NOAA Technical Memorandum NMFS-XXX-##

Analyses to support the Dolphinfish Management Strategy Evaluation in the U.S. Atlantic

Mandy Karnauskas

January 2023

U.S. DEPARTMENT OF COMMERCE

National Oceanic and Atmospheric Administration
National Marine Fisheries Service
Northwest Fisheries Science Center

Analyses to support the Dolphinfish Management Strategy Evaluation in the U.S. Atlantic

Mandy Karnauskas[1]

1. NOAA Fisheres, Southeast Fisheries Science Center

Table of contents

List of Figures

List of Tables

1 Introduction

Here we will be displaying various data summaries and analyses related to dolphin management in the U.S. South Atlantic.

1.1 Details

The documentation of this data exploration was developed in R, using R Studio and Quarto.

2 Defining the regions for the operating model

Here we define the different regions to be used in the operating model, based on the areas that were defined in the industry working groups. The operating model is intended to encompass the Western Central Atlantic dolphin population and include international waters (FAO areas 21 and 31) which are thought to be connected with the stock that is exploited by U.S. fisheries.

We use the U.S. exclusive economic zone (EEZ) to differentiate between national and international waters. Note that some U.S. commercial activity occurs in high seas waters and we will account for those catches in the summaries.

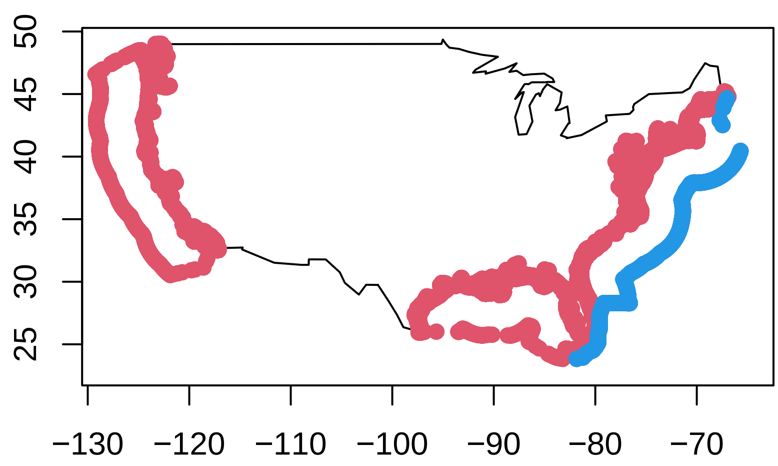
2.1 Data download for the U.S. EEZ

We download the U.S. Atlantic EEZ from Marineregions.org.

```
# clear workspace and load the libraries
if (!requireNamespace("maps", quietly = TRUE)) {
  install.packages("maps") }
library(maps)

rm(list = ls())

# read in the shapefile downloaded from Marineregions.org
#library(sf)
#shp_data <- st_read("data/eez/eez.shp")
#coords <- st_coordinates(shp_data)
#co <- as.data.frame(coords)
#save(co, file = "data/eez/Marineregions_EEZ.RData")

load("data/eez/Marineregions_EEZ.RData")
head(co)
```

```
          X        Y L1 L2 L3
1 -67.28403 45.19125  1  1  1
2 -67.28400 45.19126  1  1  1
3 -67.28387 45.19125  1  1  1
4 -67.28383 45.19125  1  1  1
5 -67.28372 45.19125  1  1  1
6 -67.28365 45.19125  1  1  1
```

```
map("usa", xlim = c(-130, -63), ylim = c(22, 50))
axis(1); axis(2); box()
points(co$X, co$Y, pch = 19, col = 2)

# remove points that are not in the U.S. South Atlantic
co <- co[which(co$X > (-82)), ]
co <- co[-which(co$X < (-81) & co$Y > 25 & co$Y < 28), ]

#plot(co$X, co$Y, col = 0)
#text(co$X, co$Y, 1:nrow(co), cex = 0.7)
#plot(co$X, co$Y, col = 0, xlim = c(-85, -80), ylim = c(22, 25))
#text(co$X, co$Y, 1:nrow(co), cex = 0.7)

# identify the northernmost and southernmost points along the Atlantic EEZ
co1 <- co[2100:3727, ]
points(co1$X, co1$Y, col = 4, pch = 19)
```

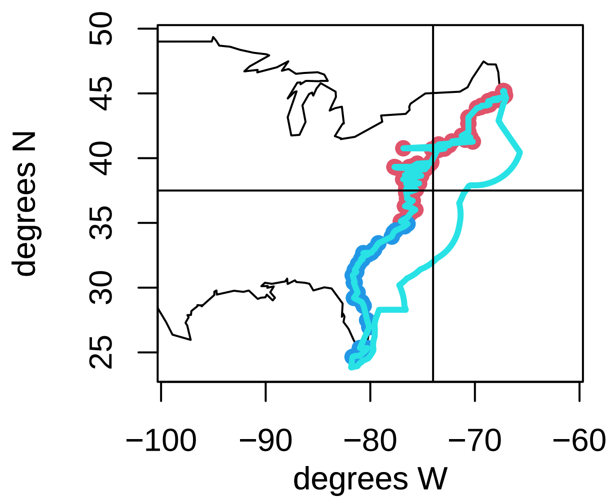


2.2 Construct the U.S. EEZ

We simplify the shapefile along the U.S. coast in order to simplify computations later in the process.

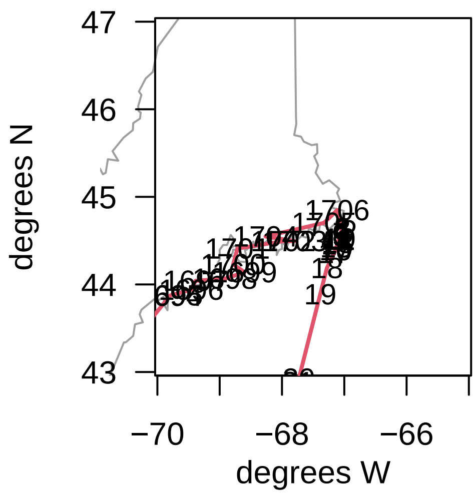
```
# subset points N and S of 35N for later delineation
co2 <- co[-c(2100:3727), ]
co3 <- co2[which(co2$Y > 35), ]
co4 <- co2[which(co2$Y <= 35), ]

# reduce number of points along the coast to make computation easier
co3 <- co3[seq(1, nrow(co3), 1100), ]
co4 <- co4[seq(1, nrow(co4), 800), ]

map("usa", xlim = c(-100, -60), ylim = c(23, 50))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2); box()
points(co3$X, co3$Y, col = 2, pch = 19)
```

```
points(co4$X, co4$Y, col = 4, pch = 19)

co2 <- rbind(co3, co4)
co2 <- co2[order(co2$Y), ]
cofin <- rbind(co1, co2)

polygon(cofin$X, cofin$Y, border = 5, lwd = 3)

# make the polygon along the coast line a bit more linear
abline(h = 37.5)
abline(v = -74)
```

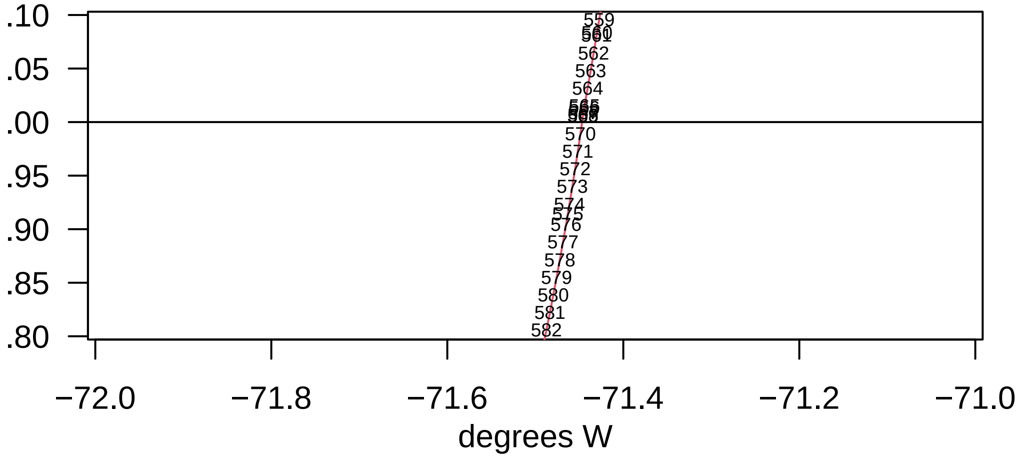



```
cofin <- cofin[-which(cofin$X < (-74) & cofin$Y > 37.6), ]
#text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.9)
lis <- c(1634, 1639, 1688, 1710, 1711, 1712)
cofin <- cofin[-lis, ]

map(database = "usa", xlim = c(-70, -65), ylim = c(43, 47), col = 8)
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
```

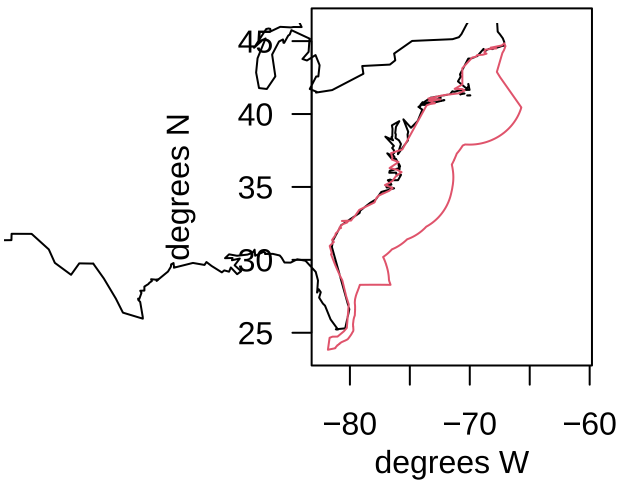
```
polygon(cofin$X, cofin$Y, border = 2, lwd = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.9)
```




```
map(database = "usa", xlim = c(-72, -71), ylim = c(34.8, 35.1))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.6)
abline(h=35)
abline(v = -60)
```




```
cofin <- cofin[, 1:2]

map(database = "usa", xlim = c(-83, -60), ylim = c(23, 47), col = 0)
map(database = "usa", xlim = c(-83, -60), ylim = c(23, 47), add = T)
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
```




```
# change the southernmost tip to exactly 24N so that it alighs with Caribbean polygon
cofin <- cofin[-which(cofin$Y < 24), ]
cofin[1624, 2] <- 24

USeez <- cofin
#write.csv(USeez, file = "eez.csv")
```

Here is a final U.S. South Atlantic EEZ polygon, with a simplified coastline to reduce computational time when subsetting data points.

2.3 Define the Caribbean region

Now we define the boundaries of the Caribbean region. Its northernmost boundary aligns with the EEZ along Florida's coast up to 28.3 degrees N, at which point the boundary cuts southeast until it intersects with the 24N boundary just east of the Bahamas, approximating the Bahamas EEZ. Its western boundary is -90 degrees W and its southern boundary is 8 degrees N (essentially the land boundary of the Caribbean Sea). Its eastern boundary is -71.1 degrees W.

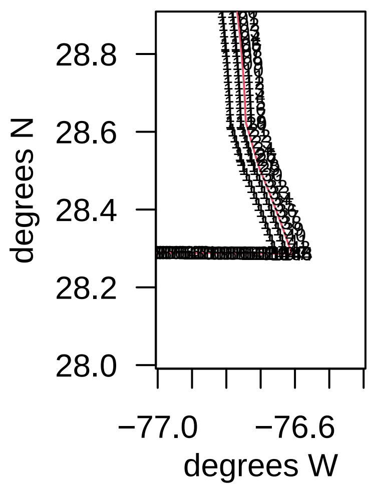
```r
which.min(cofin$Y)
```

```
[1] 1624
```

```r
cofin[1624, ]
```

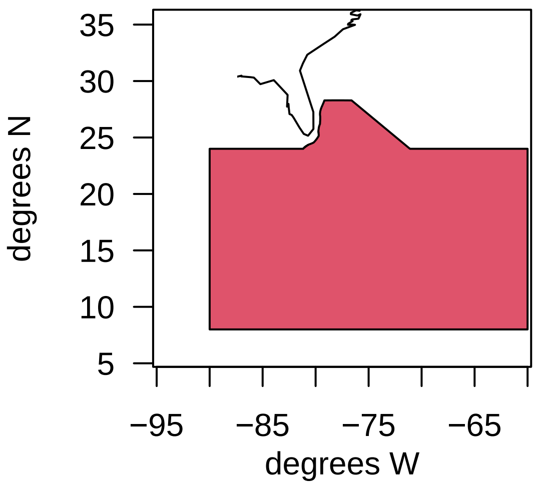
```
             X  Y
3723 -81.16148 24
```

```r
map(database = "usa", xlim = c(-77, -76.4), ylim = c(28, 28.9))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.6)
```




```r
car1 <- cofin[1143:1624, ]
car2 <- data.frame(cbind(c(-90, -90, -60, -60, -71.1), c(24, 8, 8, 24, 24)))
names(car2) <- c("X", "Y")
CAR <- rbind(car1, car2)
```

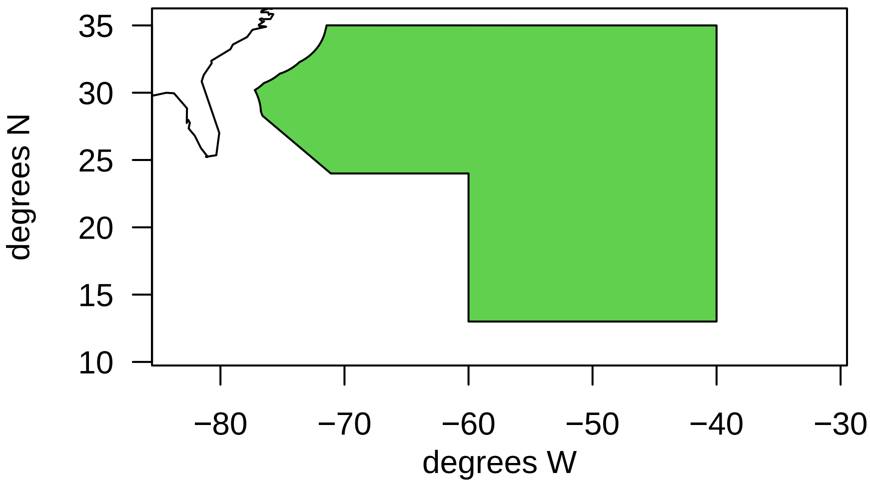
```
names(CAR) <- c("X", "Y")

map(database = "usa", xlim = c(-95, -60), ylim = c(5, 36))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(CAR$X, CAR$Y, col = 2)
```



3 Define the NCA region

The North Central Atlantic (NCA) region, otherwise known as the Western Central Atlantic, is defined according to the boundaries of FAO area 31, but excluding the U.S. EEZ and Caribbean territorial seas to the West.

```
nca1 <- cofin[569:1143, ]  # subset points along EEZ
nca2 <- data.frame(cbind(c(-71.1, -60, -60, -40, -40), c(24, 24, 13, 13, 35))) # define bou
names(nca2) <- c("X", "Y")
NCA <- rbind(nca1, nca2)
names(NCA) <- c("X", "Y")

map(database = "usa", xlim = c(-85, -30), ylim = c(10, 36))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(NCA$X, NCA$Y, col = 3)
```


35
30
25
20
15
10
degrees N
−80
−70
−60
−50
−40
−30
degrees W

4 Define the NED region

The Northeast Distant Waters (NED) region, otherwise known as the Northwesetern Atlantic, is defined according to the boundaries of FAO area 21, but excluding the U.S. EEZ territorial seas to the West. It does include Canadian and French territorial EEZs. The northern boundary is 60 degrees north, as dolphin are not caught above this latitude.

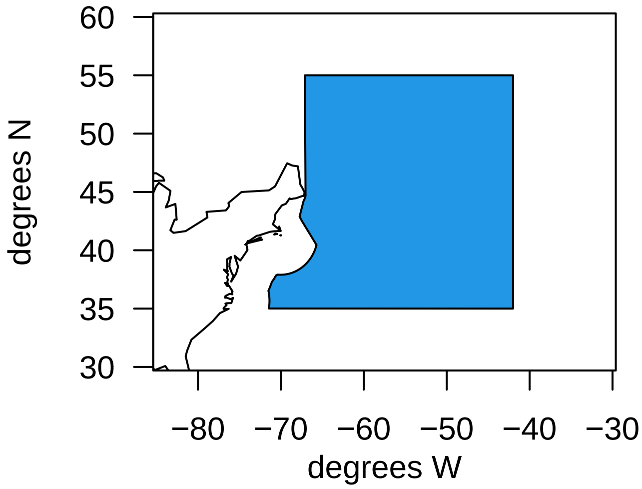
```
which.max(cofin$Y)  # find northernmost point of U.S. EEZ
```

```
[1] 1702
```

```
cofin[1702, ]
```

```
              X        Y
77844 -67.11553 44.84764
```

```
ned1 <- cofin[1:569, ]  # subset points along EEZ
ned2 <- data.frame(cbind(c(-42, -42, -67.1), c(35, 55, 55)))  # define boundaries according
names(ned2) <- c("X", "Y")
NED <- rbind(ned1, ned2)
names(NED) <- c("X", "Y")

map(database = "usa", xlim = c(-85, -30), ylim = c(30, 60))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(NED$X, NED$Y, col = 4)
```


60
55
50
45
40
35
30
degrees N
−80
−70
−60
−50
−40
−30
degrees W

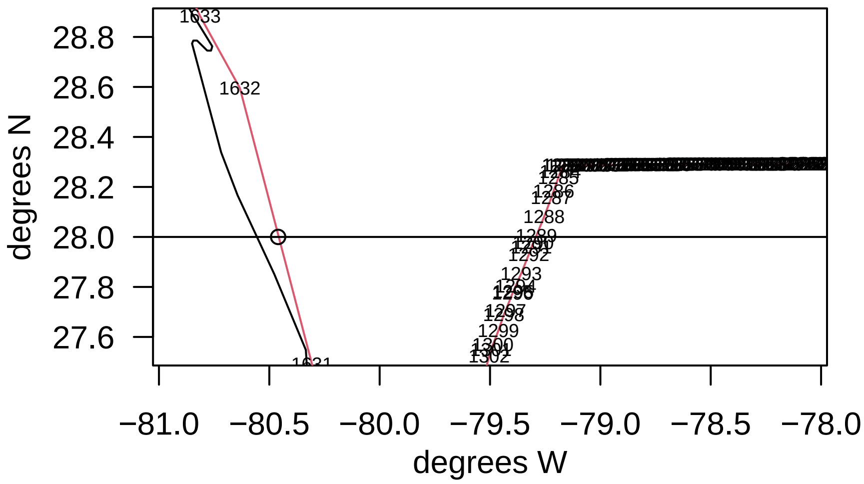
5 Define the Florida Keys region

The southernmost region of the U.S. EEZ is called the FLK region, and includes waters within the EEZ boundary, north up to 28 degrees N. The boundary of 28N is chosen because it aligns with commercial logbook reporting boundaries and aligns roughly with the Brevard - Indian River County border. This is convenient because U.S. recreational statistics are summarized as groups of counties, with SE FL encompassing Miami-Dade to Indian River County and NE FL encompassing Brevard to Nassau County.

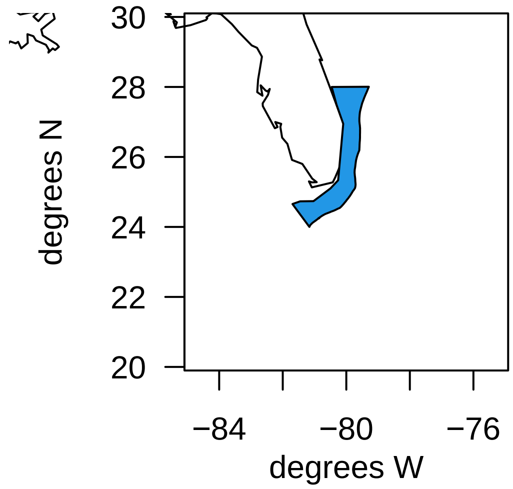
```
which.min(abs(co1$Y - 28))  # find the southernmost point
```

```
[1] 1289
```

```
map(database = "usa", xlim = c(-81, -78), ylim = c(27.5, 28.9))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.6)   # find points falling at 28N
points(-80.46, 28)
cofin[1632, ] <- c(-80.46, 28)
abline(h = 28)
```



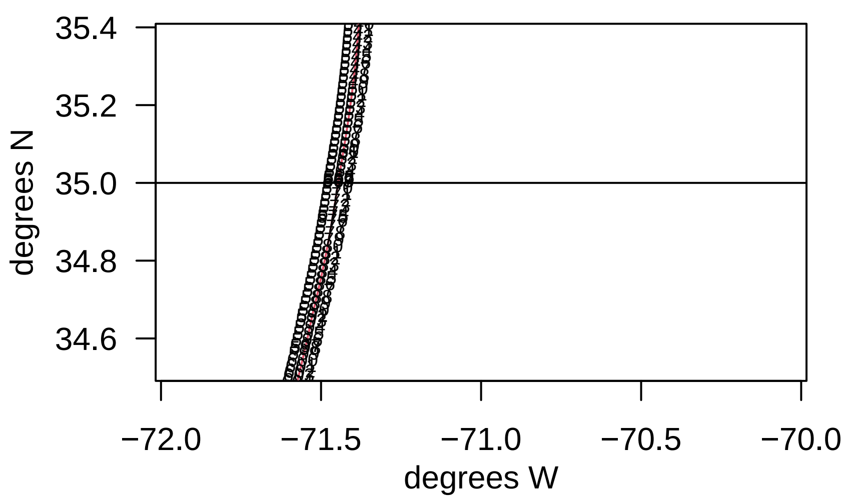

```
FLK <- cofin[1289:1632, ]  # subset points along EEZ

map(database = "usa", xlim = c(-85, -75), ylim = c(20, 30))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(FLK$X, FLK$Y, col = 4)
```


30
28
26
24
22
20
degrees N
−84
−80
−76
degrees W

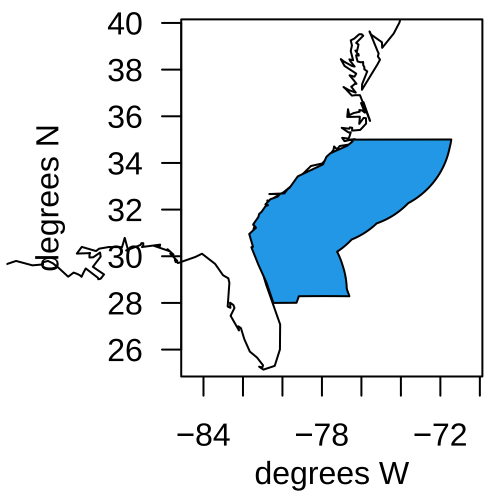
6 Define the North Florida - South NC region

The second area that was decided upon to be defined within the operating model was northern Florida to Southern North Carolina. We form a boundary at 35 degrees north because this is the delineation between the NED and NCA regions. Also, 35N is just south of Cape Hatteras, which is a boundary for NMFS recreational data collection, as well as a boundary of commercial logbook data reporting. Therefore, all sources of data (international, U.S. commercial, and U.S. recreational) can be easily parsed along this boundary.

```
map(database = "usa", xlim = c(-72, -70), ylim = c(34.5, 35.4))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.6)  # find points that fall at 35N
abline(h = 35)
```



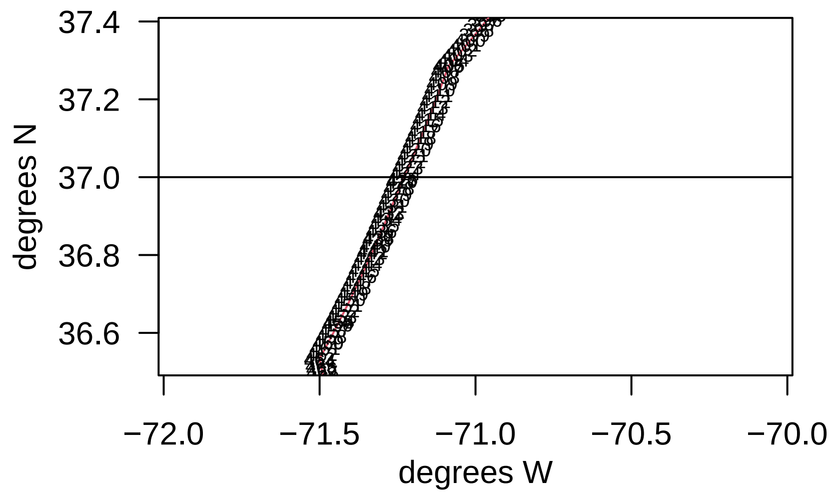
```
cofin[1663, 2] <- 35
NCFL <- rbind(cofin[1632:1663, ], cofin[569:1289, ])  # subset along EEZ

map(database = "usa", xlim = c(-85, -70), ylim = c(25, 40))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(NCFL$X, NCFL$Y, col = 4)
```



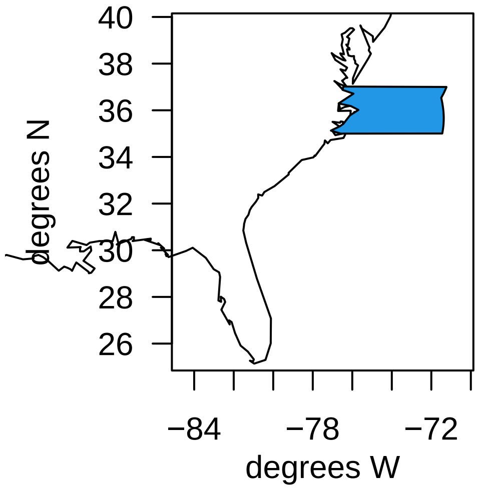
7 Define the northern North Carolina region

The third U.S. area that was decided upon to be defined within the operating model was northern North Carolina, which was intended to encompass the northern part of the state up to the Virginia border. We form a boundary at 37 degrees north because this is a boundary for commercial logbook data reporting and roughly aligns with the latitude of the state boundary between North Carolina and Virginia. Note there is a slight misalignment between this boundary and the way that U.S. recreational statistics are reported (by state), so the commercial and recreational data will be subset along slightly different boundaries for this region.

```
map(database = "usa", xlim = c(-72, -70), ylim = c(36.5, 37.4))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(cofin$X, cofin$Y, border = 2)
text(cofin$X, cofin$Y, 1:nrow(cofin), cex = 0.6)
abline(h = 37)
```




```
cofin[425, 2] <- 37
NNC <- rbind(cofin[1663:1673, ], cofin[425:569, ])

map(database = "usa", xlim = c(-85, -70), ylim = c(25, 40))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(NNC$X, NNC$Y, col = 4)
```

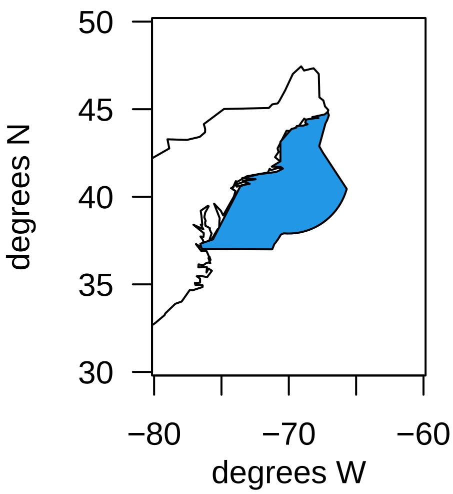


degrees N
40
38
36
34
32
30
28
26
−84
−78
−72
degrees W

8 Define the Virginia and north region

The fourth region within the U.S. EEZ is the remainder of the area, which encompasses national waters from Virginia up to Maine.

```
VBM <- rbind(cofin[1673:1702, ], cofin[1:425, ])

map(database = "usa", xlim = c(-80, -60), ylim = c(30, 50))
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()
polygon(VBM$X, VBM$Y, col = 4)
```



8.1 Plot all of the regions

Now we will put all of the regions on a single map and check that the boundaries align.

```
png(filename = "data/eez/new_spatial_zones1.png", width= 800, height = 900)

map(database = "world", xlim = c(-92, -38), ylim = c(5, 55), col = 8)
mtext(side = 1, "degrees W", line = 2); mtext(side = 2, "degrees N", line = 3)
axis(1); axis(2, las = 2); box()

polygon(CAR$X, CAR$Y, border = 1)
polygon(NCA$X, NCA$Y, border = 1)
polygon(NED$X, NED$Y, border = 1)
polygon(FLK$X, FLK$Y, border = 1)
polygon(NCFL$X, NCFL$Y, border = 1)
polygon(NNC$X, NNC$Y, border = 1)
polygon(VBM$X, VBM$Y, border = 1)

map(database = "world", add = T, col = 8, lwd = 2)

text(-75, 15, "CAR", cex = 1.2)
text(-57, 28, "NCA", cex = 1.2)
text(-57, 42, "NED", cex = 1.2)
text(mean(FLK$X), mean(FLK$Y), "FLK", cex = 1.2)
text(-79.3, mean(NCFL$Y), "NCFL", cex = 1.2)
text(-73.8, mean(NNC$Y), "NNC", cex = 1.2)
text(-71.5, mean(VBM$Y), "VBM", cex = 1.2)

dev.off()
```

```
pdf
  2
```

```
#class(CAR)

CAR$region <- "CAR"
FLK$region <- "FLK"
NCFL$region <- "NCFL"
```

```
NNC$region <- "NNC"
VBM$region <- "VBM"
NCA$region <- "NCA"
NED$region <- "NED"

zones <- rbind(FLK, NCFL, NNC, VBM, NED, NCA, CAR)

# plot(zones$X, zones$Y, col = as.numeric(as.factor(zones$region)))

# output the coordinates to a .csv file
write.csv(zones, file = "data/eez/dolphin_OM_areas.csv", row.names = FALSE)
```

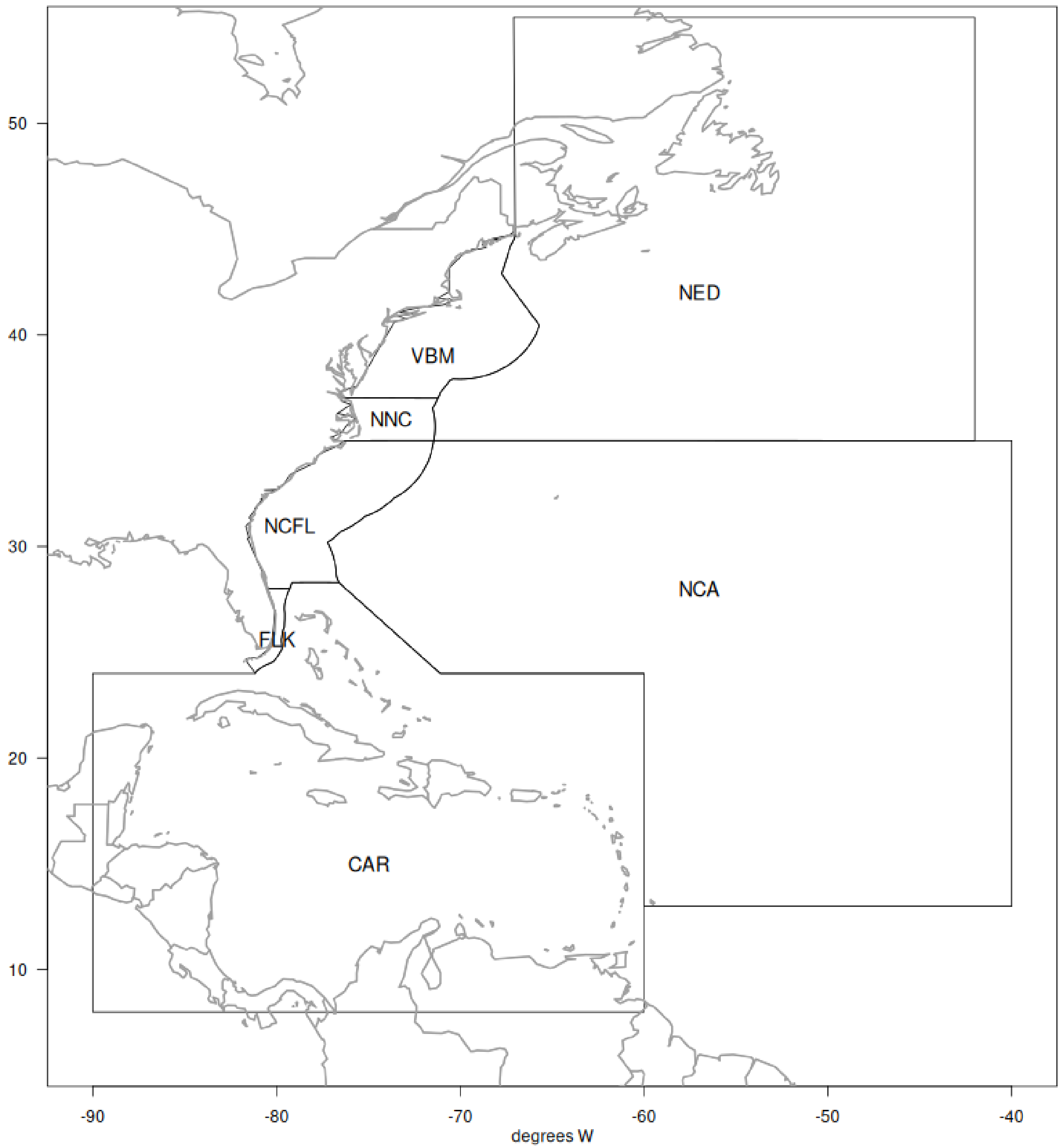


Figure 8.1: Areas defined in the operating model.

