    xlab(xlab) + ylab(ylab) + ggtitle(titl) + guides(fill = guide_legend(ncol = 1))
}

```

```

    theme(axis.title = element_text(size = 16), axis.text = element_text(size = 15),
          plot.title = element_text(size = 20), strip.text = element_text(size = 15),
          legend.title = element_blank())

  print(p)
}

# Recruit per spawner (all stocks in one figure)
plot_stackbarcpts_single(YEAR = all_epu$Time, var2bar = all_epu$Var,
  x = all_epu$value, titl = "", xlab = "", ylab = "Small fish per large fish biomass (anomaly)",
  height = 7, width = 9)

```

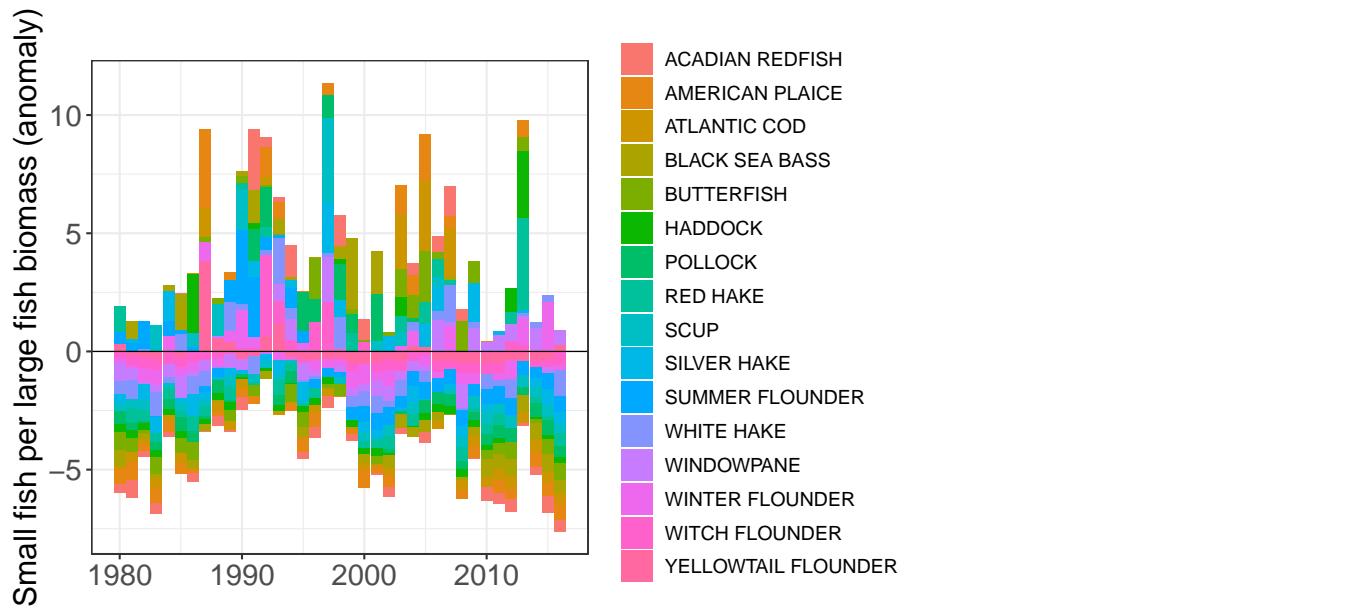


Figure 23.1: Groundfish productivity across all stocks in the Northeast Large Marine Ecosystem.

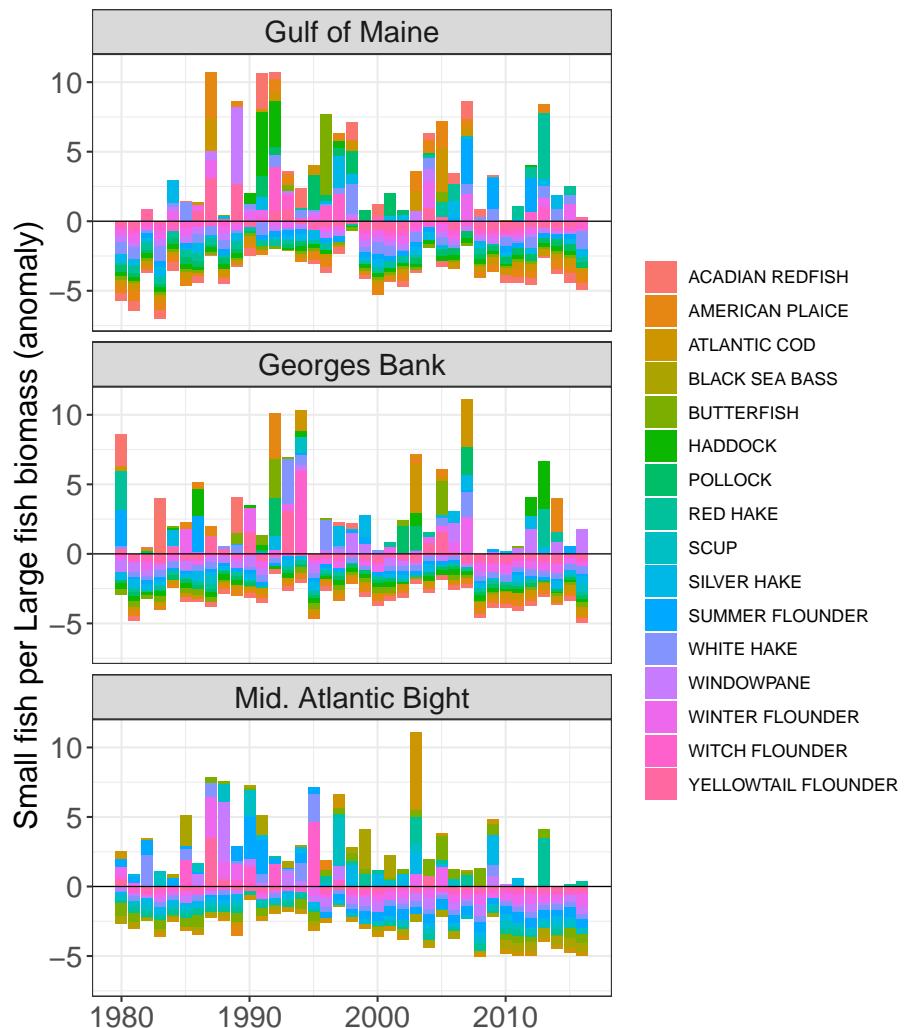


Figure 23.2: Groundfish productivity visualized by EPU.

Chapter 24

Recreational Fishing Indicators

Description: A variety of indicators derived from MRIP Recreational Fisheries Statistics, including total recreational catch, total angler trips by region, annual diversity of recreational fleet effort, and annual diversity of managed species.

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull with analysis

Contributor(s): Geret DePiper, Scott Steinbeck

Data steward: Geret DePiper, geret.depiper@noaa.gov

Point of contact: Geret DePiper, geret.depiper@noaa.gov

Public availability statement: Data sets are publicly available (see Data Sources below).

24.1 Methods

We used total recreational harvest as an indicator of seafood production and total recreational trips and total recreational anglers as proxies for recreational value generated from the Mid-Atlantic and New England regions respectively. We estimated both recreational catch diversity in species managed by the Fisheries Management Councils; Mid-Atlantic (MAFMC), New England (NEFMC) and Atlantic States (ASFMC) and fleet effort diversity using the effective Shannon index.

24.1.1 Data sources

All recreational fishing indicator data, including number of recreationally harvested fish, number of angler trips, and number of anglers, were downloaded from the Marine Recreational Information Program MRIP Recreational Fisheries Statistics Queries portal. Relevant metadata including information regarding data methodology updates are available at the query site. Note that 2017 data were considered preliminary at the time of the data pull.

Data sets were queried by region on the MRIP site, and for the purposes of the State of the Ecosystem reports, the “NORTH ATLANTIC” and “MID-ATLANTIC” regions were mapped to the New England and Mid-Atlantic report versions respectively. All query pages are accessible through the MRIP Recreational Fisheries Statistics site.

The number of recreationally harvested fish was found by selecting “TOTAL HARVEST (A + B1)” on the Catch Time Series Query page. Catch diversity estimates were also derived from the total catch time series (see below). Species included in the diversity of catch analysis included American eel *Anguilla rostrata*, Atlantic cod *Gadus morhua*, Atlantic mackerel *Scomber scombrus*, Atlantic sturgeon *Acipenser oxyrinchus*, black drum *Pogonias cromis*, black sea bass *Centropristes striata*, bluefish *Pomatomus saltatrix*, cobia *Rachycentron canadum*, haddock *Melanogrammus aeglefinus*, pollock *Pollachius virens*, red drum *Sciaenops ocellatus*, scup *Stenotomus chrysops*, Spanish mackerel *Scomberomorus maculatus*, spiny dogfish *Squalus acanthias*, spot *Leiostomus xanthurus*, spotted seatrout *Cynoscion nebulosus*, striped bass *Morone saxatilis*, summer flounder *Paralichthys dentatus*, tautog *Tautoga onitis*, tilefish *Lopholatilus chamaeleonticeps*, weakfish *Cynoscion regalis*, winter flounder *Pseudopleuronectes americanus*, and “All Other Species”.

Angler trips (listed as “TOTAL” trips) were pulled from the MRIP Effort Time Series Query page, and included data from 1981 - 2019. Time series of recreational fleet effort diversity were calculated from this data set (see below). The number of anglers was total number of anglers from the Marine Recreational Fishery Statistics Survey (MRFSS) Participation Time Series Query, and includes data from 1981 - 2016.

24.1.2 Data analysis

Recreational fleet effort diversity

```
REC_DATA <- read.csv("X:/gdepiper/ESR2018/SOE/Data/Rec_Days_Fished_2018.csv",
  as.is = TRUE)

REC_DATA$P_Shore <- -(as.numeric(gsub("", "", "\\\"\\\"", REC_DATA$Shore)) / as.numeric(gsub(
  "", "\\\"\\\"", REC_DATA$All_Modes))) * log(as.numeric(gsub("", "\\\"\\\"", REC_DATA$All_Modes)))
```

```

    "", "\\"\"", REC_DATA$Shore))/as.numeric(gsub("", "", "\\"\"",  

REC_DATA$All_Modes)))  
  

REC_DATA$P_Private <- -(as.numeric(gsub("", "", "\\"\"", REC_DATA$Private_Rental))/as.numeric(gsub(  

    "", "\\"\"", REC_DATA$All_Modes))) * log(as.numeric(gsub("",  

    "", "\\"\"", REC_DATA$Private_Rental))/as.numeric(gsub("",  

    "", "\\"\"", REC_DATA$All_Modes)))  
  

REC_DATA$P_Party <- -(as.numeric(gsub("", "", "\\"\"", REC_DATA$Party_Charter))/as.numeric(gsub(  

    "", "\\"\"", REC_DATA$All_Modes))) * log(as.numeric(gsub("",  

    "", "\\"\"", REC_DATA$Party_Charter))/as.numeric(gsub("",  

    "", "\\"\"", REC_DATA$All_Modes)))  
  

REC_DATA$Value <- exp(REC_DATA$P_Shore + REC_DATA$P_Private +  

REC_DATA$P_Party)  
  

REC_DATA$Region[REC_DATA$Region == "\"MID-ATLANTIC\"] <- "MA"  

REC_DATA$Region[REC_DATA$Region == "\"NORTH ATLANTIC\"] <- "NE"  
  

E_SHANNON <- subset(REC_DATA, select = c("Time", "Region", "Value"))  

E_SHANNON$Units <- "Effective Shannon"  

E_SHANNON$Var <- "Recreational fleet effort diversity across modes"  

E_SHANNON$Source <- "MRIP effort time series, processed to generate diversity measure."

```

Recreational catch diversity

```

REC_CATCH <- read.csv("X:/gdepiper/ESR2018/SOE/Data/Rec_Species_Quantity_2018.csv",  

  as.is = TRUE)  
  

REC_CATCH$Value <- as.numeric(gsub(", ", "", REC_CATCH$Value))  

TOT_REC_CATCH <- aggregate(Value ~ Time + Region, data = REC_CATCH,  

  FUN = sum)  

names(TOT_REC_CATCH) <- c("Time", "Region", "Tot_Catch")  

REC_CATCH <- merge(REC_CATCH, TOT_REC_CATCH, by = c("Time", "Region"))  

REC_CATCH$P_Catch <- -(REC_CATCH$Value/REC_CATCH$Tot_Catch *  

  log(REC_CATCH$Value/REC_CATCH$Tot_Catch))  

REC_CATCH <- aggregate(P_Catch ~ Time + Region, data = REC_CATCH,  

  FUN = sum)  

REC_CATCH$Value <- exp(REC_CATCH$P_Catch)  

REC_CATCH <- subset(REC_CATCH, select = c("Time", "Region", "Value"))  

REC_CATCH$Region[REC_CATCH$Region == "MID-ATLANTIC"] <- "MA"  

REC_CATCH$Region[REC_CATCH$Region == "NORTH ATLANTIC"] <- "NE"  

REC_CATCH$Units <- "Effective Shannon"  

REC_CATCH$Var <- "Recreational Diversity of Catch"

```

```

## Species include: American Eel, Atlantic Cod, Atlantic
## Mackerel, Atlantic Sturgeon, Black Drum, Black Sea Bass,
## Bluefish, Cobia, Haddock, Pollock, Red Drum, Scup, Spanish
## Mackerel, Spiny Dogfish, Spot, Spotted Seatrout, Striped
## Bass, Summer Flounder, Tautog, Tilefish, Weakfish, Winter
## Flounder, and All Other Species.

REC_CATCH$Source <- "MRIP catch time series."

```

24.1.3 Data processing

Recreational fishing indicators were formatted for inclusion in the `ecodata` R package using the following code.

```

# Processing for recreational fishing indicators

library(dplyr)
library(tidyr)
library(stringr)

raw.dir <- here::here("data-raw")

get_rec <- function(save_clean = F) {

  files = list.files(raw.dir, pattern = "REC_HARVEST|Rec_angler|Rec_Species")
  for (i in 1:length(files)) assign(files[i], read.csv(file.path(raw.dir,
    files[i])))

  recdat <- NULL
  for (i in ls()) {
    if (stringr::str_detect(i, "REC_|Rec_")) {
      d <- get(i) %>% dplyr::select(Time, EPU = Region,
        Value, Units, Var)

      assign("recdat", rbind(recdat, d))
    }
  }

  recdat <- recdat %>% filter(!is.na(EPU))

  if (save_clean) {
    usethis::use_data(recdat, overwrite = T)
  } else {

```

```

        return(recdat)
    }
}

get_rec(save_clean = T)

```

24.1.4 Plotting

```

recdat <- ecodata::recdat %>%
  filter(EPU == "NE") %>%
  group_by(Var) %>%
  mutate(hline = mean(Value))

ylim_re <- c(5e6, 30e6)
ylim_rd <- c(1.75,2.75)
ylim_ra <- c(0, 2e6)

series.col <- "black"

rec_div <- recdat %>%
  filter(Var == "Recreational fleet effort diversity across modes") %>%
  ggplot() +
  #Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min , xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  #label
  # annotate("text",
  #           x = label_loc[label_loc$Var == "Recreational fleet effort diversity across modes",],
  #           y = label_loc[label_loc$Var == "Recreational fleet effort diversity across modes",],
  #           label = label_loc[label_loc$Var == "Recreational fleet effort diversity across modes",
  #                             size = letter_size]+
  geom_gls(aes(x = Time, y = Value,
                group = Var),
            alpha = trend.alpha, size = trend.size) +
  geom_line(aes(x = Time, y = Value, color = Var), size = lwd) +
  geom_point(aes(x = Time, y = Value, color = Var), size = pex) +
  ylim(ylim_rd) +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  scale_color_manual(values = series.col, aesthetics = "color")+
  guides(color = FALSE) +
  ggtitle("Rec. fleet effort diversity")+
  ylab(expression("Effective Shannon")) +
  xlab("")+

```

```

geom_hline(aes(yintercept = hline,
                color = Var),
            size = hline.size,
            alpha = hline.alpha,
            linetype = hline.lty) +
theme_ts()

rec_divCatch <- recdat %>%
  filter(Var == "Recreational Diversity of Catch") %>%
  ggplot() +
#Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min , xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
#  annotate("text",
#           x = label_loc[label_loc$Var == "Recreational anglers",]$xloc,
#           y = label_loc[label_loc$Var == "Recreational anglers",]$yloc,
#           label = label_loc[label_loc$Var == "Recreational anglers",]$labels,
#           size = letter_size)+

  geom_gls(aes(x = Time, y = Value,
                group = Var),
            alpha = trend.alpha, size = trend.size) +
  geom_line(aes(x = Time, y = Value, color = Var), size = lwd) +
  geom_point(aes(x = Time, y = Value, color = Var), size = pcex) +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  scale_color_manual(values = series.col, aesthetics = "color")+
  guides(color = FALSE) +
  ggtitle("Rec. diversity of catch")+
  ylab(expression("Effective Shannon")) +
  xlab("Time")+
  geom_hline(aes(yintercept = hline,
                color = Var),
            size = hline.size,
            alpha = hline.alpha,
            linetype = hline.lty) +
theme_ts()

cowplot::plot_grid(#rec_effort,
                   rec_div,
                   #rec_anglers,
                   rec_divCatch,
                   ncol = 1,
                   align = "hv") +

```

```
theme(plot.margin = unit(c(0.1, 0, 0, 0), "cm"))
```

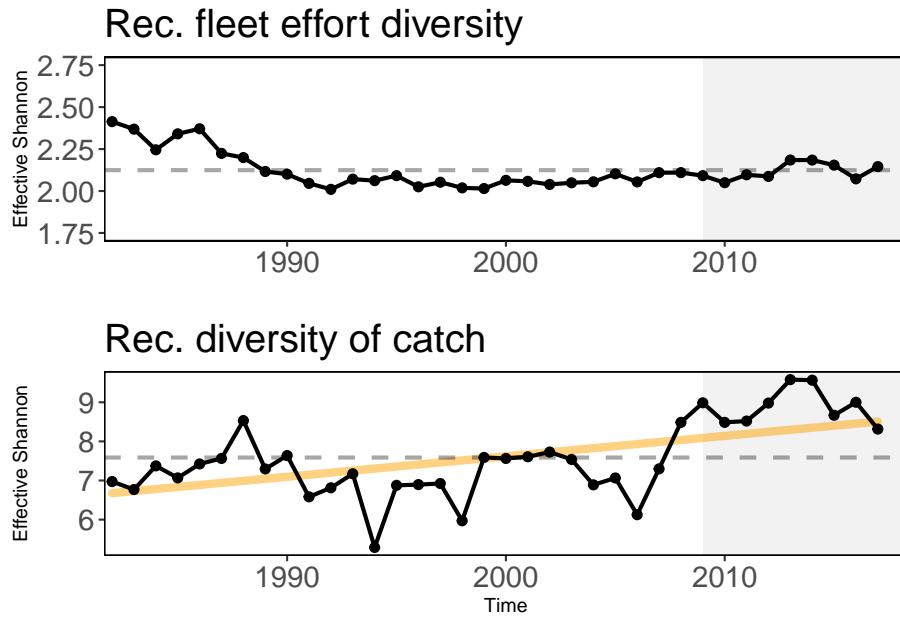


Figure 24.1: Recreational effort diversity and diversity of recreational catch in New England.

```
landings_rec <- ecodata::recdat %>%
  filter(EPU == "NE",
         Var == "Recreational Seafood") %>%
  mutate(hline = mean(Value))

series.col <- "black"

ggplot(data = landings_rec) +
  #Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min , xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  geom_gls(aes(x = Time, y = Value,
                group = Var),
            alpha = trend.alpha, size = trend.size) +
  geom_line(aes(x = Time, y = Value, color = Var), size = lwd) +
```

```

geom_point(aes(x = Time, y = Value, color = Var), size = pcex) +
  ggtitle("NE Recreational seafood harvest") +
  scale_y_continuous(labels = function(l){trans = 1 / 1000000}) +
  scale_x_continuous(breaks = seq(1985, 2015, by = 5), expand = c(0.01, 0.01)) +
  scale_color_manual(values = series.col, aesthetics = "color") +
  guides(color = FALSE) +
  ylab(expression("Fish caught (10^6*n)")) +
  geom_hline(aes(yintercept = hline,
                 color = Var),
              size = hline.size,
              alpha = hline.alpha,
              linetype = hline.lty) +
  theme_ts()

```

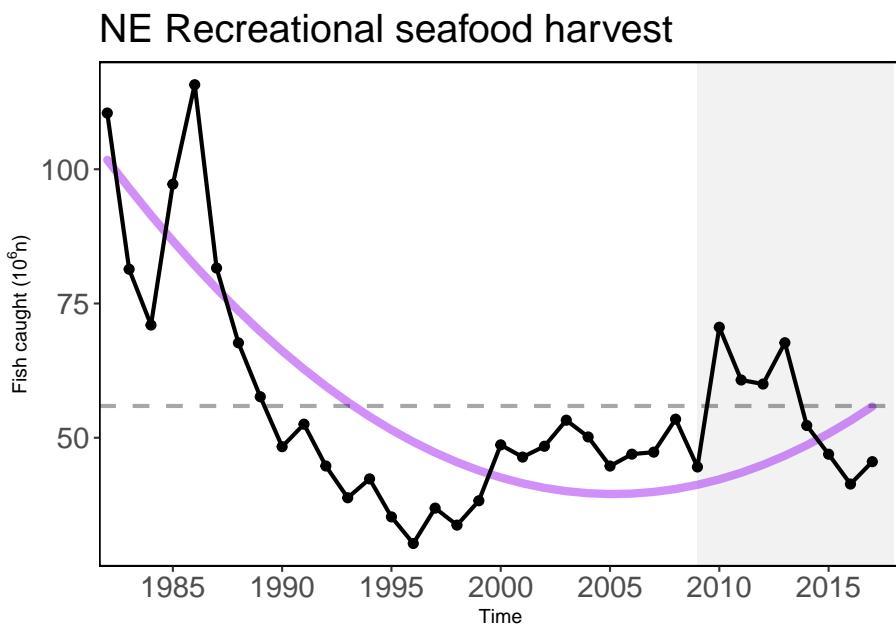


Figure 24.2: Total recreational seafood harvest in New England.

```

recdat <- ecodata::recdat %>%
  filter(EPU == "NE") %>%
  group_by(Var) %>%
  mutate(hline = mean(Value))

```

```

ylim_re <- c(5e6, 30e6)
ylim_rd <- c(1.75, 2.75)
ylim_ra <- c(0, 2e6)

#Create dataframe for label locations
# label_loc <- data.frame(xloc = min(recdat$Time)+0.3,
#                           yloc = c(ylim_re[2]*0.975,
#                                   ylim_rd[2]*0.975,
#                                   ylim_ra[2]*0.975),
#                           labels = LETTERS[1:3],
#                           Var = c("Recreational Effort",
#                                  "Recreational fleet effort diversity across modes",
#                                  "Recreational anglers"))

series.col <- "black"
#x.shade.min <- max(recdat$Time, na.rm = T) - 9
#x.shade.max <- max(recdat$Time, na.rm = T)

series.col <- "black"

rec_effort <- recdat %>%
  filter(Var == "Recreational Effort") %>%
  ggplot() +
  #Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min , xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  #label
  # annotate("text",
  #           x = label_loc[label_loc$Var == "Recreational Effort",]$xloc,
  #           y = label_loc[label_loc$Var == "Recreational Effort",]$yloc,
  #           label = label_loc[label_loc$Var == "Recreational Effort",]$labels,
  #           size = letter_size) +
  geom_gls(aes(x = Time, y = Value,
                group = Var),
            alpha = trend.alpha, size = trend.size) +
  geom_line(aes(x = Time, y = Value, color = Var), size = lwd) +
  geom_point(aes(x = Time, y = Value, color = Var), size = pex) +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  scale_y_continuous(labels = function(l){trans = l / 1000000}, limits = ylim_re) +
  scale_color_manual(values = series.col, aesthetics = "color") +
  guides(color = FALSE) +
  ggtitle("Recreational effort") +
  ylab(expression("Days fished (10^6* days)"))

```

```

xlab("")+
  geom_hline(aes(yintercept = hline,
                  color = Var),
              size = hline.size,
              alpha = hline.alpha,
              linetype = hline.lty) +
  theme_ts()

rec_anglers <- recdat %>%
  filter(Var == "Recreational anglers") %>%
  ggplot() +
  #Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha,
           xmin = x.shade.min , xmax = x.shade.max,
           ymin = -Inf, ymax = Inf) +
  # annotate("text",
  #           x = label_loc[label_loc$Var == "Recreational anglers",]$xloc,
  #           y = label_loc[label_loc$Var == "Recreational anglers",]$yloc,
  #           label = label_loc[label_loc$Var == "Recreational anglers",]$labels,
  #           size = letter_size) +
  geom_gls(aes(x = Time, y = Value,
                group = Var),
            alpha = trend.alpha, size = trend.size) +
  geom_line(aes(x = Time, y = Value, color = Var), size = lwd) +
  geom_point(aes(x = Time, y = Value, color = Var), size = pcex) +
  scale_x_continuous(expand = c(0.01, 0.01)) +
  scale_y_continuous(labels = function(l){trans = l / 1000000}, limits = ylim_ra) +
  scale_color_manual(values = series.col, aesthetics = "color") +
  guides(color = FALSE) +
  ggtitle("Recreational anglers") +
  ylab(expression("Anglers (10^6 * n)")) +
  xlab("Time") +
  geom_hline(aes(yintercept = hline,
                  color = Var),
              size = hline.size,
              alpha = hline.alpha,
              linetype = hline.lty) +
  theme_ts()

cowplot::plot_grid(rec_effort,
                    #rec_div,
                    rec_anglers,
                    #rec_div_catch,

```

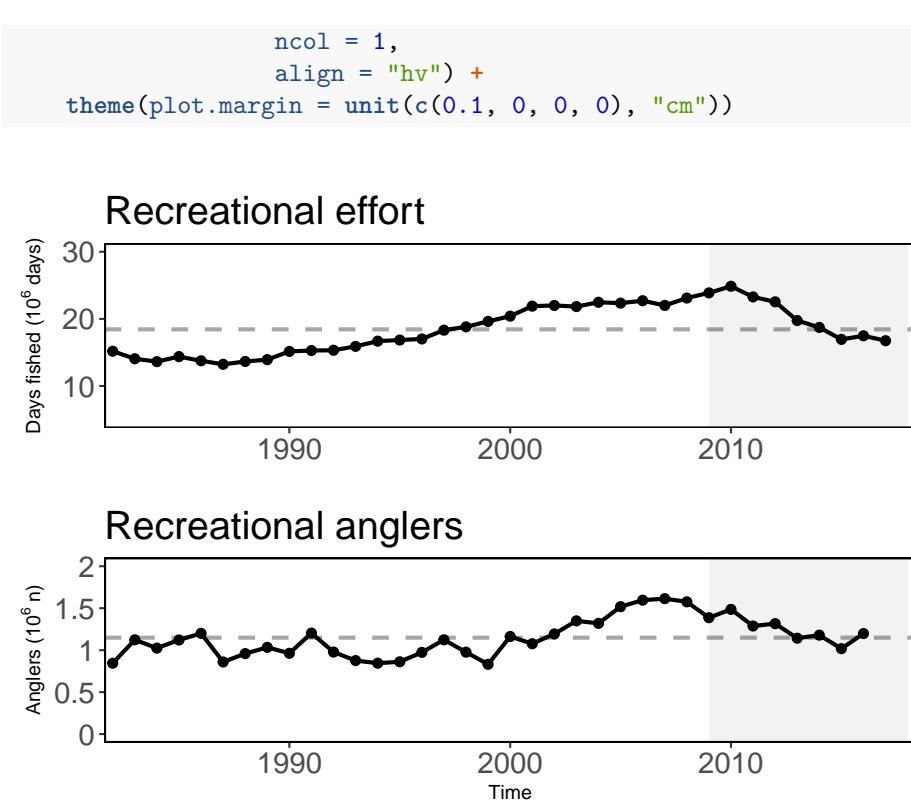


Figure 24.3: Recreational effort and number of recreational anglers in New England.

Chapter 25

Right Whale Abundance

Description: Right Whale

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Synthesis of published information; Published methods

Contributor(s): Christopher D. Orphanides

Data steward: Chris Orphanides, chris.orphanides@noaa.gov

Point of contact: Richard Pace, richard.pace@noaa.gov

Public availability statement: Source data are available from the New England Aquarium upon request. Derived data are available here

25.1 Methods

25.1.1 Data sources

The North Atlantic right whale abundance estimates were taken from a published document (see Pace, Corkeron, and Kraus 2017), except for the most recent 2016 and 2017 estimates. Abundance estimates from 2016 and 2017 were taken from the 2016 National Oceanographic and Atmospheric Administration marine mammal stock assessment (Hayes et al. 2017) and an unpublished 2017 stock assessment.

25.1.2 Data extraction

Data were collected from existing reports and validated by report authors.

25.1.3 Data analysis

Analysis for right whale abundance estimates is provided by Pace, Corkeron, and Kraus (2017), and code can be found in the supplemental materials.

25.1.4 Data processing

Time series of right whale abundance estimates were formatted for inclusion in the `ecodata` R package using the following R code.