9 NOAA Fisheries recreational landings data

Here we will partition recreational landings data from NOAA Fisheries' Marine Recreational Information Program (MRIP), a standardized, peer-reviewed suite of recreational fishing surveys used to produce catch and effort estimates for quota monitoring and stock assessment in the United States. MRIP is a standardized survey suite that employs a complex statistical sampling design stratified across time, regions of the coast, and different fishing modes, which allows the intercept data to be scaled up by effort estimates to calculate total catches.

9.1 Data access and upload

Data are downloaded directly from the Miami server using the files provided to SERO for ACL monitoring. The current location is: M://SFD/SECM-SFD/ACL/2025_Jun23_MRIP_-MRFSS/. The landings are in FES MRIP and the units are in pounds whole weight.

```
# clear workspace
rm(list = ls())

# read in data file
load("data/mrip_fes_rec81_25wv2_23June25.RData")  # most recent file on SFD server
head(dat)
```

```
  REC_ACL_MRIP_FES_ID YEAR MONTH WAVE SUB_REG NEW_ST NEW_STA NEW_MODE NEW_MODEN
1            73407425 1989    NA    4       6      9      NC        4      Priv
2            73407426 1989    NA    4       6      9      NC        4      Priv
3            73407427 1989    NA    4       6      9      NC        4      Priv
4            73407428 1989    NA    4       6      9      NC        4      Priv
5            73407429 1989    NA    4       6      9      NC        4      Priv
6            73407430 1989    NA    4       6      9      NC        4      Priv
  AREA_X NEW_AREA  NEW_AREAN FL_REG NC_REG   DS              NEW_SCI
1      5        5    Inshore     NA      S MRIP Centropristis striata
2      5        5    Inshore     NA      N MRIP Centropristis striata
```

```
3      2        2  Ocean>3mi     NA      N MRIP Centropristis striata
4      2        2  Ocean>3mi     NA      S MRIP Centropristis striata
5      1        1 Ocean<=3mi     NA      N MRIP Centropristis striata
6      1        1 Ocean<=3mi     NA      S MRIP Centropristis striata
        NEW_COM    SP_CODE SPECIES_ITIS SPECIES        AB1          B2
1 black sea bass 8835020301       167687         34181.4033 121019.1368
2 black sea bass 8835020301       167687           795.8612   4445.0527
3 black sea bass 8835020301       167687           549.5390   2919.4840
4 black sea bass 8835020301       167687         71823.2390  24548.2535
5 black sea bass 8835020301       167687           415.7642     87.5928
6 black sea bass 8835020301       167687         48728.5632  11859.6428
           A         B1 LBSEST_SECWWT LBSEST_SECSOURCE SAMPLE_SIZE_USED
1 31761.4755  2419.9278    11457.8425          srysmwa               41
2   584.4155   211.4457      266.7782          srysmwa               41
3   549.5390     0.0000      484.4319          srysmwa              119
4 33343.1844 38480.0546    63313.9276          srysmwa              119
5   415.7642     0.0000      294.0623          srysmwa               71
6 40585.0152  8143.5480    34464.8156          srysmwa               71
   AVG_LEN   AVG_WGT SRYSMWA_AVG_LEN SRYSMWA_AVG_WGT SRYSMWA_SAMPLE_SIZE
1 203.4572 0.3352069        203.4572       0.3352069                  41
2 203.4572 0.3352069        203.4572       0.3352069                  41
3 296.2590 0.8815243        296.2590       0.8815243                 119
4 296.2590 0.8815243        296.2590       0.8815243                 119
5 261.9949 0.7072816        261.9949       0.7072816                  71
6 261.9949 0.7072816        261.9949       0.7072816                  71
  SRYSMWA_FLAGGED SRYSMW_AVG_LEN SRYSMW_AVG_WGT SRYSMW_SAMPLE_SIZE
1              NA       269.2563      0.7310037                231
2              NA       269.2563      0.7310037                231
3              NA       269.2563      0.7310037                231
4              NA       269.2563      0.7310037                231
5              NA       269.2563      0.7310037                231
6              NA       269.2563      0.7310037                231
  SRYSMW_FLAGGED SRYSM_AVG_LEN SRYSM_AVG_WGT SRYSM_SAMPLE_SIZE SRYSM_FLAGGED
1             NA      259.7694     0.6948228               556            NA
2             NA      259.7694     0.6948228               556            NA
3             NA      259.7694     0.6948228               556            NA
4             NA      259.7694     0.6948228               556            NA
5             NA      259.7694     0.6948228               556            NA
6             NA      259.7694     0.6948228               556            NA
  SRYS_AVG_LEN SRYS_AVG_WGT SRYS_SAMPLE_SIZE SRYS_FLAGGED SRY_AVG_LEN
1      270.955    0.8099641              982           NA    269.1276
```

```
2      270.955   0.8099641              982           NA   269.1276
3      270.955   0.8099641              982           NA   269.1276
4      270.955   0.8099641              982           NA   269.1276
5      270.955   0.8099641              982           NA   269.1276
6      270.955   0.8099641              982           NA   269.1276
  SRY_AVG_WGT SRY_SAMPLE_SIZE SRY_FLAGGED SR_AVG_LEN SR_AVG_WGT SR_SAMPLE_SIZE
1   0.8085211            1703          NA   303.6871   1.050403          46055
2   0.8085211            1703          NA   303.6871   1.050403          46055
3   0.8085211            1703          NA   303.6871   1.050403          46055
4   0.8085211            1703          NA   303.6871   1.050403          46055
5   0.8085211            1703          NA   303.6871   1.050403          46055
6   0.8085211            1703          NA   303.6871   1.050403          46055
  SR_FLAGGED S_AVG_LEN S_AVG_WGT S_SAMPLE_SIZE S_FLAGGED     VAR_B2   VAR_AB1
1         NA  344.4373  1.364082        305089        NA 4072492794 310099094
2         NA  344.4373  1.364082        305089        NA    2866289    258947
3         NA  344.4373  1.364082        305089        NA          0         0
4         NA  344.4373  1.364082        305089        NA  105731413 534702898
5         NA  344.4373  1.364082        305089        NA          0         0
6         NA  344.4373  1.364082        305089        NA   22265768 286196169
         SA_LABEL GOM_LABEL REC_ACL MIGRA_GRP AGG_MODEN JURISDICTION PS_REG
1 Snapper Grouper                 1             Private        State      6
2 Snapper Grouper                 1             Private        State      5
3 Snapper Grouper                 1             Private      Federal      5
4 Snapper Grouper                 1             Private      Federal      6
5 Snapper Grouper                 1             Private        State      5
6 Snapper Grouper                 1             Private        State      6
  COUNCILREG GFMC_FMP SAFMC_FMU LBSEST_SECGWT AGGR_AREA AREA FIRST_MONTH
1          6                  Y     9710.0360             NA          NA
2          5                  Y      226.0832             NA          NA
3          5                  Y      410.5355             NA          NA
4          6                  Y    53655.8708             NA          NA
5          5                  Y      249.2054             NA          NA
6          6                  Y    29207.4709             NA          NA
  LAST_MONTH ALT_FLAG CHTS_CL CHTS_H CHTS_RL CHTS_WAB1C CHTS_WAB1H MODE_FX
1         NA        0       0      0       0          0          0       7
2         NA        0       0      0       0          0          0       7
3         NA        0       0      0       0          0          0       7
4         NA        0       0      0       0          0          0       7
5         NA        0       0      0       0          0          0       7
6         NA        0       0      0       0          0          0       7
  NEW_SEASN SEASON
```

```
1               NA
2               NA
3               NA
4               NA
5               NA
6               NA
```

```
#apply(dat[2:18], 2, table, useNA = "always")

# confirm nomenclature for dolphin and subset by species --------------------------
table(dat$NEW_COM[grep("dolphin", dat$NEW_COM)])
```

```
dolphin
   9377
```

```
table(dat$NEW_SCI[grep("dolphin", dat$NEW_COM)])
```

```
Coryphaena hippurus
               9377
```

```
d <- dat[which(dat$NEW_COM == "dolphin"), ]
#apply(d[2:18], 2, table, useNA = "always")
d <- d[which(d$YEAR < 2025), ]    # take out 2025 because incomplete year
tot <- tapply(d$LBSEST_SECWWT, d$YEAR, sum, na.rm = T)
```

9.2 Separate Gulf and Atlantic recreational landings

Now we filter the data set to separate out the Gulf versus Atlantic landings.

```
# codes for WFL regions: 1 (NW FL Panhandle- Escambia to Dixie co.), 2 (SW FL Peninsula- Le
# subregional codes: 6 is Atlantic and 7 is Gulf
table(d$FL_REG, d$SUB_REG, useNA = "always")
```

```
        4    5    6    7 <NA>
  1     0    0    0  387    0
  2     0    0    0  110    0
  3     0    0    0  568    0
  4     0    0 1022    0    0
  5     0    0  312    0    0
  <NA> 96  585 3836 2426    0
```

```
unique(d$NEW_STA)
```

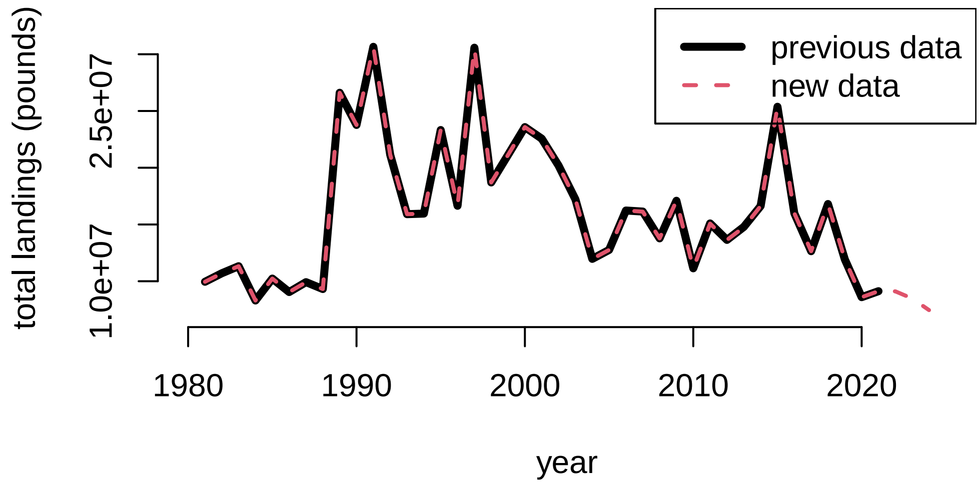
```
 [1] "NC"      "FLE"     "FLW"     "SC"      "GA"      "MS"      "AL"      "LA"
 [9] "TX"      "FLE/GA"  "RI"      "VA"      "DE"      "MD"      "NJ"      "CT"
[17] "MA"      "NY"      "LA/MS"   "AL/FLW"
```

```
d1 <- d[which(d$COUNCILREG == 6), ]
dg <- d[which(d$COUNCILREG == 7), ]
table(d1$COUNCILREG)
```

```

   6
6419
```

Now we can summarize the total dolphin landings in weight for the U.S. Atlantic, and compare to previous values sent for quota monitoring. We see that the values are almost identical.

```
# summarize by year and compare --------------------------

tot1 <- tapply(d1$LBSEST_SECWWT, d1$YEAR, sum, na.rm = T)
#head(tot1)

old <- read.csv("data/RecreationalDolphinLandings_SAandGOM_MLarkin.csv", skip = 6)
plot(old$Year, old$ATL, type = "l", xlab = "year", ylab = "total landings (pounds)", lwd =
     xlim = c(1980, 2025), ylim = c(7*10^6, 3.3*10^7), bty = "n")
lines(as.numeric(names(tot1)), tot1, lwd = 2, col = 2, lty = 2)
legend("topright", c("previous data", "new data"), col = c(1, 2), lwd = c(4, 2), lty = c(1,
```



9.3 Partition landings by area and sector

Now we will partition out the total landings according to the dolphin MSE areas. The areas are as follows:

- FLK: Florida Keys to Indian River County County (FL)
- NCFL: Brevard (FL) to Southern NC (South of Hatteras)
- NNC: Northern NC (North of Hatteras to NC/VA border)
- VBM: Virginia to Maine

```
# separate out 4 regions --------------------------

d1$region <- "VBM"

# select for FLK
# FL_REG codes: 4: SE FL- Miami-Dade to Indian River County.; 5: NE FL- Brevard to Nassau C
table(d1$FL_REG, d1$NEW_STA, useNA = "always")
```

```
          CT    DE   FLE FLE/GA   FLW    GA    MA    MD    NC    NJ    NY    RI    SC    VA
  3        0     0     0      0   568     0     0     0     0     0     0     0     0     0
  4        0     0  1022      0     0     0     0     0     0     0     0     0     0     0
  5        0     0   312      0     0     0     0     0     0     0     0     0     0     0
  <NA>    14   114     0   1794     0    54    20   125  1344   108    73    62   644   165
      
        <NA>
  3        0
  4        0
  5        0
  <NA>     0
```

```
d1$region[which(d1$FL_REG == 3 | d1$FL_REG == 4)] <- "FLK"

# select for NCFL
d1$region[which(d1$FL_REG == 5)] <- "NCFL"
lis <- c("SC", "GA", "FLE/GA")            # NCFL states
d1$region[which(d1$NEW_STA %in% lis)] <- "NCFL"
# NC_REG codes: N: North of Cape Hatteras; S: South of Cape Hatteras
table(d1$NC_REG, d1$NEW_STA, useNA = "always")
```

```
          CT    DE   FLE FLE/GA   FLW    GA    MA    MD    NC    NJ    NY    RI    SC    VA
          14   114  1334   1794   568    54    20   125   586   108    73    62   644   165
  N        0     0     0      0     0     0     0     0   420     0     0     0     0     0
  S        0     0     0      0     0     0     0     0   338     0     0     0     0     0
  <NA>     0     0     0      0     0     0     0     0     0     0     0     0     0     0
      
        <NA>
           0
  N        0
  S        0
  <NA>     0
```

```
d1$region[which(d1$NC == "S")] <- "NCFL"

# select for NNC
d1$region[which(d1$NC == "N")] <- "NNC"
```

```r
table(d1$FL_REG, d1$NEW_STA, d1$region)
```

```
, ,  = FLK

    
       CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
  3     0    0    0      0  568    0    0    0    0    0    0    0    0    0
  4     0    0 1022      0    0    0    0    0    0    0    0    0    0    0
  5     0    0    0      0    0    0    0    0    0    0    0    0    0    0

, ,  = NCFL

    
       CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
  3     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  4     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  5     0    0  312      0    0    0    0    0    0    0    0    0    0    0

, ,  = NNC

    
       CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
  3     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  4     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  5     0    0    0      0    0    0    0    0    0    0    0    0    0    0

, ,  = VBM

    
       CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
  3     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  4     0    0    0      0    0    0    0    0    0    0    0    0    0    0
  5     0    0    0      0    0    0    0    0    0    0    0    0    0    0
```

```r
table(d1$NC_REG, d1$NEW_STA, d1$region)  # check classification
```

```
, ,  = FLK
```

```
      CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
       0    0 1022      0  568    0    0    0    0    0    0    0    0    0
  N    0    0    0      0    0    0    0    0    0    0    0    0    0    0
  S    0    0    0      0    0    0    0    0    0    0    0    0    0    0

, ,  = NCFL


      CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
       0    0  312   1794    0   54    0    0    0    0    0    0  644    0
  N    0    0    0      0    0    0    0    0    0    0    0    0    0    0
  S    0    0    0      0    0    0    0    0  338    0    0    0    0    0

, ,  = NNC


      CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
       0    0    0      0    0    0    0    0    0    0    0    0    0    0
  N    0    0    0      0    0    0    0    0  420    0    0    0    0    0
  S    0    0    0      0    0    0    0    0    0    0    0    0    0    0

, ,  = VBM


      CT   DE  FLE FLE/GA  FLW   GA   MA   MD   NC   NJ   NY   RI   SC   VA
      14  114    0      0    0    0   20  125  586  108   73   62    0  165
  N    0    0    0      0    0    0    0    0    0    0    0    0    0    0
  S    0    0    0      0    0    0    0    0    0    0    0    0    0    0
```

```r
# summarize by Gulf and Atlantic regions and check against total -----------------------
atl <- tapply(d1$LBSEST_SECWWT, list(d1$YEAR, d1$region), sum, na.rm = T)
GULF <- tapply(dg$LBSEST_SECWWT, dg$YEAR, sum, na.rm = T)

table(rownames(atl) == names(GULF))
```

```
TRUE
  44
```

```
rec <- cbind(GULF, atl)

#plot(rowSums(rec), tot)
rowSums(rec)/tot
```

```
1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996
   1    1    1   NA    1    1    1    1    1    1    1    1    1    1    1    1
1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012
   1    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024
   1    1    1    1    1    1    1    1    1    1    1    1
```

```
write.csv(rec, file = "data/recLandings.csv")
```

Now we will separate out the landings by recreational sector and plot the totals by region and sector.

```
# separate out by for hire vs. private rec --------------------------
d1$sector <- "ForHire"

d1$sector[which(d1$NEW_MODEN == "Shore")] <- "Rec"
d1$sector[which(d1$NEW_MODEN == "Priv")]  <- "Rec"

table(d1$sector, d1$NEW_MODEN, useNA = "always")    # check classification
```

```
          Cbt Cbt/Hbt  Hbt Priv Shore <NA>
  ForHire 1616     119 2912    0     0    0
  Rec        0       0    0 1750    22    0
  <NA>       0       0    0    0     0    0
```

```
tot <- tapply(d1$LBSEST_SECWWT, list(d1$YEAR, d1$region, d1$sector), sum, na.rm = T)

head(tot)
```

```
, , ForHire

          FLK      NCFL        NNC       VBM
```

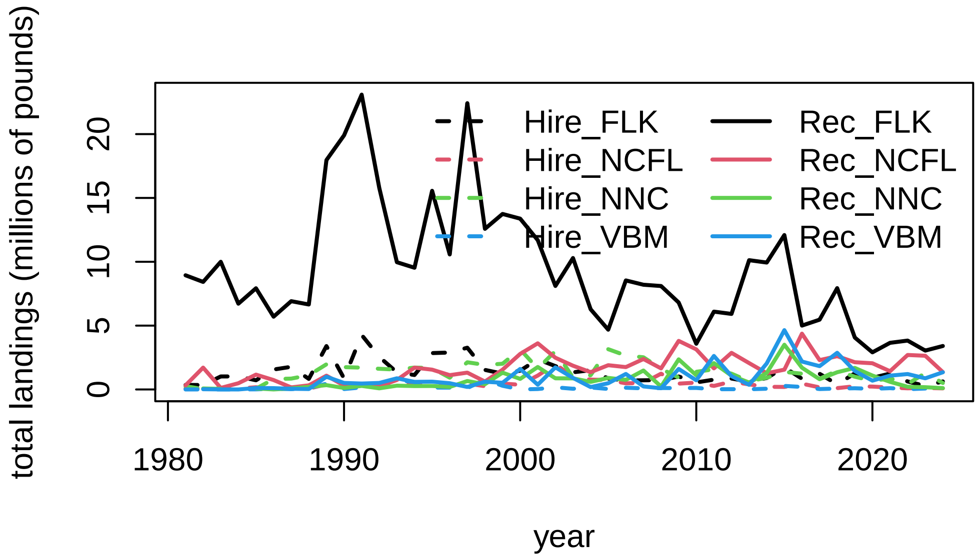
```
1981  354639.5  74702.50 256755.24  1400.247
1982  347458.2  92271.42  27578.44  2749.224
1983 1020005.8  41784.04  88690.53  5379.177
1984 1021086.8  74658.81        NA  4203.048
1985  717523.0 172989.65  58514.13  8880.857
1986 1562354.0  63555.89 817192.26 61658.193

, , Rec

          FLK      NCFL       NNC        VBM
1981  8942641  322505.7        NA         NA
1982  8422673 1711834.7 78367.161  18727.290
1983 10000662  119370.4 44226.032   5456.237
1984  6717044  486079.7        NA      0.000
1985  7925301 1162708.4 77477.096 112944.274
1986  5699311  736872.0  2549.586 103945.134
```

```r
fin <- data.frame(cbind(tot[, , 1], tot[, , 2]))
names(fin) <- c("Hire_FLK", "Hire_NCFL", "Hire_NNC", "Hire_VBM",
                "Rec_FLK", "Rec_NCFL", "Rec_NNC", "Rec_VBM")
head(fin)
```

```
      Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM  Rec_FLK  Rec_NCFL   Rec_NNC
1981  354639.5  74702.50 256755.24  1400.247  8942641  322505.7        NA
1982  347458.2  92271.42  27578.44  2749.224  8422673 1711834.7 78367.161
1983 1020005.8  41784.04  88690.53  5379.177 10000662  119370.4 44226.032
1984 1021086.8  74658.81        NA  4203.048  6717044  486079.7        NA
1985  717523.0 172989.65  58514.13  8880.857  7925301 1162708.4 77477.096
1986 1562354.0  63555.89 817192.26 61658.193  5699311  736872.0  2549.586
        Rec_VBM
1981         NA
1982  18727.290
1983   5456.237
1984      0.000
1985 112944.274
1986 103945.134
```

```r
matplot(rownames(fin), fin/10^6, lty = c(rep(2, 4), rep(1, 4)),
        col = rep(1:4, 2), type = "l", lwd = 2,
```

```
        xlab = "year", ylab = "total landings (millions of pounds)")
legend("topright", colnames(fin), lty = c(rep(2, 4), rep(1, 4)), col = rep(1:4, 2), lwd = 2
```




```
# check that totals match
cor(rowSums(fin, na.rm = T), tot1)
```

```
[1] 1
```

```
mean(rowSums(fin, na.rm = T) - tot1)    # very tiny differences
```

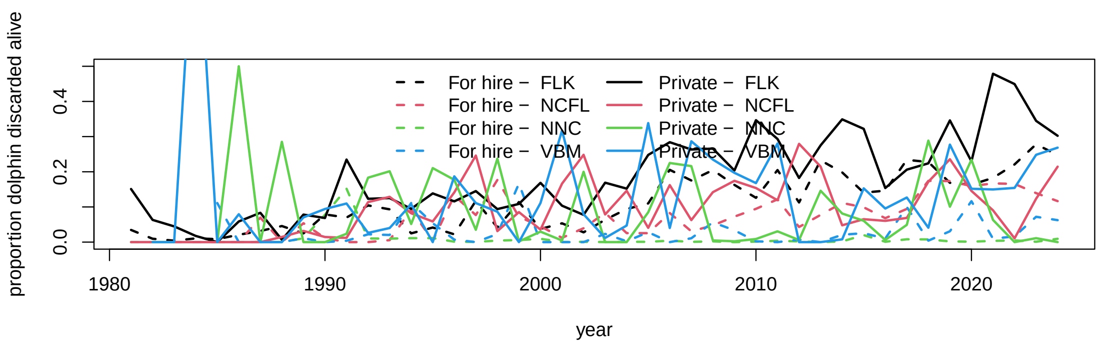
```
[1] 4.233284e-11
```

Here are the landings (in pounds) for years 1981 to 2024.

9.4 Looking at extent of discarding

Discard estimates (known as B2 estimates) are provided in numbers by the MRIP survey but are not available in the Southeast Region Headboat Survey. We can look at the overall numbers of dolphinfish caught versus reported released from the MRIP survey to get an understanding of the extent of discarding.

```
d1$perdis <- d1$B2 / (d1$AB1 + d1$B2)  # percent discarded is B2 (fish released alive) / to

# calculate mean discard rate by year, region, sector ----------------
dis <- tapply(d1$perdis, list(d1$YEAR, d1$region, d1$sector), mean, na.rm = T)
disr <- tapply(d1$perdis, list(d1$region, d1$sector), mean, na.rm = T)

matplot(rownames(dis), cbind(dis[, , 1], dis[, , 2]), lty = c(rep(2, 4), rep(1, 4)), col =
        type = "l", lwd = 2, xlab = "year", ylab = "proportion dolphin discarded alive", yl
legend("top", c(paste("For hire - ", colnames(dis)), paste("Private - ", colnames(dis))),
       lty = c(rep(2, 4), rep(1, 4)), col = rep(1:4, 2), lwd = 2, ncol = 2, bty = "n")
```




```
round(disr * 100, 2)  # discarding rates by region and sector
```

```
     ForHire   Rec
FLK    10.79 18.76
NCFL    9.44 11.23
NNC     0.94  9.62
VBM     2.94 12.69
```

According to the MRIP survey and the reported numbers, discarding generally increases over time, excluding some highly anomalous estimates from the early years which are driven by low sample sizes. Discarding is generally higher in the private sector compared to the for-hire sector, and is highest in the Florida Keys region.

9.5 Split the landings by season

The dolphin MSE operates on a seasonal time step and thus the landings and discards must by parsed apart by individual season and year. This is not straightforward because MRIP estimates are reported in two-month waves (Jan - Feb, Mar - Apr, May - Jun, etc.), and the seasons are 3-month periods (Dec - Feb, Mar - May, Jun - Aug, Sep - Nov). For the two waves that have to be split into seasons (e.g., December is lumped with January and February; June is lumped with July and August), we split the wave 50/50%, i.e, we make the assumption that half of the landings in that wave occurs in one season and the other half occurs in the next season.

```
# define seasons ----------------------------
table(d1$WAVE, useNA = "always")
```

```

   0    1    2    3    4    5    6 <NA>
   5  407  718 1230 1439 1095  581  944
```

```
d1$YRWAVE <- paste0(d1$YEAR, "_", d1$WAVE)

totw <- tapply(d1$LBSEST_SECWWT, list(d1$YRWAVE, d1$region, d1$sector), sum, na.rm = T)

f <- data.frame(cbind(totw[, , 1], totw[, , 2]))
names(f) <- c("Hire_FLK", "Hire_NCFL", "Hire_NNC", "Hire_VBM", "Rec_FLK", "Rec_NCFL", "Rec_
head(f)
```

```
          Hire_FLK Hire_NCFL   Hire_NNC Hire_VBM   Rec_FLK  Rec_NCFL Rec_NNC
1981_2          NA        NA         NA       NA 2793285.5 270968.46      NA
1981_3  300873.764        NA 254049.420       NA 3915698.3        NA      NA
1981_4   50017.460        NA   2705.821       NA  905621.2  51537.26      NA
1981_5    3748.301        NA         NA       NA  479834.0        NA      NA
1981_6          NA        NA         NA       NA  848201.5        NA      NA
1981_NA         NA   74702.5         NA 1400.247        NA        NA      NA
        Rec_VBM
1981_2       NA
1981_3       NA
1981_4       NA
1981_5       NA
1981_6       NA
1981_NA      NA
```

```r
f$YEAR <- as.numeric(substr(rownames(f), 1, 4))
f$WAVE <- as.numeric(substr(rownames(f), 6, 8))
head(f)   # table with landings separated by wave and year
```

```
          Hire_FLK Hire_NCFL   Hire_NNC Hire_VBM   Rec_FLK  Rec_NCFL Rec_NNC
1981_2          NA        NA         NA       NA 2793285.5 270968.46      NA
1981_3  300873.764        NA 254049.420       NA 3915698.3        NA      NA
1981_4   50017.460        NA   2705.821       NA  905621.2  51537.26      NA
1981_5    3748.301        NA         NA       NA  479834.0        NA      NA
1981_6          NA        NA         NA       NA  848201.5        NA      NA
1981_NA         NA   74702.5         NA 1400.247        NA        NA      NA
        Rec_VBM YEAR WAVE
1981_2       NA 1981    2
1981_3       NA 1981    3
1981_4       NA 1981    4
1981_5       NA 1981    5
1981_6       NA 1981    6
1981_NA      NA 1981   NA
```

```r
f[is.na(f)] <- 0

f <- f[which(f$YEAR >= 1985), ]
f <- f[which(f$YEAR <= 2024), ]

table(f$YEAR)
```

```
1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000
   7    7    7    7    7    7    7    7    7    7    7    7    6    6    6    6
2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016
   6    6    6    6    6    6    6    6    6    6    6    6    7    7    7    6
2017 2018 2019 2020 2021 2022 2023 2024
   6    6    6    6    6    6    6    6
```

```r
table(f$WAVE)
```

```
 0  1  2  3  4  5  6
15 40 40 40 40 40 40
```

```
# deal with NAs in many locations ---------------
# calculate percentages of landings by wave, for each region and fleet
avw <- tapply(d1$LBSEST_SECWWT, list(d1$WAVE, d1$region, d1$sector), sum, na.rm = T)
avw <- avw[-1, , ]
avw <- data.frame(cbind(avw[, , 1], avw[, , 2]))
names(avw) <- c("Hire_FLK", "Hire_NCFL", "Hire_NNC", "Hire_VBM", "Rec_FLK", "Rec_NCFL", "Re
avwp <- apply(avw, 2, function (x) x / sum(x, na.rm = T))
#colSums(avwp)

# go through each column, find NAs and fill in waves by percentages
pre_colsum <- colSums(f, na.rm = T)  # column sums to check later

lis <- which(f$WAVE == 0)  # list of rows with NA wave data

for (i in 1:8) {      # loop through each column
  for(j in lis) {     # loop through list of NA waves
    if (f[j, i] > 0) {  # if there are NA wave landings, split by known proportion from oth
      miscatch <- avwp[, i] * f[j, i]   # missing landings are known proportion * NA wave l
      f[which(f$YEAR == f$YEAR[j] & f$WAVE != 0), i] <- f[which(f$YEAR == f$YEAR[j] & f$WAV
      f[j, i] <- 0   # set to zero after landings is attributed to waves
    }
  }
}
f[lis, 1:8] # should be all zeros now
```

```
        Hire_FLK Hire_NCFL Hire_NNC Hire_VBM Rec_FLK Rec_NCFL Rec_NNC Rec_VBM
1985_NA        0         0        0        0       0        0       0       0
1986_NA        0         0        0        0       0        0       0       0
1987_NA        0         0        0        0       0        0       0       0
1988_NA        0         0        0        0       0        0       0       0
1989_NA        0         0        0        0       0        0       0       0
1990_NA        0         0        0        0       0        0       0       0
1991_NA        0         0        0        0       0        0       0       0
1992_NA        0         0        0        0       0        0       0       0
1993_NA        0         0        0        0       0        0       0       0
1994_NA        0         0        0        0       0        0       0       0
1995_NA        0         0        0        0       0        0       0       0
1996_NA        0         0        0        0       0        0       0       0
2013_0         0         0        0        0       0        0       0       0
2014_0         0         0        0        0       0        0       0       0
```

```
2015_0          0          0          0          0          0          0          0          0
```

```
pos_colsum <- colSums(f, na.rm = T)
pre_colsum[1:8] == pos_colsum[1:8]  # check that landings were parsed correctly - column su
```

```
 Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM   Rec_FLK  Rec_NCFL   Rec_NNC   Rec_VBM
     TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```
f <- f[which(f$WAVE != 0), ]  # remove NA waves which now contain no landings

# split waves 3 and 6 50/50% to parse into seasons ------------------
lis <- which(f$WAVE == 3 | f$WAVE == 6)  # list of 3 and 6 waves for all years

for (i in lis) {           # go through list of waves
  temp <- f[i, ] / 2       # split landings in half
  temp[9] <- f$YEAR[i]     # reformat year and wave data columns
  temp[10] <- f$WAVE[i]
  temp <- rbind(temp, temp)   # create two half-waves and relabel with new wave reference n
  temp[1, 10] <- temp[1, 10] - 0.5
  temp[2, 10] <- temp[2, 10] + 0.5
  f <- rbind(f, temp)         # concatenate with main data table
  }

f <- f[-lis, ]  # remove waves that have been split in half

pos_colsum <- colSums(f, na.rm = T)
pre_colsum[1:8] == pos_colsum[1:8]  # check that landings were parsed correctly - column su
```

```
 Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM   Rec_FLK  Rec_NCFL   Rec_NNC   Rec_VBM
     TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```
f <- f[order(f$YEAR, f$WAVE), ]  # reorder table in chronological order

f$YEAR[which(f$WAVE == 6.5)] <- f$YEAR[which(f$WAVE == 6.5)] + 1  # add year to December wa
f$WAVE[which(f$WAVE == 6.5)] <- 0.5  # change December wave to small number so it gets summ

table(f$WAVE)
```

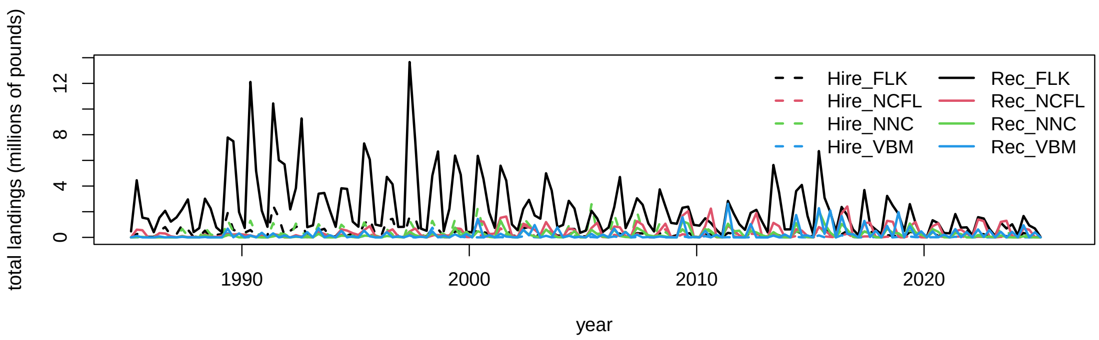
```
0.5   1   2 2.5 3.5   4   5 5.5
 40  40  40  40  40  40  40  40
```

```
f$seas <- cut(f$WAVE, breaks = c(0, 1.5, 3, 4.5, 6))
#table(f$seas)
f$seas1 <- NA
f$seas1[which(f$seas == "(0,1.5]")] <- "DJF"  # Dec Jan Feb is 0.5 and 1
f$seas1[which(f$seas == "(1.5,3]")] <- "MAM"  # Mar Apr May is 2 and 2.5
f$seas1[which(f$seas == "(3,4.5]")] <- "JJA"  # Jun Jul Aug is 3.5 and 4
f$seas1[which(f$seas == "(4.5,6]")] <- "SON"  # Sep Oct Nov is 5 and 5.5

temp <- tapply(f[, 1], list(f$seas1, f$YEAR), sum, na.rm = T)
temp1 <- matrix(temp)
for (i in 2:8) {
  temp2 <- matrix(tapply(f[, i], list(f$seas1, f$YEAR), sum, na.rm = T))
  temp1 <- cbind(temp1, temp2)
  }

findat <- data.frame(sort(rep(as.numeric(colnames(temp)), 4)),
                     rep(rownames(temp), ncol(temp)), temp1)
names(findat) <- c("year", "season", names(f)[1:8])

colSums(f[, 1:8], na.rm = T) == colSums(findat[, 3:10], na.rm = T)  # check that total land
```

```
 Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM   Rec_FLK  Rec_NCFL   Rec_NNC   Rec_VBM
     TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

We plot the results by region and sector, over the time series of year and season. We can clearly see the seasonality in the landings by region.