```
# Processing for North Atlantic Right Whale data

# See full documentation for these data at
# https://noaa-edab.github.io/tech-doc/right-whale-abundance.html

library(dplyr)
library(tidyr)
library(stringr)

raw.dir <- here::here("data-raw")

get_narw <- function(save_clean = F) {
  narw <- read.csv(file.path(raw.dir, "narw_numbers.csv")) %>%
    gather(., Var, Value, -YEAR) %>% mutate(Var = tolower(paste("right whale abundance", Var)), Units = "n", EPU = "All") %>% mutate(Var = ifelse(str_detect(Var, "median"), "right whale abundance median", ifelse(str_detect(Var, "lcl"), "right whale abundance lcl", "right whale abundance ucl")))) %>%
    dplyr::rename(Time = YEAR)

  if (save_clean) {
    usethis::use_data(narw, overwrite = T)
  } else {
    return(narw)
  }
}
```

25.1.5 Plotting

```
# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")
```

```
# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

par(mar = c(5, 5, 4, 1))
soe.plot(SOE.data, "Time", "right_whale_median", rel.y.num = 1.2,
  full.trend = F, lwd = 1.5, point.cex = 1, end.start = 2007,
  x.label = "Year", y.label = "Abundance, n", rel.y.text = 1)

lw_CI <- SOE.data[Var == "right_whale_lower_95", list(Time, Value)]
up_CI <- SOE.data[Var == "right_whale_upper_95", list(Time, Value)]
points(lw_CI, type = "l", lty = 2, col = adjustcolor("darkorange",
  0.6), lwd = 2.5)
points(up_CI, type = "l", lty = 2, col = adjustcolor("darkorange",
  0.6), lwd = 2.5)
```

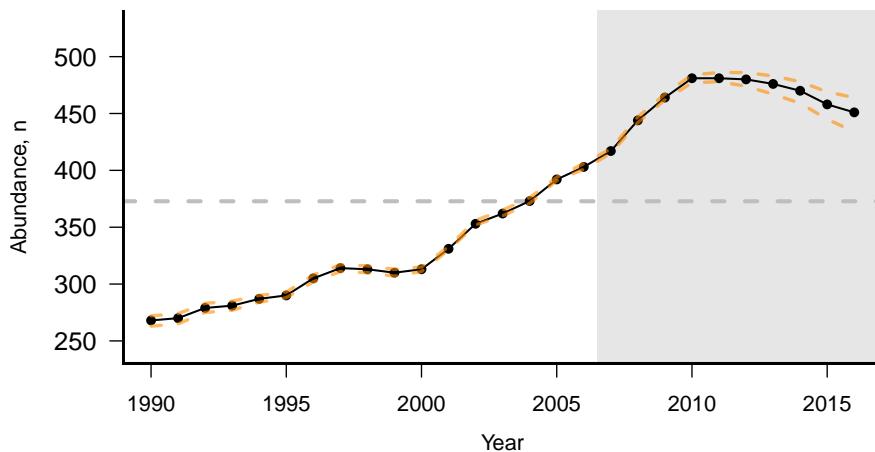


Figure 25.1: North Atlantic right whale population estimates shown with 95% credible intervals.

Chapter 26

Seabird diet and productivity

Description: Common tern annual diet and productivity at seven Gulf of Maine colonies managed by the National Audubon Society's Seabird Restoration Program

Indicator category: Published method

Found in: State of the Ecosystem - New England (2019)

Contributor(s): Don Lyons, Steve Kress, Paula Shannon, Sue Schubel

Data steward: Don Lyons, dlyons@audubon.org

Point of contact: Don Lyons, dlyons@audubon.org

Public availability statement: Please email dlyons@audubon.org for further information and queries on this indicator source data.

26.1 Methods

Chick diet

Common tern (*Sterna hirundo*) chick diet was quantified at each of the seven nesting sites (Fig. 26.2) by observing chick provisioning from portable observation blinds. The locations of observation blinds within each site were chosen to maximize the number of visible nests, and provisioning observations took place between mid-June and early August annually. Observations of chick diet were made during one or two, three to four hour periods throughout the day, but typically proceed according to nest activity levels (moreso in the morning

hours). Observations began with chicks as soon as they hatched, and continue until the chicks fledged or died.

Most common tern prey species were identifiable to the species level due to distinct size, color and shape. However, when identification was not possible or was unclear, prey species were listed as “unknown” or “unknown fish”. More detailed methods can be found in Hall, Kress, and Griffin (2000).

Nest productivity

Common tern nest productivity, in terms of the number of fledged chicks per nest, was collected annually from fenced enclosures at island nesting sites (known as “productivity plots”). Newly hatched chicks within these enclosures were weighed, marked or banded, and observed until fledging, death, or until a 15 day period had passed when chicks were assumed to have fledged. Productivity was also quantified from observer blinds for nests outside of the productivity plots where chicks were marked for identification. More detailed methods for quantifying nest productivity can be found in Hall and Kress (2004)

26.1.1 Data sources

Common tern diet and nest productivity data were provided by the National Audubon Society’s Seabird Restoration Program.

26.1.2 Data processing

Diet and productivity data were formatted for inclusion in the `ecodata` R package using the following R code.

```
# Process raw common tern productivity and diet data

# If save == TRUE, processed data are saved as a .Rds file to
# the data directory.

# Function returns a processed data frame containing time
# series of productivity and diet composition for common tern
# at different islands in southern Gulf of Maine. The first
# three letters in the variable 'Var' correspond to the
# island at which sampling occurred, with COTE referring to
# Common Tern. If 'Var' includes productivity, then 'Value'
# is the average number of fledged chicks. Otherwise, if
# 'Var' specifies diet, then 'Value' is equal to the count of
# observed prey type. For example, 'EER COTE Diet Amphipod'
# refers to the number of preyed upon amphipods observed at
# Eastern Egg Rock in a given year.
```

```

library(dplyr)
library(tidyr)

# Get raw data
raw.dir <- here::here("data-raw")

get_commontern <- function(save_clean = F) {

  d <- read.csv(file.path(raw.dir, "Audubon_Common_Tern.csv"))

  # Process
  common_tern <- d %>% filter(Island != "") %>% tidyr::gather(.,
    Var, Value, -Year, -Species, -Island) %>% mutate(Species = ifelse(Var !=
    "Productivity", paste(Species, "Diet"), "COTE")) %>%
    unite(., "Var", c("Island", "Species", "Var"), sep = " ") %>%
    dplyr::rename(Time = Year) %>% mutate(EPU = "GOM", Units = ifelse(stringr::str_detect(Var,
    "Productivity"), "fledged chicks per nest", "N"))

  if (save_clean) {

    usethis::use_data(common_tern, overwrite = T)
  } else {
    return(common_tern)
  }
}
get_commontern(save_clean = T)

```

26.1.3 Data analysis

Raw diet data were used to create time series of mean shannon diversity through time and across study sites using the `vegan` R package (Oksanen et al. 2019). Diet diversity is presented along with nest productivity (+/- 1 SE) below.

```

# Calculating time series of diversity indices using the
# vegan package.
diet_div <- ecodata::common_tern %>% filter(str_detect(Var, "Diet"),
  !str_detect(Var, "Sum")) %>% mutate(Island = word(Var, 1),
  Var = word(Var, 4)) %>% group_by(Island, Time) %>% dplyr::summarise(evenness = diversity(Value),
  shannon = diversity(Value), simpson = diversity(Value, index = "simpson")) %>%
  gather(., Var, Value, -Island, -Time) %>% group_by(Var, Time) %>%
  dplyr::summarize(Value = mean(Value, na.rm = T), sd = sd(Value,
  na.rm = T), n = n()) %>% group_by(Var) %>% mutate(hline = mean(Value,
  na.rm = T))

```

```

aggregate_prod <- ecodata::common_tern %>% filter(!str_detect(Var,
  "Diet|Sum")) %>% mutate(Island = word(Var, 1), Var = word(Var,
  3), Island = plyr::mapvalues(Island, from = c("EER", "JI",
  "MR", "OGI", "PINWR", "SINWR", "STI"), to = c("Eastern Egg Rock",
  "Jenny Island", "Matinicu Rock", "Outer Green Island", "Pond Island",
  "Seal Island", "Stratton Island"))) %>% group_by(Time) %>%
  dplyr::summarise(Mean = mean(Value, na.rm = T), SE = sd(Value,
  na.rm = T)/sqrt(n()), SD = sd(Value, na.rm = T), n = n()) %>%
  mutate(Mean = ifelse(is.na(SE), NA, Mean), se.low = Mean -
  SE, se.high = Mean + SE, hline = mean(Mean, na.rm = T))

prodplot <- aggregate_prod %>% ggplot() + # Highlight last ten years
  annotate("rect", fill = shade.fill, alpha = shade.alpha, xmin = x.shade.min,
  xmax = x.shade.max, ymin = -Inf, ymax = Inf) + geom_line(aes(x = Time,
  y = Mean), size = lwd - 0.75) + geom_point(aes(x = Time,
  y = Mean), size = pcox - 0.75) + geom_gls(aes(x = Time, y = Mean)) +
  geom_errorbar(aes(x = Time, ymin = se.low, ymax = se.high),
  width = 0.25) + scale_x_continuous(expand = c(0.01, 0.01),
  limits = c(1991, 2018)) + guides(color = FALSE) + ggtitle("Common tern productivity")
  ylab(expression("Fledged chicks per nest")) + xlab("Time") +
  geom_hline(aes(yintercept = hline), color = "black", size = hline.size,
  alpha = hline.alpha, linetype = hline.lty) + labs(tag = "a") +
  theme_ts()

diet_div <- ecodata::common_tern %>% filter(str_detect(Var, "Diet"),
  !str_detect(Var, "Sum")) %>% mutate(Island = word(Var, 1),
  Var = word(Var, 4)) %>% group_by(Island, Time) %>% dplyr::summarise(evenness = diversity(
  shannon = diversity(Value), simpson = diversity(Value, index = "simpson")) %>%
  gather(., Var, Value, -Island, -Time) %>% group_by(Var, Time) %>%
  dplyr::summarize(Value = mean(Value, na.rm = T), sd = sd(Value,
  na.rm = T), n = n()) %>% group_by(Var) %>% mutate(hline = mean(Value,
  na.rm = T))

# evenness <- diet_div %>% filter(Var == 'evenness') %>%
# ggplot(aes(x = Time, y = Value)) + annotate('rect', fill =
# shade.fill, alpha = shade.alpha, xmin = x.shade.min , xmax
# = x.shade.max, ymin = -Inf, ymax = Inf) + geom_line() +
# geom_point() + # geom_gls() + scale_x_continuous(expand =
# c(0.01, 0.01),limits = c(1992,2018)) + ggtitle('Evenness')+
# ylab(expression('Evenness')) + xlab('')+
# geom_hline(aes(yintercept = hline), size = hline.size,
# alpha = hline.alpha, linetype = hline.lty) + theme_ts()

shannon <- diet_div %>% filter(Var == "shannon") %>% ggplot(aes(x = Time,
  y = Value)) + annotate("rect", fill = shade.fill, alpha = shade.alpha,

```

```

xmin = x.shade.min, xmax = x.shade.max, ymin = -Inf, ymax = Inf) +
  geom_line() + geom_point() + # geom_gls() +
scale_x_continuous(expand = c(0.01, 0.01), limits = c(1992, 2018)) +
  ggtitle("Common tern diet diversity") + ylab(expression("Shannon Diversity")) +
  xlab("") + geom_hline(aes(yintercept = hline), size = hline.size,
  alpha = hline.alpha, linetype = hline.lty) + labs(tag = "b") +
  theme_ts()

prodplot + shannon + plot_layout(ncol = 2, widths = c(4, 4))

```

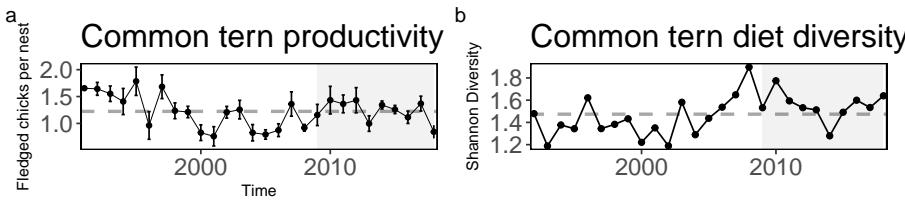


Figure 26.1: a. Mean common tern productivity at nesting sites in Gulf of Maine. Error bars show +/- 1 SE of the mean. b. Shannon diversity of common tern diets observed at nesting sites in Gulf of Maine. Diversity of common tern diets has been predominantly above the long-term mean since 2006.

Along with nest productivity and diet diversity indices, we presented maps of sampling sites in Gulf of Maine and mean prey frequencies across sites. Prey occurring in less than 5% of common tern diets was excluded for visual clarity.

```

# Map subfig Set lat/lon window for maps
xmin = -70.5
xmax = -68.25
ymin = 43
ymax = 44.5
xlims <- c(xmin, xmax)
ylims <- c(ymin, ymax)
island_loc <- data.frame(Island = c("EER", "JI", "MR", "OGI",
  "PINWR", "SINWR", "STI"), Latitude = c(43.861, 43.764, 43.785,
  43.65, 43.739, 43.886, 43.505), Longitude = c(-69.382, -69.909,
  -68.854, -70.124, -69.771, -68.742, -70.312))
islands <- ggplot() + geom_sf(data = new_england, size = map.lwd) +
  geom_point(data = island_loc, aes(x = Longitude, y = Latitude)) +
  geom_label_repel(data = island_loc, aes(x = Longitude, y = Latitude,
  label = Island), nudge_y = -0.1, size = 2.5) + coord_sf(crs = crs,
  xlim = xlims, ylim = ylims) + theme_map() + ggtitle("Common tern study sites") +
  xlab("Longitude") + ylab("Latitude") + labs(tag = "a")

```

```

theme(panel.border = element_rect(colour = "black", fill = NA,
  size = 0.75), legend.key = element_blank(), axis.title = element_text(size = 11),
  strip.background = element_blank(), strip.text = element_text(hjust = 0),
  axis.text = element_text(size = 8))
# inset_states <- new_england %>% filter(STATE_NAME %in%
# c('Maine', 'New Hampshire', 'Massachusetts')) Set lat/lon
# window for maps
xmin2 = -72
xmax2 = -66.25
ymin2 = 42.5
ymax2 = 47.5
xlims2 <- c(xmin2, xmax2)
ylims2 <- c(ymin2, ymax2)
NE <- ggplotGrob(ggplot() + geom_sf(data = new_england, size = map.lwd) +
  coord_sf(crs = crs, xlim = xlims2, ylim = ylims2) + annotate("rect",
  xmin = xmin, xmax = xmax, ymin = ymin, ymax = ymax, color = "black",
  size = 0.1, fill = "transparent") + theme_map() + xlab("") +
  ylab("") + theme(panel.border = element_rect(colour = "black",
  fill = "transparent", size = 0.75), plot.background = element_rect(fill = "transparent",
  colour = NA), legend.key = element_blank(), axis.title = element_text(size = 11),
  strip.background = element_blank(), strip.text = element_text(hjust = 0),
  axis.text = element_blank()))
full_map <- islands + annotation_custom(grob = NE, xmin = -69,
  xmax = -68.25, ymin = 42.9, ymax = 43.6)

# Prey freq stacked bar
prey_freq <- ecodata::common_tern %>% filter(str_detect(Var,
  "Diet"), !str_detect(Var, "Sum")) %>% mutate(Island = word(Var,
  1), Var = word(Var, 4)) %>% group_by(Var, Time) %>% dplyr::summarise(Value = sum(Va
  na.rm = T)) %>% group_by(Time) %>% mutate(Freq = Value/sum(Value,
  na.rm = T)) %>% group_by(Var) %>% mutate(hline = mean(Freq,
  na.rm = T))

diet_freq_bar <- prey_freq %>% filter(Freq > 0.05) %>% dplyr::mutate(Prey = gsub("\\\\.",
  " ", Var)) %>% dplyr::mutate(Prey = gsub("Invertebrate",
  "Invert.", Prey)) %>% # dplyr::rename(Prey = Var) %>%
ggplot(aes(x = Time, y = Freq, fill = Prey)) + geom_bar(stat = "identity") +
  # geom_bar_interactive(aes(tooltip =
  # paste(Prey, round(Freq, 2), Time)), stat = 'identity') +
  scale_fill_manual(values = RColorBrewer::brewer.pal(10, "Paired")) +
  ggtitle("Prey composition") + ylab("Proportion of prey items") +
  labs(tag = "b") + theme_ts() + theme(legend.text = element_text(size = 8))

```

```
# diet_freq_bar <- girafe(ggobj = diet_freq_bar)
# diet_freq_bar <- girafe_options(diet_freq_bar,
#   opts_zoom(max = 5) )

full_map + diet_freq_bar + plot_layout(ncol = 2, widths = c(4,
  4))
```

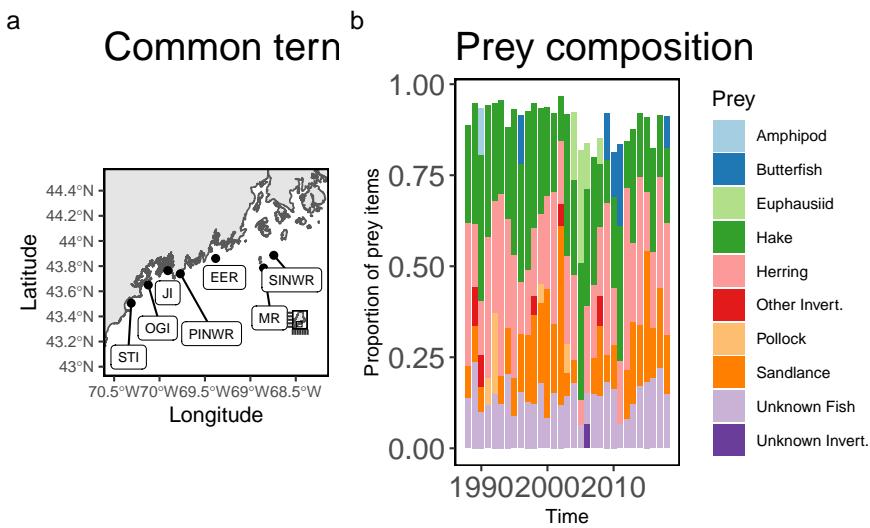


Figure 26.2: Common terns: a. Locations of the seven sampled common tern nesting sites in Gulf of Maine (EER = Eastern Egg Rock, JI = Jenny Island, MR = Matinicus Rock, OGI = Outer Green Island, PINWR = Pond Island National Wildlife Refuge, SINWR = Seal Island National Wildlife Refuge, STI = Stratton Island), and b. Prey frequencies in the diets of common tern observed across the seven colonies in Gulf of Maine. Prey occurring in $<\backslash 5\%$ of common tern diets were excluded for clarity.

Chapter 27

Seasonal SST Anomalies

Description: Seasonal SST Anomalies

Indicator category: Database pull with analysis

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Contributor(s): Sean Hardison, Vincent Saba

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Kimberly Bastille, kimberly.bastille@noaa.gov

Public availability statement: Source data are available here.

27.1 Methods

27.1.1 Data sources

Data for seasonal sea surface temperature anomalies (Fig. 27.1) were derived from the National Oceanographic and Atmospheric Administration optimum interpolation sea surface temperature high resolution data set (NOAA OISST V2) provided by NOAA Earth System Research Laboratory's Physical Science Division, Boulder, CO. The data extend from 1981 to present, and provide a $0.25^\circ \times 0.25^\circ$ global grid of SST measurements (Reynolds et al. 2007).

27.1.2 Data extraction

Individual files containing daily mean SST data for each year during the period of 1981-present were downloaded from the OI SST V5 site. Yearly data provided as layered rasters were masked according to the extent of Northeast US

Continental Shelf. Data were split into three month seasons for (Winter = Jan, Feb, Mar; Spring = Apr, May, Jun; Summer = July, August, September; Fall = Oct, Nov, Dec).

27.1.3 Data analysis

We calculated the long-term mean (LTM) for each season-specific stack of rasters over the period of 1982-2010, and then subtracted the (LTM) from daily mean SST values to find the SST anomaly for a given year. The use of climatological reference periods is a standard procedure for the calculation of meteorological anomalies (WMO 2017). Prior to 2019 State of the Ecosystem reports, SST anomaly information made use of a 1982-2012 reference period. A 1982-2010 reference period was adopted to facilitate calculating anomalies from a standard NOAA ESRL data set.

R code used in extraction and processing:

```
# Processing for spatial SST anomaly

library(dplyr)
library(raster)
library(sf)
library(ggplot2)
library(ncdf4)
library(reshape2)

rast_prep <- function(r) {
  r <- rotate(r) #Rotate
  r <- crop(r, extent(-77, -60, 35, 46)) #Crop
  return(r)
}

raw.dir <- "~/git/ecodata/inst/extdata/gridded"
crs <- "+proj=longlat +lat_1=35 +lat_2=45 +lat_0=40
+lon_0=-77 +x_0=0 +y_0=0 +datum=NAD83 +no_defs +ellps=GRS80 +towgs84=0,0,0"

# These data are large files that are not included among
# ecodata source files. They are accessible here:
# https://www.esrl.noaa.gov/psd/data/gridded/data.noaa.oisst.v2.highres.html
sst.2018 <- rast_prep(stack(file.path(raw.dir, "sst.day.mean.2018.nc")))
ltm <- rast_prep(stack(file.path(raw.dir, "sst.day.mean.ltm.1982-2010.nc")))

# save(sst.2018, file =
# '~/git/ecodata/inst/extdata/gridded/SST.2018.rdata')
```

```

# save(sst.2018, file =
#   '~/git/ecodata/inst/extdata/gridded/SST.LTM.rdata')

winter.ltm <- ltm[[1:90]]
spring.ltm <- ltm[[91:181]]
summer.ltm <- ltm[[182:273]]
fall.ltm <- ltm[[274:365]]

winter.anom <- sst.2018[[1:90]] - winter.ltm
spring.anom <- sst.2018[[91:181]] - spring.ltm
summer.anom <- sst.2018[[182:273]] - summer.ltm
fall.anom <- sst.2018[[274:365]] - fall.ltm

rast_process <- function(r, season) {
  r <- stackApply(r, indices = rep(1, nlayers(r)), mean) #Find mean anomaly
  crs(r) <- crs #Add SOE CRS
  r <- disaggregate(r, 5) #interpolate step 1 - create higher res grid
  r <- focal(r, w = matrix(1, nrow = 5, ncol = 5), fun = mean,
    na.rm = TRUE, pad = TRUE) #interpolate step 2 - moving window
  r <- as(r, "SpatialPointsDataFrame") #Convert to ggplot-able object
  r <- as.data.frame(r)
  r <- r %>% reshape2::melt(id = c("y", "x")) %>% dplyr::rename(Latitude = y,
    Longitude = x) %>% dplyr::select(-variable) %>% mutate(Season = season) %>%
    dplyr::rename(Value = value)

  return(r)
}

seasonal_sst_anomaly_gridded <- rbind(rast_process(winter.anom,
  season = "Winter"), rast_process(spring.anom, season = "Spring"),
  rast_process(summer.anom, season = "Summer"), rast_process(fall.anom,
  season = "Fall"))

usethis::use_data(seasonal_sst_anomaly_gridded, overwrite = T)

```

27.1.4 Plotting

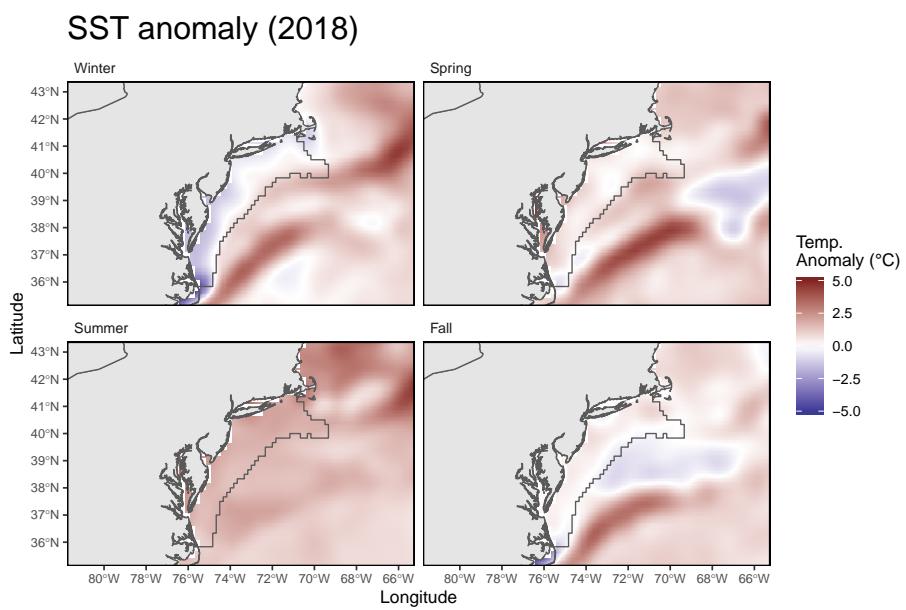


Figure 27.1: Seasonal sea-surface temperature anomalies in the Mid-Atlantic Bight.

Chapter 28

Single Species Status Indicator

Description: Summary of the most recent stock assessment results for each assessed species.

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Synthesis of published information

Contributor(s): Sarah Gaichas, based on code and spreadsheets originally provided by Chris Legault

Data steward: Sarah Gaichas sarah.gaichas@noaa.gov

Point of contact: Sarah Gaichas sarah.gaichas@noaa.gov

Public availability statement: All stock assessment results are publicly available (see Data Sources). Summarized data are available here.

28.1 Methods

28.1.1 Data sources

“Data” used for this indicator are the outputs of stock assessment models and review processes, including reference points (proxies for fishing mortality limits and stock biomass targets and limits), and the current fishing mortality rate and biomass of each stock. The spreadsheet summarizes the most recent stock assessment updates for each species, which are available on the Northeast Fisheries Science Center (NEFSC) website at: <https://www.nefsc.noaa.gov/saw/>

reports.html
<https://www.nefsc.noaa.gov/publications/crd/crd1717/>

Additional assessments are reported directly to the New England Fishery Management Council (NEFMC): <http://s3.amazonaws.com/nefmc.org/Document-2-SAFE-Report-for-Fishing-Year-2016.pdf>
http://s3.amazonaws.com/nefmc.org/4_NEFSC_SkateMemo_July_2017.pdf

28.1.2 Data extraction

Each assessment document was searched to find the following information (often but not always summarized under a term of reference to determine stock status in the executive summary):

- **Bcur:** current year biomass, (most often spawning stock biomass (SSB) or whatever units the reference points are in)
- **Fcur:** current year fishing mortality, F
- **Bref:** biomass reference point, a proxy of Bmsy (the target)
- **Fref:** fishing mortality reference point, a proxy of Fmsy

28.1.3 Data processing

The following R code was used to process the stock status data set for inclusion in the **ecodata** R package.

```
# Processing single-species stock mortality and biomass data

# Read more about this data at https://noaa-edab.github.io/tech-doc/stockstatus.html

library(dplyr)
library(tidyr)
library(ggplot2)

data.dir <- here::here('data-raw')

get_stocks <- function(save_clean = F){
  assess <- read.csv(file.path(data.dir, "2019assess.csv"))
  decode <- read.csv(file.path(data.dir, "2019decoder.csv"))

  stock_status <-
  assess %>%
  group_by(Entity.Name) %>%
```

```

filter(Assessment.Year == max(Assessment.Year)) %>% #Find last year assessment occurred for each species
ungroup() %>%
left_join(.,decode, by = "Entity.Name") %>% #Join in list of managed species
dplyr::select(Entity.Name, Assessment.Year, F.Fmsy, B.Bmsy, Council, Code) %>% #select columns
mutate(id = 1:length(Entity.Name)) %>%
gather(.,Var, Value,-id,-Entity.Name,-Assessment.Year,-Council,-Code) %>% #wide to long
dplyr::select(-id) %>%
dplyr::rename(`Last assessment` = Assessment.Year,
              Stock = Entity.Name) %>% #rename variables for clarity
mutate(Units = "unitless")

if (save_clean){
  usethis::use_data(stock_status, overwrite = T)
} else {
  return(stock_status)
}
}
get_stocks(save_clean = T)

```

28.1.4 Data analysis

For each assessed species, Bcur is divided by Bref and Fcur is divided by Fref. They are then plotted for each species on an x-y plot, with Bcur/Bref on the x axis, and Fcur/Fref on the y axis.

28.1.5 Plotting

The script used to develop the figure in the State of the Ecosystem report is below, with an example figure. Different lines are commented out of the script to produce the Mid- Atlantic or New England figures. The positioning table used to make the plot (the data frame named “decoder” below) is available here.