```
findat$yrseas <- findat$year + as.numeric(as.factor(findat$season))/4-0.125
matplot(findat$yrseas, findat[, 3:10]/10^6, lty = c(rep(2, 4), rep(1, 4)), col = rep(1:4, 2
        xlab = "year", ylab = "total landings (millions of pounds)")
legend("topright", colnames(findat[, 3:10]), lty = c(rep(2, 4), rep(1, 4)), col = rep(1:4,
```




```
# check that totals match
cbind(colSums(findat[, 3:10], na.rm = T), colSums(fin[-c(1:5), ], na.rm = T))
```

```
                [,1]      [,2]
Hire_FLK    55604999  54887476
Hire_NCFL   16909916  16736926
Hire_NNC    57944903  57886388
Hire_VBM     5836953   5828072
Rec_FLK    371462933 363537632
Rec_NCFL    70803866  69641157
Rec_NNC     32556596  32479119
Rec_VBM     40081529  39968585
```

```
# the values should be close but slightly >1 because the last data set includes December 19
colSums(findat[, 3:10], na.rm = T) / colSums(fin[-c(1:5), ], na.rm = T)
```

```
 Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM   Rec_FLK  Rec_NCFL   Rec_NNC   Rec_VBM
 1.013073  1.010336  1.001011  1.001524  1.021800  1.016696  1.002385  1.002826
```

Here we will format the data for input to the operating model.

```
tab3 <- findat

flt <- c(rep("Hire", nrow(tab3)*4), rep("Rec", nrow(tab3)*4))

yrmon <- as.numeric(tab3$yrseas)
yrs <- floor(yrmon)
```

```r
qrt <- (yrmon - floor(yrmon)) * 4 + 0.5
area <- c(rep("FLK", nrow(tab3)), rep("NCFL", nrow(tab3)),
          rep("NNC", nrow(tab3)), rep("VBM", nrow(tab3)))
area <- rep(area, 2)

findat1 <- data.frame(yrs, qrt, flt, area, matrix(as.matrix(tab3[,3:10])))
names(findat1) <- c("Year", "Quarter", "Fleet", "Area", "Catch_lbs")
head(findat1)
```

```
  Year Quarter Fleet Area Catch_lbs
1 1985       1  Hire  FLK   4462.09
2 1985       2  Hire  FLK 253487.38
3 1985       3  Hire  FLK 380811.37
4 1985       4  Hire  FLK  41469.99
5 1986       1  Hire  FLK  40545.41
6 1986       2  Hire  FLK 501184.61
```

```r
#plot(findat1$Catch_lbs, type = "l")
#plot(findat1$Catch_lbs ~ factor(findat1$Area))
#plot(findat1$Catch_lbs ~ factor(findat1$Quarter))
#plot(findat1$Catch_lbs ~ factor(findat1$Year))

findat1 <- findat1[which(findat1$Year <= 2022), ]
findat1 <- findat1[which(findat1$Year >= 1986), ]
#apply(findat1, 2, table)

findat <- findat[which(findat$year >=1986 & findat$year <= 2022), ]
#check that the numbers are exactly the same
tapply(findat1$Catch_lbs, list(findat1$Area, findat1$Fleet), sum, na.rm = T)
```

```
         Hire       Rec
FLK  54012811 357091651
NCFL 16532764  65643193
NNC  56097190  32197622
VBM   5736592  37749487
```

```r
colSums(findat[3:10], na.rm = T)
```

```
 Hire_FLK Hire_NCFL  Hire_NNC  Hire_VBM   Rec_FLK  Rec_NCFL   Rec_NNC   Rec_VBM 
 54012811  16532764  56097190   5736592 357091651  65643193  32197622  37749487 
```

```
# output final numbers in correct format
write.csv(findat1, file = "data/FINAL_files/rec_catch_TomFormat.csv", row.names = FALSE)
```

10 NOAA Fisheries commercial data

We do not display the commercial data analysis here because it is based on logbook reporting and trip ticket databases, which both contain confidential data. The code for processing the commercial data can be seen in the file "commercial.R" and the steps are outlined below.

10.1 Analysis of logbook data

We first use the pelagic longline logbook data to understand how the fleet operates in the high seas on a seasonal basis. These numbers are used later to estimate the seasonality of the international landings, because those are only reported on an annual basis but quarterly values are needed.

The steps are as follows:

1. Read in the pelagic longline data which contains set level entries including location of fishing, target species, gear used, other details of the fishing trip and number of dolphin caught, discarded alive, discarded dead, and the total pounds of dolphin kept.
2. Format the longitude and latitude of the sets into decimal degrees, and find the points that fall within each of the seven polygons representing the areas of the operating model. Check that subsetting was done correctly and add and area identifier.
3. Look at the characteristics of the dolphin catch and calculate the discard rates by trip. Dead discarding rate is found to be 1.4% on average and total discard rate is 2.9% on average.
4. Standardize date formatting and add identifier for season (quarter). Specify a new year variable to allow December to be grouped with January and February.
5. Summarize the total dolphin landings (in pounds) by year, quarter, and area.

6. Convert the total pounds by year-quarter and area to proportions such that each year-quarter combination has four seasonal proportions summing to one. Save the matrix of proportions for later analysis.

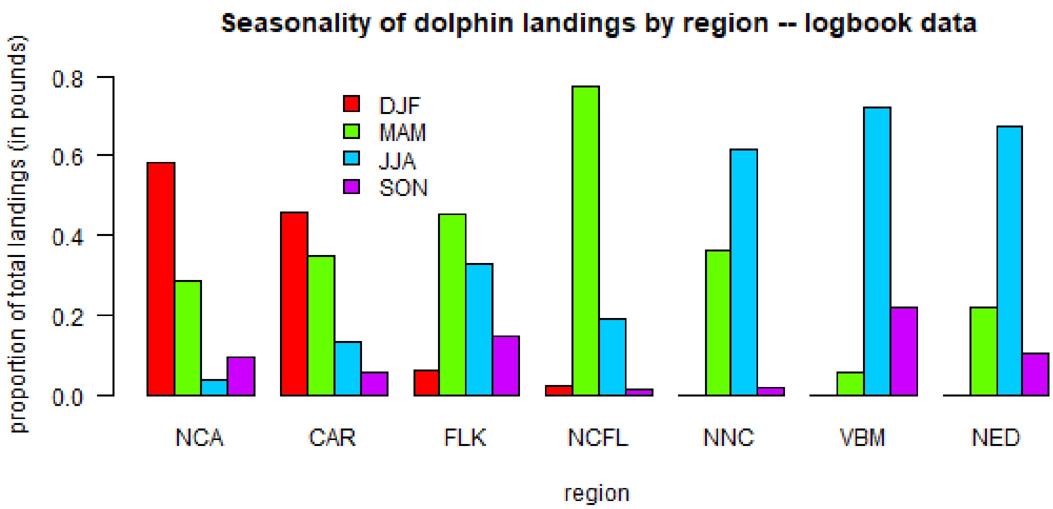


Figure 10.1: Seasonality of the pelagic longline fleet according to the logbook data. Figure shows the proportion of dolphin landings occurring in each season, for each region of the operating model.

10.2 Analysis of trip ticket data

Not all of the commercial dolphin catch activity is reported in the logbooks because dolphin are targeted via a variety of permits and gear types, not all of which have the same reporting requirements. Therefore, in order to parse the commercial landings by year, area and quarter, we use the commercial trip ticket data compiled by NOAA Fisheries. These are the same data used for quota monitoring and thus are the best representation of the total landings.

The steps are as follows:

1. Read in the trip ticket data file, which contains commercial dolphinfish landings for the Atlantic coast from 1986-2023, including all landings by year even if the gear type or area is unknown. The dataset specifies the year and month of

landing, an area of fishing, gear, and the state where landed. Pounds are in units of whole weight.

2. Read in operating model regions and classify the list of logbook areas into those seven regions. Assign each reported area to an operating model region accordingly.
3. Standardize date formatting and add identifier for season (quarter). Specify a new year variable to allow December to be grouped with January and February.
4. Summarize the landings by year-quarter, gear (PLL gear vs. other gears) and region.
5. For most combinations of year-quarter, region and gear, there are substantial landings for which the area of fishing is unknown and therefore the region cannot be defined. We assume that those landings follow the same distribution with regard to region as the other landings for that year-quarter and gear combination. For each year-quarter/gear combination, we calculate the proportion of landings that occurs in each of the seven regions. The total landings of unknown regions are multiplied by those proportions and then added to the landings from known regions. In cases where none of the areas are known for that year-quarter/gear combination, proportions from the quarter immediately previous to that quarter are used. Sum the landings by gear type to get the total landings for each year-quarter and region.
6. Reformat the data according to the specifications needed for input to the operating model (columns specifying the year, quarter, fleet, area and total landings).
7. Sum landings by year from the original data file and the final outputs to ensure that data were processed correctly and no landings are missing.

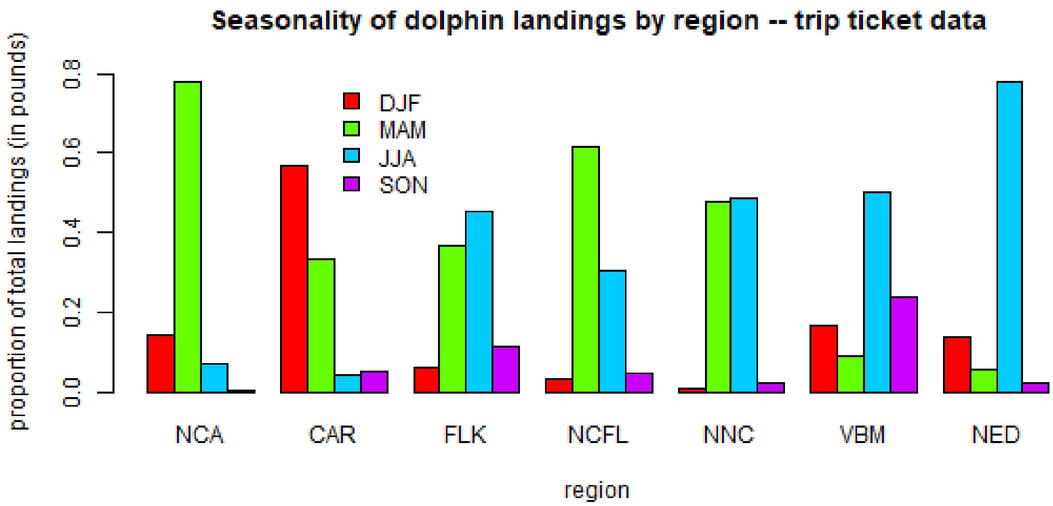


Figure 10.2: Seasonality of the commercial dolphin landings according to the trip ticket data. Figure shows the proportion of dolphin landings occurring in each season, for each region of the operating model.

11 Analysis of international catches in the high seas

Here we will explore commercial and recreational catch data as estimated by the Sea Around Us Project. This data set includes reported data for fisheries worldwide as well as reconstructed estimates for unreported catches. For each country, project scientists assemble available data on catch and use additional sociological and fishing data plus expert information and opinion, to produce their best estimates of total catch from all fishing sectors and for all species globally.

11.1 Data access and upload

The high seas data are accessed in the Basic Search High seas section of the site. The individual regions can be clicked on (e.g., Atlantic, Western Central) and then the data are downloaded using the Download Data button.

The current version of data in use is version 50.1, accessed June 2024 (the data files can be found on github in the SEFSC-dolphin-analyses/data /SAU/ folder). This was still the most recent version of data available as of September 2025.

11.2 Read in the high seas catches

First we read in the high seas catches for the Western Central Atlantic and Northwest Atlantic zones. These are equivalent to FAO areas 21 and 31.

```
# clear workspace
rm(list = ls())

if(!require("pals")) install.packages("pals")
library(pals)
```

```
# find data files
hslis <- dir("data/SAU/")[grep("HighSeas", dir("data/SAU/"))]

# identify areas 21 and 31
hslis[1]; hslis[3]
```

```
[1] "SAU HighSeas 21 v50-1.csv"
```

```
[1] "SAU HighSeas 31 v50-1.csv"
```

```
#download data
h1 <- read.csv(paste0("data/SAU/", hslis[1]))
h3 <- read.csv(paste0("data/SAU/", hslis[3]))

# confirm correct areas
#unique(h1$area_name)
#unique(h3$area_name)

hs <- rbind(h1, h3)
```

We extract the data for dolphinfish and look at the characteristics of the catches. Most of the catches are in the Western Central Atlantic, and the primary fishing entities are Venezuela and Saint Lucia. The fishing sector is all industrial, and the vast majority of the catch is reported landings, with only a tiny fraction of discards or unreported landings listed in the database. The gear type is primarily long line.

```
# check species name and filter out dolphinfish
unique(hs$common_name)[grep("dolphin", unique(hs$common_name))]
```

```
[1] "Common dolphinfish"
```

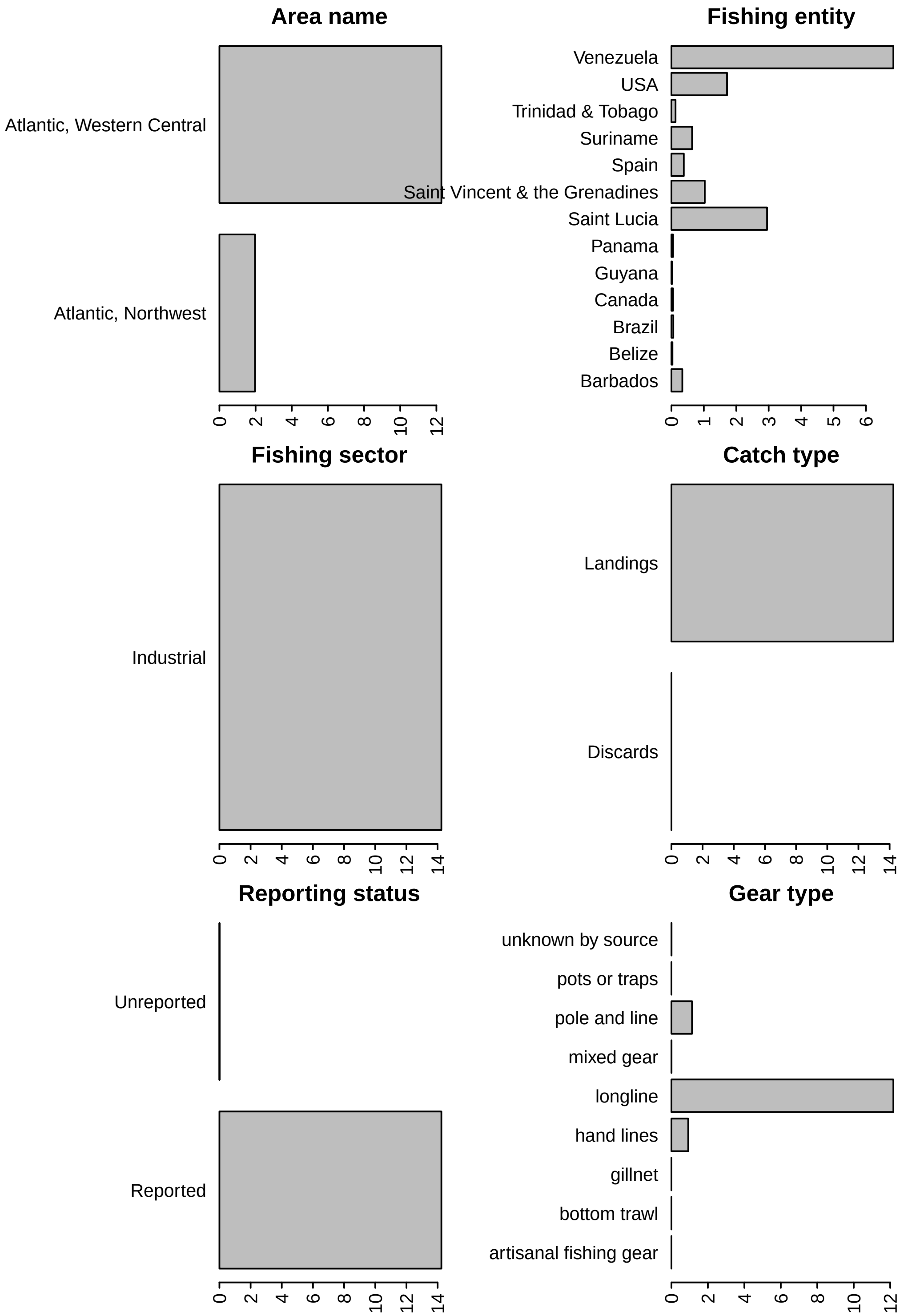
```
hs <- hs[which(hs$common_name == "Common dolphinfish"), ]

# look at the categories within each column of data
apply(hs[1:13], 2, table)
```

```
$area_name

      Atlantic, Northwest Atlantic, Western Central
                     113                        160

$area_type

high_seas
      273

$year

1985 1986 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012
   1    1    1    1    1    1    1    6    4    1    5   10    5    8   10   12
2013 2014 2015 2016 2017 2018 2019
  24   22   30   33   33   29   34

$scientific_name

Coryphaena hippurus
                273

$common_name

Common dolphinfish
               273

$functional_group

Large pelagics (>=90 cm)
                     273

$commercial_group

Perch-likes
        273

$fishing_entity

                      Barbados                          Belize
                             6                               3
```

```
                 Brazil                          Canada
                     19                              18
                 Guyana                          Panama
                      1                               1
            Saint Lucia Saint Vincent & the Grenadines
                      5                              28
                  Spain                        Suriname
                     80                               1
      Trinidad & Tobago                             USA
                     14                              74
              Venezuela
                     23

$fishing_sector

Industrial
       273

$catch_type

Discards Landings
       5      268

$reporting_status

  Reported Unreported
       253         20

$gear_type

artisanal fishing gear           bottom trawl                gillnet
                     7                      9                     11
            hand lines               longline             mixed gear
                    17                    187                      1
         pole and line         pots or traps      unknown by source
                    19                      9                     13

$end_use_type

                         Direct human consumption    Fishmeal and fish oil
                       5                      194                       37
```

```
Other
   37
```

```
# convert from metric tonnes to pounds
hs$pounds <- hs$tonnes * 1000 * 2.20462

# produce barplot of key data fields
par(mar = c(3, 20, 3, 1), mfrow = c(3, 2), mex = 0.5)
barplot(tapply(hs$pounds/10^6, hs$area_name, sum, na.rm = T), las = 2, xlab = "millions of
#barplot(tapply(hs$pounds/10^6, hs$area_type, sum, na.rm = T), las = 2, ylab = "millions of
#barplot(tapply(hs$pounds/10^6, hs$year, sum, na.rm = T), las = 2, ylab = "millions of poun
barplot(tapply(hs$pounds/10^6, hs$fishing_entity, sum, na.rm = T), las = 2, xlab = "million
barplot(tapply(hs$pounds/10^6, hs$fishing_sector, sum, na.rm = T), las = 2, xlab = "million
barplot(tapply(hs$pounds/10^6, hs$catch_type, sum, na.rm = T), las = 2, xlab = "millions of
barplot(tapply(hs$pounds/10^6, hs$reporting_status, sum, na.rm = T), las = 2, xlab = "milli
barplot(tapply(hs$pounds/10^6, hs$gear_type, sum, na.rm = T), las = 2, xlab = "millions of
```


Area name
Atlantic, Western Central
Atlantic, Northwest
0 2 4 6 8 10 12
Fishing entity
Venezuela
USA
Trinidad & Tobago
Suriname
Spain
Saint Vincent & the Grenadines
Saint Lucia
Panama
Guyana
Canada
Brazil
Belize
Barbados
0 1 2 3 4 5 6
Fishing sector
Industrial
0 2 4 6 8 10 12 14
Catch type
Landings
Discards
0 2 4 6 8 10 12 14
Reporting status
Unreported
Reported
0 2 4 6 8 10 12 14
Gear type
unknown by source
pots or traps
pole and line
mixed gear
longline
hand lines
gillnet
bottom trawl
artisanal fishing gear
0 2 4 6 8 10 12

```
round(tapply(hs$pounds, hs$catch_type, sum, na.rm = T) / sum(hs$pounds) * 100, 2)   # << 1%
```

```
Discards Landings
    0.03    99.97
```

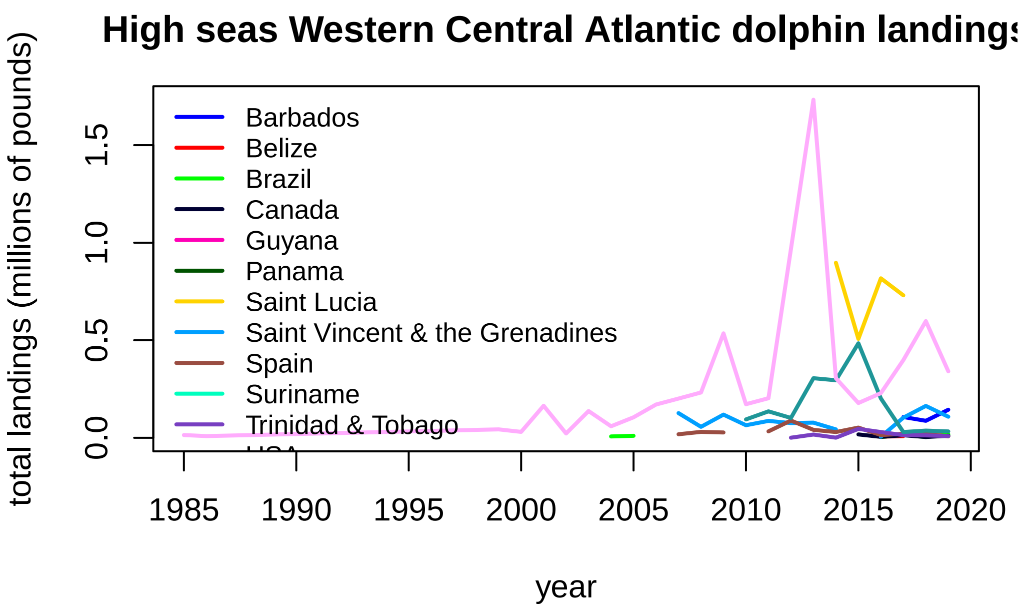
```
round(tapply(hs$pounds, hs$reporting_status, sum, na.rm = T) / sum(hs$pounds) * 100, 2)  #
```

```
  Reported Unreported
     99.97       0.03
```

11.3 Summarize the high seas landings

We plot the annual landings by fishing entity to look at trends over time.

```
hs$fishing_entity <- as.factor(hs$fishing_entity)
hs$fishing_entity <- droplevels(hs$fishing_entity)

par(mar = c(5, 5, 3, 1), mfrow = c(1, 1))
tab <- tapply(hs$pounds, list(hs$year, hs$fishing_entity), sum, na.rm = T)
matplot(as.numeric(rownames(tab)), tab/10^6, type = "l", col = glasbey(ncol(tab)), lty = 1,
        main = "High seas Western Central Atlantic dolphin landings")
legend("topleft", colnames(tab), col = glasbey(ncol(tab)), lty = 1, lwd = 2, cex = 0.8, bty
```



Finally we output the subset of high seas dolphin landings for the two areas. Because we have already accounted for the USA landings in our national reporting and are aiming to only summarize international landings here, we filter out the USA landings from the Sea Around Us database.

```
hs <- hs[which(hs$fishing_entity != "USA"), ]

hs$year <- factor(hs$year, levels = 1950:2019)
tab <- tapply(hs$pounds, list(hs$year, hs$area_name), sum, na.rm = T)
tab[is.na(tab)] <- 0
tail(tab, 40)
```

```
     Atlantic, Northwest Atlantic, Western Central
1980                0.000                    0.000
1981                0.000                    0.000
1982                0.000                    0.000
1983                0.000                    0.000
1984                0.000                    0.000
1985                0.000                14037.129
1986                0.000                 8748.399
1987                0.000                    0.000
```

```
1988                0.000                    0.000
1989                0.000                    0.000
1990                0.000                    0.000
1991                0.000                    0.000
1992                0.000                    0.000
1993                0.000                    0.000
1994                0.000                    0.000
1995                0.000                    0.000
1996                0.000                    0.000
1997                0.000                    0.000
1998                0.000                    0.000
1999                0.000                43168.296
2000                0.000                30284.627
2001                0.000               164272.223
2002                0.000                22147.477
2003                0.000               137103.181
2004                0.000                66058.258
2005                0.000               115171.863
2006                0.000               170522.101
2007            14687.527               330838.620
2008            28552.636               340863.719
2009            18719.121               662720.661
2010             4510.085               232093.837
2011            21589.424               300498.224
2012            72398.812              1059210.392
2013            34741.604              2471791.396
2014            27922.953              1247462.983
2015            90313.638               710059.125
2016            15319.433              1094893.891
2017            34806.933              1374259.739
2018            29328.358               885814.754
2019            21870.621               632747.157
```

```r
# write summary to merge with EEZ data
write.csv(tab, file = "data/SAU/high_seas_by_year.csv")
```

11.4 Distribute the high seas landings across quarters of the year

The international landings are only reported as annual landings and do not have any month or season associated with them. However, the MSE operating model requires seasonal landings. The majority of the international landings are caught by longline, so we will assume that the fleet distribution of the U.S. pelagic longline is representative of the seasonality of the international high seas fleets. Now we parse the annual landings according to the seasonality of the U.S. fleet.

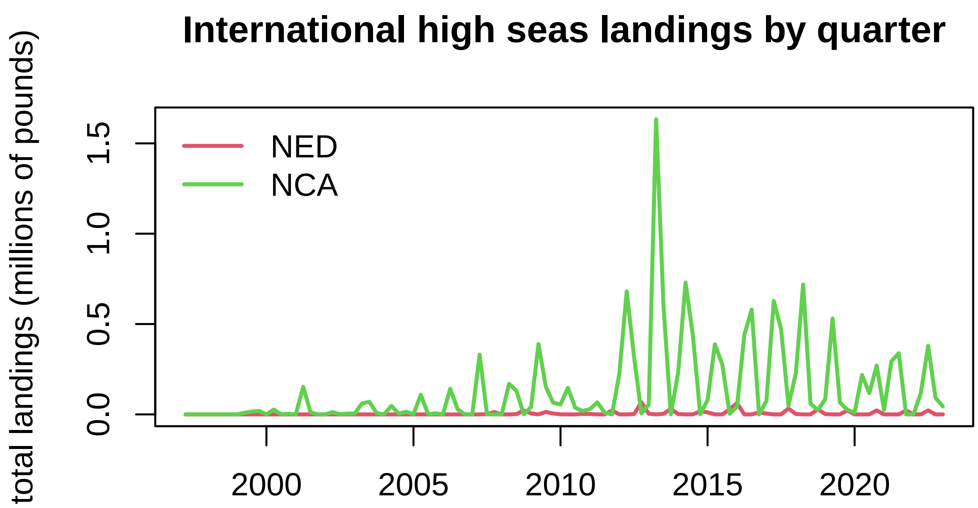
At the time of analysis, the SAU database only contained landings to 2019, so we interpolated the 2019 values to cover years 2020 - 2022.

```
# read in the percentage of the landings by area and year-quarter combinations from the U.S
load("data/outputs/per_PLLcatch_by_area_yearquarter.RData")

tab <- tab[which(rownames(tab) >= 1997), ]  # PLL data are only available from 1997 on
#tab[nrow(tab), ]
tab <- rbind(tab, tab[nrow(tab), ], tab[nrow(tab), ], tab[nrow(tab), ])  # interpolate 2019
tail(tab)
```

```
     Atlantic, Northwest Atlantic, Western Central
2017            34806.93                 1374259.7
2018            29328.36                  885814.8
2019            21870.62                  632747.2
                21870.62                  632747.2
                21870.62                  632747.2
                21870.62                  632747.2
```

```
NED <- percatch[, , 1]   # create matrices for final landings data
NCA <- percatch[, , 1]

for (i in 1:ncol(percatch))  {
  NED[, i] <- percatch[, i, 7] * tab[i, 1]   # these are the percentages for the NED (Atlan
  NCA[, i] <- percatch[, i, 1] * tab[i, 2]   # these are the percentages for the NCA (Weste
  }

# plot the parsed out data by year-quarter
yrs <- sort(rep(as.numeric(colnames(NCA)), 4))
qrt <- rep(1:4, length(yrs)/4)
```

```
matplot(yrs + qrt/4, cbind(matrix(NED), matrix(NCA))/10^6, lty = 1, lwd = 2,
        col = 2:3, type = "l", main = "International high seas landings by quarter",
        xlab = "", ylab = "total landings (millions of pounds)")
legend("topleft", c("NED", "NCA"), lwd = 2, col = 2:3, bty = "n")
```




```
# save NED data in this format because it needs to be combined with Canadian EEZ landings
save(NED, file = "data/outputs/NED_year_quarter.RData")
```

Finally, we format the data in the format for input into the operating model. We only include NCA here, as the NED high seas landings must be combined with international landings in territorial waters of Canada and France in the NED region.

```
# format for operating model
flt <- rep("Intl", length(yrs))
are <- rep("NCA", length(yrs))
findat <- data.frame(cbind(yrs, qrt, flt, are, matrix(NCA)))
names(findat) <- c("Year", "Quarter", "Fleet", "Area", "Catch_lbs")
findat$Catch_lbs <- as.numeric(findat$Catch_lbs)
head(findat)
```

```
  Year Quarter Fleet Area Catch_lbs
1 1997       1  Intl  NCA         0
2 1997       2  Intl  NCA         0
3 1997       3  Intl  NCA         0
4 1997       4  Intl  NCA         0
5 1998       1  Intl  NCA         0
6 1998       2  Intl  NCA         0
```

```
#plot(findat$Catch_lbs, type = "l")
#plot(findat$Catch_lbs ~ factor(findat$Area))
#plot(findat$Catch_lbs ~ factor(findat$Quarter))
#plot(findat$Catch_lbs ~ factor(findat$Year))
#apply(findat, 2, table)
#barplot(tapply(findat$Catch_lbs, list(findat$Quarter, findat$Area), sum, na.rm = T), besid

write.csv(findat, file = "data/FINAL_files/NCA_Intl_TomFormat.csv", row.names = FALSE)
```

12 Analysis of international catches in territorial waters

12.1 Data access and upload

For catches within each country's exclusive economic zones, data are downloaded from the Sea Around Us site (Pauly, Zeller, and Palomares, 2020.). In the Biodiversity by Taxon Tools and Data, we filter by EEZ for All regions, and then filter by Taxon at the Species level for Common dolphinfish (Coryphaena hippurus). After clicking "View graphs" one can access the full data set using the "Download Data" button.

12.2 Read in the catches within excluzive economic zones (EEZs)

Now we read in the catches reported by each country or jurisdiction's respective EEZ. Because we have extracted dolphin catches for EEZs worldwide, we need to separate out the catches for the Atlantic Ocean basin.