```

#Get data, spread for plotting, and filter
stock_status <- ecodata::stock_status %>%
  spread(.,Var,Value) %>%
  filter(Council %in% c("NEFMC","Both"))

#Plot constants
y.max <- 4.5
x.max <- 7.5

all_missing <- stock_status %>%

```

```

filter(is.na(B.Bmsy),is.na(F.Fmsy)) %>%
dplyr::select(Code, Council)

b_missing <- stock_status %>%
  filter(is.na(B.Bmsy), !is.na(F.Fmsy)) %>%
  dplyr::select(Code, Council)

f_missing <- stock_status %>%
  filter(is.na(F.Fmsy), !is.na(B.Bmsy)) %>%
  dplyr::select(Code, Council)

#A dataframe that defines custom legend for stocks with unknown status

all.df <- tibble(text = all_missing$Code,
                  x = rep(x.max,length(all_missing$Code)),
                  y = seq(4.35,2.85,-0.22),
                  color = all_missing$Council)

b.df <- tibble(text = b_missing$Code,
                 x = rep(x.max*0.8,length(b_missing$Code)),
                 y = c(4.35,4.14),
                 color = b_missing$Council)

f.df <- tibble(text = f_missing$Code,
                 x = rep(x.max*0.6,length(f_missing$Code)),
                 y = seq(4.35,2.85,-0.22),
                 color = f_missing$Council)

#Plotting code
ggplot(data = stock_status) +
  geom_vline(xintercept = 1, linetype = "dotted", color = "grey60") +
  geom_vline(xintercept = 0.5, linetype = "dashed", color = "grey60") +
  geom_hline(yintercept = 1, linetype = "dashed", color = "grey60") +
  geom_point(aes(x = B.Bmsy,
                 y = F.Fmsy,
                 color = Council,
                 shape = Council)) +
  geom_text_repel(aes(x = B.Bmsy, #geom_text_repel auto-jitters text around points
                      y = F.Fmsy,
                      label = Code,
                      color = Council), show.legend = FALSE, nudge_y = 0.05, nudge_x = 0)
  ylim(0,y.max) +
  xlim(0,x.max*1.1) +
  geom_text(data = all.df, aes(x = x, y = y, label = text, color = color), show.legend =

```

```
geom_text(data = b.df, aes(x = x, y = y, label = text, color = color), show.legend = FALSE, size = 3) +  
  geom_text(data = f.df, aes(x = x, y = y, label = text, color = color), show.legend = FALSE, size = 3) +  
  scale_color_manual(values = c("purple", "blue"), #c("purple", "blue"), #Change legend labels for clarity  
    name = "Managed by",  
    breaks = c("Both", "NEFMC"),  
    labels = c("MAFMC/NEFMC", "NEFMC")) +  
  scale_shape_manual(values = c(1, 19), #Change legend labels for clarity  
    name = "Managed by",  
    breaks = c("Both", "NEFMC"),  
    labels = c("MAFMC/NEFMC", "NEFMC")) +  
  annotate("rect", xmin = 0.924*x.max,  
    xmax = 1.08*x.max,  
    ymin = 0.645*y.max,  
    ymax = 0.98*y.max,  
    alpha = 0.01) +  
  annotate("text", x = 7.5, y = 4.5, label = "F and B missing", fontface = 2, size = 3) +  
  annotate("rect",  
    xmin = 0.729*x.max,  
    xmax = 0.871*x.max,  
    ymin = 0.905*y.max,  
    ymax = 0.98*y.max,  
    alpha = 0.01) +  
  annotate("text", x = 6, y = 4.5, label = "B missing", fontface = 2, size = 3) +  
  annotate("rect", xmin = 0.509*x.max,  
    xmax = 0.681*x.max,  
    ymin = 0.65*y.max,  
    ymax = 0.98*y.max,  
    alpha = 0.01) +  
  annotate("text", x = 4.5, y = 4.5, label = "F missing", fontface = 2, size = 3) +  
  xlab(expression(~B/B[msy])) +  
  ylab(expression(~F/F[msy])) +  
  theme_ts()
```

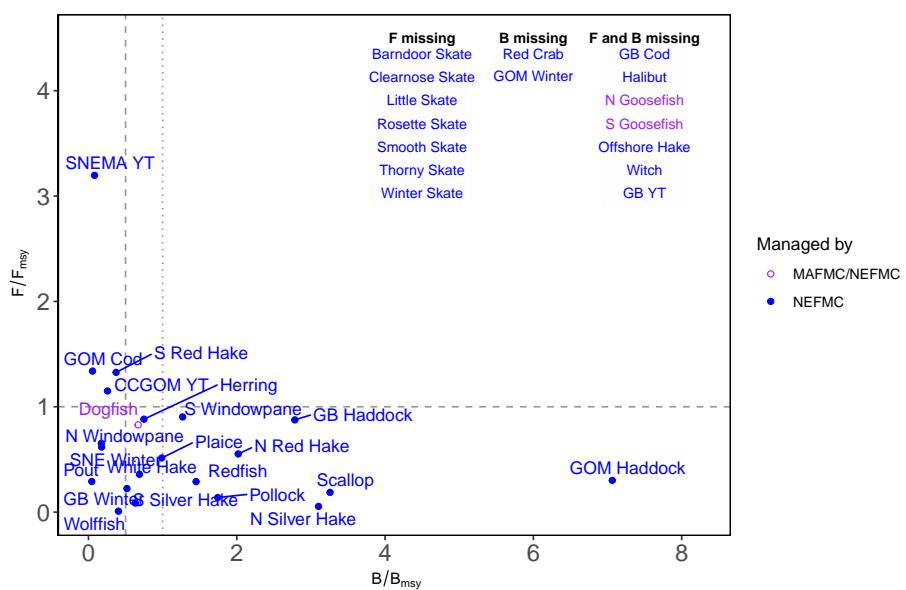


Figure 28.1: Summary of single species status for NEFMC and jointly managed stocks.

Chapter 29

Slopewater proportions

Description: Percent total of water type observed in the deep Northeast Channel (150-200 m water depth).

Indicator category: Published methods

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2019)

Contributors: Paula Fratantoni, paula.fratantoni@noaa.gov; David Mountain, NOAA Fisheries, retired.

Data steward: Kimberly Bastille, kimberly.bastille@noaa.gov

Point of contact: Paula Fratantoni, paula.fratantoni@noaa.gov

Public availability statement: Source data are publicly available at ftp://ftp.nefsc.noaa.gov/pub/hydro/matlab_files/yearly and in the World Ocean Database housed at <http://www.nodc.noaa.gov/OC5/SELECT/dbsearch/dbsearch.html> under institute code 258

29.1 Methods

29.1.1 Data sources

The slope water composition index incorporates temperature and salinity measurements collected on Northeast Fisheries Science Center surveys between 1977-present within the geographic confines of the Northeast Channel in the Gulf of Maine. Early measurements were made using water samples collected primarily with Niskin bottles at discreet depths, mechanical bathythermographs and expendable bathythermograph probes, but by 1991 the CTD – an acronym for conductivity temperature and depth – became standard equipment on all NEFSC surveys.

29.1.2 Data extraction

While all processed hydrographic data are archived in an Oracle database (OCDBS), we work from Matlab-formatted files stored locally.

29.1.3 Data analysis

Temperature and salinity measurements are examined to assess the composition of the waters entering the Gulf of Maine through the Northeast Channel. The analysis closely follows the methodology described by D. G. Mountain (2012). This method assumes that the waters flowing into the Northeast Channel between 150 and 200 meters depth are composed of slope waters, originating offshore of the continental shelf, and shelf waters, originating on the continental shelf south of Nova Scotia.

For each survey in the hydrographic archive, ocean temperature and salinity observations sampled in the area just inside the Northeast Channel (bounded by 42.2-42.6°latitude north and 66-66.8°longitude west) and between 150 - 200 meters depth are extracted and a volume-weighted average temperature and salinity is calculated. The volume weighting is accomplished by apportioning the area within the Northeast Channel polygon among the stations occupying the region, based on inverse distance squared weighting. The result of this calculation is a timeseries of volume-average temperature and salinity having a temporal resolution that matches the survey frequency in the database.

The average temperature and salinity observed at depth in the Northeast Channel is assumed to be the product of mixing between three distinct sources having the following temperature and salinity characteristics: (1) Warm Slope Water ($T=10\text{ }^{\circ}\text{C}$, $S=35$), (2) Labrador Slope Water ($T=6\text{ }^{\circ}\text{C}$, $S=34.7$) and (3) Scotian Shelf Water ($T=2\text{ }^{\circ}\text{C}$, $S=32$). As described by D. G. Mountain (2012), the relative proportion of each source is determined via a rudimentary 3-point mixing algorithm. On a temperature-salinity diagram, lines connecting the T-S coordinates for these three sources form a triangle, the sides of which represent mixing lines between the sources. A water sample that is a mixture of two sources will have a temperature and salinity that falls somewhere along the line connecting the two sources on the temperature-salinity diagram. Observations of temperature and salinity collected within the Northeast Channel would be expected to fall within the triangle if the water sampled is a mixture of the three sources. Simple geometry allows us to calculate the relative proportion of each source in a given measurement. As an example, a line drawn from the T-S point representing shelf water through an observed T-S in the center of the triangle will intersect the opposite side of the triangle (the mixing line connecting the coordinates of the two slope water sources). This intersecting T-S value may then be used to calculate the relative proportions (percentage) of the two slope water sources. Using this method, the percentage of Labrador slope water and Warm slope water are determined for the timeseries of volume-average

temperature and salinity.

It should be noted that our method assumes that the temperature and salinity properties associated with the source watermasses are constant. In reality, these may vary from year to year, modified by atmospheric forcing, mixing and/or advective processes. Likewise, other sources are periodically introduced into the Northeast Channel, including intrusions of Gulf Stream water flowing into the Gulf of Maine and modified shelf water flowing out of the Gulf of Maine along the flank of Georges Bank. These sources are not explicitly considered in the 3-point mixing algorithm and may introduce errors in the proportional estimates.

29.1.4 Data processing

Source data were formatted for inclusion in the `ecodata` R package using the following R code.

```
# Process slopewater proportion time series Slopewater
# proportions give the percent total of water type observed
# in the deep Northeast Channel (150-200 m depth). Raw data
# fields correspond to year, water mass flavor (WSW = Warm
# Slope Water, LSLW = Labrador Slope Water), and proportion
# of total expressed as a percentage.

library(dplyr)
library(tidyr)

# Get raw
raw.dir <- here::here("data-raw") #input raw

get_slopewater <- function(save_clean = F) {

  d <- read.csv(file.path(raw.dir, "slopewater_proportions.csv"))

  slopewater <- d %>% dplyr::rename(Time = year, Var = water.mass.flavor,
                                         Value = prop) %>% mutate(EPU = "GOM", Units = "unitless",
                                         Var2 = "proportion ne channel") %>% unite(., Var, c(Var,
                                         Var2), sep = " ") %>% as.data.frame()

  if (save_clean) {
    usethis::use_data(slopewater, overwrite = T)
  } else {
    return(slopewater)
  }
}
```

```

}
get_slopewater(save_clean = T)

```

29.1.5 Plotting

```

sw.df <- slopewater %>% mutate(Var, Var = plyr::mapvalues(Var,
  from = c("WSW proportion ne channel", "LSLW proportion ne channel"),
  to = c("WSW", "LSLW"))) %>% dplyr::rename(Origin = Var) %>%
  group_by(Origin) %>% mutate(hline = mean(Value))

sw.df$Origin <- factor(sw.df$Origin, levels = c("WSW", "LSLW"))

ggplot(data = sw.df) + geom_line(aes(x = Time, y = Value, color = Origin)) +
  geom_point(aes(x = Time, y = Value, color = Origin)) + ylab("Percent of Total Slope-
  water") + ggtitle("Slopewater Proportions in NE Channel") + scale_x_continuous(expand = c(0.01))
  + geom_hline(aes(yintercept = hline, color = Origin),
  size = hline.size, alpha = hline.alpha, linetype = hline.lty) +
  theme_ts() + theme(strip.text = element_text(hjust = 0, face = "italic"))

```

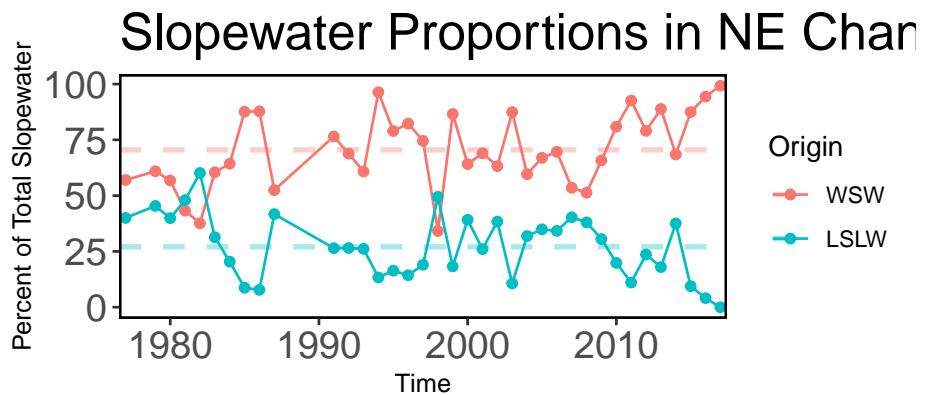


Figure 29.1: Proportion of warm slope water (WSW) and Labrador slope water (LSLW) entering the GOM through the Northeast Channel.

Chapter 30

Species Density Estimates

Description: Current and Historical Species Distributions

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018), State of the Ecosystem - Mid-Atlantic (2017, 2018)

Indicator category: Database pull; Database pull with analysis

Contributor: Kevin Friedland

Data steward: Kevin Friedland

Point of contact: Kevin Friedland, kevin.friedland@noaa.gov

Public availability statement: Source data are publicly available.

30.1 Methods

We used kernel density plots to depict shifts in species' distributions over time. These figures characterize the probability of a species occurring in a given area based on Northeast Fisheries Science Center (NEFSC) Bottom Trawl Survey data. Kernel density estimates (KDEs) of distributions are shown for the period of 1970-1979 (shaded blue) and most recent three years of survey data (shaded red) (e.g. Figure 30.1). Results are typically visualized for spring and fall bottom trawl surveys separately.

Three probability levels (25%, 50%, 75%) are shown for each time period, where the 25% region depicts the core area of the distribution and the 75% region shows the area occupied more broadly by the species. A wide array of KDEs for many ecologically and economically important species on the Northeast US Continental Shelf are available here.

30.1.1 Data sources

Current and historical species distributions are based on the NEFSC Bottom Trawl Survey data (aka “Survdat”) and depth strata. Strata are available as shapefiles that can be downloaded here (listed as “strata.shp”).

30.1.2 Data analysis

```

library(ecodata)
library(maps)
library(mapdata)
library(ks)
library(marmap)
library(raster)
library(geosphere)
library(dplyr)

# plot ks plots

#----STUFF TO SET-----
data.dir <- here::here("data")
gis.dir <- here::here("gis")

# Gives most recent period. 2015-2019 input is 2016-2018 data
rminyr <- 2015
rmaxyr <- 2019

# tlevel is density of color for KD contours areas
tlevel=75 # move later
color_b="blue"
color_r="orange3"
color_r="tomato3"

# Code to read in strata and compute areas. Or read from cache.

# readin in strata.shp and compute areas of strata
# TrawlStrata <- raster::shapefile(file.path(gis.dir, "BTS_Strata.shp"))
# AREA <- geosphere::areaPolygon(TrawlStrata, r=6371000)/10^6

# for array of strata and area and make into dataframe
# stratareas <- cbind(TrawlStrata@data$STRATA, AREA)
# colnames(stratareas) <- c("STRATA", "AREA")
# stratareas <- data.frame(stratareas)
load(file.path(data.dir, "StratAreas.Rdata"))

```

```

# Query bathymetry or load from cache
# getNOAA.bathy(lon1 = -77, lon2 = -65, lat1 = 35, lat2 = 45,
#                 resolution = 10) -> nesbath
load(file.path(data.dir, "nesbath.Rdata"))

#Load raw survey data
load(file.path(data.dir, "Survdat.RData"))

# MUST run addTrans function
# color transparency
addTrans <- function(color,trans){
  # This function adds transparency to a color.
  # Define transparency with an integer between 0 and 255
  # 0 being fully transparent and 255 being fully visable
  # Works with either color and trans a vector of equal length,
  # or one of the two of length 1.

  if (length(color) != length(trans) & !any(c(length(color),length(trans)) == 1)) stop("Vector lengths
  if (length(color) == 1 & length(trans) > 1) color <- rep(color,length(trans))
  if (length(trans) == 1 & length(color) > 1) trans <- rep(trans,length(color))

  num2hex <- function(x)
  {
    hex <- unlist(strsplit("0123456789ABCDEF",split=""))
    return(paste(hex[(x-x%>16)/16+1],hex[x%>16+1],sep=""))
  }
  rgb <- rbind(col2rgb(color),trans)
  res <- paste("#",apply(rgb,2,num2hex),2,paste,collapse=""),sep="")
  return(res)
}

plot_kd <- function(species, season, exclude_years){

  # stata to use
  # offshore strata to use
  CoreOffshoreStrata <- c(seq(1010,1300,10),1340, seq(1360,1400,10),seq(1610,1760,10))

  # inshore strata to use, still sampled by Bigelow
  CoreInshore73to12 <- c(3020, 3050, 3080 ,3110 ,3140 ,3170, 3200, 3230,
                        3260, 3290, 3320, 3350 ,3380, 3410 ,3440)
  # combine
  strata_used <- c(CoreOffshoreStrata,CoreInshore73to12)

  survdat <- survdat %>%
    dplyr::select(c(CRUISE6,STATION,STRATUM,SVSPP,YEAR,

```

```

SEASON,LAT,LON,ABUNDANCE,BIOMASS)) %>%
filter(SEASON == season,
      STRATUM %in% strata_used) %>% # delete record form non-core strata and get
# should be one per species
distinct() %>%
# add field with rounded BIOMASS scalar used to adjust distributions
mutate(LOGBIO = round(log10(BIOMASS * 10+10)))

# trim the data....to prepare to find stations only
survdat_stations <- survdat %>%
  dplyr::select(CRUISE6, STATION, STRATUM, YEAR) %>%
  distinct()

# make table of strata by year
numtowsstratyr <- table(survdat_stations$STRATUM,survdat_stations$YEAR)

# find records to keep based on core strata
rectokeep <- stratareas$STRATA %in% strata_used

# add rec to keep to survdat
stratareas <- cbind(stratareas,rectokeep)

# delete record form non-core strata
stratareas_usedonly <- stratareas[!stratareas$rectokeep=="FALSE",]

areapertow=numtowsstratyr

#compute area covered per tow per strata per year
for(i in 1:50){
  areapertow[,i]=stratareas_usedonly$AREA/numtowsstratyr[,i]
}

# change inf to NA and round and out in DF
areapertow[] [is.infinite(areapertow[])]=NA
areapertow=round(areapertow)
areapertow=data.frame(areapertow)
colnames(areapertow) <- c("STRATUM","YEAR","AREAWT")
areapertow$STRATUM <- as.numeric(as.character(areapertow$STRATUM))
areapertow$YEAR <- as.numeric(as.character(areapertow$YEAR))

survdat <- survdat %>%
  inner_join(.,areapertow, by= c("STRATUM","YEAR")) %>%
  dplyr::rename(AREAPERTOW = AREAWT)

# add col to survdat for PLOTWT

```

```

survdat$PLOTWT <- NA
survdat$PLOTWT <- ceiling(survdat$AREAPEROW/1000*survdat$LOGBIO/9)

if (!is.null(exclude_years)){
  sdat <- survdat %>% filter(!YEAR %in% exclude_years)
} else {
  sdat <- survdat
}

# read species list
sps <- ecodata::species_groupings %>% filter(!is.na(SVSPP)) %>% dplyr::select(COMNAME, SVSPP)
sps <- sps[!duplicated(sps),]
numsp <- nrow(sps)

# graph par
par(mar = c(0,0,0,0))
par(oma = c(0,0,0,0))

# index 1:numsp, or by species record number for one species, i.e. 25:25
tspe <- sps %>% filter(COMNAME == species)

# start map
map("worldHires", xlim=c(-77,-65), ylim=c(35,45), fill=T, border=0, col="gray")
map.axes()

plot(nesbath, deep=-200, shallow=-200, step=1, add=T, lwd=1, col="gray50", lty=2)

# for base period, 1970 to 1979, find call lons for species and by biomass weighting
minyr=1969;maxyr=1980
clons1 = sdat$LON[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT ==1)
clons2 = sdat$LON[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT ==2)
clons3 = sdat$LON[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT ==3)
clons4 = sdat$LON[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT ==4)
clons5 = sdat$LON[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT ==5)

# get rid of missings, KS does not like
clons1 <- na.omit(clons1)
clons2 <- na.omit(clons2)
clons3 <- na.omit(clons3)
clons4 <- na.omit(clons4)
clons5 <- na.omit(clons5)

# accumulate all lons, repeating for weighting
clons=c(clons1,clons2,clons2,clons3,clons3,clons3,clons4,clons4,clons4,clons4,

```

```

clons5,clons5,clons5,clons5,clons5)

# same for lats
clats1 = sdat$LAT[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clats2 = sdat$LAT[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clats3 = sdat$LAT[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clats4 = sdat$LAT[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clats5 = sdat$LAT[(sdat$YEAR>minyr & sdat$YEAR<maxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clats1 <- na.omit(clats1)
clats2 <- na.omit(clats2)
clats3 <- na.omit(clats3)
clats4 <- na.omit(clats4)
clats5 <- na.omit(clats5)
clats=c(clats1,clats2,clats2,clats3,clats3,clats3,clats4,clats4,clats4,
       clats5,clats5,clats5,clats5,clats5)

# combine lons and lats
x=cbind(clons,clats)
# compute KD using KS routine
Hscv1 <- Hscv.diag(x=x)
#fhat.pi1 <- kde(x=x, H=Hscv1)

fhat.pi1 <- kde(x, compute.cont=T, binned=F, xmin=c(-77, 35), xmax=c(-65, 45)) # spe

# add to plot each probability separately
contour.25 <- with(fhat.pi1, contourLines(x=eval.points[[1]],y=eval.points[[2]], z=eval.pi))
contour.50 <- with(fhat.pi1, contourLines(x=eval.points[[1]],y=eval.points[[2]], z=eval.pi))
contour.75 <- with(fhat.pi1, contourLines(x=eval.points[[1]],y=eval.points[[2]], z=eval.pi))

for (j in 1:length(contour.75)){
  polygon(unlist(contour.75[[j]][2]), unlist(contour.75[[j]][3]), col=addTrans(color_75))
}
for (j in 1:length(contour.50)){
  polygon(unlist(contour.50[[j]][2]), unlist(contour.50[[j]][3]), col=addTrans(color_50))
}
for (j in 1:length(contour.25)){
  polygon(unlist(contour.25[[j]][2]), unlist(contour.25[[j]][3]), col=addTrans(color_25))
}

clons1 = sdat$LON[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clons2 = sdat$LON[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clons3 = sdat$LON[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clons4 = sdat$LON[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
clons5 = sdat$LON[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat) | (sdat$SVSPP==tspe$SVSPP & sdat$LAT>minlat & sdat$LAT<maxlat)]
# get rid of missings, KS does not like

```

```

clons1 <- na.omit(clons1)
clons2 <- na.omit(clons2)
clons3 <- na.omit(clons3)
clons4 <- na.omit(clons4)
clons5 <- na.omit(clons5)
# accumulate all lons, repeating for weighting
clons=c(clons1,clons2,clons2,clons3,clons3,clons3,clons4,clons4,clons4,
        clons5,clons5,clons5,clons5,clons5)

# same for lats
clats1 = sdat$LAT[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT == 1)]
clats2 = sdat$LAT[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT == 2)]
clats3 = sdat$LAT[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT == 3)]
clats4 = sdat$LAT[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT == 4)]
clats5 = sdat$LAT[(sdat$YEAR>rminyr & sdat$YEAR<rmaxyr & sdat$SVSPP==tspe$SVSPP & sdat$PLOTWT == 5)]
clats1 <- na.omit(clats1)
clats2 <- na.omit(clats2)
clats3 <- na.omit(clats3)
clats4 <- na.omit(clats4)
clats5 <- na.omit(clats5)
clats=c(clats1,clats2,clats2,clats3,clats3,clats3,clats4,clats4,clats4,
        clats5,clats5,clats5,clats5,clats5)

x=cbind(clons,clats)
Hscv2 <- Hscv.diag(x=x)

#fhat.pi2 <- kde(x=x, H=Hscv2)
fhat.pi2 <- kde(x, compute.cont=T, binned=F, xmin=c(-77, 35), xmax=c(-65, 45)) # specify grid to plot

# add to plot each probability separately
contour.25 <- with(fhat.pi2, contourLines(x=eval.points[[1]],y=eval.points[[2]], z=estimate,level=.25))
contour.50 <- with(fhat.pi2,contourLines(x=eval.points[[1]],y=eval.points[[2]], z=estimate,level=.5))
contour.75 <- with(fhat.pi2, contourLines(x=eval.points[[1]],y=eval.points[[2]], z=estimate,level=.75))

for (j in 1:length(contour.75)){
  polygon(unlist(contour.75[[j]][2]), unlist(contour.75[[j]][3]), col=addTrans(color_r,tlevel),
}
for (j in 1:length(contour.50)){
  polygon(unlist(contour.50[[j]][2]), unlist(contour.50[[j]][3]), col=addTrans(color_r,tlevel),
}
for (j in 1:length(contour.25)){
  polygon(unlist(contour.25[[j]][2]), unlist(contour.25[[j]][3]), col=addTrans(color_r,tlevel),
}

```

```

text(-70,37.5, pos=4,labels = species)
segments(-69.5, 37,-68.5, 37,lwd=1,col="gray50",lty=2)
text(-68.5,37, pos=4,labels = "200m")

stline=36.5
text(-70.4,stline, pos=4,labels = "25%      50%      75%")
incline=-.3
segments(-70, stline+incline,-69,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
segments(-70, stline+incline,-68,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
segments(-70, stline+incline,-67,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
text(-66.8,stline+incline, pos=4,labels = "Base")

incline=-.7
segments(-70, stline+incline,-69,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
segments(-70, stline+incline,-68,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
segments(-70, stline+incline,-67,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
text(-66.8,stline+incline, pos=4,labels = "Recent")

incline=-1.1
segments(-70, stline+incline,-69,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
segments(-70, stline+incline,-68,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
segments(-70, stline+incline,-67,stline+incline ,lwd=20,col=addTrans(color_b,tlevel))
segments(-70, stline+incline,-69,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
segments(-70, stline+incline,-68,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
segments(-70, stline+incline,-67,stline+incline ,lwd=20,col=addTrans(color_r,tlevel))
text(-66.8,stline+incline, pos=4,labels = "Overlap")

}

```

30.1.3 Plotting

```
plot_kd(species = "SEA SCALLOP", season = "SPRING", exclude_years = NULL)
```

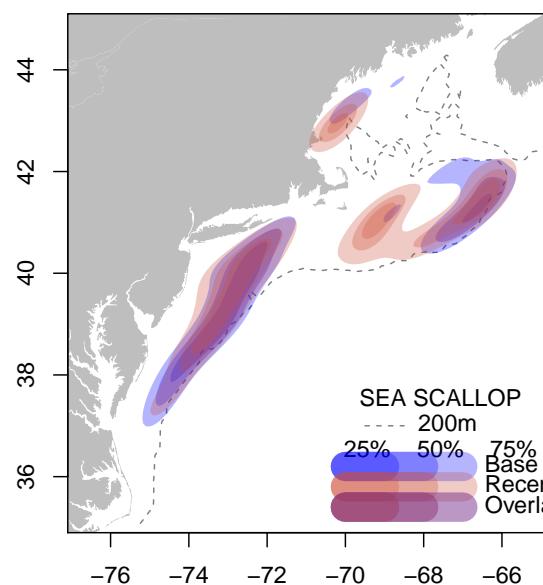


Figure 30.1: Current and historical sea scallop kernel density estimates derived from spring survey data. Current estimates derived from 2016-2018 data.

Chapter 31

Species Distribution Indicators

Description: Species mean depth, along-shelf distance, and distance to coastline

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Extensive analysis; not yet published

Contributor(s): Kevin Friedland

Data steward: Kevin Friedland, kevin.friedland@noaa.gov

Point of contact: Kevin Friedland, kevin.friedland@noaa.gov

Public availability statement: Source data are available upon request (read more here). Derived data may be downloaded here.

31.1 Methods

Three metrics quantifying spatial-temporal distribution shifts within fish populations were developed by Friedland et al. (2018), including mean depth, along-shelf distance, and distance to coastline. Along-shelf distance is a metric for quantifying the distribution of a species through time along the axis of the US Northeast Continental Shelf, which extends northeastward from the Outer Banks of North Carolina. Values in the derived time series correspond to mean distance in km from the southwest origin of the along-shelf axis at 0 km. The along-shelf axis begins at 76.53°W 34.60°N and terminates at 65.71°W 43.49°N.

Once mean distance is found, depth of occurrence and distance to coastline can be calculated for each species' positional center. Analyses present in the State of the Ecosystem (SOE) reports include mean depth and along-shelf distance for Atlantic cod, sea scallop, summer flounder, and black sea bass.

31.1.1 Data sources

Data for these indicators were derived from fishery-independent bottom trawl survey data collected by the Northeast Fisheries Science Center (NEFSC).

31.1.2 Data analysis

Species distribution indicators were derived using the following R code.