```
# load libraries
library(pals)

rm(list = ls())
d <- read.csv("data/SAU/SAU Taxa 600006 v50-1.csv", header = T, sep = ",")  # v. 50.1 - Jun
head(d)
```

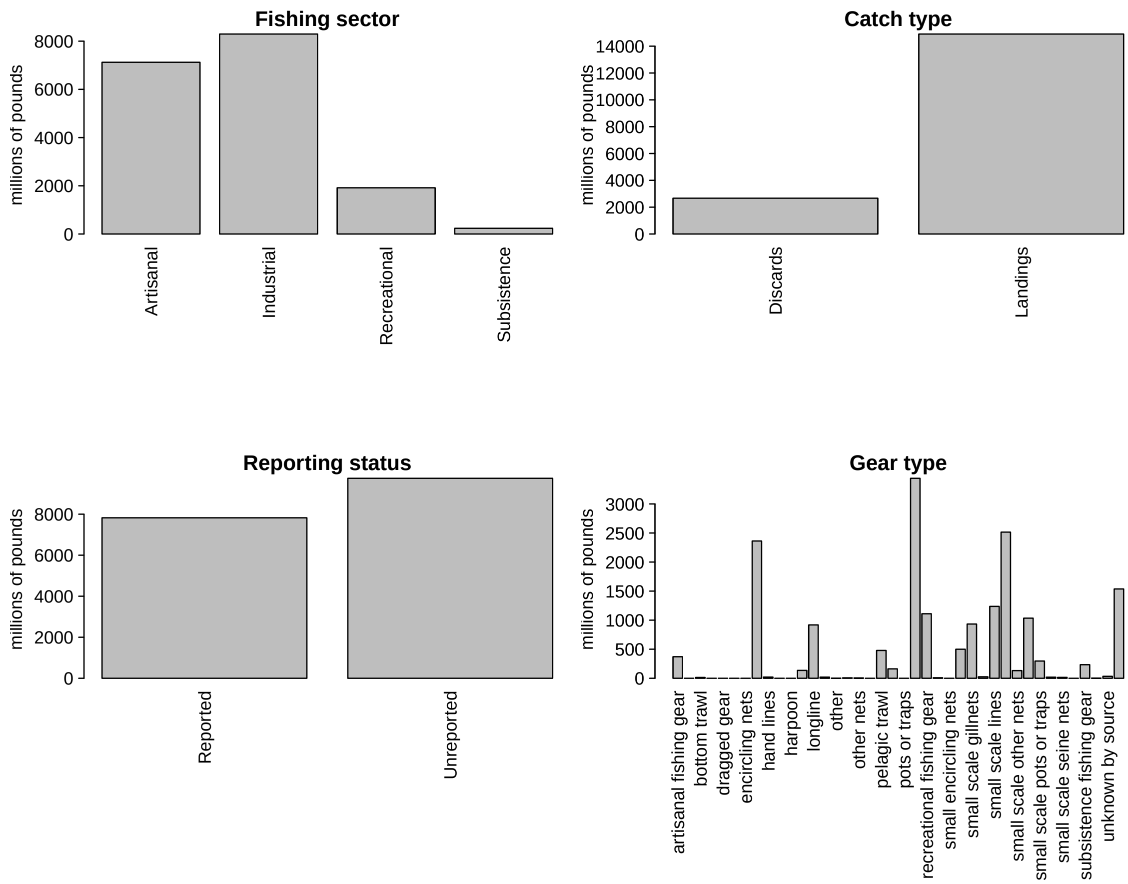
```
     area_name area_type year     scientific_name        common_name
1    Australia       eez 1950 Coryphaena hippurus Common dolphinfish
2    Australia       eez 1950 Coryphaena hippurus Common dolphinfish
3    Australia       eez 1950 Coryphaena hippurus Common dolphinfish
4      Bahamas       eez 1950 Coryphaena hippurus Common dolphinfish
5     Barbados       eez 1950 Coryphaena hippurus Common dolphinfish
6 Bermuda (UK)       eez 1950 Coryphaena hippurus Common dolphinfish
```

```
           functional_group commercial_group fishing_entity fishing_sector
1 Large pelagics (>=90 cm)       Perch-likes      Australia     Industrial
2 Large pelagics (>=90 cm)       Perch-likes      Australia     Industrial
3 Large pelagics (>=90 cm)       Perch-likes      Australia     Industrial
4 Large pelagics (>=90 cm)       Perch-likes        Bahamas   Recreational
5 Large pelagics (>=90 cm)       Perch-likes       Barbados   Recreational
6 Large pelagics (>=90 cm)       Perch-likes   Bermuda (UK)      Artisanal
  catch_type reporting_status                 gear_type
1   Landings         Reported                  longline
2   Landings         Reported                  longline
3   Landings         Reported                  longline
4   Landings       Unreported recreational fishing gear
5   Landings       Unreported recreational fishing gear
6   Landings         Reported         small scale lines
            end_use_type       tonnes landed_value
1    Fishmeal and fish oil  0.003373202 3.438170e-05
2 Direct human consumption  6.739657104 1.040254e+04
3                    Other  0.003373202 3.438170e-05
4 Direct human consumption 48.665105143 8.262728e+00
5 Direct human consumption  0.009158559 4.846886e-04
6 Direct human consumption  1.511085514 3.199403e+03
```

```r
#dim(d)
#apply(d[1:13], 2, table)

# convert tonnes to pounds
d$lbs <- d$tonnes * 1000 * 2.20462
```

We look at the characteristics of the catches within EEZs. These are largely industrial and artisanal in nature. Landings are the majority of the catch but discards are also present. There are more unreported landings than reported landings and a variety of gear types are used.

```r
# look at characteristics of data
par(mfrow = c(2, 2), mar = c(20, 7, 3, 1), mex = 0.5, mgp = c(5.5, 1, 0))
barplot(tapply(d$lbs, d$fishing_sector, sum, na.rm = T)/10^6, las = 2, ylab = "millions of
barplot(tapply(d$lbs, d$catch_type, sum, na.rm = T)/10^6, las = 2, ylab = "millions of poun
barplot(tapply(d$lbs, d$reporting_status, sum, na.rm = T)/10^6, las = 2, ylab = "millions o
barplot(tapply(d$lbs, d$gear_type, sum, na.rm = T)/10^6, las = 2, ylab = "millions of pound
```




```
round(tapply(d$lbs, d$catch_type, sum, na.rm = T) / sum(d$lbs) * 100, 2)    # 15% are discar
```

```
Discards Landings
   15.18    84.82
```

```
round(tapply(d$lbs, d$reporting_status, sum, na.rm = T) / sum(d$lbs) * 100, 2)   # 55% are u
```

```
  Reported Unreported
     44.52      55.48
```

12.3 Separating the EEZs by area

Now we have to manually assign the different EEZs to respective areas. We separate the geographical areas within the Atlantic Ocean (North and South) and lump all other EEZs into “other oceans.” This latter category is largely the Pacific Ocean and to a lesser extent the Indian Ocean. We remove the high seas catch from this database as it is not ocean-specific and we have extracted the ocean-specific high seas catch previously.

```
# remove the high seas catch from this database
d <- d[which(d$area_type != "high_seas"), ]
d$fishing_entity <- as.factor(d$fishing_entity)
d$fishing_entity <- droplevels(d$fishing_entity)
table(d$area_type)  # this should be all "eez"
```

```

  eez
63056
```

```
# Western Central Atlantic EEZs
WCA <- c("Bahamas", "Barbados", "Cayman Isl. (UK)", "Dominica", "Dominican Republic", "Gren
         "Aruba (Netherlands)", "Haiti", "Guadeloupe  (France)", "Jamaica", "Martinique (Fr
         "Montserrat (UK)", "Nicaragua (Caribbean)", "Puerto Rico (USA)", "Saint Kitts & Ne
         "Saint Lucia", "Saint Vincent & the Grenadines", "Trinidad & Tobago", "US Virgin I
         "St Barthelemy (France)", "St Martin (France)", "Curaçao (Netherlands)", "Bonaire 
         "Saba and Sint Eustatius (Netherlands)", "Sint Maarten (Netherlands)", "Honduras (
         "Colombia (Caribbean)", "Mexico (Atlantic)", "Turks & Caicos Isl. (UK)", "Guyana",
         "Antigua & Barbuda", "Belize", "British Virgin Isl. (UK)", "French Guiana", "Cuba"
         "Anguilla (UK)", "Suriname", "Costa Rica (Caribbean)", "Guatemala (Caribbean)",
         "Panama (Caribbean)", "Bermuda (UK)", "CuraÃ§ao (Netherlands)")

# Brazil
Bra <- c("Brazil (mainland)", "Fernando de Noronha (Brazil)",
         "St Paul and St. Peter Archipelago (Brazil)")

# Canada
Can <-  c("Canada (East Coast)", "Saint Pierre & Miquelon (France)")

# United States
USA <-  c("USA (East Coast)", "USA (Gulf of Mexico)")
```

```
# Northeast Atlantic EEZs
NEA <- c("Canary Isl. (Spain)", "Spain (mainland; Med and Gulf of Cadiz)", "Spain (Northwes
         "Spain (mainland, Med and Gulf of Cadiz)",
         "France (Atlantic Coast)", "Portugal (mainland)", "Azores Isl. (Portugal)")

# Eastern Central Atlantic EEZs
ECA <- c("Madeira Isl. (Portugal)", "Cape Verde", "Equatorial Guinea", "Benin",
          "Sao Tome & Principe ", "Sao Tome & Principe", "Morocco (Central)", "Ghana",
          "Côte d'Ivoire", "CÃ´te d'Ivoire", "Guinea", "Liberia", "Sierra Leone", "Nigeria"
          "Guinea-Bissau", "Senegal", "Cameroon", "Gambia", "Mauritania", "Morocco (South)"
          "Gabon", "Congo; R. of", "Congo, R. of", "Congo (ex-Zaire)")

# Southeastern Atlantic EEZs
SEA <- c("Angola", "Namibia", "South Africa (Atlantic and Cape)",
           "Ascension Isl. (UK)", "Saint Helena (UK)", "Tristan da Cunha Isl. (UK)")

# Southwest Atlantic EEZs
SWA <- c("Trindade & Martim Vaz Isl. (Brazil)", "Uruguay")

# Mediterranean EEZs
Med <- c("Italy (mainland)", "Libya", "Malta", "Syria", "Sicily (Italy)","Sardinia (Italy)"
         "Balearic Islands (Spain)", "Cyprus (North)", "Cyprus (South)",
         "Gaza Strip", "Greece (without Crete)", "Crete (Greece)", "Lebanon",
         "Turkey (Mediterranean Sea)", "Egypt (Mediterranean)", "Israel (Mediterranean)", "
         "Algeria", "Albania", "Croatia", "Montenegro", "Morocco (Mediterranean)",
         "France (Mediterranean)", "Corsica (France)")

# add variable to specify region
d$reg <- "other oceans"
d$reg[which(d$area_name %in% WCA)] <- "Western Central Atlantic EEZs"
d$reg[which(d$area_name %in% Bra)] <- "Brazil EEZ"
d$reg[which(d$area_name %in% Can)] <- "Canadian EEZ"
d$reg[which(d$area_name %in% USA)] <- "United States EEZ"
d$reg[which(d$area_name %in% NEA)] <- "Northeast Atlantic EEZs"
d$reg[which(d$area_name %in% ECA)] <- "Eastern Central Atlantic EEZs"
d$reg[which(d$area_name %in% SEA)] <- "Southeast Atlantic EEZs"
d$reg[which(d$area_name %in% SWA)] <- "Southwest Atlantic EEZs"
d$reg[which(d$area_name %in% Med)] <- "Mediterranean Sea EEZs"
```

Now we will check that we have classified the regions correctly.

```r
# check region designation
othlis <- unique(d$area_name[which(d$reg == "other oceans")])
print("EEZs of all other oceans: ")
```

```
[1] "EEZs of all other oceans: "
```

```r
othlis
```

```
 [1] "Australia"                          "China"
 [3] "Taiwan"                             "Mayotte (France)"
 [5] "French Polynesia"                   "Hong Kong (China)"
 [7] "Andaman & Nicobar Isl. (India)"     "Indonesia (Eastern)"
 [9] "Japan (main islands)"               "Japan (Daito Islands)"
[11] "Korea (South)"                      "Hawaii Northwest Islands (USA)"
[13] "Nauru"                              "Niue (New Zealand)"
[15] "Northern Marianas (USA)"            "Micronesia (Federated States of)"
[17] "Marshall Isl."                      "Palau"
[19] "Pakistan"                           "Papua New Guinea"
[21] "Philippines"                        "Seychelles"
[23] "Viet Nam"                           "Tokelau (New Zealand)"
[25] "Tonga"                              "Hawaii Main Islands (USA)"
[27] "USA (West Coast)"                   "Wake Isl. (USA)"
[29] "Wallis & Futuna Isl. (France)"      "Samoa"
[31] "Guatemala (Pacific)"                "Kiribati (Gilbert Islands)"
[33] "Mexico (Pacific)"                   "Japan (Ogasawara Islands)"
[35] "Korea (North, Yellow Sea)"          "Korea (North, Sea of Japan)"
[37] "Mauritius"                          "Solomon Isl."
[39] "Christmas Isl. (Australia)"         "Johnston Atoll (USA)"
[41] "Timor Leste"                        "Palmyra Atoll & Kingman Reef (USA)"
[43] "Jarvis Isl. (USA)"                  "Howland & Baker Isl. (USA)"
[45] "Indonesia (Central)"                "Indonesia (Indian Ocean)"
[47] "Kiribati (Line Islands)"            "Kiribati (Phoenix Islands)"
[49] "New Caledonia (France)"             "Vanuatu"
[51] "Lord Howe Isl. (Australia)"         "Myanmar"
[53] "Ecuador (mainland)"                 "El Salvador"
[55] "Fiji"                               "Malaysia (Peninsula West)"
[57] "Norfolk Isl. (Australia)"           "Peru"
[59] "Pitcairn (UK)"                      "Colombia (Pacific)"
[61] "Costa Rica (Pacific)"               "Nicaragua (Pacific)"
```

```
 [63] "Panama (Pacific)"                    "Thailand (Andaman Sea)"
 [65] "Thailand (Gulf of Thailand)"         "Cook Islands"
 [67] "Kermadec Isl. (New Zealand)"         "Tuvalu"
 [69] "Honduras (Pacific)"                  "American Samoa"
 [71] "Galapagos Isl. (Ecuador)"            "New Zealand"
 [73] "Brunei Darussalam"                   "Malaysia (Sabah)"
 [75] "Malaysia (Sarawak)"                  "Guam (USA)"
 [77] "Clipperton Isl. (France)"            "South Africa (Indian Ocean Coast)"
 [79] "Singapore"                           "Easter Isl. (Chile)"
 [81] "Madagascar"                          "Réunion (France)"
 [83] "Chagos Archipelago (UK)"             "Comoros Isl."
 [85] "Mozambique Channel Isl. (France)"    "Kenya"
 [87] "Mozambique"                          "Somalia"
 [89] "Tanzania"                            "Glorieuse Islands (France)"
 [91] "Yemen (Socotra)"                     "Oman"
 [93] "St Paul & Amsterdam Isl. (France)"  "Iran (Sea of Oman)"
 [95] "Desventuradas Isl. (Chile)"          "India (mainland)"
 [97] "Maldives"                            "Tromelin Isl. (France)"
 [99] "Cambodia"                            "Sri Lanka"
[101] "Malaysia (Peninsula East)"           "Bangladesh"
```

```
print("U.S. EEZs outside the Atlantic Ocean: ")
```

```
[1] "U.S. EEZs outside the Atlantic Ocean: "
```

```
othlis[grep("USA", othlis)]
```

```
 [1] "Hawaii Northwest Islands (USA)"     "Northern Marianas (USA)"
 [3] "Hawaii Main Islands (USA)"          "USA (West Coast)"
 [5] "Wake Isl. (USA)"                    "Johnston Atoll (USA)"
 [7] "Palmyra Atoll & Kingman Reef (USA)" "Jarvis Isl. (USA)"
 [9] "Howland & Baker Isl. (USA)"         "Guam (USA)"
```

```
print("French EEZs outside the Atlantic Ocean: ")
```

```
[1] "French EEZs outside the Atlantic Ocean: "
```

```r
othlis[grep("France", othlis)]
```

```
[1] "Mayotte (France)"                   "Wallis & Futuna Isl. (France)"
[3] "New Caledonia (France)"             "Clipperton Isl. (France)"
[5] "Réunion (France)"                   "Mozambique Channel Isl. (France)"
[7] "Glorieuse Islands (France)"         "St Paul & Amsterdam Isl. (France)"
[9] "Tromelin Isl. (France)"
```

```r
unique(d$area_name[which(d$reg == "Western Central Atlantic EEZs")])   # Western Central EE
```

```
 [1] "Bahamas"
 [2] "Barbados"
 [3] "Bermuda (UK)"
 [4] "Cayman Isl. (UK)"
 [5] "Dominica"
 [6] "Dominican Republic"
 [7] "Grenada"
 [8] "Guadeloupe  (France)"
 [9] "Haiti"
[10] "Jamaica"
[11] "Martinique (France)"
[12] "Montserrat (UK)"
[13] "Puerto Rico (USA)"
[14] "Saint Kitts & Nevis"
[15] "Saint Lucia"
[16] "Saint Vincent & the Grenadines"
[17] "Trinidad & Tobago"
[18] "US Virgin Isl."
[19] "St Barthelemy (France)"
[20] "St Martin (France)"
[21] "Curaçao (Netherlands)"
[22] "Bonaire (Netherlands)"
[23] "Saba and Sint Eustatius (Netherlands)"
[24] "Sint Maarten (Netherlands)"
[25] "Honduras (Caribbean)"
[26] "Colombia (Caribbean)"
[27] "Mexico (Atlantic)"
[28] "Turks & Caicos Isl. (UK)"
[29] "Costa Rica (Caribbean)"
```

```
[30] "Panama (Caribbean)"
[31] "Antigua & Barbuda"
[32] "Belize"
[33] "British Virgin Isl. (UK)"
[34] "Cuba"
[35] "French Guiana"
[36] "Guyana"
[37] "Anguilla (UK)"
[38] "Suriname"
[39] "Venezuela"
[40] "Guatemala (Caribbean)"
[41] "Aruba (Netherlands)"
[42] "Nicaragua (Caribbean)"
```

```
unique(d$area_name[which(d$reg == "Brazil EEZ")])   # Brazil EEZs:
```

```
[1] "Brazil (mainland)"
[2] "Fernando de Noronha (Brazil)"
[3] "St Paul and St. Peter Archipelago (Brazil)"
```

```
unique(d$area_name[which(d$reg == "Canadian EEZ")])   # Canadian EEZs:
```

```
[1] "Canada (East Coast)"              "Saint Pierre & Miquelon (France)"
```

```
unique(d$area_name[which(d$reg == "United States EEZ")])  # United States EEZs:
```

```
[1] "USA (East Coast)"     "USA (Gulf of Mexico)"
```

```
unique(d$area_name[which(d$reg == "Northeast Atlantic EEZs")])   # Northeast Atlantic EEZs:
```

```
[1] "Azores Isl. (Portugal)"
[2] "Spain (mainland, Med and Gulf of Cadiz)"
[3] "Canary Isl. (Spain)"
[4] "Portugal (mainland)"
[5] "France (Atlantic Coast)"
[6] "Spain (Northwest)"
```

```
unique(d$area_name[which(d$reg == "Eastern Central Atlantic EEZs")])   # Eastern Central At
```

```
 [1] "Cape Verde"                "Equatorial Guinea"
 [3] "Sao Tome & Principe"       "Madeira Isl. (Portugal)"
 [5] "Togo"                      "Ghana"
 [7] "Guinea"                    "Liberia"
 [9] "Guinea-Bissau"             "Senegal"
[11] "Sierra Leone"              "Morocco (Central)"
[13] "Morocco (South)"           "Côte d'Ivoire"
[15] "Gabon"                     "Benin"
[17] "Gambia"                    "Mauritania"
[19] "Nigeria"                   "Cameroon"
[21] "Congo, R. of"              "Congo (ex-Zaire)"
```

```
unique(d$area_name[which(d$reg == "Southeast Atlantic EEZs")])   # Southeast Atlantic EEZs:
```

```
[1] "Ascension Isl. (UK)"              "South Africa (Atlantic and Cape)"
[3] "Saint Helena (UK)"                "Angola"
[5] "Namibia"                          "Tristan da Cunha Isl. (UK)"
```

```
unique(d$area_name[which(d$reg == "Southwest Atlantic EEZs")])   # Southwest Atlantic EEZs:
```

```
[1] "Trindade & Martim Vaz Isl. (Brazil)" "Uruguay"
```

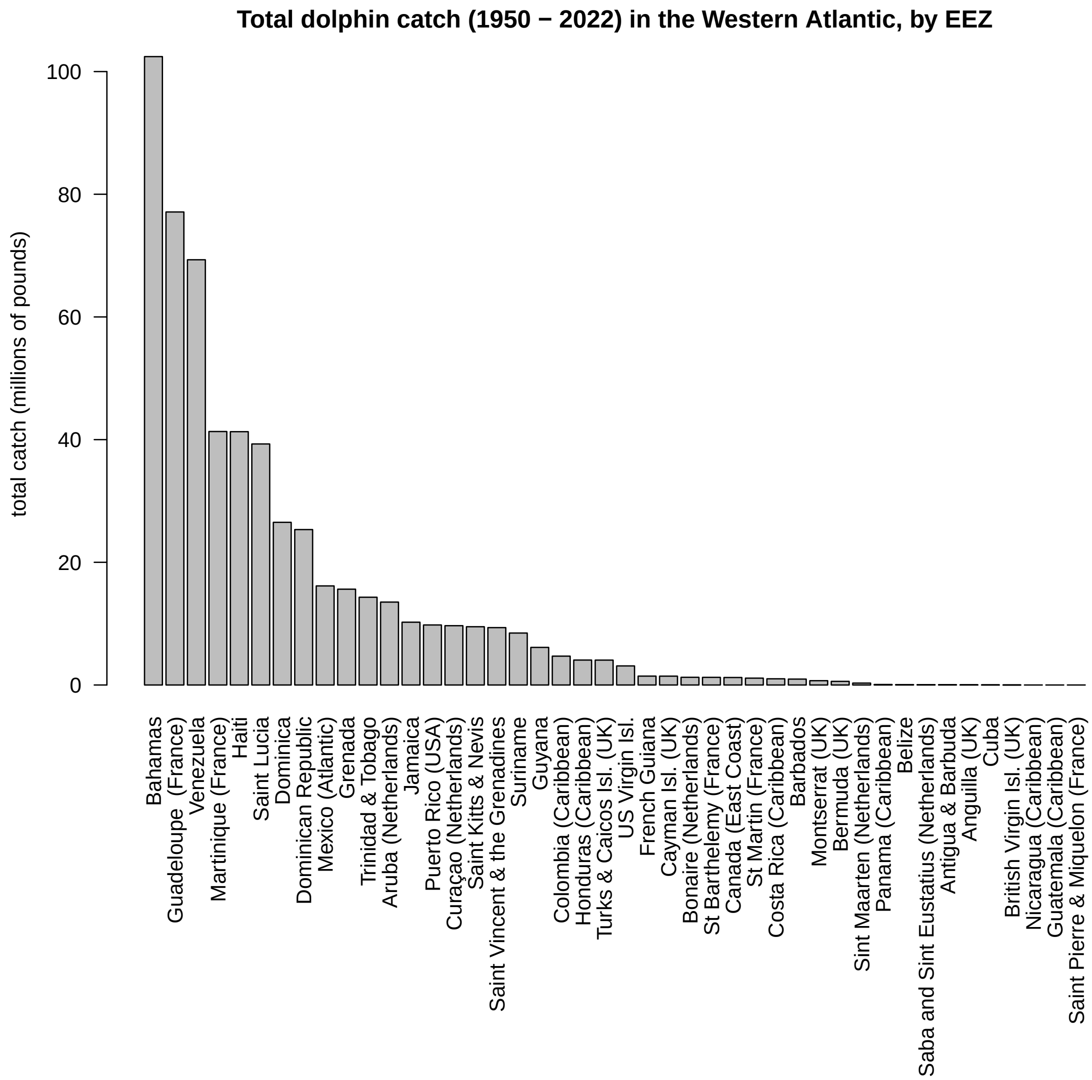
```
unique(d$area_name[which(d$reg == "Mediterranean Sea EEZs")])   # Mediterranean Sea EEZs:
```

```
 [1] "Italy (mainland)"         "Libya"
 [3] "Malta"                    "Syria"
 [5] "Sicily (Italy)"           "Sardinia (Italy)"
 [7] "Balearic Islands (Spain)" "Tunisia"
 [9] "Algeria"                  "Morocco (Mediterranean)"
[11] "France (Mediterranean)"   "Corsica (France)"
[13] "Croatia"                  "Lebanon"
[15] "Albania"                  "Greece (without Crete)"
[17] "Crete (Greece)"           "Cyprus (South)"
```

12.4 Summarize catch by regional EEZs

Now we summarize the catch by region by grouping the catch across the EEZs by region. We save these data for later analysis, and move on to analyzing only the Western Central and Northwest Atlantic jurisdictions.

```
# summarize EEZ catch by region
tab <- tapply(d$lbs, list(d$year, d$reg), sum, na.rm = T)
write.csv(data.frame(tab), file = "data/outputs/intl_catches_allAtlantic.csv", row.names =

dwest <- d[which(d$reg == "Canadian EEZ" | d$reg == "Western Central Atlantic EEZs"), ]
save(dwest, file = "data/outputs/SAU_EEZs_WCA.RData")

tab <- tapply(dwest$lbs, list(dwest$year, dwest$area_name), sum, na.rm = T)

# eight countries are responsible for vast majority of catch with total catches across all
par(mar = c(17, 4, 3, 1))
barplot(sort(colSums(tab, na.rm = T)/10^6, decreasing = T), horiz = F, las = 2,
        main = "Total dolphin catch (1950 - 2022) in the Western Atlantic, by EEZ",
        ylab = "total catch (millions of pounds)")
```



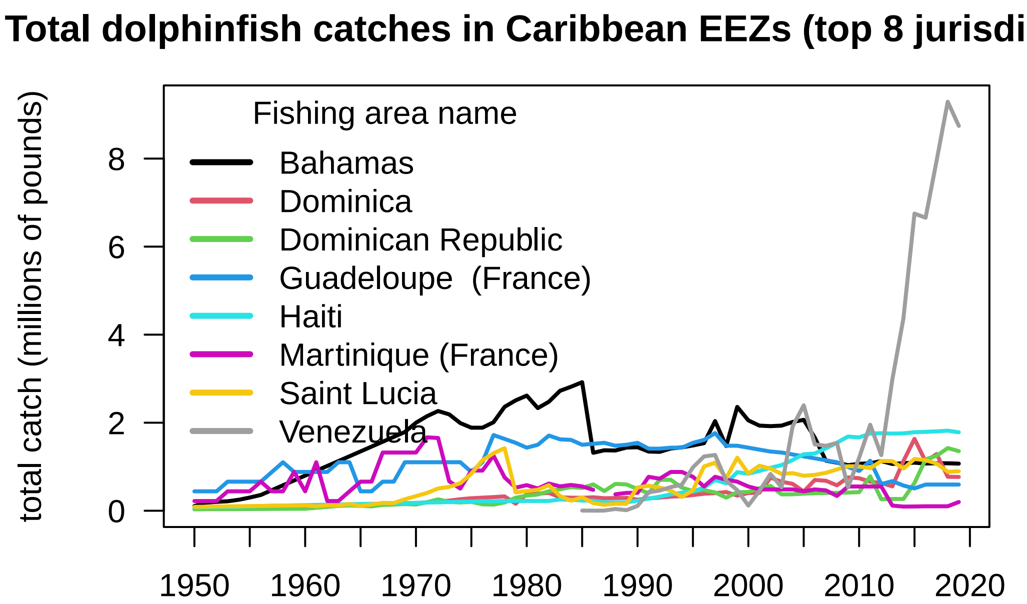
Note that Bermuda is actually in the NCA region and not the Greater Caribbean region. However, its catches constitute 0.1% of the total catch which is essentially negligible, and to avoid complicating the operating model and code unnecessarily we will leave those catches accounted for in the CAR region.

```
round(colSums(tab, na.rm = T)[7] / sum(colSums(tab, na.rm = T), na.rm = T) * 100, 2)
```

```
Bermuda (UK)
```

```
      0.1
```

```
lis <- names(sort(colSums(tab, na.rm = T), decreasing = T)[1:8]) # list of top 8 jurisdicti
tab1 <- tab[, colnames(tab) %in% lis]

par(mar = c(3, 5, 3, 1))
matplot(rownames(tab1), tab1/10^6, las = 2, type = "l", lty = 1, lwd = 2, pch = 19, col = 1
        xlab = "", ylab = "total catch (millions of pounds)", axes = F,
        main = "Total dolphinfish catches in Caribbean EEZs (top 8 jurisdictions)")
axis(1, at = seq(1950, 2020, 5)); axis(2, las = 2); box()
legend("topleft", colnames(tab1), col = 1:8, lty = 1, lwd = 3, bty = "n", title = "Fishing
```

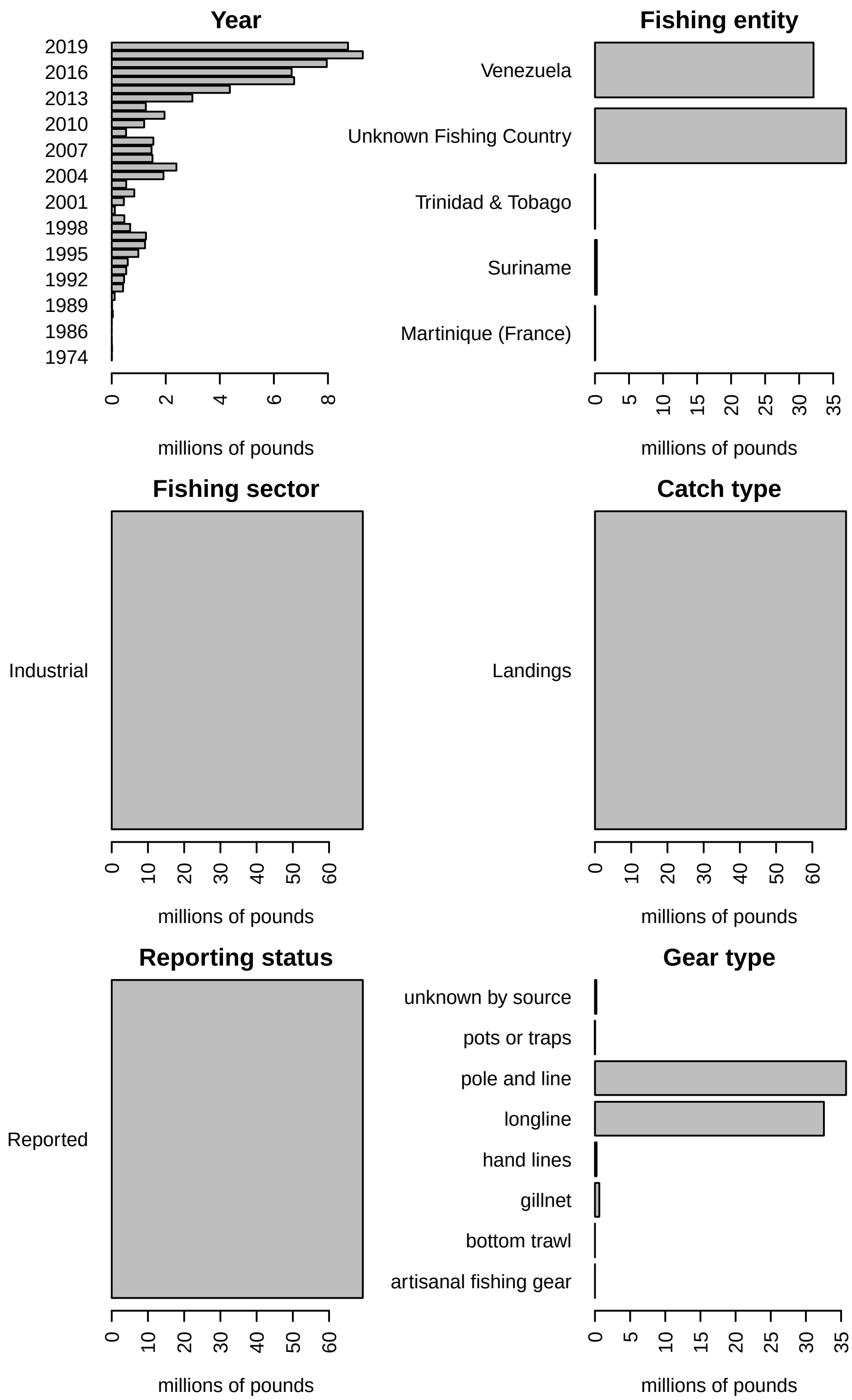


Note the rapid eight-fold increase in catches listed within the Venezuelan EEZ which occurs from 2013 - 2019. This increase is responsible for a rapid increase in estimated international catches within the Western Atlantic.

12.5 Look at Venezuela catches in detail

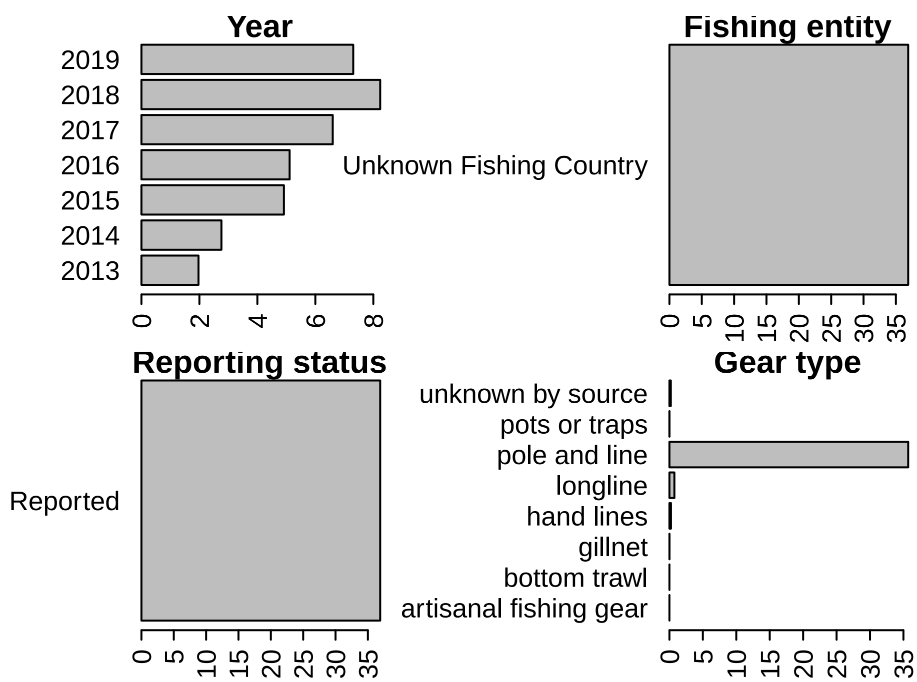
The figures above indicate that the database shows a rapid increase in international catches within the Western Atlantic, driven by a sudden increase in catches reported within Sea Around Us database within the Venezuelan EEZ. We investigate these catches in detail, as it seems odd that a relatively small fleet from a single country would be able to ramp up effort so rapidly. Data can be downloaded on a country-specific basis, the catches by taxon in the waters of Venezuela are accessed here.