```

# CODE TO READ IN STRATA COMPUTE AREAS, NOW JUST READ IN
# STRATAREAS dataframe readin in strata.shp and compute areas
# of strata TrawlStrata<-shapefile(file.choose())
# plot(TrawlStrata) AREA<-areaPolygon(TrawlStrata,
# r=6371000)/10^6

# for array of strata and area and make into dataframe
# stratareas=cbind(TrawlStrata@data$STRATA, AREA)
# colnames(stratareas) <- c('STRATA', 'AREA')
# stratareas=data.frame(stratareas) save(stratareas,
# file='stratareas.rdata')

# load stratareas
load("stratareas.rdata")

# get survey datafile Survdat.RData load(file.choose())
# load(file='Survdat.RData')
load(file = surfilename)

# trim the data.... need to choose season
retvars <- c("CRUISE6", "STATION", "STRATUM", "SVSPP", "YEAR",
           "SEASON", "LAT", "LON", "DEPTH", "ABUNDANCE", "BIOMASS")
survdat <- survdat[retvars]
survdat <- survdat[(survdat$SEASON == selseason), ]

# stata to use offshore strata to use
CoreOffshoreStrata <- c(seq(1010, 1300, 10), 1340, seq(1360,
              1400, 10), seq(1610, 1760, 10))
# inshore strata to use, still sampled by Bigelow
CoreInshore73to12 = c(3020, 3050, 3080, 3110, 3140, 3170, 3200,
                     3230, 3260, 3290, 3320, 3350, 3380, 3410, 3440)
# combine
strata_used = c(CoreOffshoreStrata, CoreInshore73to12)

# find records to keep based on core strata
rectokeep = survdat$STRATUM %in% strata_used

# table(rectokeep) add rec to keep to survdat
survdat = cbind(survdat, rectokeep)

# delete record form non-core strata
survdat = survdat[!survdat$rectokeep == "FALSE", ]

# get rid of species

```

```

survdat$rectokeep = survdat$SVSPP %in% spptokeep
survdat = survdat[!survdat$rectokeep == "FALSE", ]

# find unique tow records only, since length and weight
# removed unique deletes to one record per species
survdat <- unique(survdat)

# add field with rounded BIOMASS scalar used to adjust
# distributions
survdat$LOGBIO <- round(log10(survdat$BIOMASS * 10 + 10))
# take a look go from 1 to 5?
table(survdat$LOGBIO)

# trim the data.... to prepare to find stations only
retvars <- c("CRUISE6", "STATION", "STRATUM", "YEAR")
survdat_stations <- survdat[retvars]
# unique reduces to a record per tow
survdat_stations <- unique(survdat_stations)

# make table of strata by year
numtowsstratyr = table(survdat_stations$STRATUM, survdat_stations$YEAR)

# find records to keep based on core strata
rectokeep = stratareas$STRATA %in% strata_used

# add rec to keep to survudat
stratareas = cbind(stratareas, rectokeep)

# delete record form non-core strata
stratareas_usedonly = stratareas[!stratareas$rectokeep == "FALSE",
]

# creat areapertow
areapertow = numtowsstratyr

# compute area covered per tow per strata per year
for (i in 1:47) {
  areapertow[, i] = stratareas_usedonly$AREA/numtowsstratyr[, i]
}

# change inf to NA and round and out in DF
areapertow[] [is.infinite(areapertow[])] = NA

```

```

areapertow = round(areapertow)
areapertow = data.frame(areapertow)
colnames(areapertow) <- c("STRATA", "YEAR", "AREAWT")

# add col to survdat for strata weight
survdat$AREAPERTOW = NA

# fill AREAPERTOW
dimsurvdat = dim(survdat)
for (i in 1:dimsurvdat[1]) {
    survdat$AREAPERTOW[i] = areapertow$AREAWT[which(survdat$STRATUM[i] ==
        areapertow$STRATA & survdat$YEAR[i] == areapertow$YEAR)]
}

table(ceiling(survdat$AREAPERTOW/1000))
table(survdat$LOGBIO)
table(ceiling(survdat$AREAPERTOW/1000 * survdat$LOGBIO/9))

# add col to survdat for PLOTWT
survdat$PLOTWT = NA
survdat$PLOTWT = ceiling(survdat$AREAPERTOW/1000 * survdat$LOGBIO/9)

table(survdat$PLOTWT)

# Plot stations
plot(survdat$LON[survdat$YEAR == 1974], survdat$LAT[survdat$YEAR ==
    1974])

# put in shorter name
sdat = survdat

# clear some space
remove(survdat)

# number of records to evaluate
numrecs = nrow(sdat)

# =====

# TAKEN OUT SINCE THE SAME AS GASDIST

# blank array for ASDIST d = array(data = NA, dim =
# nrow(diag))

```

```

# block to calculate diag distance ASDIST for (j in
# 1:numrecs) { print(numrecs-j) lat1=sdat$LAT[j]* radt
# long1=sdat$LON[j]* radt for (i in 1:150){ lat2=diag$LAT[i]*
# radt long2=diag$LON[i]* radt d[i] <-
# acos(sin(lat1)*sin(lat2) + cos(lat1)*cos(lat2) *
# cos(long2-long1)) * R } dindex=which(d==min(d)) lat1=34.60*
# radt long1=-76.53* radt lat2=diag$LAT[dindex]* radt
# long2=diag$LON[dindex]* radt sdat$ASDIST[j] =
# acos(sin(lat1)*sin(lat2) + cos(lat1)*cos(lat2) *
# cos(long2-long1)) * R }

# =====

# TAKEN OUT SINCE THE SAME AS GDTOC

# blank array for DTOC d = array(data = NA, dim =
# nrow(nescoast))

# block to calculate diag distance DTOC for (j in 1:numrecs)
# { print(numrecs-j)

# lat1=sdat$LAT[j]* radt long1=sdat$LON[j]* radt for (i in
# 1:nrow(nescoast)){ lat2=nescoast$LAT[i]* radt
# long2=nescoast$LON[i]* radt d[i] <-
# acos(sin(lat1)*sin(lat2) + cos(lat1)*cos(lat2) *
# cos(long2-long1)) * R } sdat$DTOC[j] = d[which(d==min(d))]

# }

# =====

print("distance to coast using geosphere")

##### Geosphere package to calc distance to coastline from pts
##### (lon,lat), returns meters
dd = array(data = NA, dim = nrow(sdat))
pts = data.frame(sdat$LON, sdat$LAT)
# line = t(rbind(nescoast$Longitude, nescoast$Latitude)) line
# = t(rbind(nesc2$LON, nesc2$LAT))
line_nescoast2 = t(rbind(nescoast2$LON, nescoast2$LAT))

# dd=dist2Line(pts[,], line)
dd = dist2Line(pts[, ], line_nescoast2)

```

```

sdat$GDTOC = dd/1000 # convert meters to KM

# TESTING (look at sdat to compare dtc (geosphere) to DTOC
# (from loop)) plot(line) lines(diag) lines(line)
# points(nescoast) ddtest=data.frame(dd) ddtest$distance=NULL
# plot(nescoast2) lines(nescoast2) ptt=105
# points(sdat$LON[ptt], sdat$LAT[ptt], col='red');
# sdat$DTOC[ptt]; sdat$dtc[ptt]; points(ddtest[ptt,],
# col='green')

# =====

print("diag distance using geosphere")
# Find distance to diagonal line (diag), use coordinates of
# nearest point to find distance to NC outerbanks (min(diag))
dd2 = array(data = NA, dim = nrow(sdat))
dd2 = dist2Line(pts[, ], diag, distfun = distHaversine)
# Distance of closest point to data along diag line to NC
# coast
p1 = diag[1, ] #start of line
p3 = diag[150, ] #end of line
p2 = data.frame(dd2[, 2], dd2[, 3])
distNC = distCosine(p1, p2, r = 6378137)/1000 # convert to KM (Great circle distance)
sdat$GASDIST = distNC

# =====

# create column for missing depth data intially with depth
# data
sdat$MISDEPTH = sdat$DEPTH

# find cases with missing depth data
missingdepth = which(is.na(sdat$DEPTH))

# fill only those records in misdepth with depth from grid
for (k in missingdepth) {
    sdat$MISDEPTH[k] = extract(gdepth, cbind(sdat$LON[k], sdat$LAT[k])) *
        -1
}

# =====

```

```

# outline=paste('YR', ', ', 'SVSPP', ', ', 'mASDIST', ', ', 'mDTOC', ', ', 'mMISDEPTH', ', ', 'mLAT', ', ', 'mLON',
# write.table(outline,file=outfile,row.name=F,col.names=F,append=TRUE)

out_data = array(NA, c((max(sdat$YEAR) - min(sdat$YEAR) + 1) *
  numsp, 7))

row_c = 0

for (i in 1:numsp) {
  print(i)
  for (j in min(sdat$YEAR):max(sdat$YEAR)) {

    row_c = row_c + 1
    # sumdist=sum(sdat$ASDIST[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]] *sdat$PLOTWT[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]]) 
    # lendist=sum(sdat$PLOTWT[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]]) mASDIST =sumdist / lendist

    # sumdist=sum(sdat$DTOC[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]] *sdat$PLOTWT[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]]) 
    # lendist=sum(sdat$PLOTWT[sdat$YEAR==j] &
    # sdat$SVSPP==sps$SVSPP[i]]) mDTOC =sumdist / lendist

    sumdepth = sum(sdat$MISDEPTH[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]] * sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]])
    lendepth = sum(sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]])
    mMISDEPTH = sumdepth/lendepth

    sumdepth = sum(sdat$LAT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]] * sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]])
    lendepth = sum(sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]])
    mLAT = sumdepth/lendepth

    sumdepth = sum(sdat$LON[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]] * sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
      sps$SVSPP[i]])
  }
}

```

```

lendepth = sum(sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]])
mLON = sumdepth/lendepth

sumdepth = sum(sdat$GASDIST[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]] * sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]])
lendepth = sum(sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]])
mGASDIST = sumdepth/lendepth

sumdepth = sum(sdat$GDTOC[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]] * sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]])
lendepth = sum(sdat$PLOTWT[sdat$YEAR == j & sdat$SVSPP ==
  sps$SVSPP[i]])
mGDTOC = sumdepth/lendepth

out_data[row_c, 1] = j
out_data[row_c, 2] = sps$SVSPP[i]
out_data[row_c, 3] = mMISDEPTH
out_data[row_c, 4] = mLAT
out_data[row_c, 5] = mLON
out_data[row_c, 6] = mGASDIST
out_data[row_c, 7] = mGDTOC

}

}

# outline= paste(j, ', ', sps$SVSPP[i], ', ', mMISDEPTH, ', ', mLAT, ', ', mLON, ', ', mGASDIST, ', ', mGDTOC)
# write.table(outline, file=outfile, row.name=F, col.names=F, append=TRUE)

out_data = data.frame(out_data)

names(out_data)[names(out_data) == "X1"] <- "YR"
names(out_data)[names(out_data) == "X2"] <- "SP"
names(out_data)[names(out_data) == "X3"] <- "DEPTH"
names(out_data)[names(out_data) == "X4"] <- "LAT"

```

```

names(out_data)[names(out_data) == "X5"] <- "LON"
names(out_data)[names(out_data) == "X6"] <- "ASDIST"
names(out_data)[names(out_data) == "X7"] <- "DTEOC"

write.csv(out_data, file = outfile)

```

31.1.3 Data processing

Distribution indicators were further formatted for inclusion in the `ecodata` R package using the following R code.

```

# Aggregate species distribution metrics for the Northeast
# Shelf

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

get_species_dist <- function(save_clean = F) {

  species_dist <- read.csv(file.path(raw.dir, "sp dist.csv")) %>%
    dplyr::rename(depth = DEPTH, Latitude = LAT, Longitude = LON,
      `along-shelf distance` = ASD, `distance to coast` = DTC,
      Time = Year) %>% gather(., Var, Value, -Time) %>%
    mutate(EPU = "All", Units = ifelse(str_detect(Var, "distance"),
      "km", ifelse(str_detect(Var, "Latitude"), "degreesN",
        ifelse(str_detect(Var, "Longitude"), "degreesW",
          ifelse(str_detect(Var, "depth"), "m", NA)))))

  if (save_clean) {
    usethis::use_data(species_dist, overwrite = T)
  } else {
    return(species_dist)
  }
}

```

31.1.4 Plotting

```

# Relative working directories
data.dir <- here::here("data")
r.dir <- here::here("R")

```

```

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mfrow = c(2, 1), mar = c(0, 0, 0, 0), oma = c(3.5,
  5, 2, 4))

soe.plot(SOE.data, "Time", "black sea bass spring mean along-shelf dist",
  stacked = "A", rel.y.num = 1.1, suppressAxis = T, line.forward = T,
  tol = 0.15, x.start = 1963, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "black sea bass fall mean along-shelf dist",
  stacked = "B", rel.y.num = 1.1, tol = 0.15, end.start = 2007,
  cex.stack = 1.5)
soe.stack.axis("Year", "Mean along-shelf distance, km", y.line = 3)

```

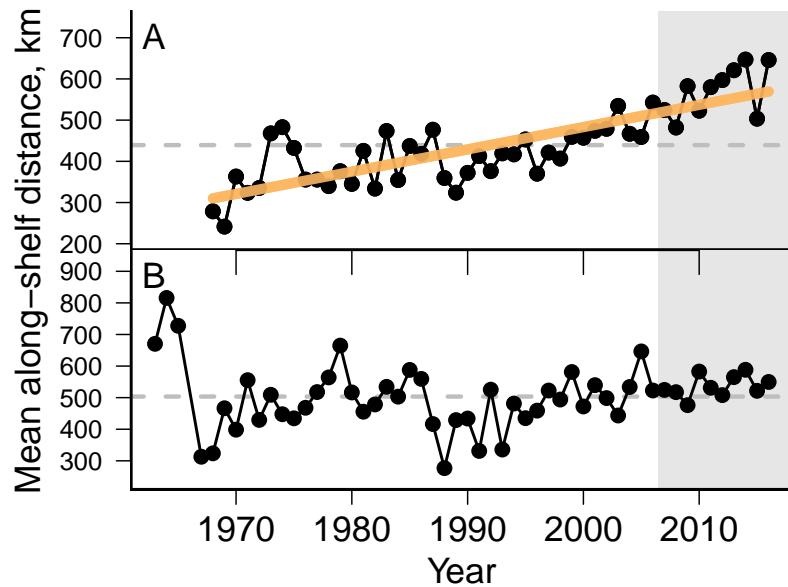


Figure 31.1: Black sea bass along-shelf distance trends in spring (A) and fall (B).

```
opar <- par(mfrow = c(2, 1), mar = c(0, 0, 0, 0), oma = c(3.5,
  5, 2, 4))

soe.plot(SOE.data, "Time", "black sea bass spring mean depth",
  stacked = "A", rel.y.num = 1.1, suppressAxis = T, tol = 0.15,
  x.start = 1963, end.start = 2007, cex.stacked = 1.5)
soe.plot(SOE.data, "Time", "black sea bass fall mean depth",
  stacked = "B", rel.y.num = 1.1, tol = 0.15, end.start = 2007,
  line.forward = T, cex.stacked = 1.5)
soe.stacked.axis("Year", "Mean depth, m", y.line = 3)
```

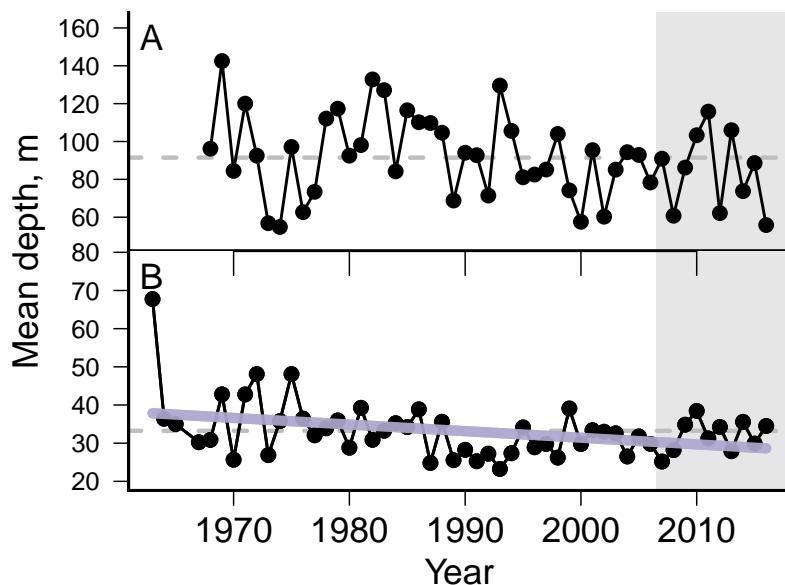


Figure 31.2: Mean depth of black sea bass in spring (A) and fall (B) on the Northeast Continental Shelf.

Chapter 32

Survey Data

Description: Survdat (Survey database)

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull

Contributor(s): Sean Lucey

Data steward: Sean Lucey sean.lucey@noaa.gov

Point of contact: Sean Lucey sean.lucey@noaa.gov

Public availability statement: Source data are available to qualified researchers upon request (see “Access Information” here). Derived data used in SOE reports are available here.

32.1 Methods

The Northeast Fisheries Science Center (NEFSC) has been conducting standardized bottom trawl surveys in the fall since 1963 and spring since 1968. The surveys follow a stratified random design. Fish species and several invertebrate species are enumerated on a tow by tow basis (Azarovitz 1981).

The data are housed in the NEFSC’s survey database (SVDBS) maintained by the Ecosystem Survey Branch.

Direct pulls from the database are not advisable as there have been several gear modifications and vessel changes over the course of the time series (Miller et al. 2010). Survdat was developed as a database query that applies the appropriate calibration factors for a seamless time series since the 1960s. As such, it is the base for many of the other analyses conducted for the State of the Ecosystem report that involve fisheries independent data.

Table 32.1: Calibration factors for NEFSC trawl survey data

Name	Code	Applied
Door Conversion	DCF	<1985
Net Conversion	GCF	1973 - 1981 (Spring)
Vessel Conversion I	VCF	Delaware II records
Vessel Conversion II	BCF	Henry Bigelow records

The Survdat script can be broken down into two sections. The first pulls the raw data from SVDBS. While the script is able to pull data from more than just the spring and fall bottom trawl surveys, for the purposes of the State of the Ecosystem reports only the spring and fall data are used. Survdat identifies those research cruises associated with the seasonal bottom trawl surveys and pulls the station and biological data. Station data includes tow identification (cruise, station, and stratum), tow location and date, as well as several environmental variables (depth, surface/bottom salinity, and surface/bottom temperature). Stations are filtered for representativeness using a station, haul, gear (SHG) code for tows prior to 2009 and a tow, operations, gear, and acquisition (TOGA) code from 2009 onward. The codes that correspond to a representative tow (SHG \leq 136 or TOGA \leq 1324) are the same used by assessment biologists at the NEFSC. Biological data includes the total biomass and abundance by species, as well as lengths and number at length.

The second section of the Survdat script applies the calibration factors. There are four calibration factors applied (Table 32.1). Calibration factors are pulled directly from SVDBS. Vessel conversions were made from either the NOAA Ship *Delaware II* or NOAA Ship *Henry Bigelow* to the NOAA Ship *Albatross IV* which was the primary vessel for most of the time series. The Albatross was decommissioned in 2009 and the Bigelow is now the primary vessel for the bottom trawl survey.

The output from Survdat is an RData file that contains all the station and biological data, corrected as noted above, from the NEFSC Spring Bottom Trawl Survey and NEFSC Fall Bottom Trawl Survey. The RData file is a data.table, a powerful wrapper for the base data.frame (<https://cran.r-project.org/web/packages/data.table/data.table.pdf>). There are also a series of tools that have been developed in order to utilize the Survdat data set (<https://github.com/slicey/RSurvey>).

32.1.1 Data sources

Survdat is a database query of the NEFSC survey database (SVDBS). These data are available to qualified researchers upon request. More information on the data request process is available under the “Access Information” field here.

32.1.2 Data extraction

Extraction methods are described above. The following is the R code used in the data extraction process.

```
# Survdat.r This script will generate data from the NEFSC
# bottom trawl surveys SML

#-----
# User parameters
if (Sys.info()["sysname"] == "Windows") {
  data.dir <- "L:\\Rworkspace\\RSurvey"
  out.dir <- "L:\\EcoAP\\Data\\survey"
  memory.limit(4000)
}
if (Sys.info()["sysname"] == "Linux") {
  data.dir <- "/home/slucey/slucey/Rworkspace/RSurvey"
  out.dir <- "/home/slucey/slucey/EcoAP/Data/survey"
  uid <- "slucey"
  cat("Oracle Password: ")
  pwd <- scan(stdin(), character(), n = 1)
}

shg.check <- "y"  # y = use only SHG <=136 or TOGA <= 1324 (>2008)
raw.check <- "n"  # y = save data without conversions (survdat.raw), will still
# save data with conversions (survdat)
all.season <- "n" # y = save data with purpose code 10 not just spring/fall
# (survdat.allseason), will not save survdat regular
use.SAD <- "y"  # y = grab data from Survey Analysis Database (SAD) for
# assessed species
#-----
# Required packages
library(RODBC)
library(data.table)

#-----
# Created functions Convert output to text for RODBC query
sqltext <- function(x) {
  out <- x[1]
  if (length(x) > 1) {
    for (i in 2:length(x)) {
      out <- paste(out, x[i], sep = "'", "'")
    }
  }
  out <- paste("'", out, "'", sep = "") 
  return(out)
}
```

```

}

#-----
# Begin script
if (Sys.info()["sysname"] == "Windows") {
  channel <- odbcDriverConnect()
} else {
  channel <- odbcConnect("sole", uid, pwd)
}

# Generate cruise list
if (all.season == "n") {
  cruise.qry <- "select unique year, cruise6, svvessel, season
from mstr_cruise
where purpose_code = 10
and year >= 1963
and (season = 'FALL'
      or season = 'SPRING')
order by year, cruise6"
}

if (all.season == "y") {
  cruise.qry <- "select unique year, cruise6, svvessel, season
from mstr_cruise
where purpose_code = 10
and year >= 1963
order by year, cruise6"
}

cruise <- as.data.table(sqlQuery(channel, cruise.qry))
cruise <- na.omit(cruise)
setkey(cruise, CRUISE6, SVVESSEL)

# Use cruise codes to select other data
cruise6 <- sqltext(cruise$CRUISE6)

# Station data
if (shg.check == "y") {
  preHB.station.qry <- paste("select unique cruise6, svvessel, station, stratum,
                                tow, decdeg_beglat as lat, decdeg_beglon as lon,
                                begin_est_towdate as est_towdate, avgdepth as depth,
                                surftemp, surfsalin, bottemp, botsalin
                                from Union_fscts_svsta
                                where cruise6 in (",
  cruise6, ")")
}

```

```

        and SHG <= 136
        and cruise6 <= 200900
        order by cruise6, station",
sep = "")

HB.station.qry <- paste("select unique cruise6, svvessel, station, stratum,
                        tow, decdeg_beglat as lat, decdeg_beglon as lon,
                        begin_est_towdate as est_towdate, avgdepth as depth,
                        surftemp, surfsalin, bottemp, botsalin
                        from Union_fscs_svsta
                        where cruise6 in (",
cruise6, ")
                        and TOGA <= 1324
                        and cruise6 > 200900
                        order by cruise6, station",
sep = "")

preHB.sta <- as.data.table(sqlQuery(channel, preHB.station.qry))
HB.sta <- as.data.table(sqlQuery(channel, HB.station.qry))
station <- rbindlist(list(preHB.sta, HB.sta))
}

if (shg.check == "n") {
  station.qry <- paste("select unique cruise6, svvessel, station, stratum, tow,
                        decdeg_beglat as lat, decdeg_beglon as lon,
                        begin_est_towdate as est_towdate, avgdepth as depth,
                        surftemp, surfsalin, bottemp, botsalin
                        from UNION_FSCS_SVSTA
                        where cruise6 in (",
cruise6, ")
                        order by cruise6, station",
sep = "")
  station <- as.data.table(sqlQuery(channel, station.qry))
}

setkey(station, CRUISE6, SVVESSEL)

# merge cruise and station
survdat <- merge(cruise, station)

# Catch data
catch.qry <- paste("select cruise6, station, stratum, tow, svspp, catchsex,
                     expcatchnum as abundance, expcatchwt as biomass
                     from UNION_FSCS_SVCAT"

```

```

        where cruise6 in (",
cruise6, ")
        and stratum not like 'YT%'
        order by cruise6, station, svsp",
sep = "")

catch <- as.data.table(sqlQuery(channel, catch.qry))
setkey(catch, CRUISE6, STATION, STRATUM, TOW)

# merge with survdat
setkey(survdat, CRUISE6, STATION, STRATUM, TOW)
survdat <- merge(survdat, catch, by = key(survdat))

# Length data
length.qry <- paste("select cruise6, station, stratum, tow, svsp, catchsex,
length, expnumlen as numlen
from UNION_FSCS_SVLEN
where cruise6 in (",
cruise6, ")
        and stratum not like 'YT%'
        order by cruise6, station, svsp, length",
sep = "")

len <- as.data.table(sqlQuery(channel, length.qry))
setkey(len, CRUISE6, STATION, STRATUM, TOW, SVSP, CATCHSEX)

# merge with survdat
setkey(survdat, CRUISE6, STATION, STRATUM, TOW, SVSP, CATCHSEX)
survdat <- merge(survdat, len, all.x = T)

if (raw.check == "y") {
  survdat.raw <- survdat
  save(survdat.raw, file = paste(out.dir, "Survdat_raw.RData",
    sep = ""))
}

# Conversion Factors need to make abundance column a double
# instead of an integer
survdat[, `:=`(ABUNDANCE, as.double(ABUNDANCE))]

# Grab all conversion factors off the network
convert.qry <- "select *
  from survan_conversion_factors"

convert <- as.data.table(sqlQuery(channel, convert.qry))

```

```

# DCF < 1985 Door Conversion
dcf.spp <- convert[DCF_WT > 0, SVSPP]
for (i in 1:length(dcf.spp)) {
  survdat[YEAR < 1985 & SVSPP == dcf.spp[i], `:=` (BIOMASS,
    BIOMASS * convert[SVSPP == dcf.spp[i], DCF_WT])]
}
dcf.spp <- convert[DCF_NUM > 0, SVSPP]
for (i in 1:length(dcf.spp)) {
  survdat[YEAR < 1985 & SVSPP == dcf.spp[i], `:=` (ABUNDANCE,
    round(ABUNDANCE * convert[SVSPP == dcf.spp[i], DCF_NUM]))]
}

# GCF Spring 1973-1981 Net Conversion
gcf.spp <- convert[GCF_WT > 0, SVSPP]
for (i in 1:length(gcf.spp)) {
  survdat[SEASON == "SPRING" & YEAR > 1972 & YEAR < 1982 &
    SVSPP == gcf.spp[i], `:=` (BIOMASS, BIOMASS/convert[SVSPP ==
      gcf.spp[i], GCF_WT])]
}
gcf.spp <- convert[GCF_NUM > 0, SVSPP]
for (i in 1:length(gcf.spp)) {
  survdat[SEASON == "SPRING" & YEAR > 1972 & YEAR < 1982 &
    SVSPP == gcf.spp[i], `:=` (ABUNDANCE, round(ABUNDANCE/convert[SVSPP ==
      gcf.spp[i], GCF_NUM]))]
}

# VCF SVVESSEL = DE Vessel Conversion
vcf.spp <- convert[VCF_WT > 0, SVSPP]
for (i in 1:length(vcf.spp)) {
  survdat[SVVESSEL == "DE" & SVSPP == vcf.spp[i], `:=` (BIOMASS,
    BIOMASS * convert[SVSPP == vcf.spp[i], VCF_WT])]
}
vcf.spp <- convert[VCF_NUM > 0, SVSPP]
for (i in 1:length(vcf.spp)) {
  survdat[SVVESSEL == "DE" & SVSPP == vcf.spp[i], `:=` (ABUNDANCE,
    round(ABUNDANCE * convert[SVSPP == vcf.spp[i], VCF_NUM]))]
}

# Bigelow >2008 Vessel Conversion - need flat files (not on
# network) Use Bigelow conversions for Pisces as well (PC)
big.fall <- as.data.table(read.csv(file.path(data.dir, "bigelow_fall_calibration.csv")))
big.spring <- as.data.table(read.csv(file.path(data.dir, "bigelow_spring_calibration.csv")))

bf.spp <- big.fall[pW != 1, svsp]
for (i in 1:length(bf.spp)) {

```

```

survdat[SVVESSEL %in% c("HB", "PC") & SEASON == "FALL" &
        SVSPP == bf.spp[i], `:=` (BIOMASS, BIOMASS/big.fall[svspp ==
        bf.spp[i], pW]))]
}
bf.spp <- big.fall[pw != 1, svspp]
for (i in 1:length(bf.spp)) {
  survdat[SVVESSEL %in% c("HB", "PC") & SEASON == "FALL" &
          SVSPP == bf.spp[i], `:=` (ABUNDANCE, round(ABUNDANCE/big.fall[svspp ==
          bf.spp[i], pw]))]
}

bs.spp <- big.spring[pw != 1, svspp]
for (i in 1:length(bs.spp)) {
  survdat[SVVESSEL %in% c("HB", "PC") & SEASON == "SPRING" &
          SVSPP == bs.spp[i], `:=` (BIOMASS, BIOMASS/big.spring[svspp ==
          bs.spp[i], pW]))]
}
bs.spp <- big.spring[pw != 1, svspp]
for (i in 1:length(bs.spp)) {
  survdat[SVVESSEL %in% c("HB", "PC") & SEASON == "SPRING" &
          SVSPP == bs.spp[i], `:=` (ABUNDANCE, round(ABUNDANCE/big.spring[svspp ==
          bs.spp[i], pw]))]
}

if (use.SAD == "y") {
  sad.qry <- "select svspp, cruise6, stratum, tow, station, sex as catchsex,
              catch_wt_B_cal, catch_no_B_cal, length, length_no_B_cal
              from STOCKEFF.I_SV_MERGED_CATCH_CALIB_0"
  sad <- as.data.table(sqlQuery(channel, sad.qry))

  setkey(sad, CRUISE6, STRATUM, TOW, STATION, SVSPP, CATCHSEX,
         LENGTH)
  sad <- unique(sad)
  survdat <- merge(survdat, sad, by = key(sad), all.x = T)

  # Carry over SAD values to survdat columns and delete SAD
  # columns
  survdat[!is.na(CATCH_WT_B_CAL), `:=` (BIOMASS, CATCH_WT_B_CAL)]
  survdat[!is.na(CATCH_NO_B_CAL), `:=` (ABUNDANCE, CATCH_NO_B_CAL)]
  survdat[, `:=` (NUMLEN, as.double(NUMLEN))]
  survdat[!is.na(LENGTH_NO_B_CAL), `:=` (NUMLEN, LENGTH_NO_B_CAL)]
  survdat[, `:=` (c("CATCH_WT_B_CAL", "CATCH_NO_B_CAL", "LENGTH_NO_B_CAL"),
                 NULL)]
}

```

```
odbcClose(channel)

if (all.season == "n") save(survdat, file = file.path(out.dir,
  "Survdat.RData"))
if (all.season == "y") save(survdat, file = file.path(out.dir,
  "Survdat_allseason.RData"))
```

32.1.3 Data analysis

The fisheries independent data contained within the Survdat is used in a variety of products; the more complicated analyses are detailed in their own sections. The most straightforward use of this data is for the resource species aggregate biomass indicators. For the purposes of the aggregate biomass indicators, fall and spring survey data are treated separately. Additionally, all length data is dropped and species separated by sex at the catch level are merged back together.

For the aggregate biomass indicators, Survdat is first post stratified into Ecological Production Units. Stations are labeled by the EPU they fall within using the `over` function from the `rdga` R package (Bivand, Keitt, and Rowlingson 2018). Next, the total number of stations within each EPU per year is counted using unique station records. Biomass is summed by species per year per EPU. Those sums are divided by the appropriate station count to get the EPU mean. Finally, the mean biomasses are summed by aggregate groups. These steps are encompassed in the processing code shown below, which also includes steps taken to format the data set for inclusion in the `ecodata` R package.

```
# Set up for finding survey proportions

# This script was adapted from code written by Sean Lucey at
# the NEFSC. Note that you will need the latest survdat file
# to get this to run successfully.

data.dir <- here::here("data-raw")
#-----
# Required packages
# devtools::install_github('slucey/RSurvey/Survdat')

library(data.table)
library(rgdal)
library(Survdat)
library(dplyr)
library(sf)
library(tidyr)
#-----
```

```

# Load raw data and get 'strata' aka EPUs here
load(file.path(data.dir, "Survdat.RData"))

strata <- ecodata::epu_sf %>% as("Spatial")

# Generate area table
strat.area <- getarea(strata, "EPU")

# Post stratify data
survdat.EPU <- poststrat(survdat, strata, "EPU")
setnames(survdat.EPU, "newstrata", "EPU")

# Subset by season/ strata set
fall <- survdat.EPU[SEASON == "FALL", ]
spring <- survdat.EPU[SEASON == "SPRING", ]

# Run stratification prep
fall.prep <- stratprep(fall, strat.area, strat.col = "EPU", area.col = "Area")
spring.prep <- stratprep(spring, strat.area, strat.col = "EPU",
                         area.col = "Area")

# Calculate mean weight/tow by aggregate groups n tows
n.tows.fall <- unique(fall.prep[, list(YEAR, EPU, ntows)])
n.tows.spring <- unique(spring.prep[, list(YEAR, EPU, ntows)])

# drop length data
setkey(fall.prep, YEAR, EPU, STATION, STRATUM, SVSPP, CATCHSEX)
fall.prep <- unique(fall.prep, by = key(fall.prep))
fall.prep[, `:=`(c("LENGTH", "NUMLEN"), NULL)]

setkey(spring.prep, YEAR, EPU, STATION, STRATUM, SVSPP, CATCHSEX)
spring.prep <- unique(spring.prep, by = key(spring.prep))
spring.prep[, `:=`(c("LENGTH", "NUMLEN"), NULL)]

# Merge Sexed species
setkey(fall.prep, YEAR, EPU, STATION, STRATUM, SVSPP)
fall.prep <- fall.prep[, sum(BIOMASS, na.rm = T), by = key(fall.prep)]

setkey(spring.prep, YEAR, EPU, STATION, STRATUM, SVSPP)
spring.prep <- spring.prep[, sum(BIOMASS, na.rm = T), by = key(spring.prep)]

# Sum biomass within an EPU
fall.sum <- fall.prep[, sum(V1), by = c("YEAR", "EPU", "SVSPP")]
spring.sum <- spring.prep[, sum(V1), by = c("YEAR", "EPU", "SVSPP")]

```

```

# Merge sum with station count
fall.sum <- merge(fall.sum, n.tows.fall, by = c("YEAR", "EPU"))
spring.sum <- merge(spring.sum, n.tows.spring, by = c("YEAR",
    "EPU"))

# Calculate mean weight per tow
fall.sum[, `:=`(kg.per.tow, V1/ntows)]
fall.mean <- fall.sum[, list(YEAR, EPU, SVSPP, kg.per.tow)]

spring.sum[, `:=`(kg.per.tow, V1/ntows)]
spring.mean <- spring.sum[, list(YEAR, EPU, SVSPP, kg.per.tow)]

# get species groupings
groups <- ecodata::species_groupings %>% dplyr::select(group = SOE_18,
    SVSPP, comname = COMNAME) %>% filter(!is.na(SVSPP)) %>% as.data.table()

# Aggregate by conceptual model groupings
fall <- merge(fall.mean, unique(groups[, list(SVSPP, group, comname)]),
    by = "SVSPP", all.x = T)
spring <- merge(spring.mean, unique(groups[, list(SVSPP, group,
    comname)])), by = "SVSPP", all.x = T)

# Fix NA group to other
fall[is.na(group), `:=`(group, "Other")]
spring[is.na(group), `:=`(group, "Other")]

# Before aggregating, save data with species ids
spring$season <- "spring"
fall$season <- "fall"

survey_biomass <- rbind(spring, fall) %>% filter(!is.na(SVSPP))

managed <- ecodata::species_groupings %>% dplyr::select(comname = COMNAME,
    Fed_Managed)

nefsc_survey_disaggregated <- survey_biomass %>% group_by(EPU,
    group, season, YEAR) %>% mutate(Total = sum(kg.per.tow)) %>%
    mutate(Prop = kg.per.tow/Total) %>% filter(!group %in% c("Apex Predator",
    "Other"), EPU != "SS") %>% dplyr::rename(Time = YEAR) %>%
    as.data.frame() %>% left_join(., managed, by = c("comname")) %>%
    distinct() %>% complete(Time = full_seq(min(.Time):max(.Time),
    1), nesting(EPU, Fed_Managed, season, group, comname, SVSPP)) %>%
    dplyr::rename(Management = Fed_Managed, `Feeding guild` = group,
        Season = season, Proportion = Prop)

```

```

usethis::use_data(nefsc_survey_disaggregated, overwrite = TRUE)

## GET AGGREGATED SURVEY
## DATA-----

fall.agg <- fall[, sum(kg.per.tow), by = c("YEAR", "EPU", "group")]
spring.agg <- spring[, sum(kg.per.tow), by = c("YEAR", "EPU",
                                               "group")]

# #Total
fall.agg[, `:=` (Total, sum(V1)), by = c("YEAR", "EPU", "group")]
spring.agg[, `:=` (Total, sum(V1)), by = c("YEAR", "EPU", "group")]

# Get in correct long format for SOE By feeding guild
fall.tot <- copy(fall.agg)
fall.tot[, `:=` (Var, paste(group, "Fall Biomass Index"))]
setnames(fall.tot, c("YEAR", "EPU", "V1"), c("Time", "Region",
                                              "Value"))
fall.tot[, `:=` (Units, "kg tow^-1")]
fall.tot <- unique(fall.tot)
fall.tot <- mutate(fall.tot, Season = "Fall")

spring.tot <- copy(spring.agg)
spring.tot[, `:=` (Var, paste(group, "Spring Biomass Index"))]
setnames(spring.tot, c("YEAR", "EPU", "V1"), c("Time", "Region",
                                                "Value"))
spring.tot[, `:=` (Units, "kg tow^-1")]
spring.tot <- unique(spring.tot)
spring.tot <- mutate(spring.tot, Season = "Spring")

agg_survey <- rbind(spring.tot, fall.tot)

# Merge into one data set
nefsc_survey <- agg_survey %>% dplyr::select(Time, EPU = Region,
                                                 Var, Units, Value) %>% distinct() %>% complete(Time = full_seq(min(.T$Time):max(.T$Time),
                                                               1), nesting(EPU, Var))

usethis::use_data(nefsc_survey, overwrite = T)

```

32.1.4 Plotting

```

# Relative working directories
data.dir <- here::here("data")

```

```

r.dir <- here::here("R")

# Load data
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

# Source plotting functions
source(file.path(r.dir, "BasePlot_source.R"))

opar <- par(mfrow = c(4, 1), mar = c(0, 0, 0, 0), oma = c(4,
  6.5, 2, 6))

soe.plot(SOE.data, "Time", "Piscivore Fall Biomass Index MAB",
  stacked = "A", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Planktivore Fall Biomass Index MAB",
  stacked = "B", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Benthivore Fall Biomass Index MAB",
  stacked = "C", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Benthos Fall Biomass Index MAB",
  stacked = "D", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
# soe.plot(SOE.data, 'Time', 'Total Fall Biomass MAB',
# stacked = 'E', status = F, endshade = T, rel.y.num = 1.1,
# full.trend = T, scale.axis = 1)
soe.stack.axis("Year", expression("Biomass, kg tow"^-1), y.line = 3,
  x.line = 2.7)

opar <- par(mfrow = c(4, 1), mar = c(0, 0, 0, 0), oma = c(4,
  6.5, 2, 6))

soe.plot(SOE.data, "Time", "Piscivore Spring Biomass Index MAB",
  stacked = "A", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Planktivore Spring Biomass Index MAB",
  stacked = "B", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Benthivore Spring Biomass Index MAB",
  stacked = "C", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
soe.plot(SOE.data, "Time", "Benthos Spring Biomass Index MAB",
  stacked = "D", rel.y.num = 1.1, end.start = 2007, cex.stack = 1.5)
# soe.plot(SOE.data, 'Time', 'Total Spring Biomass MAB',
# stacked = 'E', status = F, endshade = T, rel.y.num = 1.1,
# full.trend = T, scale.axis = 1)
soe.stack.axis("Year", expression("Biomass, kg tow"^-1 * "")),
  y.line = 3, x.line = 2.7)

```

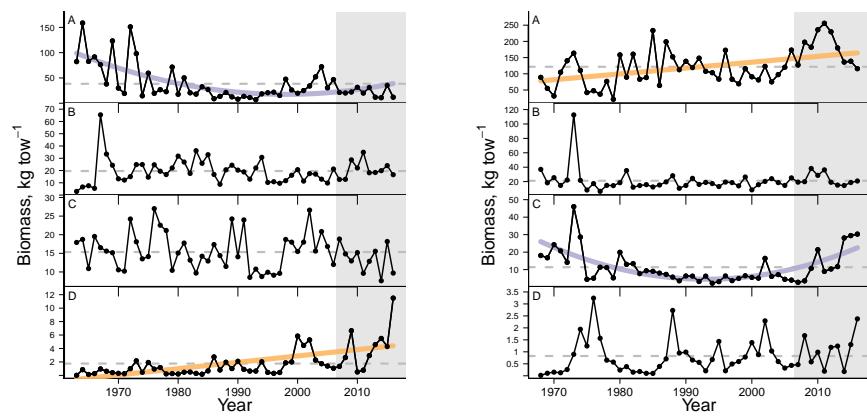


Figure 32.1: Fall (left) and spring (right) MAB Survey Biomass (A: Piscivore, B: Planktivore, C: Benthivore, D: Benthos).

Chapter 33

Thermal Habitat Projections

Description: Species Thermal Habitat Projections

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018),
State of the Ecosystem - Mid-Atlantic (2018)

Indicator category: Published methods

Contributor(s): Vincent Saba

Data steward: Vincent Saba, vincent.saba@noaa.gov

Point of contact: Vincent Saba, vincent.saba@noaa.gov

Public availability statement: Source data are available to the public. Model outputs for thermal habitat projections are available here.

33.1 Methods

This indicator is based on work reported in Kleisner et al. (2017).

33.1.1 Data sources

33.1.1.1 Global Climate Model Projection

We used National Oceanographic and Atmospheric Administration's Geophysical Fluid Dynamics Laboratory (NOAA GFDL) CM2.6 simulation consisting of (1) a 1860 pre-industrial control, which brings the climate system into near-equilibrium with 1860 greenhouse gas concentrations, and (2) a transient climate

response (2xCO₂) simulation where atmospheric CO₂ is increased by 1% per year, which results in a doubling of CO₂ after 70 years. The climate change response from CM2.6 was based on the difference between these two experimental runs. Refer to Saba et al. (2016) for further details.

33.1.1.2 Modeling Changes in Suitable Thermal Habitat

The NOAA Northeast Fisheries Science Center, U.S. Northeast Shelf (NES) bottom trawl survey, which has been conducted for almost 50-years in the spring and fall, provides a rich source of data on historical and current marine species distribution, abundance, and habitat, as well as oceanographic conditions (Azarowitz 1981). The survey was implemented to meet several objectives: (1) monitor trends in abundance, biomass, and recruitment, (2) monitor the geographic distribution of species, (3) monitor ecosystem changes, (4) monitor changes in life history traits (e.g., trends in growth, longevity, mortality, and maturation, and food habits), and (5) collect baseline oceanographic and environmental data. These data can be leveraged for exploring future changes in the patterns of abundance and distribution of species in the region.

33.1.2 Data analysis

33.1.2.1 Global Climate Model Projection

The CM2.6 80-year projections can be roughly assigned to a time period by using the International Panel on Climate Change (IPCC) Representative Concentration Pathways (RCPs), which describe four different 21st century pathways of anthropogenic greenhouse gas emissions, air pollutant emissions, and land use (IPCC 2014). There are four RCPs, ranging from a stringent mitigation scenario (RCP2.6), two intermediate scenarios (RCP4.5 and RCP6.0), and one scenario with very high greenhouse gas emissions (RCP8.5). For RCP8.5, the global average temperature at the surface warms by 2C by approximately 2060-2070 relative to the 1986-2005 climatology (see Figure SPM.7a in IPCC, 2013). For CM2.6, the global average temperature warms by 2C by approximately years 60-80 (see Fig. 1 in Winton et al. (2014)). Therefore, the last 20 years of the transient climate response simulation roughly corresponds to 2060-2080 of the RCP8.5 scenario.

Here, the monthly differences in surface and bottom temperatures ('deltas') for spring (February-April) and fall (September- November) are added to an average annual temperature climatology for spring and fall, respectively, derived from observed surface and bottom temperatures to produce an 80-year time series of future bottom and surface temperatures in both seasons. The observed temperatures come from the NEFSC spring and fall bottom trawl surveys conducted from 1968 to 2013 and represent approximately 30,000 observations over the time series.

33.1.2.2 Modeling Changes in Suitable Thermal Habitat

We modeled individual species thermal habitat across the whole U.S. NES and not by sub-region because we did not want to assume that species would necessarily maintain these assemblages in the future. Indeed, the goal here is to determine future patterns of thermal habitat availability for species on the U.S. NES in more broad terms. We fit one generalzaed additive model (GAM) based on both spring and fall data (i.e., an annual model as opposed to separate spring and fall models) and use it to project potential changes in distribution and magnitude of biomass separately for each season for each species. By creating a single annual model based on temperature data from both spring and fall, we ensure that the full thermal envelope of each species is represented. For example, if a species with a wide thermal tolerance has historically been found in cooler waters in the spring, and in warmer waters in the fall, an annual model will ensure that if there are warmer waters in the spring in the future, that species will have the potential to inhabit those areas. Additionally, because the trawl survey data are subject to many zero observations, we use delta-lognormal GAMs (Wood 2011), which model presence-absence separately from logged positive observations. The response variables in each of the GAMs are presence/absence and logged positive biomass of each assemblage or individual species, respectively. A binomial link function is used in the presence/absence models and a Gaussian link function is used in the models with logged positive biomass. The predictor variables are surface and bottom temperature and depth (all measured by the survey at each station), fit with penalized regression splines, and survey stratum, which accounts for differences in regional habitat quality across the survey region. Stratum may be considered to account for additional information not explicitly measured by the survey (e.g., bottom rugosity). Predictions of species abundance are calculated as the product of the predictions from the presence-absence model, the exponentiated predictions from the logged positive biomass model, and a correction factor to account for the retransformation bias associated with the log transformation (Duan 1983; and see Pinsky et al. 2013).

We calculated the suitable thermal habitat both in terms of changes in ‘suitable thermal abundance’, defined as the species density possible given appropriate temperature, depth and bathymetric conditions, and changes in ‘suitable thermal area’, defined as the size of the physical area potentially occupied by a species given appropriate temperature, depth and bathymetric conditions. Suitable thermal abundance is determined from the predictions from the GAMs (i.e., a prediction of biomass). However, this quantity should not be interpreted directly as a change in future abundance or biomass, but instead as the potential abundance of a species in the future given changes in temperature and holding all else (e.g., fishing effort, species interactions, productivity, etc.) constant. Suitable thermal area is determined as a change in the suitable area that a species distribution occupies in the future and is derived from the area of the kernel density of the distribution. To ensure that the estimates are conservative, we select all points with values greater than one standard deviation above the

mean. We then compute the area of these kernels using the `gArea` function from the `rgeos` package in R (Bivand et al. 2011).

33.1.3 Plotting

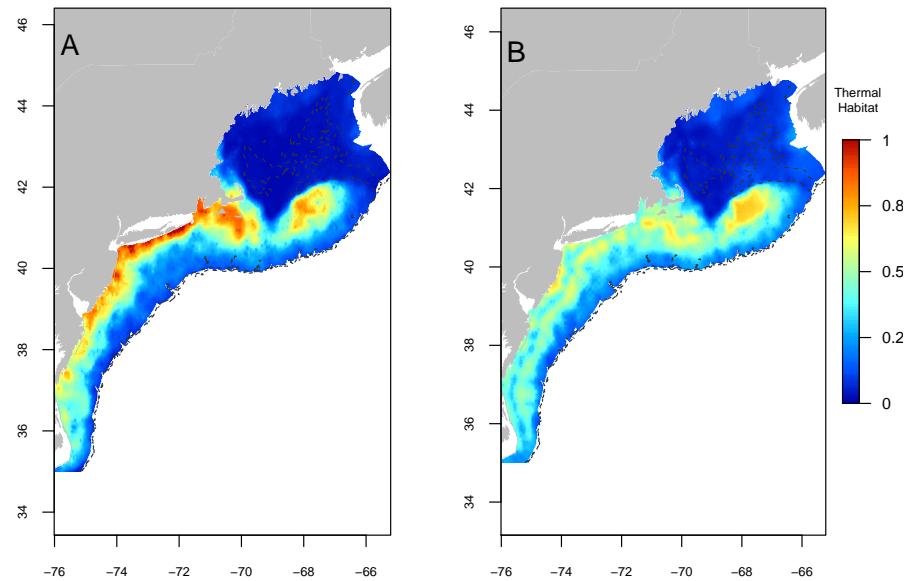


Figure 33.1: Current thermal habitat estimate (A), and 20-40 year thermal habitat projection (B) for summer flounder on the Northeast Continental Shelf.

Note: The thermal habitat model output for all species presented in State of the Ecosystem reports is accessible through the NEFSC ERDDAP server.

Chapter 34

Trend Analysis

Description: Time series trend analysis

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2018, 2019), State of the Ecosystem - Mid-Atlantic (2018, 2019)

Indicator category: Extensive analysis, not yet published

Contributor(s): Sean Hardison, Charles Perretti, Geret DePiper

Data steward: NA

Point of contact: Kimberly Bastille, kimberly.bastille@noaa.gov

Public availability statement: NA

34.1 Methods

Summarizing trends for ecosystem indicators is desirable, but the power of statistical tests to detect a trend is hampered by low sample size and autocorrelated observations (see Nicholson and Jennings 2004; Wagner et al. 2013; Storch 1999). Prior to 2018, time series indicators in State of the Ecosystem reports were presented with trend lines based on a Mann-Kendall test for monotonic trends to test significance ($p < 0.05$) of both long term (full time series) and recent (2007–2016) trends, although not all time series were considered for trend analysis due to limited series lengths. There was also concern that a Mann-Kendall test would not account for any autocorrelation present in State of the Ecosystem (SOE) indicators.

In a simulation study (Hardison et al. 2019), we explored the effect of time series length and autocorrelation strength on statistical power of three trend detection methods: a generalized least squares model selection approach, the Mann-

Kendall test, and Mann-Kendall test with trend-free pre-whitening. Methods were applied to simulated time series of varying trend and autocorrelation strengths. Overall, when sample size was low ($N = 10$) there were high rates of false trend detection, and similarly, low rates of true trend detection. Both of these forms of error were further amplified by autocorrelation in the trend residuals. Based on these findings, we selected a minimum series length of $N = 30$ for indicator time series before assessing trend.

We also chose to use a GLS model selection (GLS-MS) approach to evaluate indicator trends in the 2018 (and future) State of the Ecosystem reports, as this approach performed best overall in the simulation study. GLS-MS also allowed for both linear and quadratic model fits and quantification of uncertainty in trend estimates. The model selection procedure for the GLS approach fits four models to each time series and selects the best fitting model using AICc. The models are, 1) linear trend with uncorrelated residuals, 2) linear trend with correlated residuals, 3) quadratic trend with uncorrelated residuals, and 4) quadratic trend with correlated residuals. I.e., the models are of the form

$$Y_t = \alpha_0 + \alpha_1 X_t + \alpha_2 X_t^2 + \epsilon_t$$

$$\epsilon_t = \rho \epsilon_{t-1} + \omega_t$$

$$\omega_t \sim N(0, \sigma^2)$$

Where Y_t is the observation in time t , X_t is the time index, ϵ_t is the residual in time t , and ω_t is a normally distributed random variable. Setting $\alpha_2 = 0$ yields the linear trend model, and $\rho = 0$ yields the uncorrelated residuals model.

The best fit model was tested against the null hypothesis of no trend through a likelihood ratio test ($p < 0.05$). All models were fit using the R package `nlme` (Pinheiro et al. 2017) and AICc was calculated using the R package `AICcmodavg` (Mazerolle 2017). In SOE time series figures, significant positive trends were colored orange, and negative trends purple.

34.1.1 Data source(s)

NA

34.1.2 Data extraction

NA

34.1.3 Data analysis

```

# R packages
library(dplyr)
library(nlme)
library(AICcmodavg)
library(data.table)

# data.dir <- "./data"
# load(file.path(data.dir, "SOE_data_2018.Rdata"))

#-----GLS Model Selection-----#
fit_lm <- function(dat) {

  constant_norm <-
    nlme::gls(series ~ 1,
               data = dat)

  constant_ar1 <-
    try(nlme::gls(series ~ 1,
                  data = dat,
                  correlation = nlme::corAR1(form = ~time)))
  if (class(constant_ar1) == "try-error"){
    return(best_lm <- data.frame(model = NA,
                                   aicc = NA,
                                   coefs..Intercept = NA,
                                   coefs.time = NA,
                                   coefs.time2 = NA,
                                   pval = NA))
  }

  # Linear model with normal error
  linear_norm <-
    nlme::gls(series ~ time,
               data = dat)

  # Linear model with AR1 error
  linear_ar1 <-
    try(nlme::gls(series ~ time,
                  data = dat,
                  correlation = nlme::corAR1(form = ~time)))
  if (class(linear_ar1) == "try-error"){

```

```

    return(best_lm <- data.frame(model = NA,
                                   aicc = NA,
                                   coefs..Intercept = NA,
                                   coefs.time = NA,
                                   coefs.time2 = NA,
                                   pval = NA))

}

# Polynomial model with normal error
dat$time2 <- dat$time^2
poly_norm <-
  nlme::gls(series ~ time + time2,
             data = dat)

# Polynomial model with AR1 error
poly_ar1 <-
  try(nlme::gls(series ~ time + time2,
                 data = dat,
                 correlation = nlme::corAR1(form = ~time)))
if (class(poly_ar1) == "try-error"){
  return(best_lm <- data.frame(model = NA,
                                 aicc = NA,
                                 coefs..Intercept = NA,
                                 coefs.time = NA,
                                 coefs.time2 = NA,
                                 pval = NA))
}

# Calculate AICs for all models
df_aicc <-
  data.frame(model = c("poly_norm",
                      "poly_ar1",
                      "linear_norm",
                      "linear_ar1"),
             aicc = c(AICc(poly_norm),
                      AICc(poly_ar1),
                      AICc(linear_norm),
                      AICc(linear_ar1)),
             coefs = rbind(coef(poly_norm),
                           coef(poly_ar1),
                           c(coef(linear_norm), NA),
                           c(coef(linear_ar1), NA)),
             # Calculate overall signifiance (need to use

```

```

# ML not REML for this
pval = c(anova(update(constant_norm, method = "ML"),
               update(poly_norm, method = "ML"))$`p-value`[2],
          anova(update(constant_ar1, method = "ML"),
               update(poly_ar1, method = "ML"))$`p-value`[2],
          anova(update(constant_norm, method = "ML"),
               update(linear_norm, method = "ML"))$`p-value`[2],
          anova(update(constant_ar1, method = "ML"),
               update(linear_ar1, method = "ML"))$`p-value`[2]))
```

```

best_lm <-
  df_aicc %>%
  dplyr::filter(aicc == min(aicc))
```

```

if (best_lm$model == "poly_norm") {
  model <- poly_norm
} else if (best_lm$model == "poly_ar1") {
  model <- poly_ar1
} else if (best_lm$model == "linear_norm") {
  model <- linear_norm
} else if (best_lm$model == "linear_ar1") {
  model <- linear_ar1
}

return(list(p = best_lm$pval,
            model = model))
}
```

```

#-----Plotting code-----
soe.plot <- function(data, x.var, y.var, x.label = '^', y.label = '^', tol = 0.1,
                      x.start = NA, x.end = NA, end.start = 2008, bg.col = background, mean_line =
                      end.col = recent, stacked = NA, x.line = 2.5, y.line = 3.5, scale.axis = 1,
                      rel.y.num = 1.5, rel.y.text = 1.5, suppressAxis = FALSE, status = F, anomaly
                      endshade = TRUE, full.trend = TRUE, point.cex = 1.5, lwd = 2, ymax = TRUE,
                      y.upper = y.upper, y.lower = y.lower, extra = FALSE, x.var2 = x.var2, y.var2
                      line.forward = FALSE, mean_line.2 = T, cex.stack = 1, website = T) {
```

```

#Select Data
x <- data[Var == y.var, ]
x <- x[order(x[, get(x.var)]), ]
setnames(x, x.var, 'X')
```

```

#Set common time step if necessary
```

```

if(is.na(x.start)) x.start <- min(x[, X])
if(is.na(x.end))   x.end    <- max(x[, X])
x <- x[X >= x.start, ]

#Set up plot parameters
if (ymax == TRUE){
  y.max <- max(x[, Value]) + tol * max(x[, Value])
} else {
  y.max <- as.numeric(y.upper)
}

if (ymin == TRUE){
  y.min <- min(x[, Value]) - tol * abs(min(x[, Value]))
} else if (ymin == FALSE){
  y.min <- as.numeric(y.lower)
}

y.mean <- mean(x[, Value])
y.sd <- sd(x[, Value])

#Plot blank plot
plot(x[X >= x.start, list(X)], xlim = c(x.start, x.end),
      ylim = c(y.min,y.max), xlab = '^', ylab = '^', axes = F)

#Add background
u <- par('usr')
rect(u[1], u[3], u[2], u[4], border = NA, col = bg.col)

#Add end period shading
if (endshade == TRUE){
  rect(end.start - 0.5, u[3], u[2], u[4], border = NA, col = end.col)
}

#Add mean line
if (anomaly == F){
  if (mean_line == TRUE){
    abline(h = y.mean, col = 'grey', lwd = 3, lty = 2)
  }
} else if (anomaly == TRUE){
  abline(h = 0, col = 'grey', lwd = 3, lty = 2)
}

#Add x y lines
abline(h = u[3], lwd=3)

```

```

abline(v = u[1], lwd=3)

#Add data points/lines
points(x[, list(X, Value)], pch = 16, cex = point.cex)
lines( x[, list(X, Value)], lwd = lwd)