```
rm(list = ls())  # clear the workspace

# download the data for waters of Venezuela
ven <- read.csv("data/SAU/SAU EEZ 862 v50-1.csv")

# subset dolphinfish from the data
v <- ven[which(ven$common_name == "Common dolphinfish"), ]

#apply(v[1:13], 2, table)

#convert to pounds
v$pounds <- v$tonnes * 1000 * 2.20462

# look at the characteristics of the data
par(mar = c(5, 10, 1, 1), mfrow = c(3, 2), mex = 0.9)
#barplot(tapply(v$pounds/10^6, v$area_name, sum, na.rm = T), las = 2, xlab = "millions of p
#barplot(tapply(v$pounds/10^6, v$area_type, sum, na.rm = T), las = 2, xlab = "millions of p
barplot(tapply(v$pounds/10^6, v$year, sum, na.rm = T), las = 2, xlab = "millions of pounds"
barplot(tapply(v$pounds/10^6, v$fishing_entity, sum, na.rm = T), las = 2, xlab = "millions
barplot(tapply(v$pounds/10^6, v$fishing_sector, sum, na.rm = T), las = 2, xlab = "millions
barplot(tapply(v$pounds/10^6, v$catch_type, sum, na.rm = T), las = 2, xlab = "millions of p
barplot(tapply(v$pounds/10^6, v$reporting_status, sum, na.rm = T), las = 2, xlab = "million
barplot(tapply(v$pounds/10^6, v$gear_type, sum, na.rm = T), las = 2, xlab = "millions of po
```


Year
2019
2016
2013
2010
2007
2004
2001
1998
1995
1992
1989
1986
1974
0 2 4 6 8
millions of pounds
Fishing entity
Venezuela
Unknown Fishing Country
Trinidad & Tobago
Suriname
Martinique (France)
0 5 10 15 20 25 30 35
millions of pounds
Fishing sector
Industrial
0 10 20 30 40 50 60
millions of pounds
Catch type
Landings
0 10 20 30 40 50 60
millions of pounds
Reporting status
Reported
0 10 20 30 40 50 60
millions of pounds
Gear type
unknown by source
pots or traps
pole and line
longline
hand lines
gillnet
bottom trawl
artisanal fishing gear
0 5 10 15 20 25 30 35
millions of pounds

Note that a large portion of the catch in the database is listed as coming from an "Unknown Fishing Country." We extract these catches and further investigate the characteristics.

```
# extract the Unknown Fishing Country catch
v1 <- v[which(v$fishing_entity == "Unknown Fishing Country"), ]
#apply(v1[1:15], 2, table)

# look at characteristics
par(mar = c(3, 10, 1, 5), mfrow = c(2, 2), mex = 0.6)
barplot(tapply(v1$pounds/10^6, v1$year, sum, na.rm = T), las = 2, xlab = "millions of pound
barplot(tapply(v1$pounds/10^6, v1$fishing_entity, sum, na.rm = T), las = 2, xlab = "million
barplot(tapply(v1$pounds/10^6, v1$reporting_status, sum, na.rm = T), las = 2, xlab = "milli
barplot(tapply(v1$pounds/10^6, v1$gear_type, sum, na.rm = T), las = 2, xlab = "millions of
```

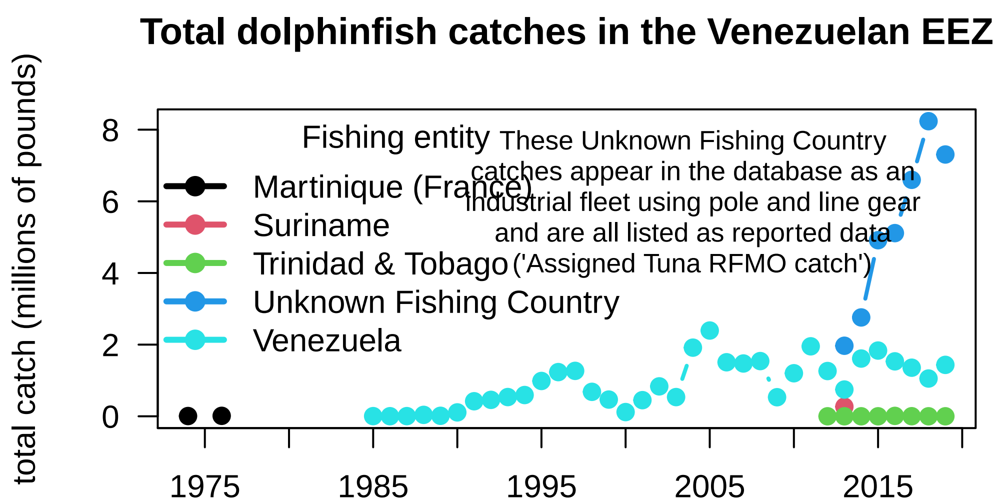


Here is another look at the catches within the venezuelan EEZ, separated by fishing entity.

```
# summarize Venezuela EEZ catches by fishing entity
tab <- tapply(v$pounds, list(v$year, v$fishing_entity), sum, na.rm = T)
head(tab)
```

```
     Martinique (France) Suriname Trinidad & Tobago Unknown Fishing Country
1974            7320.429       NA                NA                      NA
1976           12814.641       NA                NA                      NA
1985                  NA       NA                NA                      NA
1986                  NA       NA                NA                      NA
1987                  NA       NA                NA                      NA
1988                  NA       NA                NA                      NA
     Venezuela
1974        NA
1976        NA
1985  4699.563
1986  2229.484
1987  4585.610
1988 39960.149
```

```
matplot(rownames(tab), tab/10^6, las = 2, type = "b", lty = 1, lwd = 2, pch = 19, col = 1:5
        xlab = "", ylab = "total catch (millions of pounds)", axes = F,
        main = "Total dolphinfish catches in the Venezuelan EEZ")
axis(1, at = seq(1950, 2020, 5)); axis(2, las = 2); box()
legend("topleft", colnames(tab), col = 1:5, pch = 19, lty = 1, lwd = 3, bty = "n", title =

text(2004, 6, cex = 0.8,
     "These Unknown Fishing Country\ncatches appear in the database as an\nindustrial fleet
```



The very substantial increase in international Western Atlantic dolphin catches can be attributed solely to this Unknown Fishing Country catches which appear in the database beginning in 2013. The documentation of the catch reconstruction can be found in the Sea Around Us documentation files (see Mendoza et al. 2015). We spoke to Jeremy Mendoza and other experts who were involved in the reconstruction of domestic catches in Venezuela, and they noted that the "Assigned tuna RFMO catch" actually emerge from spatially reshuffled landing reports to ICCAT, according to the methods of Coulter et al. (2020). In this work, the Sea Around Us team developed a global spatial reallocation of landings of tunas and associated species that were reported to ICCAT. In short, this work uses the fraction of reported spatially disaggregated data to reallocate all data reported to ICCAT. This step is necessary because there are not many countries that report spatial data to ICCAT in the region, and this method may lead to a significant reallocation of landings from the Caribbean to the Venezuelan EEZ, where the local pole and line fishery is significant (J. Mendoza, personal communication). Thus, it is likely that landings from other EEZs, that were included in the reconstructed catch data layer, could have been reassigned to the Venezuelan EEZ.

Because there may be some overlap between the reconstructed data used in the national EEZ estimates, and the ICCAT reporting that was reassigned, there could potentially be a "double counting" of these catch estimates in the Sea Around Us database. We verified with Sea Around Us database managers that their method led to an error resulting in an overestimate of the catch, and they advised us to remove these Unknown Fishing Country catches as they are difficult to trace back to the data source.

12.6 Compile catch in CAR and NED jurisdictional waters by discards and reporting type

Now that we have separated the EEZ catch by region, we will summarize it according to the areas used in the dolphin management strategy evaluation operating model. We will separate the catch by discards versus landings and reported versus unreported data so that we can use these data in various sensitivity analyses.

```
rm(list = ls())  # clear the workspace

# load data
load("data/outputs/SAU_EEZs_WCA.RData")  # SAU EEZ data for NED and NCA saved from earlier
load("data/outputs/erroneous_catch_to_be_removed.RData")   # double-counted catch values to

d <- dwest

# summarize the catch by area, discards/landings and reported/unreported
tapply(d$lbs, list(d$catch_type, d$reporting_status, d$reg), sum, na.rm = T)
```

```
, , Canadian EEZ

         Reported Unreported
Discards       NA         NA
Landings  1220146         NA

, , Western Central Atlantic EEZs

          Reported Unreported
Discards        NA    2179444
Landings 393279187  176215080
```

The catch from the Canadian EEZs is 100% landings, with no unreported data and no discards. In the WCA there are no reported discards, only unreported discards. Thus, we only need to separate out unreported discards and landings for the WCA.

```
# summarize the catch by year, area, discards/landings and reported/unreported

d$year <- factor(d$year, levels = 1950:2019)
dc <- d[which(d$reg == "Canadian EEZ"), ]
dw <- d[which(d$reg == "Western Central Atlantic EEZs"), ]

# verify proper identification of the erroneous Venezuelan Unknown Fishing Country catches
bad <- dw[which(dw$area_name == "Venezuela" & dw$fishing_entity == "Unknown Fishing Country
#dim(bad)
bad_totals <- tapply(bad$lbs, bad$year, sum, na.rm = T)

cbind(tail(bad_totals, 10), tail(removecatch, 10)) # the numbers match exactly
```

```
        [,1]    [,2]
2010      NA      NA
2011      NA      NA
2012      NA      NA
2013 1968566 1968566
2014 2759603 2759603
2015 4915156 4915156
2016 5112450 5112450
2017 6599737 6599737
2018 8236857 8236857
2019 7307245 7307245
```

```
# now remove those bad catches from the database prior to summarizing
#dim(dw)
dw <- dw[-which(dw$area_name == "Venezuela" & dw$fishing_entity == "Unknown Fishing Country
#dim(dw)

CAN <- tapply(dc$lbs, list(dc$year, dc$catch_type, dc$reporting_status), sum, na.rm = T)
tail(CAN)
```

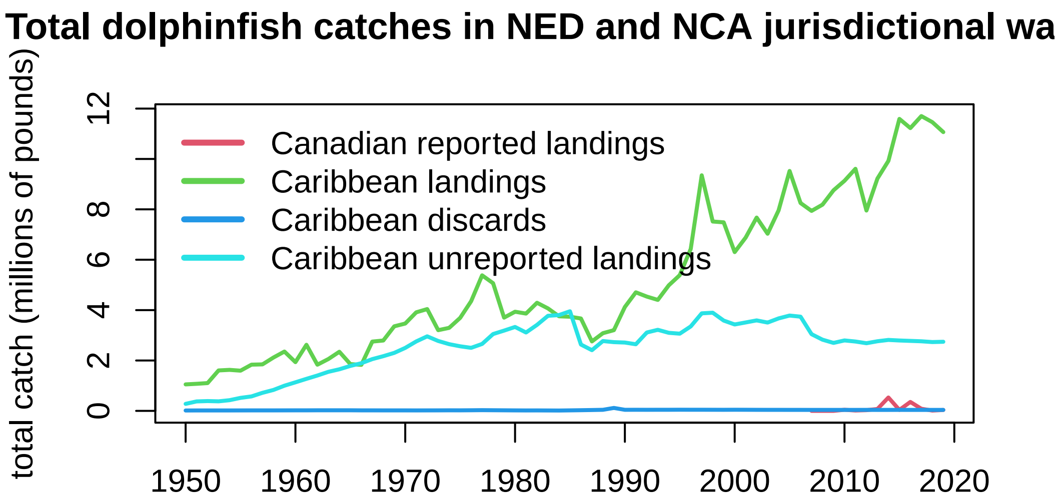
```
, , Reported

      Landings
2014 531022.55
2015  38945.76
2016 354370.65
```

```
2017  83776.88
2018  12588.88
2019  36112.19
```

```
tab_dw <- tapply(dw$lbs, list(dw$year, dw$catch_type, dw$reporting_status), sum, na.rm = T)
tail(tab_dw)
```

```
, , Reported

     Discards Landings
2014       NA  9923818
2015       NA 11590469
2016       NA 11225710
2017       NA 11702729
2018       NA 11460758
2019       NA 11066090

, , Unreported

     Discards Landings
2014 38062.63  2816498
2015 39134.61  2794662
2016 38973.17  2778493
2017 38728.81  2762926
2018 38769.93  2732351
2019 38655.51  2743182
```

```
# combine CAR reported landings, unreported discards, unreported landings
CAR <- data.frame(cbind(tab_dw[, 2, 1], tab_dw[, , 2]))
names(CAR) <- c("landings", "discards", "unreported")

matplot(rownames(CAR), cbind(CAN, CAR)/10^6, type = "l", lwd = 2, lty = 1, col = 2:5,
        xlab = "", ylab = "total catch (millions of pounds)",
        main = "Total dolphinfish catches in NED and NCA jurisdictional waters")
legend("topleft", col = 2:5, c("Canadian reported landings", "Caribbean landings",
                    "Caribbean discards", "Caribbean unreported landings"), lty = 1, lwd =
```




```
round(sum(CAR$discards, na.rm = T) / (sum(CAR$discards, na.rm = T) + sum(CAR$landings, na.r
```

```
[1] 0.61
```

Note that the Caribbean unreported discards in the database are negligible (< 1%) and we exclude them from further analysis.

12.7 Distribute the territorial international catches across quarters of the year

The international catches are only reported as annual catches and do not have any month or season associated with them. However, the MSE operating model requires seasonal catches. Lacking other information on the seasonality of these fleets, we will assume that the fleet distribution of the U.S. pelagic longline is representative of the seasonality of the jurisdictional fleets. Now we parse the annual catches according to the seasonality of the U.S. fleet.

The pelagic longline logbook data is only available going back to 1997, but there are Caribbean catches extending back to the 1950s. To interpolate the seasonality for

1986 - 1996, we take the average seasonality of the oldest 10 years of logbook data (1997 - 2006) and use those averages for the unknown years.

At the time of analysis, the SAU database only contained catches to 2019, so we interpolated the 2019 values to cover years 2020 - 2022.

```
#par(mar = c(5, 10, 3, 1))
#barplot(tapply(dw$lbs, dw$gear_type, sum, na.rm = T), horiz = T, las = 2)
#barplot(tapply(dc$lbs, dc$gear_type, sum, na.rm = T), horiz = T, las = 2)

# read in the percentage of the catch by area and year-quarter combinations from the U.S. P
load("data/outputs/per_PLLcatch_by_area_yearquarter.RData")

names(percatch[1, 1, ]) # matrix 2 == Caribbean, matrix 7 == NED (Atlantic Northwest)
```

```
[1] "NCA"  "CAR"  "FLK"  "NCFL" "NNC"  "VBM"  "NED"
```

```
CAN <- CAN[which(rownames(CAN) >= 1997)]  # PLL data are only available from 1997 on
CAN <- c(CAN, CAN[length(CAN)], CAN[length(CAN)], CAN[length(CAN)])  # interpolate 2019 cat

CANqrt <- percatch[, , 7]   # create matrix for final catch data
for (i in 1:ncol(percatch))  {
  CANqrt[, i] <- percatch[, i, 7] * CAN[i]  }

percatch_car <- percatch[, , 2] # create matrix for Caribbean percentages
int <- rowMeans(percatch_car[, 1:10]) # calculate the average seasonality using the oldest
percatch_car <- cbind(matrix(rep(int, length(1986:1996)), nrow = 4), percatch_car)  # attac
colnames(percatch_car) <- 1986:2022   # full data frame for Caribbean seasonality 1986 - 20

CAR <- CAR[which(rownames(CAR) >= 1986), ]  # only need Caribbean data from 1986 on
CAR <- rbind(CAR, CAR[nrow(CAR), ], CAR[nrow(CAR), ], CAR[nrow(CAR), ])  # interpolate 2019

#dim(CAR)
#dim(percatch_car)

CARrep <- percatch_car   # create matrices for final catch data
CARunr <- percatch_car

for (i in 1:ncol(percatch_car))  {
  CARrep[, i] <- percatch_car[, i] * CAR[i, 1]
```

```
  CARunr[, i] <- percatch_car[, i] * CAR[i, 3]
}

# ensure sums are equal after parsing
sum(CARrep); sum(CAR$landings)
```

```
[1] 286220280
```

```
[1] 286220280
```

```
sum(CARunr); sum(CAR$unreported)
```

```
[1] 112754888
```

```
[1] 112754888
```

```
sum(CAN, na.rm = T); sum(CANqrt, na.rm = T)
```

```
[1] 1328482
```

```
[1] 1328482
```

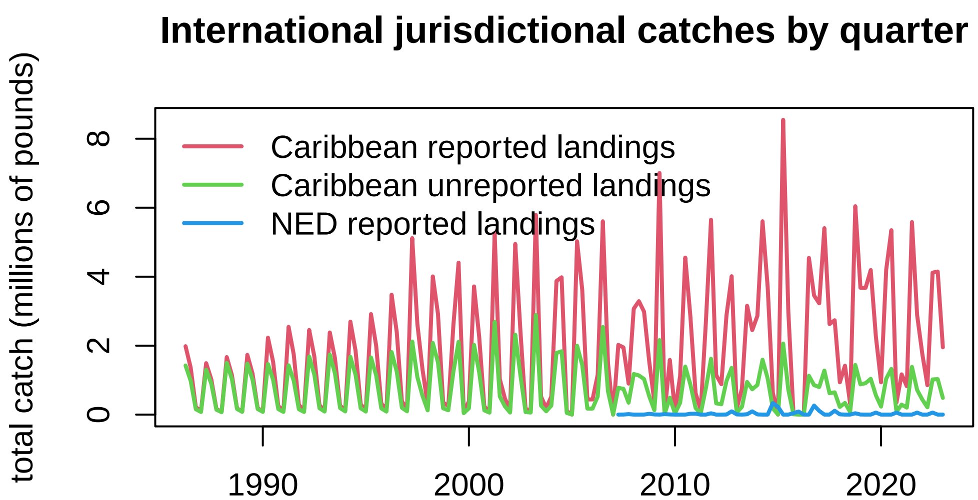
The Canada jurisdictional data need to be combined with the NED high seas data calculated in the previous section, because both of these are a component of the international catches in the NED area.

```
load("data/outputs/NED_year_quarter.RData")

NEDtot <- NED + CANqrt
# ensure sums are correct
sum(NED, na.rm = T) + sum(CANqrt, na.rm = T); sum(NEDtot, na.rm = T)
```

```
[1] 1808855
```

```
[1] 1808855
```

```
NEDtot <- NEDtot[, !is.na(colSums(NEDtot))] # remove NA columns from CAN data

# plot the parsed out data by year-quarter
yrs <- sort(rep(as.numeric(colnames(CARrep)), 4))
qrt <- rep(1:4, length(yrs)/4)
yrs1 <- sort(rep(as.numeric(colnames(NEDtot)), 4))
qrt1 <- rep(1:4, length(yrs1)/4)

matplot(yrs + qrt/4, cbind(matrix(CARrep), matrix(CARunr))/10^6, lty = 1, lwd = 2,
        col = 2:3, type = "l", main = "International jurisdictional catches by quarter",
        xlab = "", ylab = "total catch (millions of pounds)")
lines(yrs1 + qrt1/4, as.numeric(matrix(NEDtot))/10^6, lwd = 2, col = 4)
legend("topleft", c("Caribbean reported landings", "Caribbean unreported landings", "NED re
       lwd = 2, col = 2:4, bty = "n")
```



Finally, we format the data in the format for input into the operating model.

```
# format for operating model - Caribbean
flt <- rep("Intl", length(yrs))
are <- rep("CAR", length(yrs))
fin1 <- data.frame(cbind(yrs, qrt, flt, are, matrix(CARrep)))
```

```r
flt <- rep("UnRep", length(yrs))
fin2 <- data.frame(cbind(yrs, qrt, flt, are, matrix(CARunr)))

findat <- data.frame(rbind(fin1, fin2), stringsAsFactors = FALSE)
names(findat) <- c("Year", "Quarter", "Fleet", "Area", "Catch_lbs")
findat$Catch_lbs <- as.numeric(findat$Catch_lbs)
head(findat)
```

```
  Year Quarter Fleet Area Catch_lbs
1 1986       1  Intl  CAR 1983610.3
2 1986       2  Intl  CAR 1363432.3
3 1986       3  Intl  CAR  216194.7
4 1986       4  Intl  CAR  104564.4
5 1987       1  Intl  CAR 1491251.1
6 1987       2  Intl  CAR 1025009.7
```

```r
#plot(findat$Catch_lbs, type = "l")
#plot(findat$Catch_lbs ~ factor(findat$Fleet))
#plot(findat$Catch_lbs ~ factor(findat$Quarter))
#plot(findat$Catch_lbs ~ factor(findat$Year))
#apply(findat, 2, table)

# write the output file
write.csv(findat, file = "data/FINAL_files/CAR_Intl_unrep_TomFormat.csv", row.names = FALSE

# format for operating model - NED
flt <- rep("Intl", length(yrs1))
are <- rep("NED", length(yrs1))
fin <- data.frame(cbind(yrs1, qrt1, flt, are, matrix(NEDtot)), stringsAsFactors = FALSE)
names(fin) <- c("Year", "Quarter", "Fleet", "Area", "Catch_lbs")
fin$Catch_lbs <- as.numeric(fin$Catch_lbs)
head(fin)
```

```
  Year Quarter Fleet Area Catch_lbs
1 2007       1  Intl  NED     0.000
2 2007       2  Intl  NED  1450.366
3 2007       3  Intl  NED 13279.912
4 2007       4  Intl  NED     0.000
5 2008       1  Intl  NED     0.000
6 2008       2  Intl  NED  1312.750
```

```
#apply(fin, 2, table)

# write output file
write.csv(fin, file = "data/FINAL_files/NED_Intl_TomFormat.csv", row.names = FALSE)

#barplot(tapply(fin$Catch_lbs, list(fin$Quarter), sum, na.rm = T), col = rainbow(4))
```

12.8 References

Pauly D., Zeller D., Palomares M.L.D. (Editors), 2020. Sea Around Us Concepts, Design and Data (seaaroundus.org).

13 Data compilation for all fleets in the operating model

In this last step we will compile all of the output data files from the different data sources (NMFS commercial trip tickets, NMFS recreational statistics, Sea Around Us) and check the files for accuracy.

Recall that there are seven defined areas in the operating model, over which the landings are summed. Four of these areas are within the U.S. EEZ and the other three areas are international waters (high seas and jurisdictional waters of other countries).

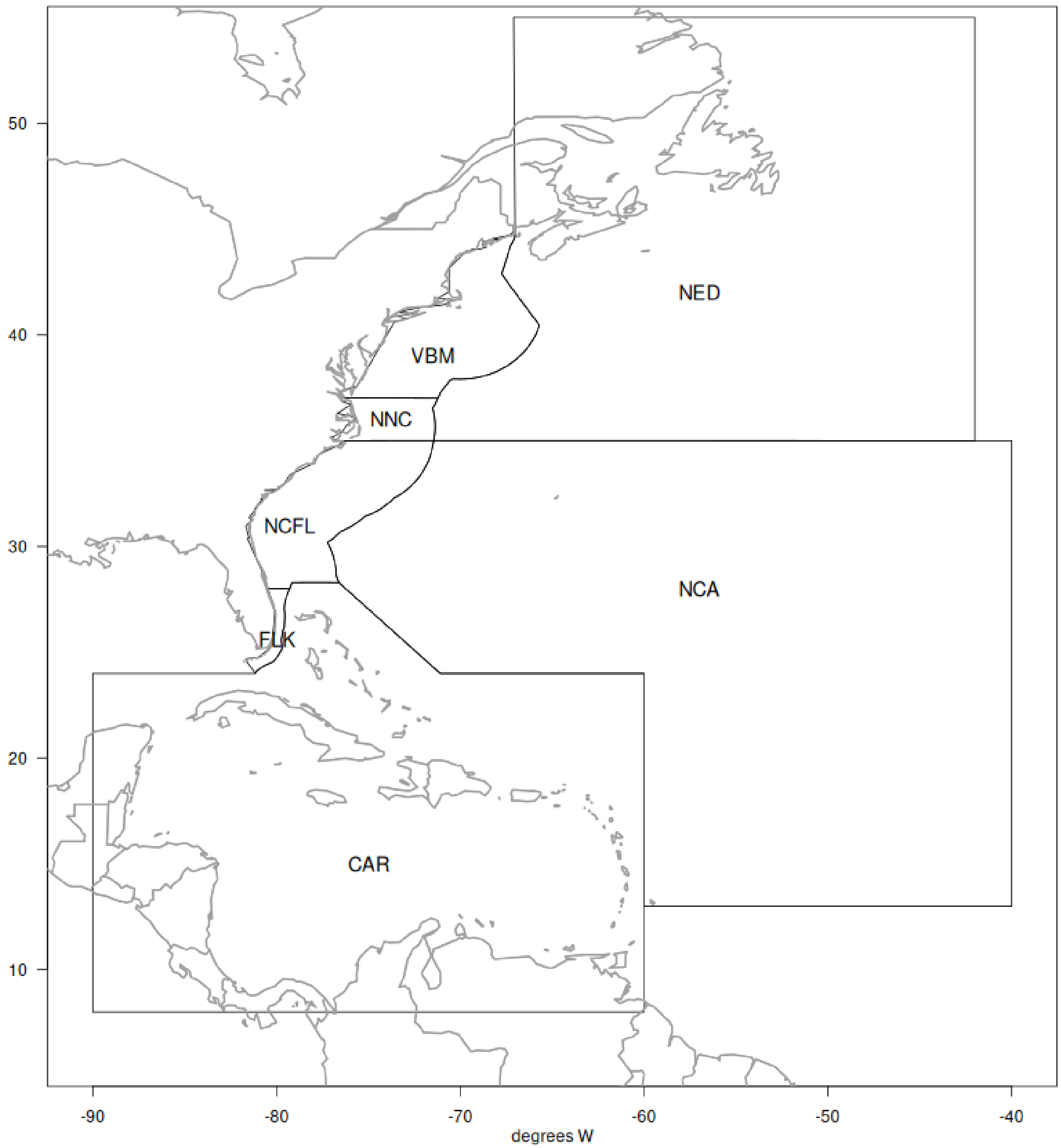


Figure 13.1: Areas defined in the operating model.

13.1 Data upload

First we access all of the final data files from the previous steps.

```
# clear workspace
rm(list = ls())

# view data files in directory
dir("data/FINAL_files/")
```

```
[1] "CAR_Intl_unrep_TomFormat.csv"
[2] "commercial_TomFormat.csv"
[3] "complete_catch_dataset_09052025b.csv"
[4] "NCA_Intl_TomFormat.csv"
[5] "NED_Intl_TomFormat.csv"
[6] "rec_catch_TomFormat.csv"
```

```
intc <- read.csv("data/FINAL_files/CAR_Intl_unrep_TomFormat.csv")
intn <- read.csv("data/FINAL_files/NCA_Intl_TomFormat.csv")
intw <- read.csv("data/FINAL_files/NED_Intl_TomFormat.csv")
com <- read.csv("data/FINAL_files/commercial_TomFormat.csv")
rec <- read.csv("data/FINAL_files/rec_catch_TomFormat.csv")
```

This file contains all international landings and unreported catches for the greater Caribbean region (jurisdictional waters of all countries). Recall that estimated discards were very small (< 1%) and were removed from the summary.

```
apply(intc[1:4], 2, table, useNA = "always")
```

```
$Year

1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001
   8    8    8    8    8    8    8    8    8    8    8    8    8    8    8    8
2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017
   8    8    8    8    8    8    8    8    8    8    8    8    8    8    8    8
2018 2019 2020 2021 2022 <NA>
   8    8    8    8    8    0
```

```
$Quarter

   1    2    3    4 <NA>
  74   74   74   74    0

$Fleet

 Intl UnRep  <NA>
  148   148     0

$Area

 CAR <NA>
 296    0
```

This file contains all international landings for the NCA region (i.e., FAO region 31), also known as the Western Central Atlantic. There are no territorial waters in this area (except for Bermuda, which as discussed previously is summarized with the CAR region) and there are no unreported catches or discards in the database.

```
apply(intn[1:4], 2, table, useNA = "always")
```

```
$Year

1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012
   4    4    4    4    4    4    4    4    4    4    4    4    4    4    4    4
2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 <NA>
   4    4    4    4    4    4    4    4    4    4    0

$Quarter

   1    2    3    4 <NA>
  26   26   26   26    0

$Fleet

Intl <NA>
 104    0

$Area
```

```
 NCA <NA> 
 104    0 
```

This file contains all international landings for the NED region (i.e., FAO region 21), also known as the Northwest Atlantic. This includes NED high seas landings, as well as landings from Canadian and French territorial waters which fall in this region. There are no unreported catches or discards for this region in the database.

```r
apply(intw[1:4], 2, table, useNA = "always")
```

```
$Year

2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 
   4    4    4    4    4    4    4    4    4    4    4    4    4    4    4    4 
<NA> 
   0 

$Quarter

   1    2    3    4 <NA> 
  16   16   16   16    0 

$Fleet

Intl <NA> 
  64    0 

$Area

 NED <NA> 
  64    0 
```

This file contains all landings for the U.S. commercial fleet, including landings from U.S. EEZs and high seas. There are no unreported catches in this region. Recall that estimated dead discards were very small (< 2%) and were removed from the summary.

```r
apply(com[1:4], 2, table, useNA = "always")
```

```
$Year

1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001
  28   28   28   28   28   28   28   28   28   28   28   28   28   28   28   28
2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017
  28   28   28   28   28   28   28   28   28   28   28   28   28   28   28   28
2018 2019 2020 2021 2022 <NA>
  28   28   28   28   28    0

$Quarter

   1    2    3    4 <NA>
 259  259  259  259    0

$Fleet

UScom  <NA>
 1036     0

$Area

 CAR  FLK  NCA NCFL  NED  NNC  VBM <NA>
 148  148  148  148  148  148  148    0
```

This file contains all landings for the U.S. recreational fleet, which are all assumed to occur within the U.S. EEZs. There are no unreported catches in this region.

```r
apply(rec[1:4], 2, table, useNA = "always")
```

```
$Year

1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001
  32   32   32   32   32   32   32   32   32   32   32   32   32   32   32   32
2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017
  32   32   32   32   32   32   32   32   32   32   32   32   32   32   32   32
2018 2019 2020 2021 2022 <NA>
  32   32   32   32   32    0
```

```
$Quarter

   1    2    3    4 <NA> 
 296  296  296  296    0 

$Fleet

Hire  Rec <NA> 
 592  592    0 

$Area

 FLK NCFL  NNC  VBM <NA> 
 296  296  296  296    0 
```

Now we combine the data files and create some figures with the final composite data file.

```
d <- rbind(intc, intn, intw, com, rec)

# how many rows contain zeros
#hist(d$Catch_lbs)
table(d$Catch_lbs == 0)
```

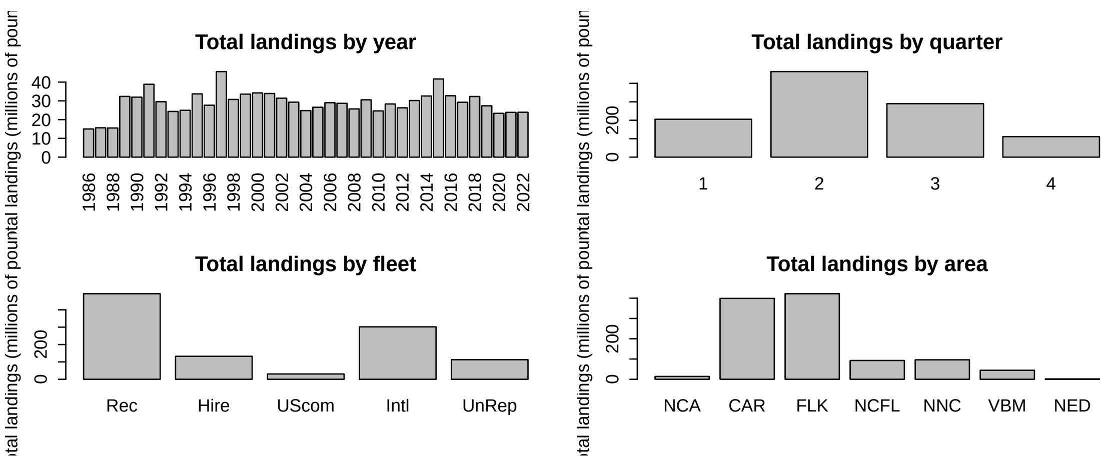
```
FALSE  TRUE 
 2047   637 
```

```
# resort levels geographically
d$Area <- factor(d$Area, levels = c("NCA", "CAR", "FLK", "NCFL", "NNC", "VBM", "NED"))
d$Fleet <- factor(d$Fleet, levels = c("Rec", "Hire", "UScom", "Intl", "UnRep"))
```

13.2 Visualizing the final data set

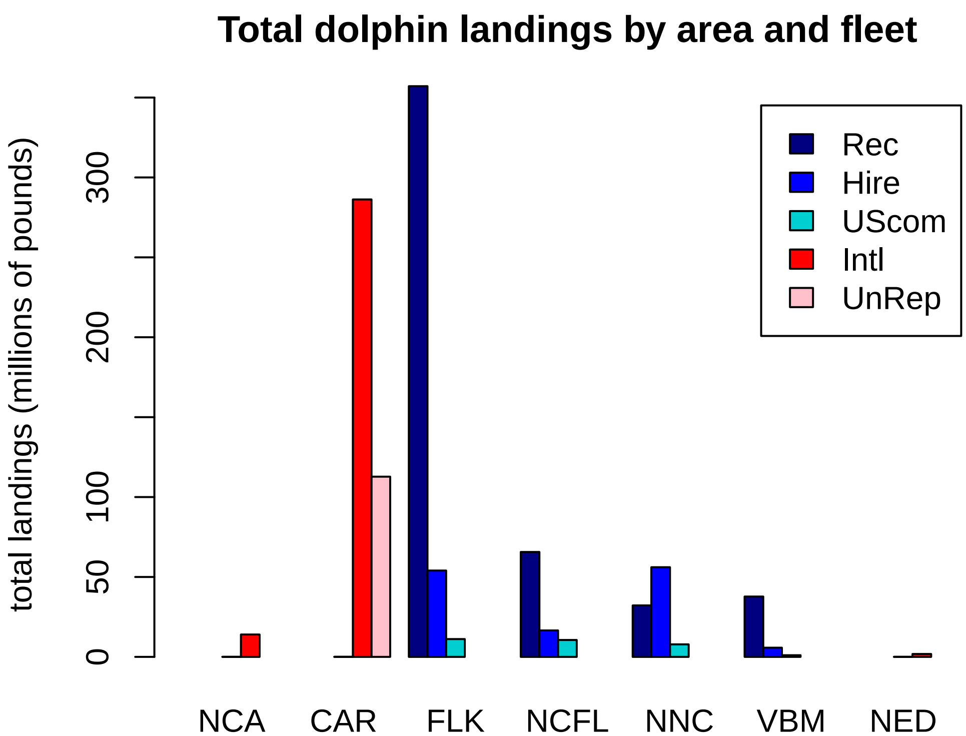
We plot the final data set in a number of different ways to detect any errors that may have occurred during processing.