#extra lines
if (extra == TRUE){
  x2 <- data[Var == y.var2, ]
  x2 <- x2[order(x2[, get(x.var2)]), ]
  setnames(x2, x.var2, 'X2')
  x2 <- x2[X2 >= x.start, ]
  if (mean_line.2 == TRUE){
    abline(h = mean(x2[, Value]), col = 'lightcoral', lwd = 3, lty = 2)
  }
  points(x2[, list(X2, Value)], pch = 16, cex = point.cex, col = "indianred")
  lines( x2[, list(X2, Value)], lwd = lwd, col = "indianred")
}

#Add axis
if (suppressAxis == FALSE){
  if(is.na(stacked)) axis(1, cex.axis = 1)
  if(!is.na(stacked)){
    if(stacked != 'A') axis(3, cex.axis = 1.5, tck = 0.1, labels = F)
  }
}

#Stacked axes with 0 overlap so need to remove
labels <- round((axTicks(2) / scale.axis), 5)
if(labels[1] == 0) labels[1] <- ''
axis(2, at = axTicks(2), labels = labels, cex.axis = rel.y.num,
     las = T)

#Add axis labels
if(!website){
  if(!is.na(stacked)) text(u[1], u[4], labels = stacked, cex = cex.stacked, adj = c(-0.5, 1.5))
} else if (website){
  text(u[1], u[4], labels = "", cex = cex.stacked, adj = c(-0.5, 1.5))
}
if(is.na(stacked)){
  mtext(1, text = x.label, line = x.line, cex = 1)
  mtext(2, text = y.label, line = y.line, cex = rel.y.text)
}

```

```

if (full.trend == T){
  #Split data into past decade and full time series
  dat <- as.data.frame(x[, list(X, Value)])

  dat <- dat %>% dplyr::rename(series = Value) %>%
    mutate(time = seq(1,nrow(dat),1))

  # Fit linear model
  lm_out <- fit_lm(dat = dat)
  p <- lm_out$p
  if (p < .05){

    newtime <- seq(min(dat$time), max(dat$time), length.out=length(dat$time))
    newdata <- data.frame(time = newtime,
                           time2 = newtime^2)
    lm_pred <- AICcmodavg::predictSE(lm_out$model,
                                       newdata = newdata,
                                       se.fit = TRUE)

    year <- seq(x$X[1],x$X[length(x$X)],length.out = length(dat$time))

    # Make plot
    if (lm_pred$fit[length(lm_pred$fit)] > lm_pred$fit[1]){
      lines(year, lm_pred$fit, col = main.pos, lwd = 7)
      points(x[, list(X, Value)], pch = 16, cex = point.cex)
      lines( x[, list(X, Value)], lwd = lwd)

      if (line.forward == TRUE){
        lines(year, lm_pred$fit, col = main.pos, lwd = 7)
      }
    } else if (lm_pred$fit[length(lm_pred$fit)] < lm_pred$fit[1]){
      lines(year, lm_pred$fit, col = main.neg, lwd = 7)
      points(x[, list(X, Value)], pch = 16, cex = point.cex)
      lines( x[, list(X, Value)], lwd = lwd)
      if (line.forward == TRUE){
        lines(year, lm_pred$fit, col = main.neg, lwd = 7)
      }
    }
  }

  if (extra == TRUE){

    # Second variable
    dat <- as.data.frame(x2[, list(X2, Value)])
  }
}

```

```

dat <- dat %>% dplyr::rename(series = Value) %>%
  mutate(time = seq(1,nrow(dat),1))

# Fit linear model
lm_out <- fit_lm(dat = dat)
p <- lm_out$p
points(x2[, list(X2, Value)], pch = 16, cex = point.cex, col = "indianred")
lines( x2[, list(X2, Value)], lwd = lwd, col = "indianred")
if (p < .05){

  newtime <- seq(min(dat$time), max(dat$time), length.out=length(dat$time))
  newdata <- data.frame(time = newtime,
                         time2 = newtime^2)
  lm_pred <- AICcmodavg::predictSE(lm_out$model,
                                    newdata = newdata,
                                    se.fit = TRUE)

  year <- seq(x2$X2[1],x2$X2[length(x2$X2)],length.out =length(dat$time))

  # Make plot
  if (lm_pred$fit[length(lm_pred$fit)] > lm_pred$fit[1] ){
    lines(year, lm_pred$fit, col = main.pos, lwd = 7)
    points(x2[, list(X2, Value)], pch = 16, cex = point.cex, col = "indianred")
    lines( x2[, list(X2, Value)], lwd = lwd, col = "indianred")
  } else if (lm_pred$fit[length(lm_pred$fit)] < lm_pred$fit[1]){
    lines(year, lm_pred$fit, col = main.neg, lwd = 7)
    points(x2[, list(X2, Value)], pch = 16, cex = point.cex, col = "indianred")
    lines( x2[, list(X2, Value)], lwd = lwd, col = "indianred")
  }
}

}

#Add axis labels for stacked plots
soe.stacked.axis <- function(x.label, y.label, x.line = 2.5, rel.x.text = 1.5,
                           y.line = 3.5, rel.y.text = 1.5, outer = TRUE){
  axis(1, cex.axis = rel.x.text)
  mtext(1, text = x.label, line = x.line, cex = rel.x.text, outer = outer)
  mtext(2, text = y.label, line = y.line, cex = rel.y.text, outer = outer)
}

```

```
}

#Background colors
background    <- 'white'
recent        <- '#E6E6E6'
main.pos <- rgb(253/255, 184/255, 99/255, alpha = .9)
main.neg <- rgb(178/255, 171/255, 210/255, alpha = .9)
```

Example plot

```
data.dir <- "data"
load(file.path(data.dir, "SOE_data_erddap.Rdata"))

opar <- par(mar = c(4, 6, 2, 6))

soe.plot(SOE.data, "Time", "Mid-Atlantic Rec catch", scale.axis = 10^6,
         end.start = 2008, x.label = "Year", rel.y.text = 1.5, rel.y.num = 1.1,
         y.line = 2.5, y.label = expression("Fish caught, 10^6 *"
                                           " n"))
```

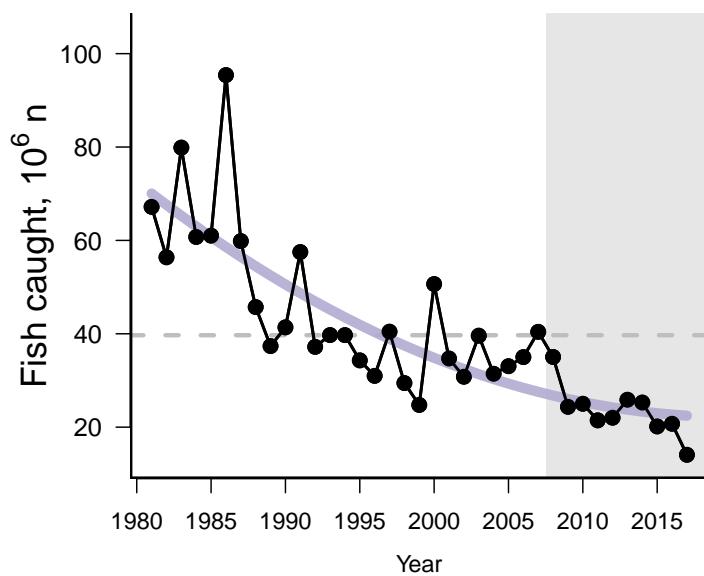


Figure 34.1: Example plot formatted for the report

Chapter 35

Zooplankton

Description: Annual time series of zooplankton abundance

Found in: State of the Ecosystem - Gulf of Maine & Georges Bank (2017, 2018, 2019), State of the Ecosystem - Mid-Atlantic (2017, 2018, 2019)

Indicator category: Database pull with analysis; Synthesis of published information; Extensive analysis, not yet published; Published methods

Contributor(s): Ryan Morse, Kevin Friedland

Data steward: Harvey Walsh, harvey.walsh@noaa.gov; Mike Jones, michael.jones@noaa.gov

Point of contact: Ryan Morse, ryan.morse@noaa.gov; Harvey Walsh, harvey.walsh@noaa.gov; Kevin Friedland, kevin.friedland@noaa.gov

Public availability statement: Source data are publicly available here. Derived data can be found here.

35.1 Methods

35.1.1 Data sources

Zooplankton data are from the National Oceanographic and Atmospheric Administration Marine Resources Monitoring, Assessment and Prediction (MARMAP) program and Ecosystem Monitoring (EcoMon) cruises detailed extensively in Kane (2007), Kane (2011), and Morse et al. (2017).

35.1.2 Data extraction

Data are from the publicly available zooplankton dataset on the NOAA File Transfer Protocol (FTP) server. The excel file has a list of excluded samples and cruises based on Kane (2007) and Kane (2011).

R code used in extraction process.

```
# load data
URL = "ftp://ftp.nefsc.noaa.gov/pub/hydro/zooplankton_data/EcoMon_Plankton_Data_v3_0.xlsx"
ZPD = openxlsx::read.xlsx(URL, sheet = "Data")
```

35.1.3 Data analysis

Annual abundance anomalies

Data are processed similarly to Kane (2007) and Perretti et al. (2017b), where a mean annual abundance by date is computed by area for each species meeting inclusion metrics set in Morse et al. (2017). This is accomplished by binning all samples for a given species to bi-monthly collection dates based on median cruise date and taking the mean, then fitting a spline interpolation between mean bi-monthly abundance to give expected abundance on any given day of the year.

Abundance anomalies (Figure 35.1) are computed from the expected abundance on the day of sample collection. Abundance anomaly time series are constructed for *Centropages typicus*, *Pseudocalanus* spp., *Calanus finmarchicus*, and total zooplankton biovolume. The small-large copepod size index is computed by averaging the individual abundance anomalies of *Pseudocalanus* spp., *Centropages hamatus*, *Centropages typicus*, and *Temora longicornis*, and subtracting the abundance anomaly of *Calanus finmarchicus*. This index tracks the overall dominance of the small bodied copepods relative to the largest copepod in the Northeast U.S. region, *Calanus finmarchicus*.

```
# libraries
library(vegan)
library(stats)
library(mgcv)
library(reshape2)
library(readxl)
library(lubridate)
library(sp)
library(maptools)
library(marmap)
library(rgeos)
```

```

# load data
URL = "ftp://ftp.nefsc.noaa.gov/pub/hydro/zooplankton_data/EcoMon_Plankton_Data_v3_0.xlsx"
ZPD = openxlsx::read.xlsx(URL, sheet = "Data")
# Fix date, time
dt = as_date(ZPD$date, origin = "1899-12-30")
DOY = yday(dt) #day of year
month = as.numeric(format(dt, "%m"))
year = as.numeric(format(dt, "%Y"))
ZPD$year = year
ZPD$month = month
ZPD$dt = dt
ZPD$DOY = DOY
ZPD$day = as.numeric(format(dt, "%d"))
ZPD$lat2 = ceiling(ZPD$lat) #use for binning into 1 degree bins for removal of undersampled bins
ZPD$lon2 = floor(ZPD$lon) #use for binning into 1 degree bins for removal of undersampled bins
# ASSIGN EPU based on GPS data load shapefiles from EDAB EPU
# analysis ## not available here

gbk = readShapeSpatial("EPU_GBKPoly.shp")
gom = readShapeSpatial("EPU_GOMPoly.shp")
mab = readShapeSpatial("EPU_MABPoly.shp")
scs = readShapeSpatial("EPU_SCSPoly.shp")
# combine shapefiles GOM and GBK
gom.gbk.shp = gUnion(gom, gbk, byid = F, id = NULL)
gom.gbk.shp = gUnion(gom, gbk, byid = F, id = NULL)
gom.scs.shp = gUnion(gom, scs, byid = F, id = NULL)
mab.gbk.shp = gUnion(mab, gbk, byid = F, id = NULL)
NES.shp = gUnion(mab.gbk.shp, gom.scs.shp, byid = F, id = NULL)
# extract just lat/lons for lines
gbk.lonlat = as.data.frame(lapply(slot(gbk, "polygons"), function(x) lapply(slot(x,
    "Polygons"), function(y) slot(y, "coords"))))
gom.lonlat = as.data.frame(lapply(slot(gom, "polygons"), function(x) lapply(slot(x,
    "Polygons"), function(y) slot(y, "coords"))))
mab.lonlat = as.data.frame(lapply(slot(mab, "polygons"), function(x) lapply(slot(x,
    "Polygons"), function(y) slot(y, "coords"))))
scs.lonlat = as.data.frame(lapply(slot(scs, "polygons"), function(x) lapply(slot(x,
    "Polygons"), function(y) slot(y, "coords"))))
gom.gbk.lonlat = as.data.frame(lapply(slot(gom.gbk.shp, "polygons"),
    function(x) lapply(slot(x, "Polygons"), function(y) slot(y,
        "coords"))))
NES.lonlat = as.data.frame(NES.shp@polygons[[1]]@Polygons[[1]]@coords) # create matrix to use in in.out function from package 'mgcv'
# create matrix to use in in.out function from package 'mgcv'
gom.mat = as.matrix(gom.lonlat)
gbk.mat = as.matrix(gbk.lonlat)
mab.mat = as.matrix(mab.lonlat)

```

```

scs.mat = as.matrix(scs.lonlat)
gom.gbk.mat = as.matrix(gom.gbk.lonlat)
# assign samples to EPU
m4 = as.matrix(ZPD[, 6:5]) #lon, lat from ZPD
ZPD$epu = NA
ZPD$epu[which(in.out(gbk.mat, m4))] = "GBK"
ZPD$epu[which(in.out(gom.mat, m4))] = "GOM"
ZPD$epu[which(in.out(scs.mat, m4))] = "SCS"
ZPD$epu[which(in.out(mab.mat, m4))] = "MAB"
test = ZPD[is.na(ZPD$epu), ] #unassigned
nms = data.frame(colnames(ZPD)) # column names of orginal data
# limit data set to zooplankton from 1977 on
ZPDb = ZPD[, c(seq(1, 14, 1), seq(290, 297, 1), seq(106, 197,
1))] # check to make sure these are correct against 'nms' if data source changes!
ZPDb = ZPDb[order(ZPDb$date), ]
ZPDb = ZPDb[which(ZPDb$year > 1976), ] # remove NA data in years prior to 1977
# Select only taxa present in yearly data > x percent of
# samples
X = 20 # percent criteria to use as minimum percent in samples
ZPDa = ZPDb
ZPDa = ZPDa[!is.na(ZPDa$zoo_gear), ] # Remove NA in zooplankton rows
# Reduce to taxa occurrence > x percent in samples
p.a = ZPDa[, 24:114]
p.a[p.a > 0] = 1 # presence/absence
count = colSums(p.a)
pct = (count/dim(ZPDa)[1]) * 100
crit = which(pct > X)
ZPDa = ZPDa[c(1:23, crit + 23)] # data limited to taxa occurring in > X percent of samples

# Take median date from each cruise and assign cruise to
# bimonth for bi-monthly means aggregation
cruises = unique(ZPDa$cruise_name)
for (i in 1:length(cruises)) {
    ZPDa$medmonth[ZPDa$cruise_name == cruises[i]] = median(ZPDa$DOY[ZPDa$cruise_name ==
cruises[i]])
}
ZPDa$bmm = NA
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(0, 59))] = 1
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(60, 120))] = 3
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(121, 181))] = 5
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(182, 243))] = 7
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(244, 304))] = 9
ZPDa$bmm[which(as.integer(ZPDa$medmonth) %in% seq(305, 366))] = 11

ZPDa[, 14] = as.numeric(ZPDa[, 14])

```

```

ZPDA[, 23] = as.numeric(ZPDA[, 23])
ZPDsave = ZPDA #from above routine, Yearly (all data)

SEASON = "Yearly"
ZPDA = ZPDsave
# LOG transform data using ZPDA from above (select season
# first)
test = log10(ZPDA[, 24:50] + 1) #choose columns with zooplankton data
ZPDlog = ZPDA
ZPDlog[, 24:50] = test
nm = matrix(colnames(ZPDlog))

area = "GBK"
gbk.yr.spln = data.frame()
for (i in 23:50) {
  num = i
  name = nm[num, 1]
  mean.loc.x = aggregate(ZPDlog[which(ZPDlog$epu == area),
    num], by = list(ZPDlog$bmm[which(ZPDlog$epu == area)]),
    FUN = mean, na.rm = T)
  func = splinefun(mean.loc.x[, 1], y = mean.loc.x[, 2], method = "natural",
    ties = mean)
  x.daily = func(seq(1, 12, 0.0302)) #365 days
  gbk.yr.spln = rbind(gbk.yr.spln, x.daily)
}
gbk.yr.spln = t(gbk.yr.spln)
rownames(gbk.yr.spln) = seq(1:365)
colnames(gbk.yr.spln) = nm[23:50, 1]

area = "GOM"
gom.yr.spln = data.frame()
for (i in 23:50) {
  num = i
  name = nm[num, 1]
  mean.loc.x = aggregate(ZPDlog[which(ZPDlog$epu == area),
    num], by = list(ZPDlog$bmm[which(ZPDlog$epu == area)]),
    FUN = mean, na.rm = T)
  func = splinefun(mean.loc.x[, 1], y = mean.loc.x[, 2], method = "natural",
    ties = mean)
  x.daily = func(seq(1, 12, 0.0302)) #365 days
  gom.yr.spln = rbind(gom.yr.spln, x.daily)
}
gom.yr.spln = t(gom.yr.spln)
rownames(gom.yr.spln) = seq(1:365)
colnames(gom.yr.spln) = nm[23:50, 1]

```

```

area = "MAB"
mab.yr.spln = data.frame()
for (i in 23:50) {
  num = i
  name = nm[num, 1]
  mean.loc.x = aggregate(ZPDlog[which(ZPDlog$epu == area),
    num], by = list(ZPDlog$bmm[which(ZPDlog$epu == area)]),
    FUN = mean, na.rm = T)
  func = splinefun(mean.loc.x[, 1], y = mean.loc.x[, 2], method = "natural",
    ties = mean)
  x.daily = func(seq(1, 12, 0.0302)) #365 days
  mab.yr.spln = rbind(mab.yr.spln, x.daily)
}
mab.yr.spln = t(mab.yr.spln)
rownames(mab.yr.spln) = seq(1:365)
colnames(mab.yr.spln) = nm[23:50, 1]

area = "SCS"
scs.yr.spln = data.frame()
for (i in 23:50) {
  num = i
  name = nm[num, 1]
  mean.loc.x = aggregate(ZPDlog[which(ZPDlog$epu == area),
    num], by = list(ZPDlog$bmm[which(ZPDlog$epu == area)]),
    FUN = mean, na.rm = T)
  func = splinefun(mean.loc.x[, 1], y = mean.loc.x[, 2], method = "natural",
    ties = mean)
  x.daily = func(seq(1, 12, 0.0302)) #365 days
  scs.yr.spln = rbind(scs.yr.spln, x.daily)
}
scs.yr.spln = t(scs.yr.spln)
rownames(scs.yr.spln) = seq(1:365)
colnames(scs.yr.spln) = nm[23:50, 1]

# Subtract mean expected value from observed abundance to get
# anomaly
gbk.anom = ZPDlog[which(ZPDlog$epu == "GBK"), ]
gom.anom = ZPDlog[which(ZPDlog$epu == "GOM"), ]
mab.anom = ZPDlog[which(ZPDlog$epu == "MAB"), ]
scs.anom = ZPDlog[which(ZPDlog$epu == "SCS"), ]

gbk.anom.b = data.frame(matrix(NA, nrow = dim(gbk.anom)[1], ncol = dim(gbk.anom)[2]))
for (i in 1:dim(gbk.anom)[1]) {
  gbk.anom.b[i, 23:50] = gbk.anom[i, 23:50] - gbk.yr.spln[which(gbk.anom$DOY[i] ==
    rownames(gbk.yr.spln)), ]

```

```

}

gom.anom.b = data.frame(matrix(NA, nrow = dim(gom.anom)[1], ncol = dim(gom.anom)[2]))
for (i in 1:dim(gom.anom)[1]) {
  gom.anom.b[i, 23:50] = gom.anom[i, 23:50] - gom.yr.spln[which(gom.anom$DOY[i] ==
    rownames(gom.yr.spln)), ]
}
mab.anom.b = data.frame(matrix(NA, nrow = dim(mab.anom)[1], ncol = dim(mab.anom)[2]))
for (i in 1:dim(mab.anom)[1]) {
  mab.anom.b[i, 23:50] = mab.anom[i, 23:50] - mab.yr.spln[which(mab.anom$DOY[i] ==
    rownames(mab.yr.spln)), ]
}
scs.anom.b = data.frame(matrix(NA, nrow = dim(scs.anom)[1], ncol = dim(scs.anom)[2]))
for (i in 1:dim(scs.anom)[1]) {
  scs.anom.b[i, 23:50] = scs.anom[i, 23:50] - scs.yr.spln[which(scs.anom$DOY[i] ==
    rownames(scs.yr.spln)), ]
}

scs.anom.b = rbind(scs.anom.b, test)
# Aggregate by Year, Yearly anomaly by epu
gbk.yr.anom = aggregate(gbk.anom.b, by = list(gbk.anom$year),
  FUN = mean, na.rm = T)
rownames(gbk.yr.anom) = gbk.yr.anom[, 1]
gbk.yr.anom[, 1] = NULL
colnames(gbk.yr.anom) = colnames(gbk.anom)
gom.yr.anom = aggregate(gom.anom.b, by = list(gom.anom$year),
  FUN = mean, na.rm = T)
rownames(gom.yr.anom) = gom.yr.anom[, 1]
gom.yr.anom[, 1] = NULL
colnames(gom.yr.anom) = colnames(gom.anom)
mab.yr.anom = aggregate(mab.anom.b, by = list(mab.anom$year),
  FUN = mean, na.rm = T)
rownames(mab.yr.anom) = mab.yr.anom[, 1]
mab.yr.anom[, 1] = NULL
colnames(mab.yr.anom) = colnames(mab.anom)
scs.yr.anom = aggregate(scs.anom.b, by = list(scs.anom$year),
  FUN = mean, na.rm = T)
rownames(scs.yr.anom) = scs.yr.anom[, 1]
scs.yr.anom[, 1] = NULL
colnames(scs.yr.anom) = colnames(scs.anom)

# Small-Large body copepod abundance anomaly
lgtx = c(25) #column for Calanus finmarchicus
smtx = c(26, 24, 28, 27) #Pseudocalanus spp, Centropages typicus, Centropages hamatus, Temora longicornis
dataTsm = gbk.yr.anom[, smtx]

```

```

dataTlg = gbk.yr.anom[, lgtx]
anom.gbk = rowMeans(dataTsm) - (dataTlg)
dataTsm = gom.yr.anom[, smtx]
dataTlg = gom.yr.anom[, lgtx]
anom.gom = rowMeans(dataTsm) - (dataTlg)
dataTsm = mab.yr.anom[, smtx]
dataTlg = mab.yr.anom[, lgtx]
anom.mab = rowMeans(dataTsm) - (dataTlg)
dataTsm = scs.yr.anom[, smtx]
dataTlg = scs.yr.anom[, lgtx]
anom.scs = rowMeans(dataTsm) - (dataTlg)

```

Seasonal abundance

Time series of zooplankton abundance in the spring and fall months have been presented in the 2019 Mid-Atlantic State of the Ecosystem report. Raw abundance data were sourced from the EcoMon cruises referenced above, and ordinary kriging was used to estimate seasonal abundance over the Northeast Shelf. These data were then aggregated further into time series of mean abundance by Ecological Production Unit. These data are presented in Figure 35.2.

35.1.4 Data processing

Zooplankton abundance indicators were formatted for inclusion in the `ecodata` R package using the following code.

Abundance anomaly

```

# Process zooplankton abundance anomalies and small-large
# index

library(dplyr)
library(tidyr)

raw.dir <- here::here("data-raw")

get_zoo_sli_anom <- function(save_clean = F) {

  load(file.path(raw.dir, "zoo_sli_anom.Rdata"))

  zoo_sli_anom <- test %>% dplyr::rename(EPU = variable, Value = value)

  if (save_clean) {
    usethis::use_data(zoo_sli_anom, overwrite = T)
  } else {

```

```

        return(zoo_sli_anom)
    }
}
get_zoo_sli_anom(save_clean = T)

```

Seasonal abundance

```

# A general function to process various OI data sets

library(dplyr)
library(tidyr)
library(lubridate)
library(raster)
library(rgdal)
library(sf)

process_oi <- function(variable, type = NULL, season, genus = NULL,
epu) {

  if (!is.null(type) & !variable %in% c("salinity", "temperature")) {
    stop("type only applicable for variables \"salinity\" and \"temperature\""
      as type = \"bottom\" or type = \"surface\"")
  }

  if (!is.null(genus) & variable != "zooplankton") {
    stop("genus only applicable for variable \"zooplankton\"")
  }

  # get compiled down-sampled raster of chosen strata from
  # shapefile.
  epumask.raster <- get(paste0(tolower(epu), "_rast"))

  # get bottom temp data and find mean for stock
  # area-----
  indir <- here::here("inst", "extdata", "gridded")

  if (variable == "salinity") {
    load(file.path(indir, paste0("sal_", type, "_", season,
      "_spdf.rdata")))
  } else if (variable == "temperature") {
    load(file.path(indir, paste0("temp_", type, "_", season,
      "_spdf.rdata")))
  } else if (variable == "chlorophyll") {
    load(file.path(indir, paste0("chl_", season, "_1997-2018.rdata")))
  }
}

```

```

} else if (variable == "zooplankton") {
  load(file.path(indir, paste0(genus, "_", season, "_zoo_1977-2016.rdata")))
}

# create null df to fill with results
data = data.frame(array(NA, dim = c(raster::nlayers(ecsa_dat),
  5)))

# loops through layers in raster brick
for (i in 1:raster::nlayers(ecsa_dat)) {
  # load raster by year

  # get file information from title
  layer_id <- stringr::str_extract(names(ecsa_dat)[[i]],
    "\\\\d.*")
  layer_id <- stringr::str_split(layer_id, "_")
  data[i, 1] <- layer_id[[1]][[1]]
  data[i, 2] <- layer_id[[1]][[2]]
  data[i, 3] <- layer_id[[1]][[3]]

  # trim to stock area
  masked.raster = ecsa_dat[[i]] * epumask.raster

  # find mean BT of stock area
  data[i, 4] = raster::cellStats(masked.raster, stat = "mean",
    na.rm = TRUE)
  data[i, 5] = raster::cellStats(masked.raster, stat = "sd",
    na.rm = TRUE)
  # if (layer_id[[1]][[1]] == '1995'){ break }
}

x <- as.numeric(data$X1)
y.out <- data$X4
y.sd <- data$X5

sd.low <- y.out - y.sd
sd.high <- y.out + y.sd

# remove
if (variable == "zooplankton") {
  if (season == "spring") {
    y.out[x %in% c(1989, 1990, 1991, 1994)] <- NA
    sd.low[x %in% c(1989, 1990, 1991, 1994)] <- NA
    sd.high[x %in% c(1989, 1990, 1991, 1994)] <- NA
  } else if (season == "fall") {
}
}

```

```

y.out[x %in% c(1989, 1990, 1992)] <- NA
sd.low[x %in% c(1989, 1990, 1992)] <- NA
sd.high[x %in% c(1989, 1990, 1992)] <- NA
}
}

if (variable == "chlorophyll") {
  type <- ""
} else if (variable == "zooplankton") {
  type <- genus
  variable <- "zoo"
}

out <- data.frame(Var = paste(paste(type, variable), season),
  Time = as.numeric(x), Value = y.out, sd.low = sd.low,
  sd.high = sd.high, Season = season, epu = epu)

out <- out[out$Time > 1968, ]
return(out)
}

# Process optimally interpolated EcoMon zooplankton data
# These data show estimated zooplankton abundances on the NE
# Shelf derived from Ecosystem Monitoring Program (EcoMon)
# sampling. EcoMon conducts shelf-wide bimonthly surveys of
# the Northeast Large Marine Ecosystem, collecting and
# ichthyoplankton to a depth of 200 m using paired Bongo
# samplers with 333 $\mu$m mesh netting. Zooplankton
# abundance data were interpolated across sampling locations
# using ordinary kriging to create a complete field. Here we
# present abundance time series for three species:
# *Centropages typicus*, *Temora longicornis*, and
# *Pseudocalanus* spp. These data are processed in a similar
# manner to optimally interpolated ocean temperature and
# salinity data sets. More information about the source data
# and processing methods used to derive these data sets can
# be found at
# https://noaa-edab.github.io/ECSA/#sec:methodszoo.

library(tidyverse)

raw.dir <- here::here("data-raw")

```

```

get_zoo_abun_anom <- function(save_clean = F) {

  load(file.path(raw.dir, "zoo_abun_anom.rdata"))

  zoo_abun_anom <- as.tibble(mtest) %>% dplyr::rename(EPU = Region,
    Value = value, Var = variable) %>% mutate(Var = plyr::mapvalues(Var,
    from = c("Pse", "Cty", "Cha", "Tlo", "Cf"), to = c("Pseudocalanus spp.",
    "Centropages typicus", "Centropages hamatus", "Temora longicornis",
    "Calanus finmarchicus")))

  if (save_clean) {
    usethis::use_data(zoo_abun_anom, overwrite = T)
  } else {
    return(zoo_abun_anom)
  }
}
get_zoo_abun_anom(save_clean = T)

source(file.path(r.dir, "process_oi.R"))

if (!all(exists("mab_rast"), exists("gb_rast"), exists("gom_rast"))) {
  message("Loading EPU rasters.")
  source(file.path(r.dir, "create_epu_mask_oi.R"))
} else {
  message("All EPU rasters exist; skipping load step.")
}

get_zoo_oi <- function(save_clean = F) {

  # MAB-
  ctyp_fall_mab <- process_oi(variable = "zooplankton", genus = "centropages",
    season = "fall", epu = "MAB")
  ctyp_spring_mab <- process_oi(variable = "zooplankton", genus = "centropages",
    season = "spring", epu = "MAB")

  tlong_fall_mab <- process_oi(variable = "zooplankton", genus = "temora",
    season = "fall", epu = "MAB")
  tlong_spring_mab <- process_oi(variable = "zooplankton",
    genus = "temora", season = "spring", epu = "MAB")
}

```