```
# plot the total landings by each factor
par(mfrow = c(2, 2), mex = 0.7)
barplot(tapply(d$Catch_lbs, d$Year, sum, na.rm = T)/10^6, las = 2,
        main = "Total landings by year", ylab = "total landings (millions of pounds)")
barplot(tapply(d$Catch_lbs, d$Quarter, sum, na.rm = T)/10^6,
        ylab = "total landings (millions of pounds)", main = "Total landings by quarter")
barplot(tapply(d$Catch_lbs, d$Fleet, sum, na.rm = T)/10^6,
        main = "Total landings by fleet", ylab = "total landings (millions of pounds)")
barplot(tapply(d$Catch_lbs, d$Area, sum, na.rm = T)/10^6,
        main = "Total landings by area", ylab = "total landings (millions of pounds)")
```



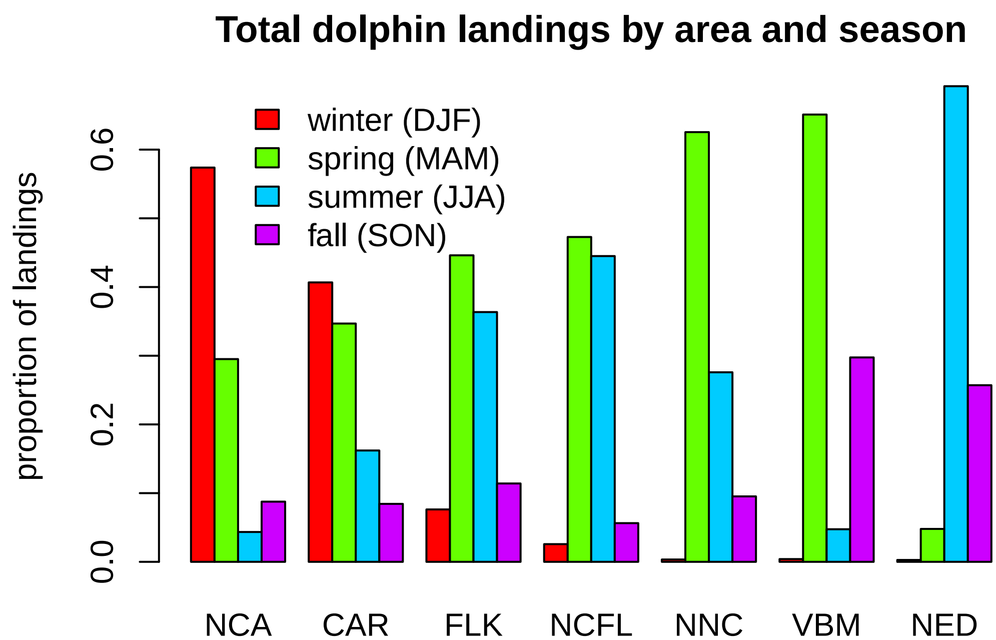
The total landings in the regions analyzed is highly variable but relatively stable over time with a slight decrease in recent years. Most of the landings are caught in quarter 2 (March - May). The largest sector is the U.S. private recreational sector, followed by international fleets (all types), the for-hire fleet and then the U.S. commercial fleet. Most of the fishing activity takes place in the Florida Keys and the Greater Caribbean, with other regions contributing much less.

```
par(mfrow = c(1, 1), mar = c(2, 5, 3, 1))
tab <- tapply(d$Catch_lbs, list(d$Fleet, d$Area), sum, na.rm = T)/10^6
barplot(tab, beside = T, col = c("navy", "blue", "darkturquoise", "red", "pink")
        , legend = rownames(tab),
        main = "Total dolphin landings by area and fleet",
        ylab = "total landings (millions of pounds)")
```



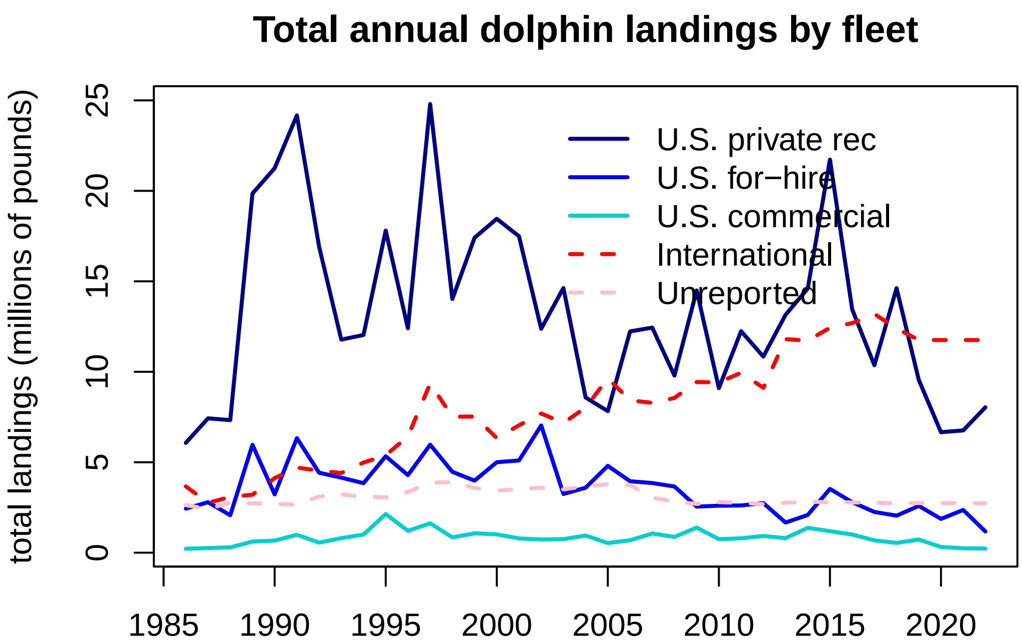
As expected, U.S. recreational fishing (private and for-hire) only takes place in the four U.S. EEZ regions (FLK, NCFL, NNC and VBM). The U.S. commercial fishery operates largely in those four U.S. EEZ regions, with small amounts of landings in the high seas of NCA, CAR and NED regions. International landings occur largely in the Caribbean, with lesser amounts in the NCA and NED high seas regions.

```
tab <- tapply(d$Catch_lbs, list(d$Quarter, d$Area), sum, na.rm = T)/10^6
tabp <- apply(tab, 2, function(x) x / sum(x, na.rm = T))

par(mar = c(2, 4, 3, 1))
barplot(tabp, beside = T, col = rainbow(4, end = 0.8),
        legend = c("winter (DJF)", "spring (MAM)", "summer (JJA)", "fall (SON)"),
        args.legend = list(x = 15, y = 0.7, bty = "n"),
        main = "Total dolphin landings by area and season",
        ylab = "proportion of landings")
```



This plot shows the seasonal movements of dolphin across the different regions, as expressed through the proportion of landings that occurs in each quarter within each area. In the NCA and CAR regions, most of the landings occur in winter, whereas in the U.S. South Atlantic regions, most of the landings occur in spring and summer. In the U.S. Mid-Atlantic the landings also occur largely in spring, and also fall. In the NED region most of the landings occur in summer.

```
tab <- tapply(d$Catch_lbs, list(d$Year, d$Fleet), sum, na.rm = T)/10^6

par(mar = c(2, 4, 3, 1))
matplot(rownames(tab), tab,
        col = c("navy", "blue", "darkturquoise", "red", "pink"),
        type = "l", lwd = 2, lty = c(rep(1, 3), 2, 2),
        main = "Total annual dolphin landings by fleet", xlab = "",
        ylab = "total landings (millions of pounds)")
legend(2002, 25, lwd = 2, lty = c(rep(1, 3), 2, 2), bty = "n",
       c("U.S. private rec", "U.S. for-hire", "U.S. commercial",
         "International", "Unreported"),
       col = c("navy", "blue", "darkturquoise", "red", "pink"))
```



This plot shows the time series of total annual dolphin landings by fleet. We can compare these figures with the plots from the previous chapters to ensure the data were processed correctly. The magnitude and variability of the landings match the expectations.

```
# output the final concatenated data file
write.csv(d, file = "data/FINAL_files/complete_catch_dataset_09052025b.csv", row.names = FA


d1 <- d[which(d$Year == 2022), ]
d1$ar2 <- ""
d1$ar2[which(d1$Area == "FLK")] <- "S"
d1$ar2[which(d1$Area == "NCFL")] <- "S"
d1$ar2[which(d1$Area == "NNC")] <- "N"
d1$ar2[which(d1$Area == "VBM")] <- "N"
d1$cat <- paste0(d1$Fleet, d1$ar2)
d1$cat[which(d1$Fleet == "UScom")] <- "UScom"

round(tapply(d1$Catch_lbs, d1$cat, sum, na.rm = T)/10^6/2.205, 3)
```

```
HireN HireS  Intl  RecN  RecS UnRep UScom
```

```
0.202 0.329 5.332 0.643 3.002 1.244 0.104
```

14 Predictive indices of dolphin abundance

Dolphin abundance in South Atlantic waters is driven by a combination of factors that are mostly outside of U.S. fishery management control, including environmental conditions and catch in international waters. In order to create a management procedure for the dolphin management strategy evaluation we need an index that can predict ahead of time whether the abundance of dolphin in U.S. waters is likely to be above or below average in a given year. Ideally this index can capture the dynamics early in the year or a season ahead in order to allow adequate time for management response. In the following sections we explore potential predictive indices of dolphin abundance.

14.1 Recreational landings as a proxy for availability

Because the recreational landings is generally unconstrained by management (the total quota is rarely reached and landings are on average well below the bag limits), the total recreational landings could serve as a proxy for dolphin availability in South Atlantic waters over time. We take the recreational data from Section 3 and summarize according to the areas specified by the management strategy evaluation: Florida Keys to Indian River County County (FLK), Brevard to Southern NC South of Hatteras (NCFL), Northern NC North of Hatteras to NC/VA border (NNC), and Virginia to Maine (VBM). We also include Gulf of America landings because dolphin are not managed in the Gulf region and thus the landings are not constrained by management and is most likely a function of availability.

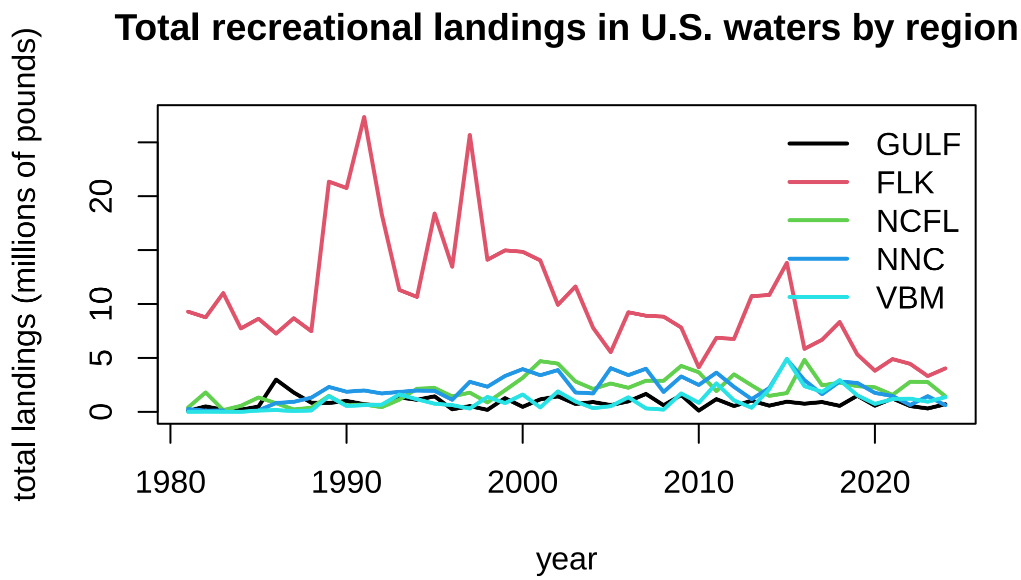
```
# clear workspace
rm(list = ls())

# install library
if(!require("readr")) install.packages("readr")
library(readr)

# read in data file
```

```
rec <- read.csv("data/recLandings.csv")
names(rec)[1] <- "Year"
```

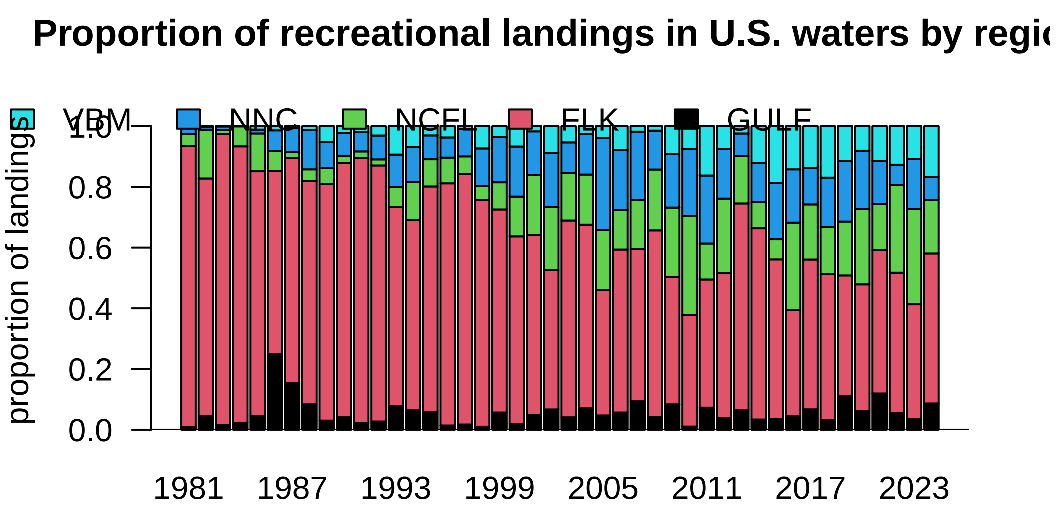
14.2 Plot the recreational landings

Plotting the total recreational landings by region and the proportion of landings by region, we can observe that most of the landings in U.S. waters came from the FLK region, particularly in the earlier years. More recently, an increasing proportion of the landings has been coming from NCFL and further north, with FLK composing only about 50% of the landings. Gulf landings have historically been a small proportion of the total landings and have remained small throughout time. Landings in the VBM region were historically very small (~1%) but in recent years have varied between 10-18% of the landings.

```
# plot landings -------------------------------------------------
matplot(rec$Year, rec[2:6]/10^6, type = "l", lty = 1, lwd = 2, col = 1:5,
        xlab = "year", ylab = "total landings (millions of pounds)",
        main = "Total recreational landings in U.S. waters by region")
legend("topright", colnames(rec)[2:6], col = 1:5, lty = 1, lwd = 2, bty = "n")
```



```
# plot in terms of percentages ---------------------------------
rec1 <- rec[2:6]
rec1 <- as.matrix(rec1/rowSums(rec1, na.rm = T))
rec1[is.na(rec1)] <- 0

b <- barplot(t(rec1), col = 1:5, names.arg = rec$Year, las = 1, ylim = c(0, 1.05),
             ylab = "proportion of landings",
             main = "Proportion of recreational landings in U.S. waters by region",
             legend.text = colnames(rec)[2:6], args.legend = list(x = 45, y = 1.15, horiz =
abline(h=0)
```




```
# calculate total landings and South Atlantic landings for later analysis
rec$ATL <- rowSums(rec[3:6], na.rm = T)
rec$ALL <- rec$GULF + rec$ATL

# subset data from 1990 forward
rec <- rec[which(rec$Year >= 1990), ]
```

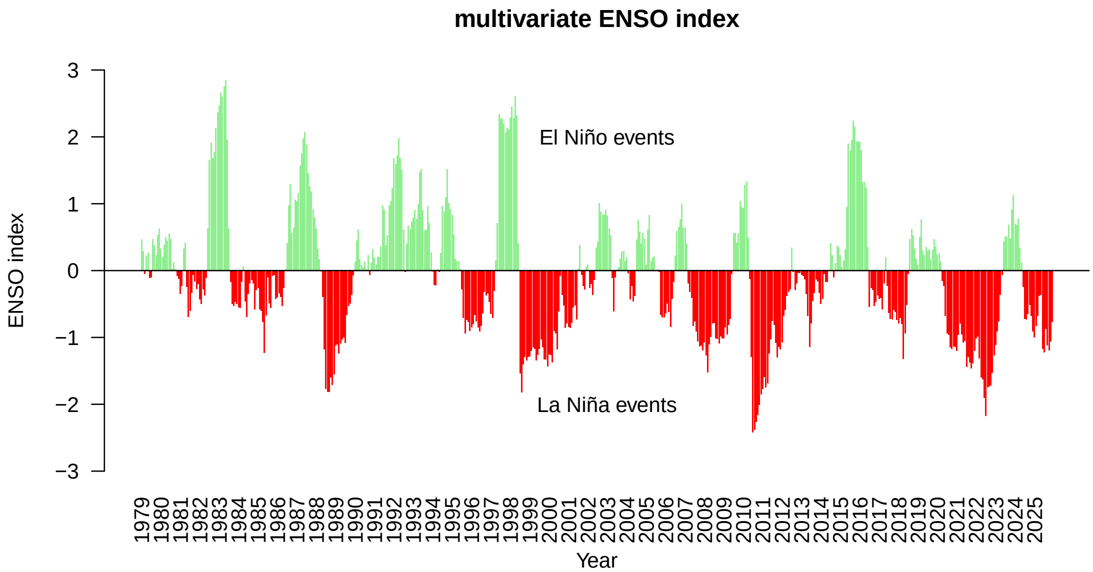
Recreational monitoring was first established in 1979 and went through a number of changes in the 1980s, particularly related to monitoring of the for-hire fleet. We can see a very large jump in landings in the late 1980s that we suspect is related to some of these changes in methodology. For our data explorations we will use recreational data from 1990 forward as a reference period in which the monitoring was more standardized.

14.3 Potential influence of El Niño Southern Oscillation

We note that many large peaks in recreational landings have occurred in years where there were severe El Niño events (e.g., 1997-1998, 2015). We first explore the El Niño Southern Oscillation (ENSO) as a potential environmental predictor of dolphin abundace. An index based on El Niño is potentially attractive because there is some ability for seasonal predictions of El Niño (3-6 months ahead of time). As large-scale disruptors of atmospheric conditions across the globe, El Niño events have impacts on the distribution of temperatures in the Atlantic and could trigger changes in fish distributions.

We use the multivariate ENSO index from NOAA's Physical Sciences Laboratory, which combines the influences of five different variables related to Pacific Ocean state into one index. The index is updated monthly and historical data can be downloaded directly from the site.

```
#download ENSO index
url <- "https://www.psl.noaa.gov/enso/mei/data/meiv2.data"
download.file(url = url, destfile = "data/enso.txt")

all_lines <- readLines("data/enso.txt")
footer_start <- grep("-999", all_lines)
e <- read.table("data/enso.txt", skip = 1, nrows = footer_start[1] - 2)
names(e) <- c("Year", "DJ", "JF", "FM", "MA", "AM", "MJ", "JJ", "JA", "AS", "SO", "ON", "ND

ets <- as.vector(as.matrix(t(e[2:13])))
years <- sort(rep(e$Year, 12)) + rep((0:11)/12, length(min(e$Year):max(e$Year)))

barplot(ets, names.arg = floor(years), las = 2,
        col = ifelse(ets >= 0, "lightgreen", "red"), border= NA, ylim = c(-3, 3),
        main = "multivariate ENSO index", xlab = "Year", ylab = "ENSO index")
abline(h = 0)
text(346, 2, "El Niño events")
text(346, -2, "La Niña events")
```

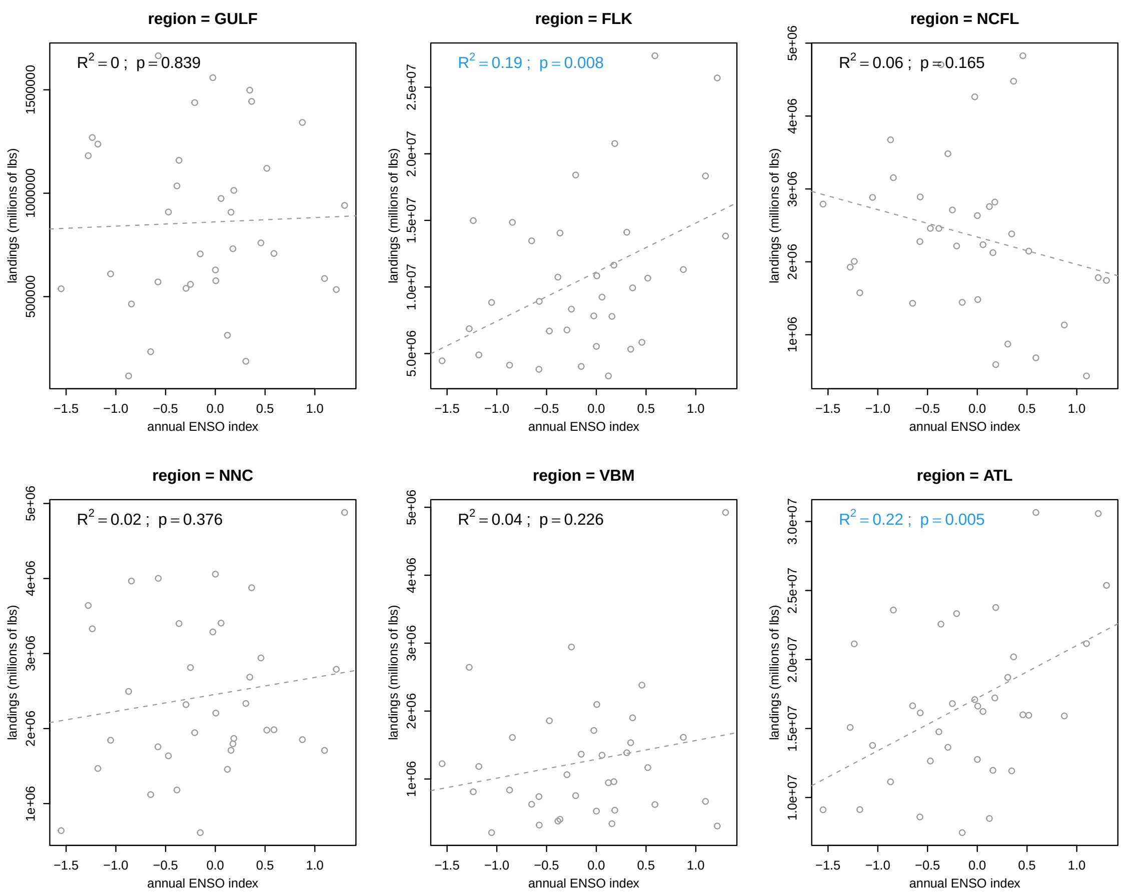


Here is the raw time series, showing positive and negative anomalies in the ENSO state.

Now we will compare the ENSO index to the recreational landings. First we compare the annual average ENSO index to the landings from the different regions. We report adjusted R-squared values and p-values for linear regression of the ENSO index versus the landings; significant relationships are highlighted in blue.

```
# calculate mean
e$mean <- rowMeans(e[2:13], na.rm = T)

# merge years with recreational landings data
d <- merge(rec, e, by = "Year")
#head(d)

lis <- names(d)[2:7]
par(mfrow = c(2, 3), mex = 0.8, mgp = c(2.5, 1, 0))
for(i in 1:6) {
  catch <- d[, which(names(d) == lis[i])]
  plot(d$mean, catch, xlab = "annual ENSO index", ylab = "landings (millions of lbs)",
       col = 8, main = paste0("region = ", lis[i]))
  out <- lm(catch ~ d$mean)
  abline(out, col = 8, lty = 2)
```

```
  r2 <- summary(out)$r.squared
  p_val <- summary(out)$coefficients[2, 4]
  p_display <- ifelse(p_val < 0.001, "p < 0.001", paste("p =", round(p_val, 3)))
  co <- 1
  if (p_val < 0.01)  { co <- 4 }
  if (p_val < 0.001) { co <- 5 }
legend("topleft",
       legend = bquote(R^2 == .(round(r2, 2)) ~ "; " ~ p == .(round(p_val, 3))),
       bty = "n", cex = 1.2, text.col = co)
}
```

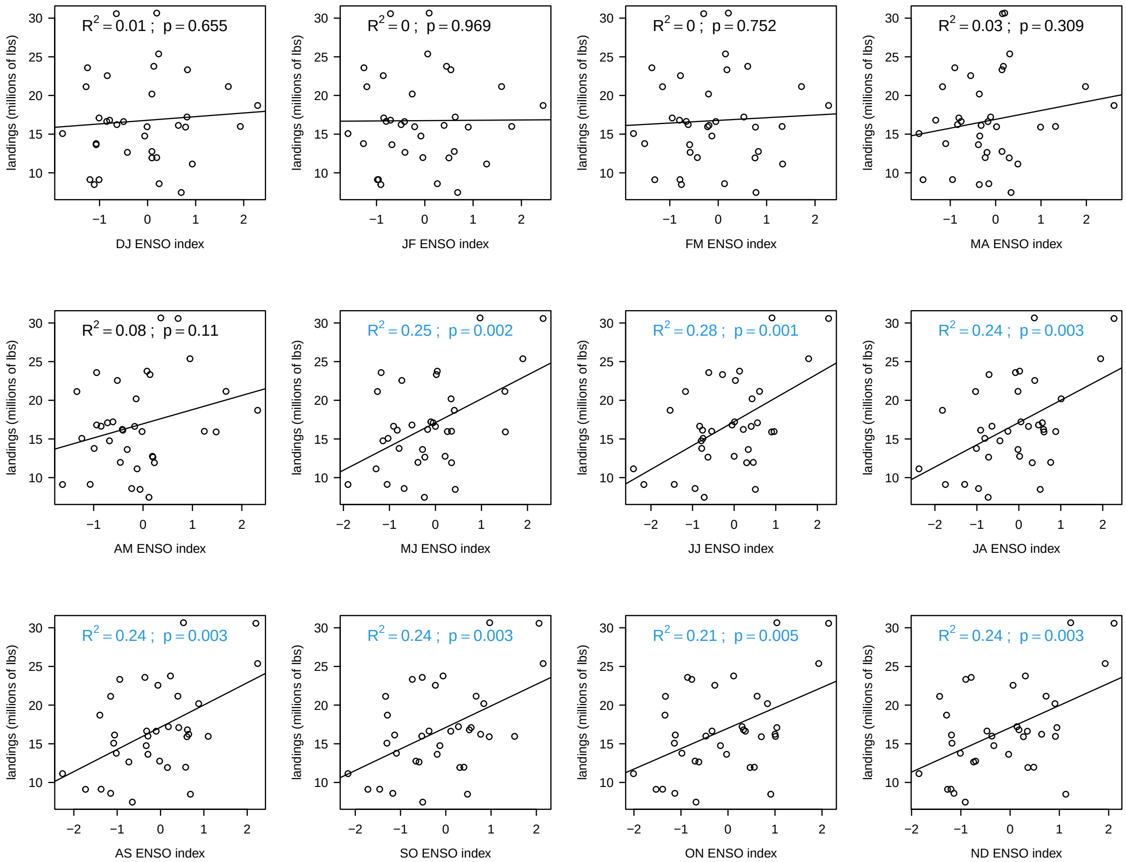


The ENSO index appears to be most closely related to the total recreational landings across the South Atlantic (driven primarily by landings in the FLK region). The relationship is highly significant and the ENSO index explains 19% of the variation in the total landings.

14.4 Seasonal influence of El Niño Southern Oscillation

Now we will consider the seasonal influence of ENSO, looking at correlations between the index during individual months and the landings. Since the strongest relationship occurs with the total South Atlantic landings, we will use these landings in the analysis.

```
lis <- names(e)[2:13]
par(mfrow = c(3, 4), mex = 0.8)
for(i in 1:12) {
  ind <- d[, which(names(d) == lis[i])]
  plot(ind, d$ATL/10^6, xlab = paste(lis[i], "ENSO index"), ylab = "landings (millions of l
  out <- lm(d$ATL/10^6 ~ ind)
  abline(out)

  r2 <- summary(out)$r.squared
  p_val <- summary(out)$coefficients[2, 4]
  p_display <- ifelse(p_val < 0.001, "p < 0.001", paste("p =", round(p_val, 3)))
  co <- 1
  if (p_val < 0.01)  { co <- 4 }
  if (p_val < 0.001) { co <- 5 }
legend("topleft",
       legend = bquote(R^2 == .(round(r2, 2)) ~ "; " ~ p == .(round(p_val, 3))),
       bty = "n", cex = 1.2, text.col = co)
}
```

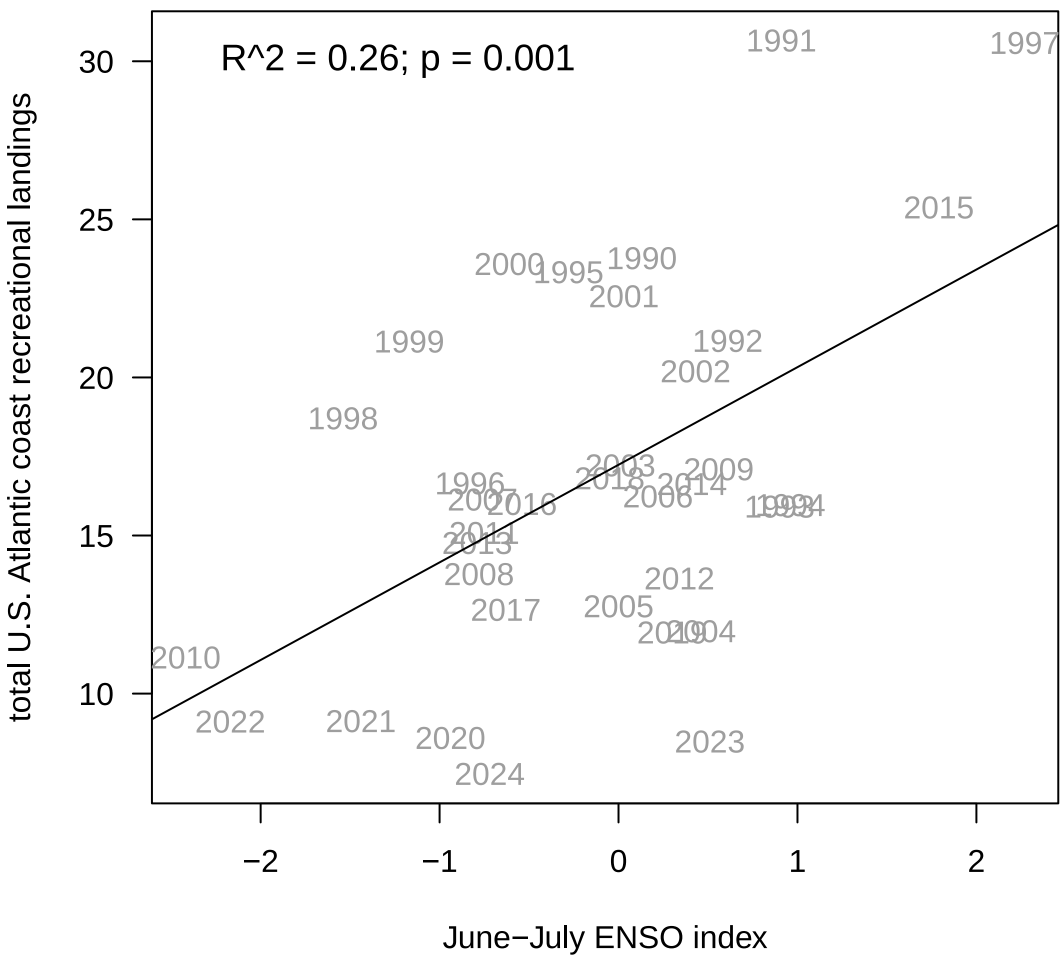


A significant relationship between the monthly ENSO index and total South Atlantic recreational landings occurs with the May-June index, but the strongest relationship occurs in June-July (with 26% of the variance in landings explained by the index in that month). The relationship in the late summer and fall months are also significant, which is expected since the ENSO index is highly correlated from month to month.

We will test the June-July ENSO index in the management procedure as it has the strongest relationship.

```
par(mfrow = c(1, 1))
plot(d$JJ, d$ATL/10^6, xlab = "June-July ENSO index", ylab = "total U.S. Atlantic coast rec
text(d$JJ, d$ATL/10^6, d$Year, col = 8)
out <- lm(d$ATL/10^6 ~ d$JJ)
abline(out, col = 1)
p <- summary(out)$coef[2, 4]
```

```r
legend("topleft", paste0("R^2 = ", round(summary(out)$adj.r.squared, 2), "; p = ",
                         round(p, 3)), cex = 1.2, bty = "n", text.col = 1)
```




```r
summary(out)
```

```
Call:
```

```
lm(formula = d$ATL/10^6 ~ d$JJ)

Residuals:
    Min      1Q  Median      3Q     Max 
-10.332  -3.886  -0.054   2.320  10.606 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  17.2374     0.8641  19.949  < 2e-16 ***
d$JJ          3.0873     0.8680   3.557  0.00116 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.046 on 33 degrees of freedom
Multiple R-squared:  0.2771,	Adjusted R-squared:  0.2552 
F-statistic: 12.65 on 1 and 33 DF,  p-value: 0.001161
```

```r
# save the index to the indices file

ind <- data.frame(cbind(d$Year, d$JJ))
write.csv(ind, file = "indices/enso_index.csv", row.names = F)
```

15 Wider Atlantic temperature distribution index

Dolphin have a circumtropical distribution, migrating to temperate waters in summer and tropical waters in winter, in a circular pattern around the Atlantic Ocean. Studies using satellite tagging have shown that, similar to other highly migratory species, they prefer waters within a narrow temperature range, of 27-30 degrees Celsius Schlenker et al. 2021. To determine whether shifting temperature distributions across the Atlantic could be impacting the abundance in U.S. waters, we analyzed Atlantic Ocean temperature patterns at the large spatial and temporal scale.

15.1 Accessing temperature data

For this analysis we need a historical temperature record that is available at large spatial (entire Atlantic) and temporal (monthly-yearly) scales. We use NOAA's Extended Reconstructed Sea Surface Temperature (ERSST) data, provided by the NOAA National Centers for Environmental Information and available through ERDDAP. This temperature record goes back to the mid-1800s, is available at monthly time steps and a 2°x2° grid size resolution, and is suitable for looking at large-scale patterns over decades. We download the data from 1980 - present and subset for the Atlantic Ocean from 100°W - 0°W longitude and 0°N - 70°N latitude.

```
# clear workspace
rm(list = ls())

# install libraries
if(!require("maps")) install.packages("maps")
# if(!require("ncdf4")) install.packages("ncdf4")
# if(!require("lubridate")) install.packages("lubridate")
# if(!require("rerddap")) install.packages("rerddap")

 library(maps)
```

```
# library(ncdf4)
# library(lubridate)
# library(rerddap)

# get ERDDAP info
# id <- info('nceiErsstv5')   #https://coastwatch.pfeg.noaa.gov/erddap/griddap/nceiErsstv5.

# download data using griddap - initially takes a long time
# uncomment lines below to update analysis
# sst_grab <- griddap(id, fields = 'sst',
#                     time = c('1990-01-01', '2025-01-01'),
#                     longitude = c(260, 358), latitude = c(0, 70))

# open nc file and extract variables
# nc <- nc_open(sst_grab$summary$filename, write=FALSE, readunlim=TRUE, verbose=FALSE)
# v1 <- nc$var[[1]]
# sst <- ncvar_get(nc, v1)
# lon <- v1$dim[[1]]$vals - 360
# lat <- v1$dim[[2]]$vals
# nc_close(nc)

# convert time variable to month and year
#tim <- as.POSIXct(v1$dim[[4]]$vals, origin = "1970-01-01")
#mon <- as.numeric(substr(tim, 6, 7))
#yr <- substr(tim, 1, 4)
#dim(sst)

#save(sst, lon, lat, tim, mon, yr, file = "data/ERSST.RData")

load("data/ERSST.RData")

# check conversions
tim[1:5]
```

```
[1] "1990-01-01 UTC" "1990-02-01 UTC" "1990-03-01 UTC" "1990-04-01 UTC"
[5] "1990-05-01 UTC"
```

```
table(mon)
```

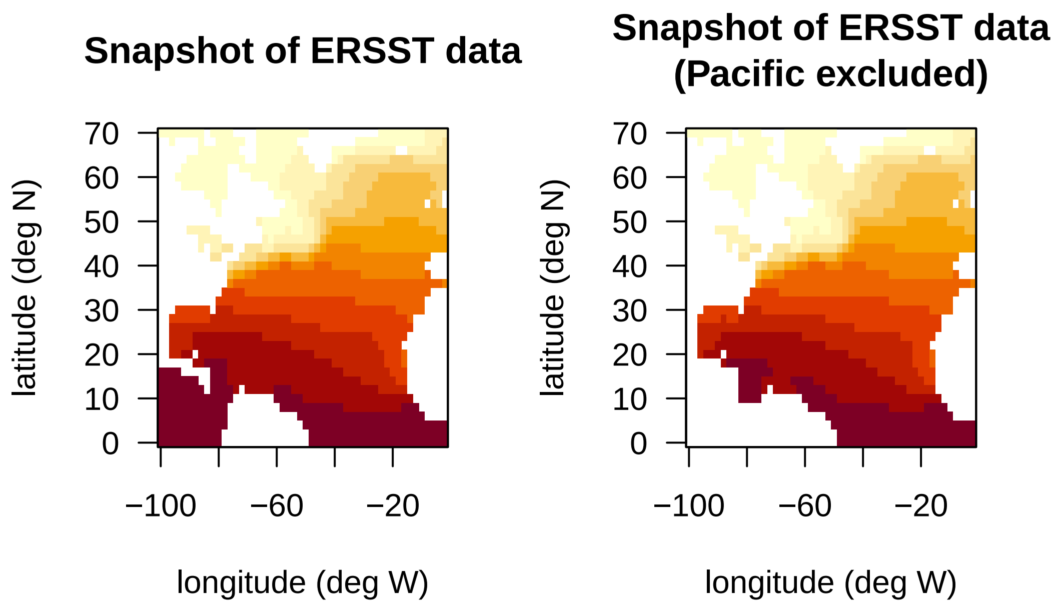
```
mon
```

```
 1  2  3  4  5  6  7  8  9 10 11 12
36 35 35 35 35 35 35 35 35 35 35 35
```

```
table(yr)
```

```
yr
1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004
   1   12   12   12   12   12   12   12   12   12   12   12   12   12   12   12
2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020
  12   12   11   12   12   12   12   12   12   12   12   12   12   12   12   12
2021 2022 2023 2024 2025
  12   12   12   12    1
```

```
# plot SST on map for first time step
par(mfrow = c(1, 2))
image(lon, lat, sst[, , 100], las = 1, xlab = "longitude (deg W)", ylab = "latitude (deg N)
      main = "Snapshot of ERSST data"); box()

# fill Pacific Ocean with NAs so values are not included
sst[which(lon <= (-92)), which(lat <= 17.5), ] <- NA
sst[which(lon <= (-85)), which(lat <= 15), ] <- NA
sst[which(lon <= (-83)), which(lat <= 10), ] <- NA
sst[which(lon <= (-77)), which(lat <= 8.5), ] <- NA
sst[which(lon <= (-78) & lon >= (-80)), which(lat <= 9), ] <- NA
image(lon, lat, sst[, , 100], las = 1, xlab = "longitude (deg W)", ylab = "latitude (deg N)
      main = "Snapshot of ERSST data\n(Pacific excluded)"); box()
```



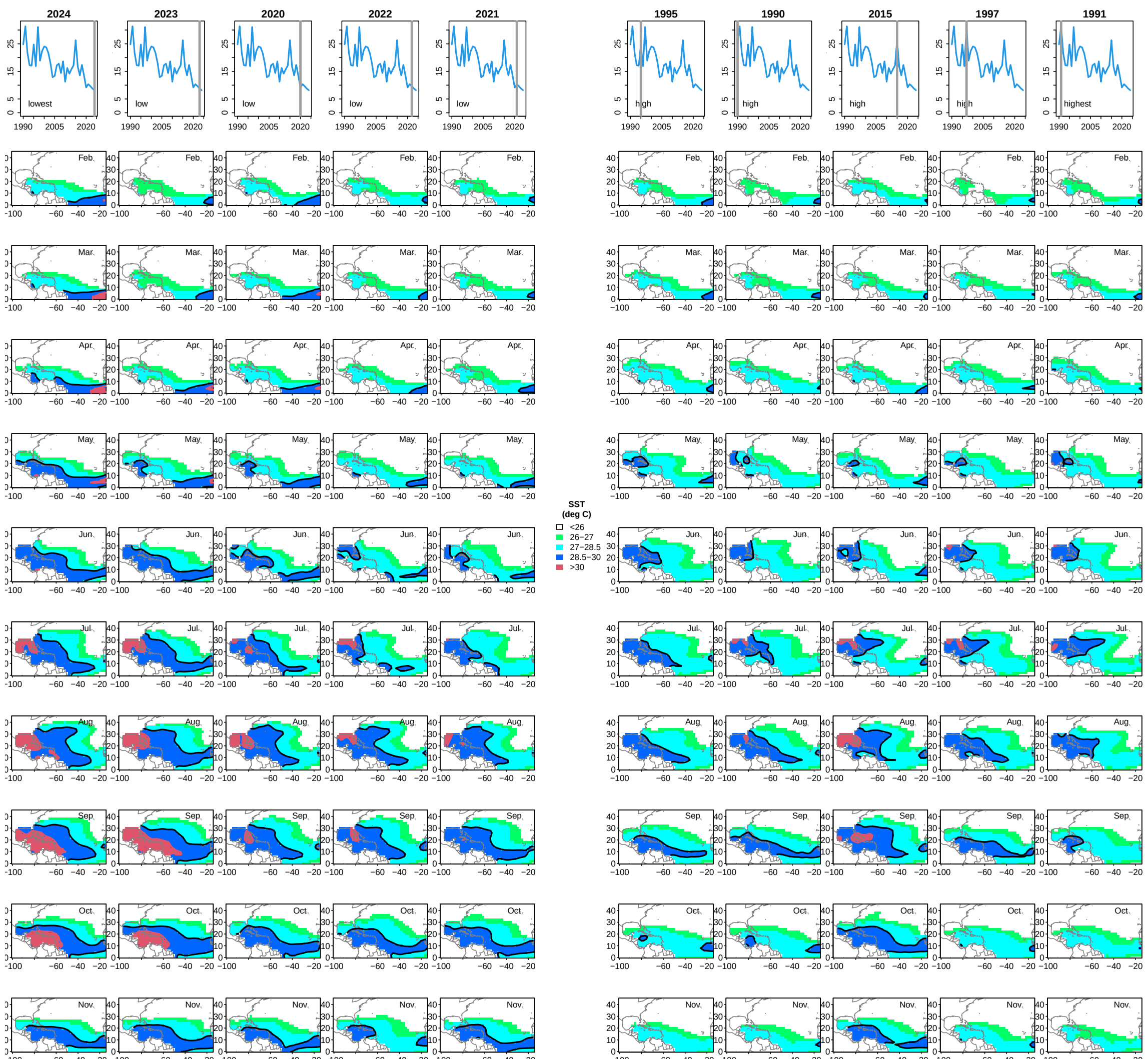
15.2 Analyzing high and low availability years

In order to identify potential temperature patterns that are influential in driving dolphin abundance in the U.S. South Atlantic waters, we identify the years of highest and lowest landings and look at the temperature distributions monthly in those years.

```
# upload recreational landings
rec <- read.csv("data/recLandings.csv")
names(rec)[1] <- "Year"

# summarize by basin and total
rec$ATL <- rowSums(rec[3:6], na.rm = T)
rec$ALL <- rec$GULF + rec$ATL

# consider 1990 and forward
rec <- rec[which(rec$Year >= 1990), ]

# sort years from lowest to higher landings
sortyr <- rec$Year[order(rec$ALL)]
```

```r
# cols <- rainbow(100, start = 0.1, end = 0.65)[100:1]
cols <- c(0, rainbow(3, start = 0.4, end = 0.6), 2)

# set up plot
nf <- layout(matrix(c(2:56, rep(1, 11), 57:111), 11, 11, byrow = FALSE),
             c(rep(3, 5), 2, rep(3, 5)), c(4, rep(3, 11)))
#layout.show(nf)

#legend
par(mar = c(0, 0, 0, 0))
plot(1, type = "n", axes = FALSE, xlab = "", ylab = "")
legend("center", fill = cols, c("<26", "26-27", "27-28.5", "28.5-30", ">30"),
       border = c(1, 0, 0, 0, 0),
       pt.cex = 2, bty = "n",
       title = "SST\n(deg C)", title.font = 2)

# sort years from lowest to highest landings
yrs <- c(sortyr[1:5], sortyr[(length(sortyr)-4): length(sortyr)])
#yrs
status <- c(rep("low", 5), rep("high", 5))
status[1] <- "lowest"
status[10] <- "highest"

# for lowest and highest years, visualize temperature patterns across the Atlantic
for (j in 1: length(yrs))  {
  k <- which(yr == yrs[j])
#  print(data.frame(tim[k], yr[k], mon[k]))
  par(mex = 0.5, mar = c(2, 2, 2, 2)+1)
  plot(rec$Year, rec$ALL/10^6, type = "l", col = 4, ylim = c(0, 32), lwd = 2, ylab = "rec l
       main = yrs[j])
  text(1990, 3, status[j], pos = 4)
  abline(v = yrs[j], col = 8, lwd = 3)

  # plot months February through November
  for (i in k[2:11])  {
    par(mex = 0.3, mar = c(2, 3, 2, 0))
    map("world", xlim = c(-100, -15), ylim = c(0, 45), col = 0)
    axis(1); axis(2, las = 2); box()
    image(lon, lat, sst[, , i], add = T, col = cols, breaks = c(-2, 26, 27, 28.5, 30, 32))
    text(-33, 40, paste(month.abb[mon[i]]), cex = 1)
```

```
    map("world", add = T, col = gray(0.5)); box()
    contour(lon, lat, sst[, , i], levels = c(30), add = T, col = c(2), lwd = 2, drawlabels
    contour(lon, lat, sst[, , i], levels = c(28.5), add = T, col = 1, lwd = 2, drawlabels =
  }
}
```

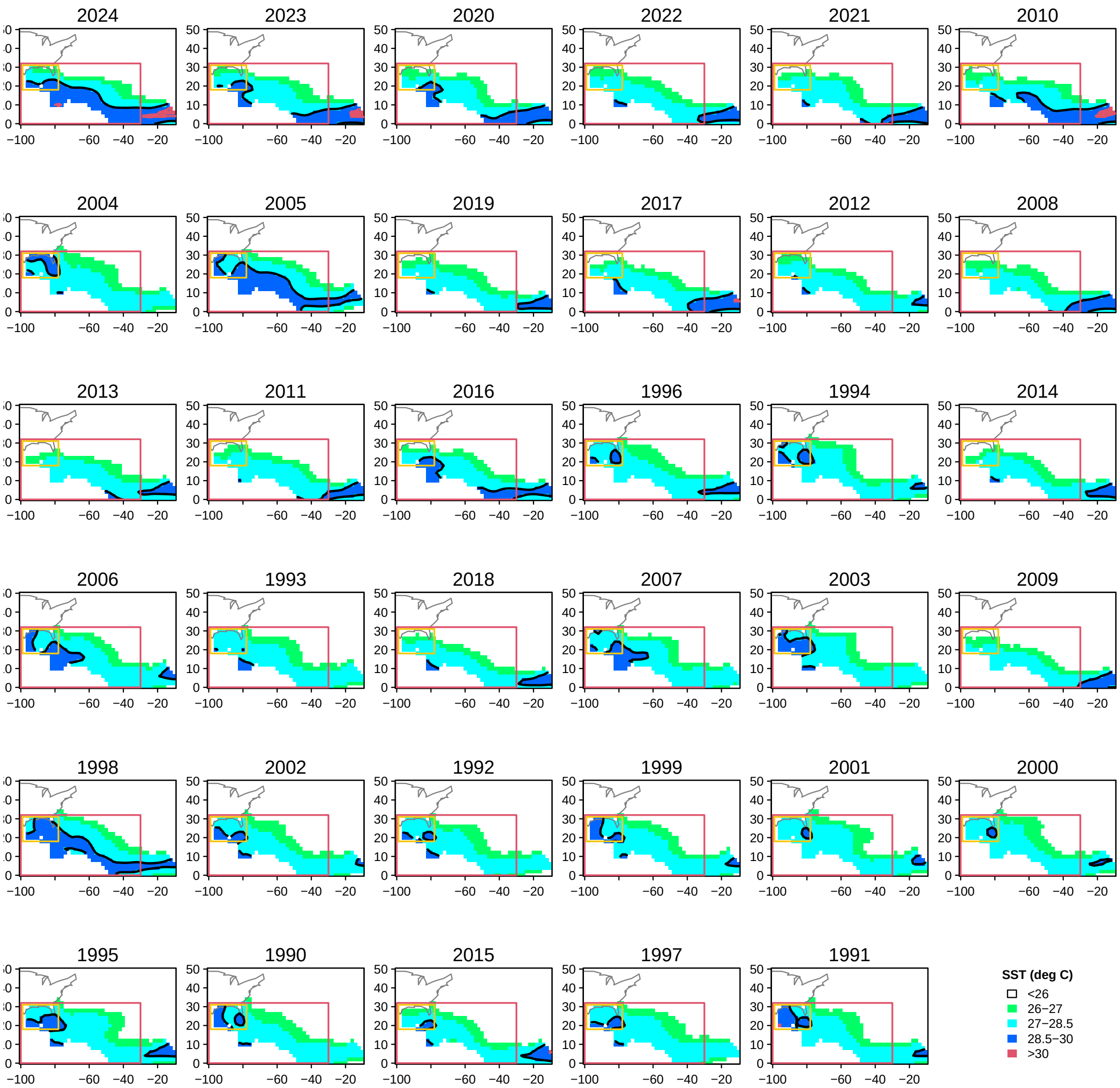


Looking at the set of plots on the left side (low abundance years) versus the right side (high abundance years), there seems to be a particularly notable pattern with respect to temperatures in the range of 28.5° - 30° Celsius. In the low abundance years, temperatures in this range are distributed in a band across the Atlantic near the equator. In the high abundance years, waters near the equator are much cooler and the 28.5° - 30° temperatures appear to be concentrated in a very small area near the Florida Keys. This pattern seems most prominent in the months of May and June.

15.3 May temperature distributions

We will take a closer look at May temperature distributions across the Atlantic for the full time series. Here we display all of the May temperatures from 1990 to present, ordered from lowest to highest years of recreational landings in the U.S. South Atlantic.

```
# look at May patterns --------------------

par(mfrow = c(6, 6), mex = 0.5, mar = c(2, 2, 2, 0))

for (j in 1: length(sortyr))  {
  k <- which(yr == sortyr[j])
  for (i in k[5])  {
    map("world", xlim = c(-100, -10), ylim = c(0, 50), col = 0)
    axis(1); axis(2, las = 2); box()
    image(lon, lat, sst[, , i], add = T, col = cols, breaks = c(-2, 26, 27, 28.5, 30, 32))
    map("usa", add = T, col = gray(0.5)); box()
    mtext(side = 3, line = 0.5, sortyr[j])
    contour(lon, lat, sst[, , i], levels = c(30), add = T, col = c(2), lwd = 2, drawlabels
    contour(lon, lat, sst[, , i], levels = c(28.5), add = T, col = 1, lwd = 2, drawlabels =
    rect(-100, 0, -30, 32, col = NA, border = 2, lwd = 1.5)
    rect(-99, 18, -78, 31, col = NA, border = 7, lwd = 1.5)
  }
}
par(mar = c(0, 0, 0, 0))
plot(1, type = "n", axes = FALSE, xlab = "", ylab = "")
legend("center", fill = cols, c("<26", "26-27", "27-28.5", "28.5-30", ">30"),
       border = c(1, 0, 0, 0, 0),
       pt.cex = 2, bty = "n",
       title = "SST (deg C)", title.font = 2)
```



The pattern with 28.5° - 30° C waters seems to hold across all years, with low abundance years (top) tending to have these waters distributed across the Atlantic, and high abundance years (bottom) tending to have these waters restricted to a small area near the Gulf and U.S. South Atlantic.

15.4 Quantifying the temperature distribution in an index

We can attempt to quantify these patterns in an index by calculating a proportion of waters covered by the dark blue water mass. It seems most useful to consider the coverage of this water mass that occurs in U.S. South Atlantic waters (approximately the yellow box; 100°W-78°W and 18°N-32°N) versus the coverage of this water mass in the larger North Atlantic region (the larger red box; 100°W-30°W and 0°-32°N). We first calculate the number of cells in which these 28.5° - 30° C waters occur in each box, and then divide to obtain an overall proportion.

```
# calculate proportion of 28.5 - 30 degree C waters in US South Atlantic waters

# subset the temperature data to 100W - 30W and 0N - 32N
lons <- which(lon >= (-100) & lon <= (-30))
lats <- which(lat >= (0) & lat <= (32))
lon[lons]
```

```
 [1] -100  -98  -96  -94  -92  -90  -88  -86  -84  -82  -80  -78  -76  -74  -
72
[16]  -70  -68  -66  -64  -62  -60  -58  -56  -54  -52  -50  -48  -46  -44  -
42
[31]  -40  -38  -36  -34  -32  -30
```

```
lat[lats]
```

```
 [1]  0  2  4  6  8 10 12 14 16 18 20 22 24 26 28 30 32
```

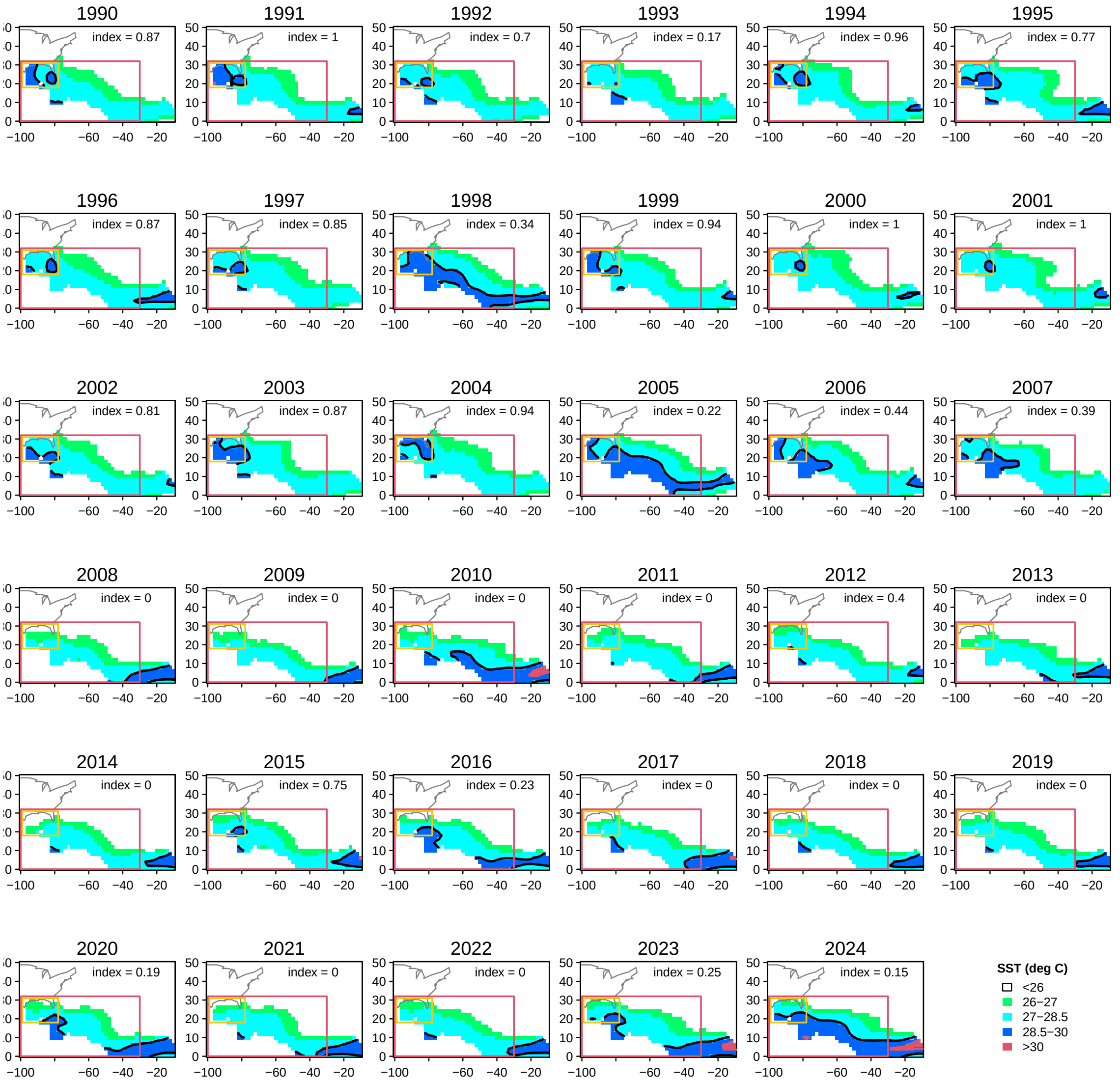
```
# find indexing for 100W - 78W and 18N - 32N
lonind <- which(lon[lons] <= (-78) & lon[lons] >= (-100))
latind <- which(lat[lats] <=32 & lat[lats] >= 18)

# subset SST data for all May months
kar <- sst[lons, lats, which(mon == 5)]

# create empty data frame
ind <- data.frame(matrix(ncol = 2, nrow = dim(kar)[3],
                         dimnames = list(NULL, c("Year", "perar"))))
ind$Year <- yr[which(mon == 5)]
```

```r
# for each year, identify SST cells between 28.5 and 30 degrees
# calculate the proportion of those waters within the small box out of the large box
for (i in 1:dim(kar)[3])  {
  k1 <- kar[, , i]
  k2 <- (k1 >28.5 & k1 < 30.0)
  ind$perar[i] <- length(which(k2[lonind, latind] == TRUE))/ length(which(k2 == TRUE))
}

# replace NA values with zero
ind$perar[which(is.na(ind$perar))] <- 0
ind
```

```
   Year     perar
1  1990 0.8666667
2  1991 1.0000000
3  1992 0.7000000
4  1993 0.1666667
5  1994 0.9629630
6  1995 0.7692308
7  1996 0.8666667
8  1997 0.8500000
9  1998 0.3402778
10 1999 0.9393939
11 2000 1.0000000
12 2001 1.0000000
13 2002 0.8076923
14 2003 0.8723404
15 2004 0.9411765
16 2005 0.2214286
17 2006 0.4444444
18 2007 0.3888889
19 2008 0.0000000
20 2009 0.0000000
21 2010 0.0000000
22 2011 0.0000000
23 2012 0.4000000
24 2013 0.0000000
25 2014 0.0000000
26 2015 0.7500000
27 2016 0.2343750
```

```
28 2017 0.0000000
29 2018 0.0000000
30 2019 0.0000000
31 2020 0.1929825
32 2021 0.0000000
33 2022 0.0000000
34 2023 0.2500000
35 2024 0.1538462
```

```
yrs <- sortyr[order(sortyr)]

par(mfrow = c(6, 6), mex = 0.5, mar = c(2, 2, 2, 0))
for (j in 1:length(yrs))  {
  k <- which(yr == yrs[j])
  for (i in k[5])  {
    map("world", xlim = c(-100, -10), ylim = c(0, 50), col = 0)
    axis(1); axis(2, las = 2); box()
    image(lon, lat, sst[, , i], add = T, col = cols, breaks = c(-2, 26, 27, 28.5, 30, 32))
    map("usa", add = T, col = gray(0.5)); box()
    mtext(side = 3, line = 0.5, yrs[j])
    text(-38, 45, paste("index =", round(ind$perar[which(ind$Year == yrs[j])], 2)))
    contour(lon, lat, sst[, , i], levels = c(30), add = T, col = c(2), lwd = 2, drawlabels
    contour(lon, lat, sst[, , i], levels = c(28.5), add = T, col = 1, lwd = 2, drawlabels =
    rect(-100, 0, -30, 32, col = NA, border = 2, lwd = 1.5)
    rect(-99, 18, -78, 31, col = NA, border = 7, lwd = 1.5)
  }
}
par(mar = c(0, 0, 0, 0))
plot(1, type = "n", axes = FALSE, xlab = "", ylab = "")
legend("center", fill = cols, c("<26", "26-27", "27-28.5", "28.5-30", ">30"),
       border = c(1, 0, 0, 0, 0),
       pt.cex = 2, bty = "n",
       title = "SST (deg C)", title.font = 2)
```

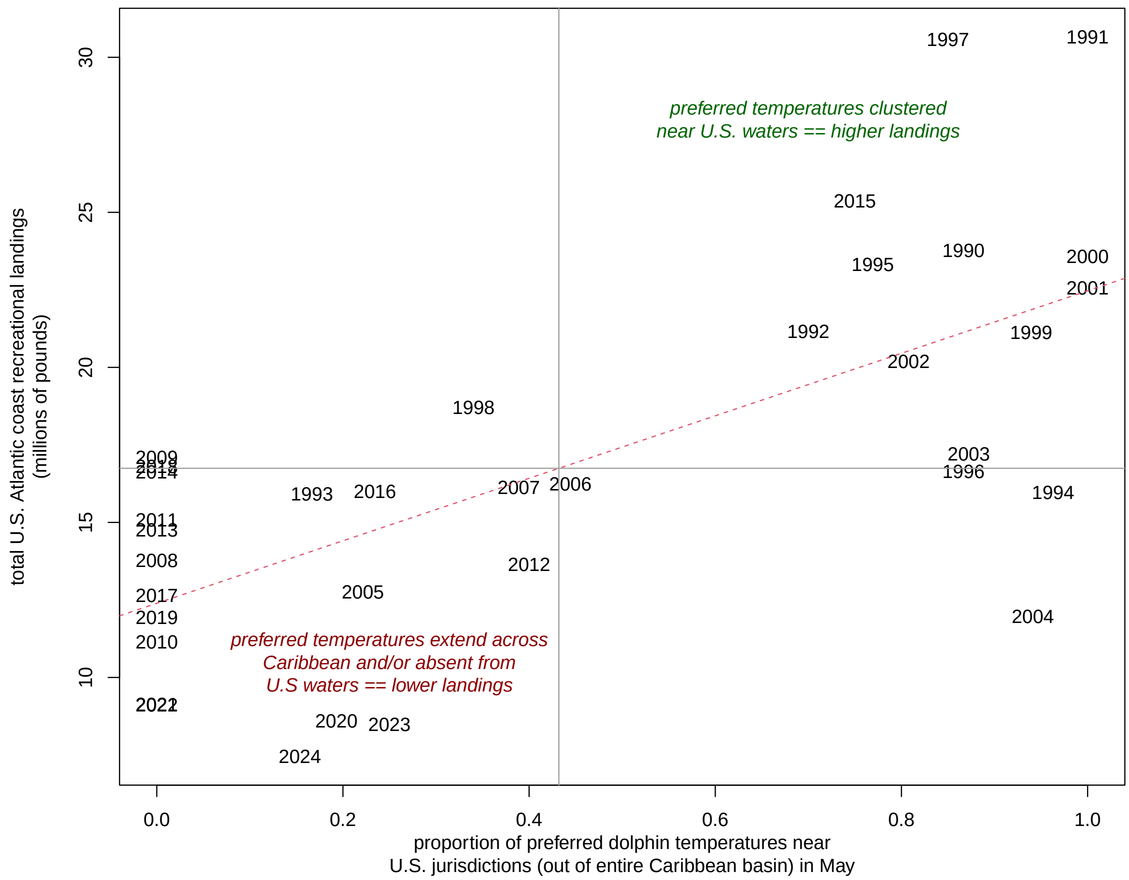


To ensure correct calculation of the index we re-display the May temperature patterns in chronological order and note the index value on each plot.

15.5 Comparison of "blue blob" index with landings

Now we plot the "blue blob" index against the recreational landings.

```
# merge new index with landings data
d <- merge(rec, ind, by = "Year")
#head(d)

# plot the "blue blob" index against recreational landings
par(mar = c(5, 5, 1, 1))
plot(d$perar, d$ATL/10^6, col = 0,
     xlab = "proportion of preferred dolphin temperatures near\nU.S. jurisdictions (out of
     ylab = "total U.S. Atlantic coast recreational landings\n(millions of pounds)")
text(d$perar, d$ATL/10^6, d$Year)
out <- lm(d$ATL/10^6 ~ d$perar)
abline(out, lty = 2, col = 2)
abline(v = mean(d$perar), col = 8)
abline(h = mean(d$ATL)/10^6, col = 8)
text(0.7, 28, "preferred temperatures clustered\nnear U.S. waters == higher landings", col
text(0.25, 10.5, "preferred temperatures extend across\nCaribbean and/or absent from\nU.S w
```



We can see that a statistical relationship does exist, whereby above average "blue blob" index values are associated with above average recreational landings in U.S. South Atlantic waters, and below average index values are associated with below average landings. Gray lines denote the mean values for the index and the landings; we see that this relationship holds true for almost all years in the time series.

The "blue blob" index explains about 45% of the variation in the landings in the U.S. South Atlantic.