```

pseudo_fall_mab <- process_oi(variable = "zooplankton", genus = "pseudocalanus",
                                 season = "fall", epu = "MAB")
pseudo_spring_mab <- process_oi(variable = "zooplankton",
                                   genus = "pseudocalanus", season = "spring", epu = "MAB")

# GB-----
ctyp_fall_gb <- process_oi(variable = "zooplankton", genus = "centropages",
                             season = "fall", epu = "GB")
ctyp_spring_gb <- process_oi(variable = "zooplankton", genus = "centropages",
                               season = "spring", epu = "GB")

tlong_fall_gb <- process_oi(variable = "zooplankton", genus = "temora",
                              season = "fall", epu = "GB")
tlong_spring_gb <- process_oi(variable = "zooplankton", genus = "temora",
                                season = "spring", epu = "GB")

pseudo_fall_gb <- process_oi(variable = "zooplankton", genus = "pseudocalanus",
                               season = "fall", epu = "GB")
pseudo_spring_gb <- process_oi(variable = "zooplankton",
                                 genus = "pseudocalanus", season = "spring", epu = "GB")

# GOM-----
ctyp_fall_gom <- process_oi(variable = "zooplankton", genus = "centropages",
                              season = "fall", epu = "GOM")
ctyp_spring_gom <- process_oi(variable = "zooplankton", genus = "centropages",
                                 season = "spring", epu = "GOM")

tlong_fall_gom <- process_oi(variable = "zooplankton", genus = "temora",
                               season = "fall", epu = "GOM")
tlong_spring_gom <- process_oi(variable = "zooplankton",
                                 genus = "temora", season = "spring", epu = "GOM")

pseudo_fall_gom <- process_oi(variable = "zooplankton", genus = "pseudocalanus",
                                 season = "fall", epu = "GOM")
pseudo_spring_gom <- process_oi(variable = "zooplankton",
                                   genus = "pseudocalanus", season = "spring", epu = "GOM")

# Bind all zoo output. Be sure to include any new species
# here
zoo_oi <- NULL
for (i in ls()) {
  if (stringr::str_detect(i, "ctyp|tlong|pseudo")) {
    assign("zoo_oi", rbind(zoo_oi, get(i)))
}

```

```

    }
}

# Process output from masked rasters (mean, +/- 1 SD). Adjust
# output in process_oi(), not here.
zoo_oi <- zoo_oi %>% dplyr::rename(EPU = epu) %>% dplyr::select(-Season) %>%
  mutate(Units = "log N m^-3")

# Split out +/- SD because too lazy to gather ;)
sd.low <- zoo_oi %>% dplyr::select(-Value, -sd.high) %>%
  mutate(Var = as.factor(paste(Var, "- 1 SD")))) %>% dplyr::rename(Value = sd.low)

sd.high <- zoo_oi %>% dplyr::select(-Value, -sd.low) %>%
  mutate(Var = as.factor(paste(Var, "+ 1 SD")))) %>% dplyr::rename(Value = sd.high)

zoo_oi <- zoo_oi %>% dplyr::select(-sd.high, -sd.low)

# Bind all data
zoo_oi <- rbind(zoo_oi, sd.high, sd.low) %>% as.data.frame()

if (save_clean) {
  usethis::use_data(zoo_oi, overwrite = T)
} else {
  return(zoo_oi)
}

}

```

35.1.5 Plotting

Abundance anomaly

Seasonal Abundance

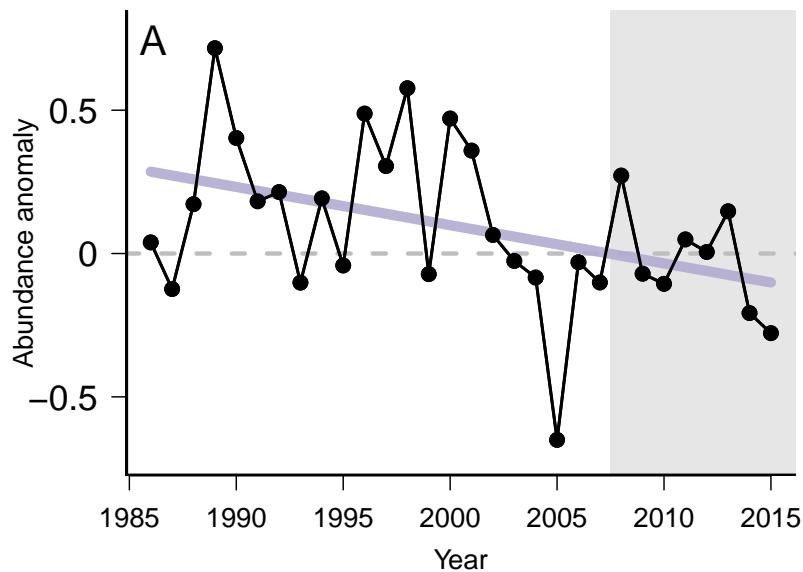


Figure 35.1: *Centropages typicus* abundance anomaly in the Mid-Atlantic Bight.

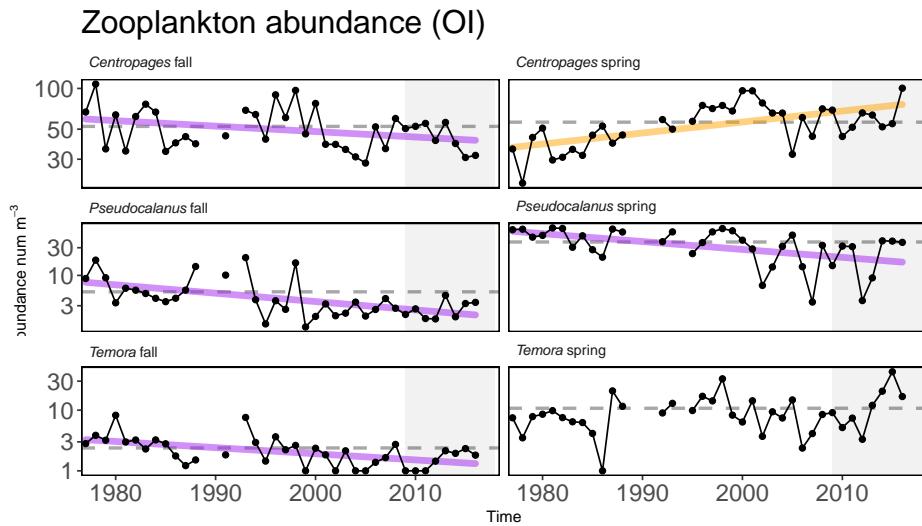


Figure 35.2: Seasonal abundance of key zooplankton species in the MAB.

References

- ACC. 2017. "Maryland Aquaculture Coordinating Council: Annual Report 2017." Annapolis, MD: Maryland Aquaculture Coordinating Council.
- Adams, Robert L. A. 1973. "Uncertainty in Nature, Cognitive Dissonance, and the Perceptual Distortion of Environmental Information: Weather Forecasts and New England Beach Trip Decisions." *Economic Geography* 49 (4): 287–97. <https://doi.org/10.2307/143232>.
- Azarovitz, Thomas R. 1981. "A brief historical review of the Woods Hole Laboratory trawl survey time series." In *Bottom Trawl Surveys*, 62–67. Woods Hole, MA: National Marine Fisheries Service. http://dmoserv3.whoi.edu/data_docs/NEFSC_Bottom_Trawl/Azarovitz1981.pdf.
- Backus, Richard H., and Donald W. Bourne, eds. 1987. *Georges Bank*. Cambridge, MA: The MIT Press.
- Balk, Bert M. 2010. "An assumption-free framework for measuring productivity change." *Review of Income and Wealth* 56 (s1): S224–S256. <https://doi.org/10.1111/j.1475-4991.2010.00388.x>.
- Beardsley, Robert C., David C. Chapman, Kenneth H. Brink, Steven R. Ramp, and Ronald Schlitz. 1985. "The Nantucket Shoals Flux Experiment (NSFE79). Part I: A Basic Description of the Current and Temperature Variability." *Journal of Physical Oceanography* 15 (6): 713–48. [https://doi.org/10.1175/1520-0485\(1985\)015%3C0713:TNSFEP%3E2.0.CO;2](https://doi.org/10.1175/1520-0485(1985)015%3C0713:TNSFEP%3E2.0.CO;2).
- Behrenfeld, Michael J., and Paul G. Falkowski. 1997. "Photosynthetic Rates Derived from Satellite-Based Chlorophyll Concentration." *Limnology and Oceanography* 42 (1): 1–20. <https://doi.org/10.4319/lo.1997.42.1.0001>.
- Bisack, K. D., and G. Magnusson. 2014. "Measuring the Economic Value of Increased Precision in Scientific Estimates of Marine Mammal Abundance and Bycatch." *North American Journal of Fisheries Management* 34 (2): 311–21. <https://doi.org/10.1080/02755947.2013.869281>.
- Bivand, Roger, Tim Keitt, and Barry Rowlingson. 2018. *Rgdal: Bindings for the 'Geospatial' Data Abstraction Library*. <https://CRAN.R-project.org/package=rgdal>.

- Bivand, Roger, Colin Rundel, Edzer Pebesma, and Karl O. Hufthammer. 2011. "rgeos: Interface to Geometry Engine—Open Source (GEOS)." [http://scholar.google.com/scholar?hl=en%7B/&%7DbtnG=Search%7B/%7Dq=intitle:Interface+to+Geometry+Engine++Open+Source+\(GEOS\)%7B/#%7D0](http://scholar.google.com/scholar?hl=en%7B/&%7DbtnG=Search%7B/%7Dq=intitle:Interface+to+Geometry+Engine++Open+Source+(GEOS)%7B/#%7D0).
- Blackwell, Brian G., Michael L. Brown, and David W. Willis. 2000. "Relative Weight (Wr) Status and Current Use in Fisheries Assessment and Management." *Reviews in Fisheries Science* 8 (1): 1–44. <https://doi.org/10.1080/10641260091129161>.
- Blasiak, Robert, James L. Anderson, Peter Bridgewater, Ken Furuya, Benjamin S. Halpern, Hisashi Kurokura, Joji Morishita, Nobuyuki Yagi, and Akane Minohara. 2014. "Paradigms of Sustainable Ocean Management." *Marine Policy* 48: 206–11. <https://doi.org/10.1016/j.marpol.2014.03.021>.
- Boundy, Michael J., Andrew I. Selwood, D. Tim Harwood, Paul S. McNabb, and Andrew D. Turner. 2015. "Development of a sensitive and selective liquid chromatography-mass spectrometry method for high throughput analysis of paralytic shellfish toxins using graphitised carbon solid phase extraction." *Journal of Chromatography A* 1387: 1–12. <https://doi.org/10.1016/j.chroma.2015.01.086>.
- Brown, Shanel Marie. 2015. "'We Have the Best Life There Ever Was': Linking Sense of Place and Adaptive Capacity in Nova Scotia's Coastal Communities." In *Thesis, Masters in Environmental Affairs*. University of Waterloo. <http://hdl.handle.net/10012/9408>.
- Calvo, Lisa M. 2017. "New Jersey Shellfish Aquaculture Situation and Outlook Report: 2015 Production Year." New Jersey Sea Grant Consortium; Haskin Shellfish Research Laboratory, Rutgers University. https://hsrl.rutgers.edu/outreach/aquaculture/AquacultureReports/NJAquaculture%7B/_%7DSurvey%7B/_%7DYr2015.pdf.
- Campbell, Janet W. 1995. "The Lognormal Distribution as a Model for Bio-Optical Variability in the Sea." *Journal of Geophysical Research: Oceans* 100 (C7): 13237–54. <https://doi.org/10.1029/95JC00458>.
- Casey, Kenneth S., Tess B. Brandon, Peter Cornillon, and Robert Evans. 2010. "The past, present, and future of the AVHRR pathfinder SST program." In *Oceanography from Space: Revisited*, 273–87. https://doi.org/10.1007/978-90-481-8681-5_16.
- Castelao, Renato, Scott Glenn, and Oscar Schofield. 2010. "Temperature, Salinity, and Density Variability in the Central Middle Atlantic Bight." *Journal of Geophysical Research: Oceans* 115 (C10). <https://doi.org/10.1029/2009JC006082>.
- Castelao, Renato, Scott Glenn, Oscar Schofield, Robert Chant, John Wilkin, and Josh Kohut. 2008. "Seasonal Evolution of Hydrographic Fields in the

- Central Middle Atlantic Bight from Glider Observations.” *Geophysical Research Letters* 35 (3). <https://doi.org/10.1029/2007GL032335>.
- Chin, Toshio Michael, Jorge Vazquez-Cuervo, and Edward M. Armstrong. 2017. “A Multi-Scale High-Resolution Analysis of Global Sea Surface Temperature.” *Remote Sensing of Environment* 200: 154–69. <https://doi.org/10.1016/j.rse.2017.07.029>.
- Clay, Patricia M., and Julia Olson. 2007. “Defining Fishing Communities: Issues in Theory and Practice.” *Napa Bulletin* 28 (1): 27–42. <https://doi.org/10.1525/napa.2007.28.1.27>.
- . 2008. “Defining‘Fishing Communities’: Vulnerability and the Magnuson-Stevens Fishery Conservation and Management Act.” *Human Ecology Review* 15 (2): 143. <https://www.humanecologyreview.org/pastissues/her152/clayolson.pdf>.
- Colburn, Lisa L., and Michael Jepson. 2012. “Social Indicators of Gentrification Pressure in Fishing Communities: A Context for Social Impact Assessment.” *Coastal Management* 40 (3): 289–300. <https://doi.org/10.1080/08920753.2012.677635>.
- Colburn, Lisa L., Michael Jepson, Changhua Weng, Tarsila Seara, Jeremy Weiss, and Jonathan A. Hare. 2016. “Indicators of Climate Change and Social Vulnerability in Fishing Dependent Communities Along the Eastern and Gulf Coasts of the United States.” *Marine Policy* 74: 323–33. <https://doi.org/10.1016/j.marpol.2016.04.030>.
- Cortes-Vazquez, Jose A., and Morgan Zedalis. 2013. “Identity and Native Species Conservation: Similar Historical Ecologies from Idaho to Spain.” *Human Ecology* 41 (6): 937–45. <https://doi.org/10.1007/s10745-013-9570-3>.
- Cren, E. D. Le. 1951. “The Length-Weight Relationship and Seasonal Cycle in Gonad Weight and Condition in the Perch (*Perca fluviatilis*).” *The Journal of Animal Ecology* 20 (2): 201. <https://doi.org/10.2307/1540>.
- Cross, Robin M, and Rolf Färe. 2009. “Value data and the Bennet price and quantity indicators.” *Economics Letters* 102 (1): 19–21. <https://doi.org/10.1016/j.econlet.2008.10.003>.
- Das, Chhandita. 2013. “Northeast Trip Cost Data - Overview, Estimation, and Predictions.” *NOAA Technical Memorandum NMFS_NE_227*: 20.
- . 2014. “An Overview of the Annual Cost Survey Protocol and Results in the Northeastern Region (2007-2009).” *NOAA Technical Memorandum NMFS-NE-226*: 34.
- Degnbol, P., and A. Jarre. 2004. “Review of Indicators in Fisheries Management – a Development Perspective.” *African Journal of Marine Science* 26 (1): 303–26. <https://doi.org/10.2989/18142320409504063>.

- DePiper, Geret S., Sarah K. Gaichas, Sean M. Lucey, Patricia Pinto da Silva, M. Robin Anderson, Heather Breeze, Alida Bundy, et al. 2017. "Operationalizing Integrated Ecosystem Assessments Within a Multidisciplinary Team: Lessons Learned from a Worked Example." *ICES Journal of Marine Science* 74 (8): 2076–86. <https://doi.org/10.1093/icesjms/fsx038>.
- Donatuto, Jamie, and Melissa R. Poe. 2015. "Evaluating Sense of Place as a Domain of Human Wellbeing for Puget Sound Restoration." Final Report. Tacoma, WA: Puget Sound Institute. https://www.eopugetsound.org/sites/default/files/Donatuto%26Poe2015_Evaluating%20Sense%20of%20Place%20Final%20Report.pdf.
- Donkersloot, Rachel. 2010. "The Politics of Place and Identity in an Irish Fishing Locale." *Journal of Maritime Studies* 9 (2): 33–53. http://www.marecentre.nl/mast/documents/Mast2010_9.2_Donkersloot.pdf.
- Duan, Naihua. 1983. "Smearing estimate: A nonparametric retransformation method." *Journal of the American Statistical Association* 78 (383): 605–10. <https://doi.org/10.1080/01621459.1983.10478017>.
- Evans, Jeffrey S., and Melanie A. Murphy. 2018. *RfUtilities*. <https://cran.r-project.org/package=rfUtilities>.
- Everett, Sally, and Cara Aitchison. 2008. "The Role of Food Tourism in Sustaining Regional Identity: A Case Study of Cornwall, South West England." *Journal of Sustainable Tourism* 16 (2): 150–67. <https://doi.org/10.2167/jost696.0>.
- Färe, Rolf, James E. Kirkley, and John B. Walden. 2006. "Adjusting Technical Efficiency to Reflect Discarding: The Case of the U.S. Georges Bank Multi-Species Otter Trawl Fishery." *Fisheries Research* 78 (2-3): 257–65. <https://doi.org/10.1016/j.fishres.2005.12.014>.
- Forsyth, Jacob Samuel Tse, Magdalena Andres, and Glen G. Gawarkiewicz. 2015. "Recent Accelerated Warming of the Continental Shelf Off New Jersey: Observations from the CMV Oleander Expendable Bathymeterograph Line." *Journal of Geophysical Research: Oceans* 120 (3): 2370–84. <https://doi.org/10.1002/2014JC010516>.
- Fratantoni, Paula S., Tamara Holzwarth-Davis, and Maureen H. Taylor. 2015. *Description of Oceanographic Conditions on the Northeast U.S. Continental Shelf During 2014*. Northeast Fisheries Science Center Reference Document 15-21. US Dep. Commer., National Marine Fisheries Service. <https://repository.library.noaa.gov/view/noaa/5047>.
- Friedland, Kevin D., and Jon A. Hare. 2007. "Long-term trends and regime shifts in sea surface temperature on the continental shelf of the northeast United States." *Continental Shelf Research* 27: 2313–28. <https://doi.org/10.1016/j.csr.2007.06.001>.

- Friedland, Kevin D, M Conor McManus, Ryan E Morse, and Jason S Link. 2018. “Event Scale and Persistent Drivers of Fish and Macroinvertebrate Distributions on the Northeast US Shelf.” *ICES Journal of Marine Science*. <https://doi.org/10.1093/icesjms/fsy167>.
- Gaichas, Sarah K., Richard J. Seagraves, Jessica M. Coakley, Geret S. DePiper, Vincent G. Guida, Jonathan A. Hare, Paul J. Rago, and Michael J. Wilberg. 2016. “A Framework for Incorporating Species, Fleet, Habitat, and Climate Interactions into Fishery Management.” *Frontiers in Marine Science* 3. <https://doi.org/10.3389/fmars.2016.00105>.
- Gamer, Matthias, Jim Lemon, and Ian Fellows Puspendra Singh. 2012. *Irr: Various Coefficients of Interrater Reliability and Agreement*. <https://CRAN.R-project.org/package=irr>.
- Garrison, Lance P, and Jason S Link. 2000. “Dietary guild structure of the fish community in the Northeast United States continental shelf ecosystem.” *Marine Ecology Progress Series* 202: 231–40. <https://doi.org/10.3354/meps202231>.
- Gawarkiewicz, Glen G., Robert E. Todd, Albert J. Plueddemann, Magdalena Andres, and James P. Manning. 2012. “Direct Interaction Between the Gulf Stream and the Shelfbreak South of New England.” *Scientific Reports* 2 (August): 553. <https://doi.org/10.1038/srep00553>.
- Georgianna, Daniel, Min-Yang Lee, and John Walden. 2017. “Contrasting trends in the Northeast United States groundfish and scallop processing industries.” *Marine Policy*. <https://doi.org/10.1016/j.marpol.2017.08.025>.
- Glenn, Scott, Robert Arnone, Trisha Bergmann, W. Paul Bissett, Michael Crowley, Jay Cullen, Joe Gryzmski, et al. 2004. “Biogeochemical Impact of Summertime Coastal Upwelling on the New Jersey Shelf.” *Journal of Geophysical Research: Oceans* 109 (C12). <https://doi.org/10.1029/2003JC002265>.
- Gong, D., J. T. Kohut, and S. M. Glenn. 2010. “Seasonal Climatology of Wind-Driven Circulation on the New Jersey Shelf.” *Journal of Geophysical Research: Oceans* 115 (C4). <https://doi.org/10.1029/2009JC005520>.
- Gray, Steven A., Stefan Gray, Linda J. Cox, and Sarah Henly-Shepard. 2013. “Mental Modeler: A Fuzzy-Logic Cognitive Mapping Modeling Tool for Adaptive Environmental Management.” In, 965–73. IEEE. <https://doi.org/10.1109/HICSS.2013.399>.
- Grifell-Tatjé, Emili, and CA Knox Lovell. 2004. “Decomposing the dividend.” *Journal of Comparative Economics* 32 (3): 500–518. <https://doi.org/10.1016/j.jce.2004.05.002>.
- Hall, C Scott, and Stephen W Kress. 2004. “Comparison of Common Tern Reproductive Performance at Four Restored Colonies Along the Maine Coast, 1991-2002.” *Waterbirds* 27 (4): 424–34. [https://doi.org/%2010.1675/1524-4695\(2004\)027%5B0424:COCTR%5D2.0.CO;2](https://doi.org/%2010.1675/1524-4695(2004)027%5B0424:COCTR%5D2.0.CO;2).

- Hall, C Scott, Stephen W Kress, and Curtice R Griffin. 2000. "Composition, Spatial and Temporal Variation of Common and Arctic Tern Chick Diets in the Gulf of Maine." *Waterbirds*, 430–39. [https://doi.org/10.1675/1524-4695\(2004\)027%5B0424:COCTR%5D2.0.CO;2](https://doi.org/10.1675/1524-4695(2004)027%5B0424:COCTR%5D2.0.CO;2).
- Halpern, Benjamin S, Catherine Longo, Darren Hardy, Karen L McLeod, Jameal F Samhouri, Steven K Katona, Kristin Kleisner, et al. 2012. "An Index to Assess the Health and Benefits of the Global Ocean." *Nature* 488 (7413): 615–20. <https://doi.org/10.1038/nature11397>.
- Hardison, Sean, Charles T Perretti, Geret S DePiper, and Andrew Beet. 2019. "A Simulation Study of Trend Detection Methods for Integrated Ecosystem Assessment." *ICES Journal of Marine Science*. <https://doi.org/10.1093/icesjms/fsz097>.
- Hare, JA, WE Morrison, MW Nelson, MM Stachura, EJ Teeters, and RB Griffis. 2016. "A Vulnerability Assessment of Fish and Invertebrates to Climate Change on the Northeast US Continental Shelf." *PLoS ONE* 11.
- Hayes, Sean Arthur, Elizabeth Josephson, Katherine Maze-Foley, Patricia E Rosel, Barbie L Byrd, Timothy VN Cole, Laura Engleby, et al. 2017. "US Atlantic and Gulf of Mexico Marine Mammal Stock Assessments - 2016." *NOAA Technical Memorandum NMFS-NE-241*, no. June: 274. http://nova.nfsc.noaa.gov/palit/Waring%20et%20al.%7B/_%7D2013%7B/_%7DNOAA%20Tech%20Memo%20NMFS-NE-223%7B/_%7DU%20.%20S%20.%20Atlantic%20and%20Gulf%20of%20Mexico%20Marine%20Mammal%20Stock%20Assessments%20-%202012.pdf.
- Health, Virginia Department of. 2011. "Virginia Recreational Water Guidance for Microcystin and Microcystis Blooms: Provisional Guidance." Richmond, VA: Virginia Department of Health. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.432.955&rep=rep1&type=pdf>.
- Hess, Jeremy J., Josephine N. Malilay, and Alan J. Parkinson. 2008. "Climate Change: The Importance of Place." *American Journal of Preventive Medicine* 35 (5): 468–78. <https://doi.org/10.1016/j.amepre.2008.08.024>.
- Hiemstra, P. H., E. J. Pebesma, C. J. W. Twenh"ofel, and G. B. M. Heuvelink. 2008. "Real-time automatic interpolation of ambient gamma dose rates from the Dutch Radioactivity Monitoring Network." *Computers & Geosciences*. <https://doi.org/10.1016/j.cageo.2008.10.011>.
- Hoagland, P., and A. Meeks. 2000. "'The Demand for Whalewatching at Stellwagen Bank National Marine Sanctuary' in: The Economic Contribution of Whalewatching to Regional Economies: Perspectives from Two National Marine Sanctuaries." In *NOAA Marine Sanctuaries Conservation Series*, 00–02.
- Host, G E, P L Polzer, D J Mladenoff, M A White, and T R Crow. 1996. "A quantitative approach to developing regional ecosystem classifications." *Ecological Applications* 6 (2): 608–18. <https://doi.org/10.2307/2269395>.

- Houghton, R. W., R. Schlitz, R. C. Beardsley, B. Butman, and J. L. Chamberlin. 1982. "Middle Atlantic Bight Cold Pool: Evolution of the Temperature Structure During Summer 1979." *J. Phys. Oceanogr.; (United States)* 12:10 (October). [https://doi.org/10.1175/1520-0485\(1982\)012%3C1019:TMABCP%3E2.0.CO;2](https://doi.org/10.1175/1520-0485(1982)012%3C1019:TMABCP%3E2.0.CO;2).
- Hu, Chuanmin, Zhongping Lee, and Bryan Franz. 2012. "Chlorophyll a Algorithms for Oligotrophic Oceans: A Novel Approach Based on Three-Band Reflectance Difference." *Journal of Geophysical Research* 117 (C1): C01011. <https://doi.org/10.1029/2011JC007395>.
- Huang, Boyin, Peter W. Thorne, Viva F. Banzon, Tim Boyer, Gennady Chepurin, Jay H. Lawrimore, Matthew J. Menne, Thomas M. Smith, Russell S. Vose, and Huai Min Zhang. 2017a. "NOAA Extended Reconstructed Sea Surface Temperature (ERSST), Version 5." <https://doi.org/10.7289/V5T72FNM>.
- Huang, Boyin, Peter W Thorne, Viva F Banzon, Tim Boyer, Gennady Chepurin, Jay H Lawrimore, Matthew J Menne, Thomas M Smith, Russell S Vose, and Huai-Min Zhang. 2017b. "Extended Reconstructed Sea Surface Temperature, Version 5 (Ersstv5): Upgrades, Validations, and Intercomparisons." *Journal of Climate* 30 (20): 8179–8205. <https://doi.org/10.1175/JCLI-D-16-0836.1>.
- Hudson, Karen. 2017. "Virginia Shellfish Aquaculture Situation and Outlook Report : Results of the 2016 Virginia Shellfish Aquaculture Crop Reporting Survey." Virginia Institute of Marine Science, College of William & Mary. <https://doi.org/10.21220/V51K6T>.
- International, AOAC. 2005. "AOAC Official Method 959.08. Paralytic shellfish poison. Biological method. Final action." In *AOAC Official Methods for Analysis*, 18th ed., 79–80. Gaithersburg.
- IPCC. 2014. "Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change." <https://doi.org/10.1017/CBO9781107415324>.
- Jennings, Simon. 2005. "Indicators to Support an Ecosystem Approach to Fisheries." *Fish and Fisheries* 6 (3): 212–32. <https://doi.org/10.1111/j.1467-2979.2005.00189.x>.
- Jepson, Michael, and Lisa L. Colburn. 2013. "Development of Social Indicators of Fishing Community Vulnerability and Resilience in the US Southeast and Northeast Regions." *NOAA Technical Memorandum NMFS-F/SPO-129 (US Dept Commerce, 2013)*. http://www.nmfs.noaa.gov/sfa/management/councils/training/2014/r_h3_fishing_community_vulnerability.pdf.
- Joyce, Terrence M, Young-Oh Kwon, Hyodae Seo, and Caroline C Ummenhofer. 2019. "Meridional Gulf Stream Shifts Can Influence Wintertime Variability in the North Atlantic Storm Track and Greenland Blocking." *Geophysical Research Letters* 46 (3): 1702–8. <https://doi.org/10.1029/2018GL081087>.

- Kane, Joseph. 2007. "Zooplankton abundance trends on Georges Bank, 1977–2004." *ICES Journal of Marine Science* 64 (5): 909–19. <https://doi.org/10.1093/icesjms/fsm066>.
- . 2011. "Multiyear variability of phytoplankton abundance in the Gulf of Maine." *ICES Journal of Marine Science* 68 (9): 1833–41. <https://doi.org/10.1093/icesjms/fsr122>.
- Khakzad, Sorna, and David Griffith. 2016. "The Role of Fishing Material Culture in Communities' Sense of Place as an Added-Value in Management of Coastal Areas." *Journal of Marine and Island Cultures* 5 (2): 95–117. <https://doi.org/10.1016/j.jmic.2016.09.002>.
- Kirk, John T. O. 1994. *Light and Photosynthesis in Aquatic Ecosystems*. Cambridge, UK: Cambridge University Press.
- Clain, Sarah C., Terre A. Satterfield, and Kai M. A. Chan. 2014. "What Matters and Why? Ecosystem Services and Their Bundled Qualities." *Ecological Economics* 107: 310–20. <https://doi.org/10.1016/j.ecolecon.2014.09.003>.
- Kleisner, Kristin M., Michael J. Fogarty, Sally McGee, Jonathan A. Hare, Skye Moret, Charles T. Perretti, and Vincent S. Saba. 2017. "Marine species distribution shifts on the U.S. Northeast Continental Shelf under continued ocean warming." *Progress in Oceanography* 153: 24–36. <https://doi.org/10.1016/j.pocean.2017.04.001>.
- Koehn, J. Zachary, Daniel R. Reineman, and John N. Kittinger. 2013. "Progress and Promise in Spatial Human Dimensions Research for Ecosystem-Based Ocean Planning." *Marine Policy* 42: 31–38. <https://doi.org/10.1016/j.marpol.2013.01.015>.
- Lee, Min-Yang. 2010. "Economic Tradeoffs in the Gulf of Maine Ecosystem: Herring and Whale-Watching." *Marine Policy* 34: 156–62. <https://doi.org/10.1016/j.marpol.2009.06.001>.
- . 2014. "Hedonic Pricing of Atlantic Cod: Effects of Size, Freshness, and Gear." *Marine Resource Economics* 29 (3). <https://doi.org/10.1086/677769>.
- Lee, Min-Yang, Scott Steinback, and Kristy Wallmo. 2017. "Applying a Bioeconomic Model to Recreational Fisheries Management: Groundfish in the Northeast United States." *Marine Resource Economics* 32 (2): 191–216. <https://doi.org/10.1086/690676>.
- Lee, Min-Yang, and Eric M. Thunberg. 2013. "An Inverse Demand System for New England Groundfish: Welfare Analysis of the Transition to Catch Share Management." *American Journal of Agricultural Economics* 95 (5). <https://doi.org/10.1093/ajae/aat061>.
- Legendre, Pierre, and Loic Legendre. 1998. *Numerical Ecology*. <https://www.elsevier.com/books/numerical-ecology/legendre/978-0-444-89249-2>.

- Lentz, S. J. 2003. "A Climatology of Salty Intrusions over the Continental Shelf from Georges Bank to Cape Hatteras." *Journal of Geophysical Research* 108 (C10). <https://doi.org/10.1029/2003JC001859>.
- . 2017. "Seasonal Warming of the Middle Atlantic Bight Cold Pool." *Journal of Geophysical Research: Oceans* 122 (2): 941–54. <https://doi.org/10.1002/2016JC012201>.
- Levin, Phillip S, Michael J Fogarty, Steven A Murawski, and David Fluharty. 2009. "Integrated Ecosystem Assessments: Developing the Scientific Basis for Ecosystem-Based Management of the Ocean." *PLoS Biol* 7 (1): e1000014. <https://doi.org/10.1371/journal.pbio.1000014>.
- Liaw, Andy, and Matthew Wiener. 2002. "Classification and Regression by randomForest." *R News* 2 (3): 18–22. <https://CRAN.R-project.org/doc/Rnews/>.
- Lim, Siew Hoon, and CA Knox Lovell. 2009. "Profit and productivity of US Class I railroads." *Managerial and Decision Economics* 30 (7): 423–42. <https://doi.org/10.1002/mde.1462>.
- Link, Jason, William Overholtz, John O'Reilly, Jack Green, David Dow, Debra Palka, Chris Legault, et al. 2008. "The Northeast U.S. Continental Shelf Energy Modeling and Analysis Exercise (EMAX): Ecological Network Model Development and Basic Ecosystem Metrics." *Journal of Marine Systems* 74 (1–2): 453–74. <https://doi.org/10.1016/j.jmarsys.2008.03.007>.
- Link, Jason S. 2005. "Translating Ecosystem Indicators into Decision Criteria." *ICES Journal of Marine Science* 62 (3): 569–76. <https://doi.org/10.1016/j.icesjms.2004.12.015>.
- Link, Jason S, Carolyn A Griswold, Elizabeth T Methratta, Jessie Gunnard, Jon Brodziak, Tarsus Kenton, Laurel A Col, et al. 2006. *Documentation for the energy modeling and analysis exercise (EMAX)*. US Department of Commerce, National Oceanic; Atmospheric Administration.
- Link, J. S., C. T. Griswold, E. T. Methratta, and J. Gunnard, eds. 2006. *Documentation for the Energy Modeling and Analysis eXercise (EMAX)*. Northeast Fish. Sci. Cent. Ref. Doc. 06-15. Northeast Fish. Sci. Cent. Ref. Doc. Woods Hole, MA: US Dep. Commer., National Marine Fisheries Service.
- Longhurst, Alan R. 2007. *Ecological Geography of the Sea*. <https://doi.org/10.1016/B978-0-12-455521-1.X5000-1>.
- Lord, Fabienne. 2011. "Understanding Social Impacts by Using New Variables and a Causal Model Diagram in New England Fisheries." *Impact Assessment and Project Appraisal* 1: 59–68. <https://doi.org/10.3152/146155111X12913679730476>.
- Maritorena, Stéphane, Odile Hembise Fanton D'Andon, Antoine Mangin, and David A Siegel. 2010. "Merged satellite ocean color data products using a

- bio-optical model: Characteristics, benefits and issues.” *Remote Sensing of Environment* 114: 1791–1804. <https://doi.org/10.1016/j.rse.2010.04.002>.
- Matzarakis, A., and C. R. de Freitas, eds. 2001. “Proceedings of the First International Workshop on Climate, Tourism and Recreation.” In. Porto Carras, Greece.
- Mazerolle, Marc J. 2017. “AICcmodavg: model selection and multimodel inference based on (Q)AIC(c).” <https://cran.r-project.org/package=AICcmodavg>.
- McGoodwin, James R. 2001. “Understanding the Cultures of Fishing Communities: A Key to Fisheries Management and Food Security.” In. <http://www.fao.org/docrep/004/y1290e/y1290e00.htm>.
- Miller, Timothy J., Jonathan A. Hare, and Larry A. Alade. 2016. “A State-Space Approach to Incorporating Environmental Effects on Recruitment in an Age-Structured Assessment Model with an Application to Southern New England Yellowtail Flounder.” *Canadian Journal of Fisheries and Aquatic Sciences* 73 (8): 1261–70. <https://doi.org/10.1139/cjfas-2015-0339>.
- Miller, T. J., C. Das, P. J. Politis, A. S. Miller, S. M. Lucey, C. M. Legault, R. W. Brown, and P. J. Rago. 2010. “Estimation of Albatross IV to Henry B. Bigelow Calibration Factors.” 10-05. Woods Hole, MA: National Marine Fisheries Service. <https://www.nefsc.noaa.gov/publications/crd/crd1005/crd1005.pdf>.
- Milligan, Glenn W., and Martha C. Cooper. 1985. “An examination of procedures for determining the number of clusters in a data set.” *Psychometrika* 50 (2): 159–79. <https://doi.org/10.1007/BF02294245>.
- Morel, André. 1991. “Light and Marine Photosynthesis: A Spectral Model with Geochemical and Climatological Implications.” *Progress in Oceanography* 26 (3): 263–306. [https://doi.org/10.1016/0079-6611\(91\)90004-6](https://doi.org/10.1016/0079-6611(91)90004-6).
- Morel, André, and Jean-François Berthon. 1989. “Surface Pigments, Algal Biomass Profiles, and Potential Production of the Euphotic Layer: Relationships Reinvestigated in View of Remote-Sensing Applications.” *Limnology and Oceanography* 34 (8): 1545–62. <https://doi.org/10.4319/lo.1989.34.8.1545>.
- Morse, R. E., K. D. Friedland, D. Tommasi, C. Stock, and J. Nye. 2017. “Distinct zooplankton regime shift patterns across ecoregions of the U.S. Northeast continental shelf Large Marine Ecosystem.” *Journal of Marine Systems* 165: 77–91. <https://doi.org/10.1016/j.jmarsys.2016.09.011>.
- Mountain, David G. 1991. “The Volume of Shelf Water in the Middle Atlantic Bight: Seasonal and Interannual Variability, 1977–1987.” *Continental Shelf Research* 11 (3): 251–67. [https://doi.org/10.1016/0278-4343\(91\)90068-H](https://doi.org/10.1016/0278-4343(91)90068-H).
- Mountain, David G. 2003. “Variability in the Properties of Shelf Water in the

- Middle Atlantic Bight, 1977–1999.” *Journal of Geophysical Research* 108 (C1). <https://doi.org/10.1029/2001JC001044>.
- . 2012. “Labrador Slope Water Entering the Gulf of Maine—Response to the North Atlantic Oscillation.” *Continental Shelf Research* 47 (September): 150–55. <https://doi.org/10.1016/j.csr.2012.07.008>.
- . 2012. “Labrador Slope Water Entering the Gulf of Maine—Response to the North Atlantic Oscillation.” *Continental Shelf Research* 47: 150–55. <https://doi.org/10.1016/j.csr.2012.07.008>.
- Mupparapu, Prashant, and Wendell S. Brown. 2002. “Role of Convection in Winter Mixed Layer Formation in the Gulf of Maine, February 1987.” *Journal of Geophysical Research: Oceans* 107 (C12): 22–21–22–18. <https://doi.org/10.1029/1999JC000116>.
- Murawski, S. 2000. “Definitions of Overfishing from an Ecosystem Perspective.” *ICES Journal of Marine Science* 57 (3): 649–58. <https://doi.org/10.1006/jmsc.2000.0738>.
- Murphy, Melanie A., Jeffrey S. Evans, and Andrew Storfer. 2010. “Quantifying Bufo boreas connectivity in Yellowstone National Park with landscape genetics.” *Ecology* 91 (1): 252–61. <https://doi.org/10.1890/08-0879.1>.
- NASA Ocean Biology Processing Group. 2017. “MODIS-Aqua Level-3 Binned Garver, Siegel, Maritorena Model (GSM) Data Version R2018.0.” NASA Ocean Biology DAAC. <https://doi.org/10.5067/AQUA/MODIS/L3B/GSM/2018>.
- . 2018. “SEAWIFS-ORBVIEW-2 Level 3 Mapped Chlorophyll Data Version R2018.0.” NASA Ocean Biology DAAC. <https://doi.org/10.5067/ORBVIEW-2/SEAWIFS/L3M/CHL/2018>.
- Nicholson, Mike D., and Simon Jennings. 2004. “Testing candidate indicators to support ecosystem-based management: The power of monitoring surveys to detect temporal trends in fish community metrics.” <https://doi.org/10.1016/j.icesjms.2003.09.004>.
- Norris-Raynbird, Carla. 2004. “‘For Hire’ in the US Gulf of Mexico: A Typology of Offshore Charter and Party Boat Operations.” *MAST (Maritime Studies)* 1 (51 – 65). <http://www.marecentre.nl/mast/documents/ArtikelCarlaNorris-Raynbird.pdf>.
- Oberg, Angela, Julia Flagg, Bonnie McCay, and Lisa L. Colburn. 2016. “Surviving Sandy: Identity and Cultural Resilience in a New Jersey Fishing Community.” In *Taking Chances on the Coast After Hurricane Sandy*, edited by Dan Van Abs and Karen O’Neill. New Brunswick, NJ: The Rutgers University Press. muse.jhu.edu/book/45532.
- Oksanen, Jari, F. Guillaume Blanchet, Michael Friendly, Roeland Kindt, Pierre Legendre, Dan McGlinn, Peter R. Minchin, et al. 2019. *Vegan: Community Ecology Package*. <https://CRAN.R-project.org/package=vegan>.

- O'Reilly, John E., Stéphane Maritorena, B. Greg Mitchell, David A. Siegel, Kendall L. Carder, Sara A. Garver, Mati Kahru, and Charles McClain. 1998. "Ocean Color Chlorophyll Algorithms for Seawifs." *Journal of Geophysical Research: Oceans* 103 (C11): 24937–53. <https://doi.org/10.1029/98JC02160>.
- Organization, World Health. 2003. *Guidelines for Safe Recreational Water Environments: Coastal and Fresh Waters*. Vol. 1. World Health Organization.
- Pace, Richard M., Peter J. Corkeron, and Scott D. Kraus. 2017. "State-space mark-recapture estimates reveal a recent decline in abundance of North Atlantic right whales." *Ecology and Evolution*. <https://doi.org/10.1002/ece3.3406>.
- Paluszynska, Aleksandra, and Przemyslaw Biecek. 2017. *randomForestExplainer: Explaining and Visualizing Random Forests in Terms of Variable Importance*. <https://CRAN.R-project.org/package=randomForestExplainer>.
- Pan, Xiaoju, Antonio Mannino, Mary E. Russ, and Stanford B. Hooker. 2008. "Remote Sensing of the Absorption Coefficients and Chlorophyll a Concentration in the United States Southern Middle Atlantic Bight from Seawifs and Modis-Aqua." *Journal of Geophysical Research* 113 (C11): C11022. <https://doi.org/10.1029/2008JC004852>.
- Pan, Xiaoju, Antonio Mannino, Mary E. Russ, Stanford B. Hooker, and Lawrence W. Harding Jr. 2010. "Remote Sensing of Phytoplankton Pigment Distribution in the United States Northeast Coast." *Remote Sensing of Environment* 114 (11): 2403–16. <https://doi.org/10.1016/j.rse.2010.05.015>.
- Pauly, Daniel. 1997. "Putting Fisheries Management Back in Places." *Reviews in Fish Biology and Fisheries* 7 (1): 125–27.
- Perretti, Charles T., Michael J. Fogarty, Kevin D. Friedland, Jon A. Hare, Sean M. Lucey, Richard S. McBride, Timothy J. Miller, et al. 2017a. "Regime Shifts in Fish Recruitment on the Northeast US Continental Shelf." *Marine Ecology Progress Series* 574: 1–11. <https://doi.org/10.3354/meps12183>.
- . 2017b. "Regime shifts in fish recruitment on the Northeast US Continental Shelf." *Marine Ecology Progress Series* 574: 1–11. <https://doi.org/10.3354/meps12183>.
- Pielou, Evelyn Chris. 1984. *The Interpretation of Ecological Data: A Primer on Classification and Ordination*. John Wiley & Sons.
- Pinheiro, J, D Bates, S DebRoy, D Sarkar, and R Core Team. 2017. "nlme: Linear and Nonlinear Mixed Effects Models." *R Package Version 3.1-131*. <https://CRAN.R-project.org/package=nlme>.
- Pinsky, Malin L., Boris Worm, Michael J. Fogarty, Jorge L. Sarmiento, and Simon A. Levin. 2013. "Marine taxa track local climate velocities." *Science* 341 (6151): 1239–42. <https://doi.org/10.1126/science.1239352>.

- Poe, Melissa R., Karma C. Norman, and Phillip S. Levin. 2014. "Cultural Dimensions of Socioecological Systems: Key Connections and Guiding Principles for Conservation in Coastal Environments." *Conservation Letters* 7 (3): 166–75. https://www.nwfsc.noaa.gov/news/events/program_reviews/2016/documents/D1_PoeNormanLevin2014.pdf.
- Pollnac, Richard B., Susan Abbott-Jamieson, Courtland Smith, Marc L. Miller, Patricia M. Clay, and Bryan Oles. 2006. "Toward a Model for Fisheries Social Impact Assessment." *Marine Fisheries Review* 68: 1–4. http://aquaticcommons.org/866/1/Pollnac_Toward.pdf.
- Potschin, Marion, and Roy Haines-Young. 2013. "Landscapes, Sustainability and the Place-Based Analysis of Ecosystem Services." *Landscape Ecology* 28 (6): 1053–65. <https://doi.org/10.1007/s10980-012-9756-x>.
- Project, JPL MUR MEaSUREs. 2015. "GHRSST Level 4 Mur Global Foundation Sea Surface Temperature Analysis (V4.1)." CA, USA: NASA PO.DAAC. <https://doi.org/10.5067/GHGMR-4FJ04>.
- Quilliam, M. A., M. Xie, and W. R. Hardstaff. 1995. "Rapid extraction and cleanup for liquid chromatography determination of domoic acid in unsalted seafood." *Journal of AOAC International* 78 (JANUARY 1995): 543–54.
- Reed, Matt, Paul Courtney, Julie Urquhart, and Natalie Ross. 2013. "Beyond Fish as Commodities: Understanding the Socio-Cultural Role of Inshore Fisheries in England." *Marine Policy* 37: 62–68. <https://doi.org/10.1016/j.marpol.2012.04.009>.
- Reynolds, Richard W., Thomas M. Smith, Chunying Liu, Dudley B. Chelton, Kenneth S. Casey, and Michael G. Schlax. 2007. "Daily high-resolution-blended analyses for sea surface temperature." *Journal of Climate* 20 (22): 5473–96. <https://doi.org/10.1175/2007JCLI1824.1>.
- Rice, Jake C., and Marie-Joëlle Rochet. 2005. "A Framework for Selecting a Suite of Indicators for Fisheries Management." *ICES Journal of Marine Science* 62 (3): 516–27. <https://doi.org/10.1016/j.icesjms.2005.01.003>.
- Roff, John C., and Mark E. Taylor. 2000. "National frameworks for marine conservation - A hierarchical geophysical approach." *Aquatic Conservation: Marine and Freshwater Ecosystems* 10 (3): 209–23. [https://doi.org/10.1002/1099-0755\(200005/06\)10:3%3C209::AID-AQC408%3E3.0.CO;2-J](https://doi.org/10.1002/1099-0755(200005/06)10:3%3C209::AID-AQC408%3E3.0.CO;2-J).
- Saba, Vincent S., Stephen M. Griffies, Whit G. Anderson, Michael Winton, Michael A. Alexander, Thomas L. Delworth, Jonathan A. Hare, et al. 2016. "Enhanced warming of the Northwest Atlantic Ocean under climate change." *Journal of Geophysical Research: Oceans* 121 (1): 118–32. <https://doi.org/10.1002/2015JC011346>.
- Saha, Korak, Xuepeng Zhao, Huai-min Zhang, Kenneth S. Casey, Dexin Zhang, Sheekela Baker-Yeboah, Katherine A. Kilpatrick, Robert H. Evans, Thomas Ryan, and John M. Relph. 2018. "AVHRR Pathfinder version 5.3 level 3

- collated (L3C) global 4km sea surface temperature for 1981-Present.” NOAA National Centers for Environmental Information. <https://doi.org/doi:10.7289/V52J68XX%20Accessed%20in%202016%20and%202018>.
- Scott, D., G McBoyle, and M. Schwartzentruber. 2004. “Climate Change and the Distribution of Climatic Resources for Tourism in North America.” *Climate Research* 27: 105–17. <https://doi.org/10.3354/cr027105>.
- Seara, Tarsila, Patricia M. Clay, and Lisa L. Colburn. 2016. “Perceived Adaptive Capacity and Natural Disasters: A Fisheries Case Study.” *Global Environmental Change* 38: 49–57. <https://doi.org/10.1016/j.gloenvcha.2016.01.006>.
- Shearman, R. Kipp, and Steven J. Lentz. 2009. “Long-Term Sea Surface Temperature Variability Along the U.S. East Coast.” *Journal of Physical Oceanography* 40 (5): 1004–17. <https://doi.org/10.1175/2009JPO4300.1>.
- Smienk, Henry G F, Dolores Calvo, Pedro Razquin, Elena Domínguez, and Luis Mata. 2012. “Single laboratory validation of A ready-to-Use phosphatase inhibition assay for detection of okadaic acid toxins.” *Toxins* 4 (5): 339–52. <https://doi.org/10.3390/toxins4050339>.
- Smith, Brian E., and Jason S. Link. 2010. *The Trophic Dynamics of 50 Fin-fish and 2 Squid Species on the Northeast US Continental Shelf*. NOAA Technichal Memorandum NMFS-NE-216. National Marine Fisheries Service, 166 Water Street, Woods Hole, MA 02543-1026. <http://www.nefsc.noaa.gov/publications/tm/tm216/>.
- Smith, Laurel A., Jason S. Link, Steven X. Cadrian, and Debra L. Palka. 2015. “Consumption by Marine Mammals on the Northeast U.S. Continental Shelf.” *Ecological Applications* 25 (2): 373–89. <https://doi.org/10.1890/13-1656.1>.
- Smith, Peter C. 1983. “The Mean and Seasonal Circulation Off Southwest Nova Scotia.” *Journal of Physical Oceanography* 13 (6): 1034–54. [https://doi.org/10.1175/1520-0485\(1983\)013%3C1034:TMASCO%3E2.0.CO;2](https://doi.org/10.1175/1520-0485(1983)013%3C1034:TMASCO%3E2.0.CO;2).
- Smith, Peter C., Robert W. Houghton, Richard G. Fairbanks, and David G. Mountain. 2001. “Interannual Variability of Boundary Fluxes and Water Mass Properties in the Gulf of Maine and on Georges Bank: 1993–1997.” *Deep Sea Research Part II: Topical Studies in Oceanography*, Coupled biological and physical studies of plankton populations on Georges Bank and related North Atlantic regions, 48 (1): 37–70. [https://doi.org/10.1016/S0967-0645\(00\)00081-3](https://doi.org/10.1016/S0967-0645(00)00081-3).
- Smith, Peter C, Neal R Pettigrew, Philip Yeats, David W Townsend, and Guoqi Han. 2012. “Regime Shift in the Gulf of Maine.” *American Fisheries Society Symposium* 79: 185–203.
- Steinback, Scott, and Eric Thunberg. 2006. “Northeast Regional Commercial

- Fishing Input-Output Model." *NOAA Technical Memorandum NMFS-NE-188*: 54. <https://www.nefsc.noaa.gov/publications/tm/tm188/tm188.pdf>.
- St Martin, Kevin. 2001. "Making Space for Community Resource Management in Fisheries." *Annals of the Association of American Geographers* 91 (1): 122–42. <https://doi.org/10.1111/0004-5608.00236>.
- Storch, Hans von. 1999. "Misuses of Statistical Analysis in Climate Research." In *Analysis of Climate Variability*, 11–26. https://doi.org/10.1007/978-3-662-03744-7_2.
- Stutts, Whitney, and Jonathon Deeds. 2017. "Single Laboratory Validation (SLV) Protocol for Submission to the Interstate Shellfish Sanitation Conference (ISSC) For Method Approval." College Park: FDA Center for Food Safety; Applied Nutrition. <http://www.issc.org/Data/Sites/1/media/00-2017biennialmeeting//taskforcei2017/17-103-supporting-documentation.pdf>.
- Sullivan, Mark C., Robert K. Cowen, and Brian P. Steves. 2005. "Evidence for Atmosphere-Ocean Forcing of Yellowtail Flounder (*Limanda ferruginea*) Recruitment in the Middle Atlantic Bight." *Fisheries Oceanography* 14 (5): 386–99. http://www.academia.edu/10197997/Evidence_for_atmosphere-ocean_forcing_of_yellowtail_flounder_Limanda_ferruginea_recruitment_in_the_Middle_Atlantic_Bight.
- Tang, Ying Zhong, and Christopher J. Gobler. 2009. "Characterization of the toxicity of *Cochlodinium polykrikoides* isolates from Northeast US estuaries to finfish and shellfish." *Harmful Algae* 8 (3): 454–62. <https://doi.org/10.1016/j.hal.2008.10.001>.
- Thunberg, Eric M., and Steven J. Correia. 2015. "Measures of Fishing Fleet Diversity in the New England Groundfish Fishery." *Marine Policy* 58 (2015): 6–14. <https://doi.org/10.1016/j.marpol.2015.04.005>.
- Townsend, David W, Andrew C Thomas, Lawrence M Mayer, Maura A Thomas, and John A Qunilan. 2006. "Oceanography of the Northwest Atlantic Continental Shelf (1,W)." In *The Sea, Volume 14A*, edited by Allan R Robinson and Kenneth H Brink, 119–68. Cambridge, MA: Harvard University Press. <https://pdfs.semanticscholar.org/bbdb/3136f6ab34fd405225fde03696feee70b36b.pdf>.
- Urquhart, Julie, and Tim Acott. 2013. "Constructing 'the Stade': Fishers' and Non-Fishers' Identity and Place Attachment in Hastings, South-East England." *Marine Policy* 37: 45–54. <https://doi.org/10.1016/j.marpol.2012.04.004>.
- USEPA. 2003. "Ambient Water Quality Criteria for Dissolve Oxygen, Water Clarity and Chlorophyll a for the Chesapeake Bay and Its Tidal Tributaries." USEPA Region III Chesapeake Bay Program Office EPA 903-R-03-002, Annapolis, Maryland. https://www.chesapeakebay.net/content/publications/cbp_13142.pdf.

- . 2017. “Ambient Water Quality Criteria for Dissolve Oxygen, Water Clarity and Chlorophyll a for the Chesapeake Bay and Its Tidal Tributaries: 2017 Addendum.” USEPA Region III Chesapeake Bay Program Office EPA 903-R-03-002, Annapolis, Maryland. https://www.chesapeakebay.net/documents/2017_Nov_ChesBayWQ_Criteria_Addendum_Final.pdf.
- Wagner, Tyler, Brian J. Irwin, James R. Bence, and Daniel B. Hayes. 2013. “Detecting Temporal Trends in Freshwater Fisheries Surveys: Statistical Power and the Important Linkages between Management Questions and Monitoring Objectives.” *Fisheries* 38 (7): 309–19. <https://doi.org/10.1080/03632415.2013.799466>.
- Walden, John B., James E. Kirkley, Rolf Färe, and Philip Logan. 2012. “Productivity Change Under an Individual Transferable Quota Management System.” *American Journal of Agricultural Economics* 94 (4): 913–28. <https://doi.org/10.1093/ajae/aas025>.
- Walsh, Harvey J., David E. Richardson, Katrin E. Marancik, and Jonathan A. Hare. 2015. “Long-Term Changes in the Distributions of Larval and Adult Fish in the Northeast U.S. Shelf Ecosystem.” Journal Article. *PLoS ONE* 10 (9): e0137382. <https://doi.org/10.1371/journal.pone.0137382>.
- Waring, Gordon T., Elizabeth Josephson, Katherine Maze-Foley, and Patricia E. Rosel, eds. 2015. *US Atlantic and Gulf of Mexico Marine Mammal Stock Assessments – 2014.*, Type = NOAA Technical Memorandum. 231. National Marine Fisheries Service. <https://doi.org/10.7289/V5TQ5ZH0>.
- Wigley, S. E., H. M. McBride, and N. J. McHugh. 2003. “Length-Weight Relationships for 74 Fish Species Collected During NEFSC Research Vessel Bottom Trawl Surveys, 1992-99.” NOAA Technical Memorandum 171. National Marine Fisheries Service. <https://www.nefsc.noaa.gov/publications/tm/tm171/tm171.pdf>.
- Winton, Michael, Whit G. Anderson, Thomas L. Delworth, Stephen M. Griffies, William J. Hurlin, and Anthony Rosati. 2014. “Has Coarse Ocean Resolution Biased Simulations of Transient Climate Sensitivity?” *Geophysical Research Letters* 41 (23): 8522–9. <https://doi.org/10.1002/2014GL061523>.
- WMO. 2017. “WMO Guidelines of the Calculation of Climate Normals.” World Meteorological Organization.
- Wood, S. 2011. “Mixed GAM Computation Vehicle with GCV/AIC/REML smoothness estimation and GAMMs by REML/PQL.” <https://stat.ethz.ch/R-manual/R-devel/library/mgcv/html/mgcv-package.html>.
- Wynveen, Christopher J., Gerard T. Kyle, and Stephen G. Sutton. 2012. “Natural Area Visitors’ Place Meaning and Place Attachment Ascribed to a Marine Setting.” *Journal of Environmental Psychology* 32 (4): 287–96. <https://doi.org/10.1016/j.jenvp.2012.05.001>.

- Zhang, Qian, Rebecca R Murphy, Richard Tian, Melinda K Forsyth, Emily M Trentacoste, Jennifer Keisman, and Peter J Tango. 2018. "Chesapeake Bay Water Quality Condition Has Been Recovering: Insights from a Multimetric Indicator Assessment of Thirty Years of Tidal Monitoring Data." *Science of the Total Environment* 637: 1617–25. <https://doi.org/10.1016/j.scitotenv.2018.05.025>.
- Zhang, Weifeng G., and Glen G. Gawarkiewicz. 2015. "Dynamics of the Direct Intrusion of Gulf Stream Ring Water onto the Mid-Atlantic Bight Shelf." *Geophysical Research Letters* 42 (18): 7687–95. <https://doi.org/10.1002/2015GL065530>.