```
summary(out)
```

```
Call:
lm(formula = d$ATL/10^6 ~ d$perar)
```

```
Residuals:
    Min      1Q  Median      3Q     Max 
-9.9082 -3.0254  0.0927  2.6638  9.6171 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   12.391      1.094  11.324 6.58e-13 ***
d$perar       10.079      1.880   5.361 6.35e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.339 on 33 degrees of freedom
Multiple R-squared:  0.4655,	Adjusted R-squared:  0.4493 
F-statistic: 28.74 on 1 and 33 DF,  p-value: 6.352e-06
```

```r
# output the index
write.csv(ind, file = "indices/blue_blob_index.csv", row.names = F)
```

16 Puerto Rico tournament index

In this section we consider tournament data from the U.S. Caribbean as a potential predictive index for dolphin abundance in the South Atlantic. Because dolphin migrate through the Caribbean region in the months before arriving in South Atlantic waters, peaks in catch-per-unit effort for fisheries operating in the Caribbean could potentially be indicative of high abundances later in the year in waters downstream of their migration patterns. Tournament data are available from Puerto Rico's Departamento de Recursos Naturales y Ambientales, División de Pesquería Recreativa y Deportiva, going back to the year 2000. Tournament data were requested and received from the institution in June 2022.

16.1 Upload and clean data set

We first input the data set, parse out the dates to extract month and year, and standardize labeling of months.

```
# clear workspace
rm(list = ls())

if(!require("dplyr")) install.packages("dplyr")
if(!require("emmeans")) install.packages("emmeans")

library(dplyr)
library(emmeans) # Best for extracting standardized indices

# import data -----------------------------------
d <- read.csv("data/PRDNER-DolTournamentData.csv")
#apply(d, 2, table)
head(d)
```

```
      Date                      Location Total.Number.Particpants
1 May-6-00 Club N\xe1utico de La Parguera                      124
```

```
2 May-6-00 Club N\xe1utico de La Parguera                    124
3 May-6-00 Club N\xe1utico de La Parguera                    124
4 May-6-00 Club N\xe1utico de La Parguera                    124
5 May-6-00 Club N\xe1utico de La Parguera                    124
6 May-6-00 Club N\xe1utico de La Parguera                    124
  Total.Number.of.Boats Average.Time.Spent.Fishing Tournament.Duration
1                    31                       10.5                   2
2                    31                       10.5                   2
3                    31                       10.5                   2
4                    31                       10.5                   2
5                    31                       10.5                   2
6                    31                       10.5                   2
   Fish.Type   Fish.name Sex Boarded Bycatch Lenght..mm. Weight..Kg.
1 8835290101 Dolphinfish   F    TRUE   FALSE        1675        9.09
2 8835290101 Dolphinfish   F    TRUE   FALSE        1675        6.81
3 8835290101 Dolphinfish   M    TRUE   FALSE        1675       10.45
4 8835290101 Dolphinfish   F    TRUE   FALSE        1625       10.00
5 8835290101 Dolphinfish   F    TRUE   FALSE        1625       12.27
6 8835290101 Dolphinfish   F    TRUE   FALSE        1625       10.00
  Distance.to.coast Zone
1                   2
2                   2
3                   2
4                   2
5                   2
6                   2
```

```r
# clean dates and extract month day year ---------
d$Date <- as.character(d$Date)
d$month <- NA
d$day <- NA
d$year <- NA

for (i in 1:nrow(d))  {# M
  d[i, 16:18] <- unlist(strsplit(d$Date[i], "-"))  }
#head(d)

# clean up errors and check outputs ---------------------
d$month <- substr(d$month, 1, 3)
d$month[which(d$month == "Abr")] <- "Apr"
d$month[which(d$month == "Arp")] <- "Apr"
```

```
d$day[which(d$day == "November" | d$day == "December")] <- NA
d$year[which(d$year == "November" | d$year == "December")] <- NA

d$day <- as.numeric(d$day)
d$year <- as.numeric(d$year)

# standardize year format
d$year[which(d$year < 83)] <- d$year[which(d$year < 83)] + 2000
d$year[which(d$year < 2000)] <- d$year[which(d$year < 2000)] + 1900

d$mon <- match(d$month, month.abb)

table(d$month)
```

```
 Apr  Aug  Dec  Feb  Jan  Jul  Jun  Mar  May  Nov  Oct  Sep
7723   57  614 2360 2160   35   23 7995 1496 1036 1344  273
```

```
table(d$day)
```

```
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
 694  829  792  689  718 1123  618  753  733 1417  493  775 1061  936  278  517
  17   18   19   20   21   22   23   24   25   26   27   28   29   30   31
 894  859  524 1213  726  875  880  655  661  808  854 1192  850  563 1050
```

```
table(d$year)
```

```
1983 1984 1985 1986 1987 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
  80  153    5   53   19  843 1254 1413 1040 1303  944 1411 1418 1209 1190  538
2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022
1263 1815 1321 1153 1512  916  944  808 1213   51   91 1070
```

In keeping with the definition of seasons in other parts of the MSE, we group December with the following January and February in a winter season.

```
# fix so Dec is grouped with following year -----------------
d$mon[which(d$mon == 12)] <- 0.5
d$year2 <- d$year
d$year2[which(d$mon == 0.5)] <- d$year[which(d$mon == 0.5)]+1
rbind(table(d$year), table(d$year2))
```

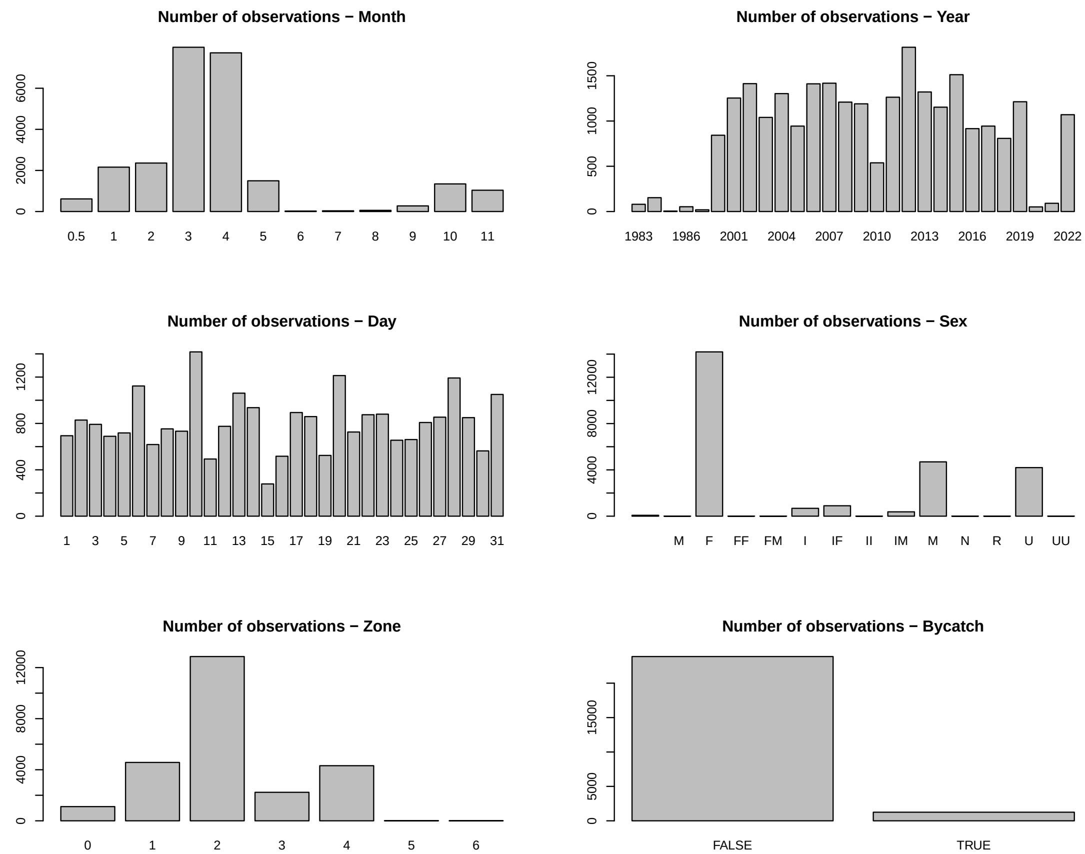
```
     1983 1984 1985 1986 1987 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009
[1,]   80  153    5   53   19  843 1254 1413 1040 1303  944 1411 1418 1209 1190
[2,]   80  153    5   53   19  789 1220 1501 1030 1313  944 1411 1418  863 1536
     2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022
[1,]  538 1263 1815 1321 1153 1512  916  944  808 1213   51   91 1070
[2,]  538 1263 1815 1321 1153 1512  915  945  801 1190   81   85 1076
```

```
#head(d)
#tail(d)
```

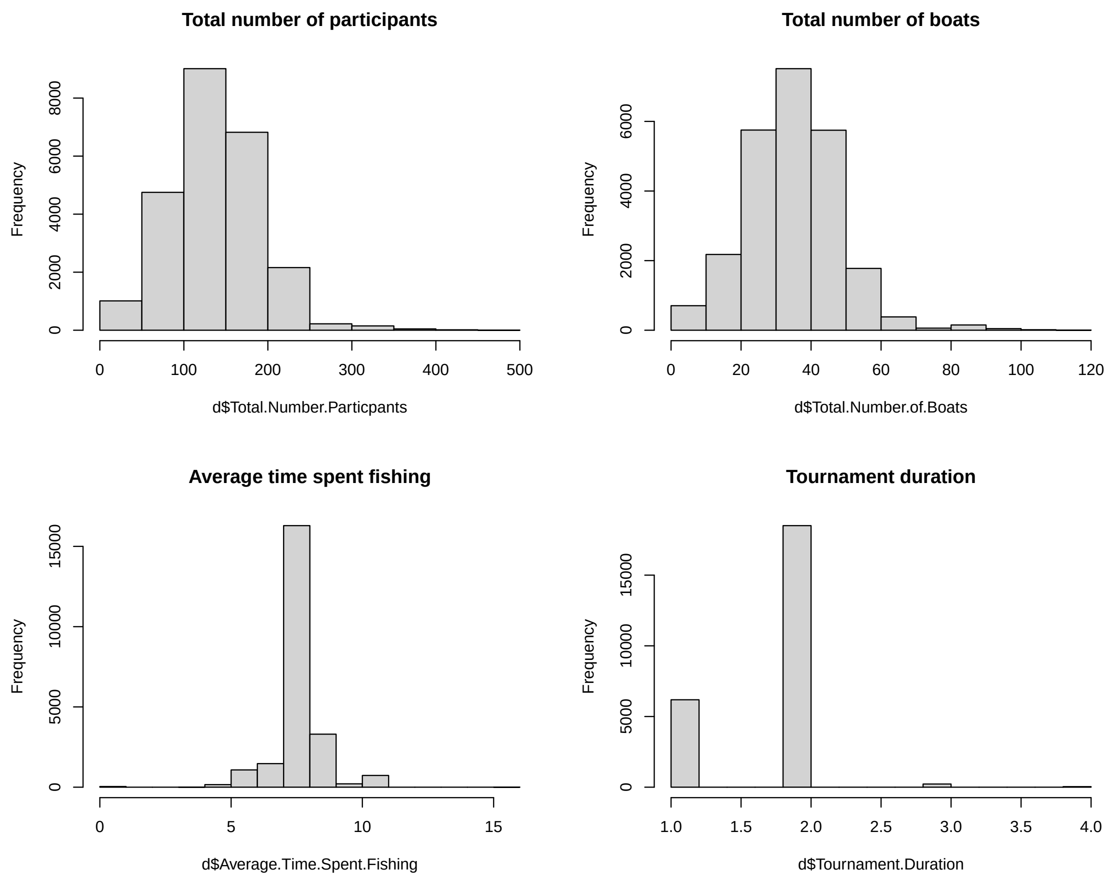
16.2 Viewing the raw data

Let's take a look at some of the data columns using barplots and histograms.

```
# look at data columns ----------------------------------

par(mfrow = c(3, 2))
barplot(table(d$mon), main = "Number of observations - Month")
barplot(table(d$year), main = "Number of observations - Year")
barplot(table(d$day), main = "Number of observations - Day")
barplot(table(d$Sex), main = "Number of observations - Sex")
barplot(table(d$Zone), main = "Number of observations - Zone")
barplot(table(d$Bycatch), main = "Number of observations - Bycatch")
```




```r
par(mfrow = c(2, 2))
hist(d$Total.Number.Particpants, main = "Total number of participants")
hist(d$Total.Number.of.Boats, main = "Total number of boats")
hist(d$Average.Time.Spent.Fishing, main = "Average time spent fishing")
hist(d$Tournament.Duration, main = "Tournament duration")
```



There are a few NAs in the tournament data columns. We need to consider what details are available and what factors we can feasibly use in the standardization process.

```
# find NAs in tournament details -------------------
table(is.na(d$Total.Number.of.Boats))
```

```
FALSE  TRUE
24331   785
```

```
table(is.na(d$Total.Number.Particpants))
```

```
FALSE  TRUE
24189   927
```

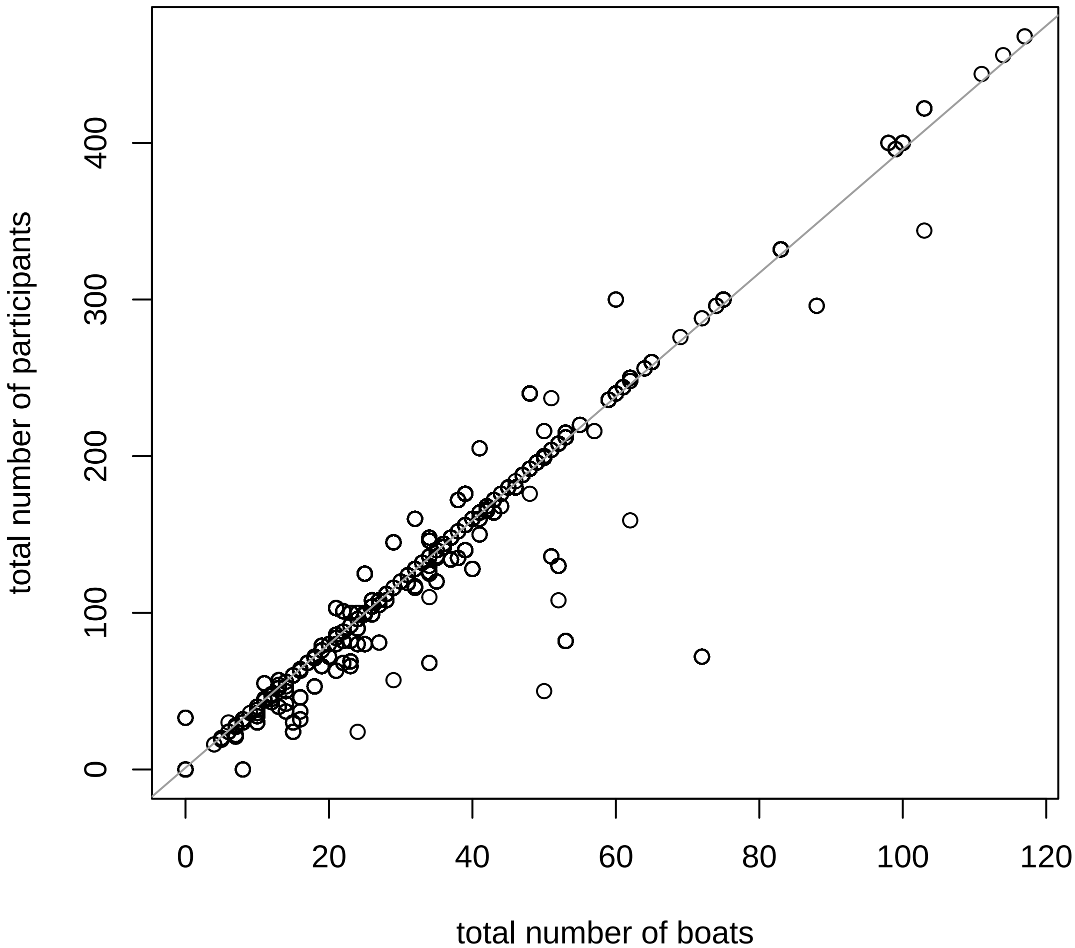
```
table(is.na(d$Total.Number.Particpants) & is.na(d$Total.Number.of.Boats))
```

```
FALSE  TRUE
24332   784
```

```
table(is.na(d$Average.Time.Spent.Fishing))
```

```
FALSE  TRUE
23290  1826
```

```
plot(d$Total.Number.of.Boats, d$Total.Number.Particpants, xlab = "total number of boats", y
out <- lm(d$Total.Number.Particpants ~ d$Total.Number.of.Boats)
abline(out, col = 8)
```



The total number of boats is highly correlated with the total number of participant and the linear regression indicates that most boats have 4 participants.

16.3 Data preparation for standardization

In preparation for creating a standardized catch-per-unit-effort (CPUE) index that can serve as a proxy for abundance, we have to specify nuisance factors in the data set that have to be accounted for. We create a unique tournament ID number with a combination of the day and number of participants.

```
# create unique tournament ID number -------------------------
d$dat <- paste0(d$day, d$month, d$year)
length(unique(paste0(d$day, d$month, d$year, d$Total.Number.Particpants)))
```

```
[1] 386
```

```
length(unique(paste0(d$day, d$month, d$year, d$Location)))
```

```
[1] 387
```

```
length(unique(paste0(d$day, d$month, d$year, d$Total.Number.of.Boats)))
```

```
[1] 386
```

```
length(unique(paste0(d$day, d$month, d$year, d$Tournament.Duration)))
```

```
[1] 366
```

```
d$datID <- paste0(d$day, d$month, d$year, d$Total.Number.Particpants)

#barplot(table(d$mon))
```

We remove all cases where dolphin is not listed as the target species; we will use only directed dolphin trips in the standardization. We also specify the months that we want to include in the index of abundance. Here we specify December to MArch With this subset of months, we have 128 unique tournaments in the data base across 121 different dates.

```
# remove non-target cases; use only directed trips -------------------
dfull <- d
d <- d[which(d$Bycatch == FALSE), ]

# subset by season, e.g. 0 - 4 is December to April -------------------
d <- d[which(d$mon >= 0 & d$mon <= 3), ]
```

```
#length(unique(d$Date))
length(unique(d$dat))
```

```
[1] 121
```

```
length(unique(d$datID))
```

```
[1] 128
```

Next we calculate the total weight by tournament, by summing the reported weights across each tournament ID. We then set up a new data frame where each tournament is a row, and columns represent the attributes of each tournament (year, month, zone, participants, number of boats, fishing time, duration, total weight and total abundance). We also calculate the average weight by dividing total weight by total abundance.

```
# calculate total by tournament ----------------------------
totWT <- tapply(d$Weight..Kg., d$datID, sum, na.rm = T)
totN <- table(d$datID)
mon <- tapply(d$mon, d$datID, mean, na.rm = T)
year <- tapply(d$year2, d$datID, mean, na.rm = T)

# checks on month and tournament assignments ----------------------
#table(d$year[d$mon == 0.5], d$year2[d$mon == 0.5])
#table(d$year[d$mon != 0.5], d$year2[d$mon != 0.5])

max(tapply(d$Total.Number.Particpants, d$datID, sd), na.rm = T)  # should all be zeros
```

```
[1] 0
```

```
max(tapply(d$Total.Number.of.Boats, d$datID, sd), na.rm = T)
```

```
[1] 0
```

```
max(tapply(d$Average.Time.Spent.Fishing, d$datID, sd), na.rm = T)
```

```
[1] 0
```

```r
# assign attributes of tournaments ------------------------------
Npart <- tapply(d$Total.Number.Particpants, d$datID, mean, na.rm = T)
Nboat <- tapply(d$Total.Number.of.Boats, d$datID, mean, na.rm = T)
Ftime <- tapply(d$Average.Time.Spent.Fishing, d$datID, mean, na.rm = T)
dura <- tapply(d$Tournament.Duration, d$datID, mean, na.rm = T)
zone <- tapply(d$Zone, d$datID, mean, na.rm = T)

# new data frame for standardization ---------------------------------
dat <- data.frame(cbind(year, mon, zone, Npart, Nboat, Ftime, dura, totWT, totN))
dat$avwt <- dat$totWT / dat$totN

head(dat)
```

```
             year mon zone Npart Nboat Ftime dura   totWT totN     avwt
10Feb2007140 2007   2    1   140    35   8.0    1  347.64   49 7.094694
10Feb200824  2008   2    1    24    15   7.0    1  264.71   33 8.021515
10Jan2009332 2009   1    1   332    83   8.0    2  770.44  147 5.241088
10Jan2014188 2014   1    1   188    47   7.5    2 1479.82  249 5.943052
10Jan2015136 2015   1    1   136    34   7.5    2  794.72  264 3.010303
10Mar2001192 2001   3    2   192    48   8.0    2 2499.33  272 9.188713
```

There are a number of missing values in this data frame. For fishing time and number of boats, we impute the median of the known values into the missing values. For missing values for the number of boats, we take the number of participants and divide by 4 (given the relationship established above). If both values are missing, we impute the median into the unknown values.

```r
# fill in NAs ----------------------------------------------------
dat$Nboat[which(dat$Nboat == 0)] <- NA
dat$Npart[which(dat$Npart == 0)] <- NA
dat$Ftime[which(dat$Ftime == 0)] <- NA

#hist(dat$Ftime)
which(is.na(dat$Ftime))
```

```
[1]  15  18  68  72  86  94 101 127 128
```

```
dat$Ftime[is.na(dat$Ftime)] <- median(dat$Ftime, na.rm = T)

#hist(dat$Nboat)
which(is.na(dat$Nboat))
```

```
[1]  15  83  94 122 128
```

```
which(is.na(dat$Npart))
```

```
[1]  11  12  15  94 101 110 122 128
```

```
dat$Npart[is.na(dat$Nboat)]
```

```
[1] NaN  33 NaN  NA NaN
```

```
#plot(dat$Nboat, dat$Npart)
#dat$Npart / dat$Nboat
dat$Nboat[is.na(dat$Nboat)] <- dat$Npart[is.na(dat$Nboat)] / 4
#dat$Npart[is.na(dat$Nboat)]
dat$Nboat[is.na(dat$Nboat)] <- median(dat$Nboat, na.rm = T)

which(is.na(dat$Nboat))
```

```
integer(0)
```

```
which(is.na(dat$Ftime))
```

```
integer(0)
```

```
which(is.na(dat$totN))
```

```
integer(0)
```

Finally, we subset the years. Data are very sparse prior to 2000 so we only include the data for 2000 and forward. We calculate the catch-per-unit-effort, defining catch as the total abundance, and effort as the number of boats multiplied by the average time spent fishing.

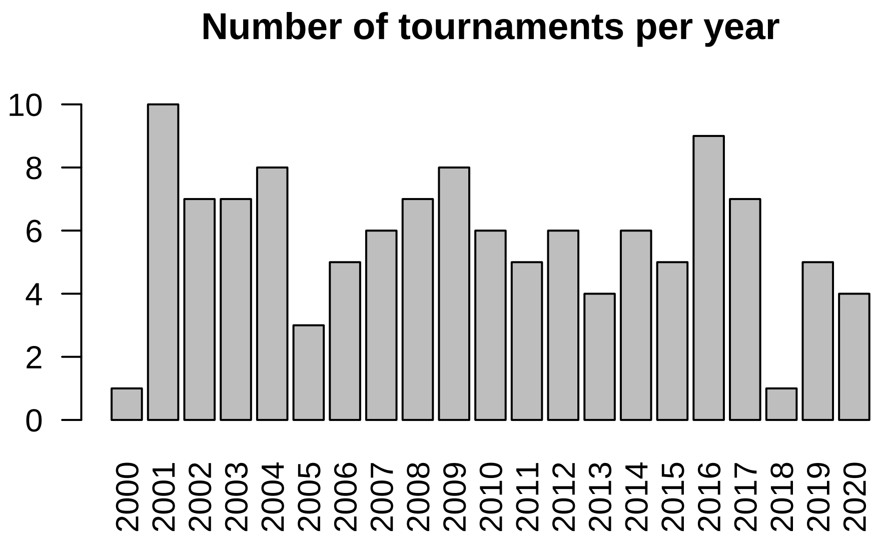
```r
# subset years -----------------------------------

yrs <- 2000:2020

dat[is.na(dat)] <- NA
dat <- dat[which(dat$year >= min(yrs)), ]
dat <- dat[which(dat$year <= max(yrs)), ]

# calculate CPUE -----------------------------------
dat$eff <- dat$Nboat * dat$Ftime   # dat$dura
dat$cpue <- dat$totN / dat$eff
dat$cpueW <- dat$totWT / dat$eff

which(dat$cpue == "Inf")
```

```
integer(0)
```

```r
which(dat$cpueW == "Inf")
```

```
integer(0)
```

```r
#dat$cpue[which(dat$cpue == "Inf")] <- NA

barplot(table(dat$year), las = 2, main = "Number of tournaments per year")
```



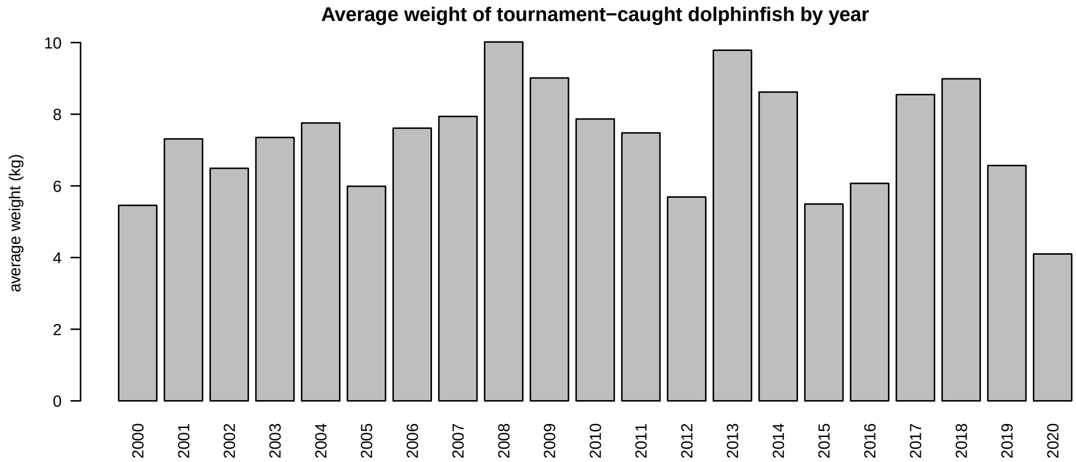
The plot shows the number of tournaments per year. We can see that there are relatively few events for each year.

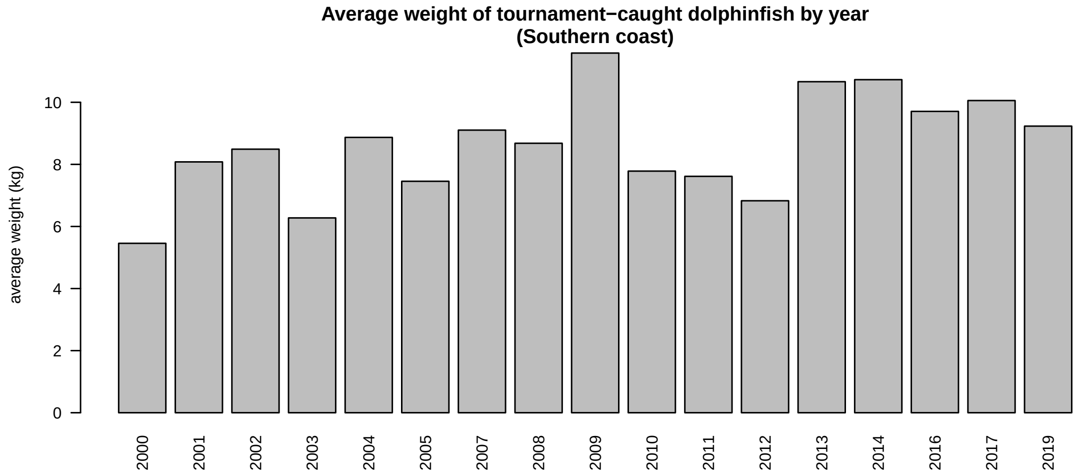
16.4 Analyze the nominal CPUE trends

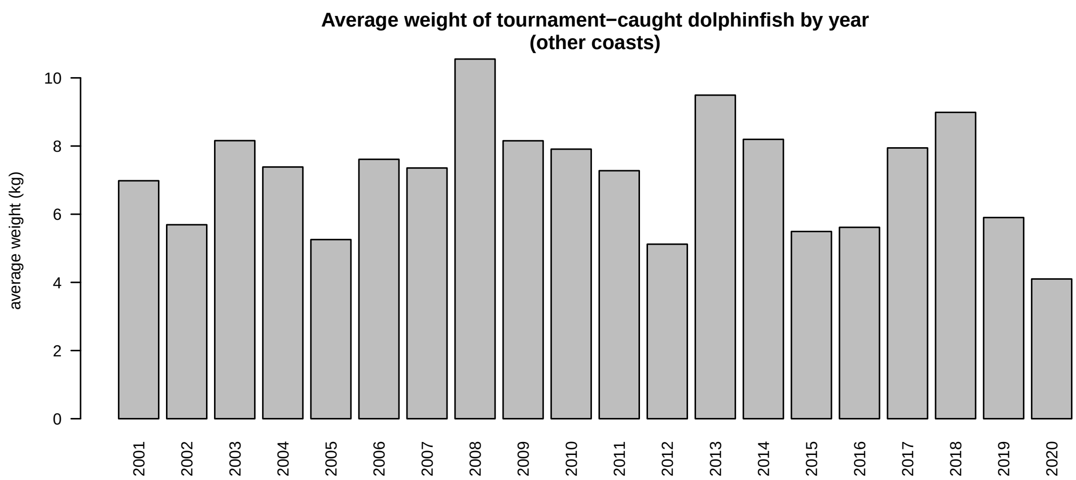
Now using this clean data set we can look at the nominal (average) CPUE by year. We first look at the average weight across all tournaments, and weight by the northern and southern coasts. The weights appear to be highly variable across year and coasts.

```
# look at nominal trends ----------------------------
par(mfrow = c(3, 1), mar = c(3, 5, 3, 1))
barplot(tapply(dat$avwt, dat$year, mean, na.rm = T), las = 2,
        main = "Average weight of tournament-caught dolphinfish by year",
        ylab = "average weight (kg)")

d2 <- dat[which(dat$zone == 2), ]
barplot(tapply(d2$avwt, d2$year, mean, na.rm = T), las = 2,
        main = "Average weight of tournament-caught dolphinfish by year\n(Southern coast)",
        ylab = "average weight (kg)")
```

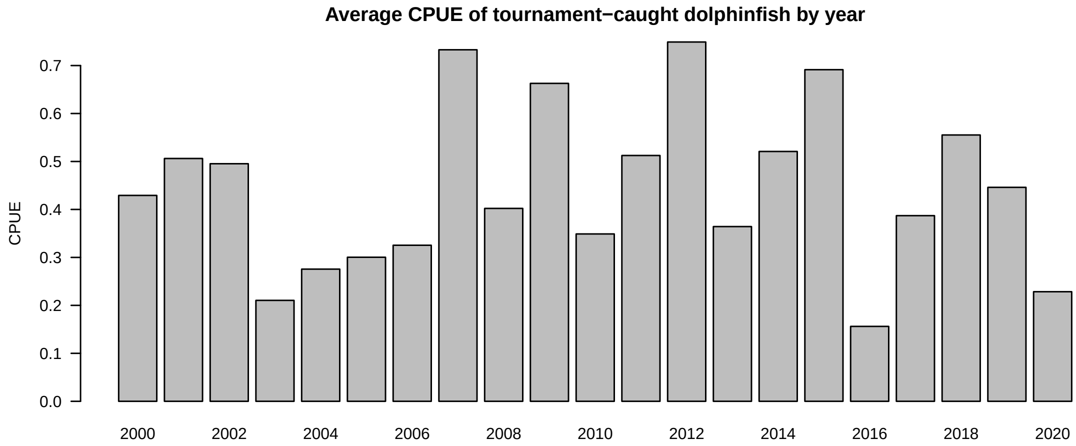
```
d2 <- dat[which(dat$zone != 2), ]
barplot(tapply(d2$avwt, d2$year, mean, na.rm = T), las = 2,
        main = "Average weight of tournament-caught dolphinfish by year\n(other coasts)",
        ylab = "average weight (kg)")
```

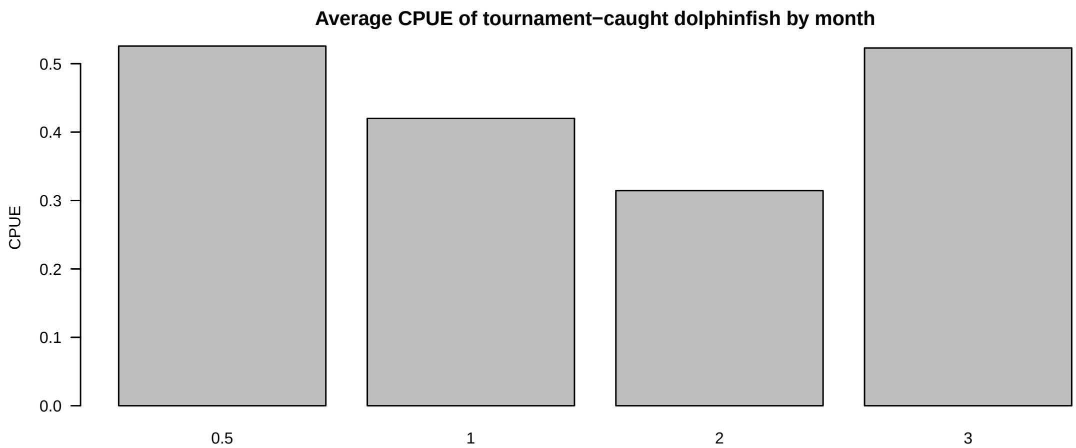

Average weight of tournament−caught dolphinfish by year
average weight (kg)
0
2
4
6
8
10
2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020

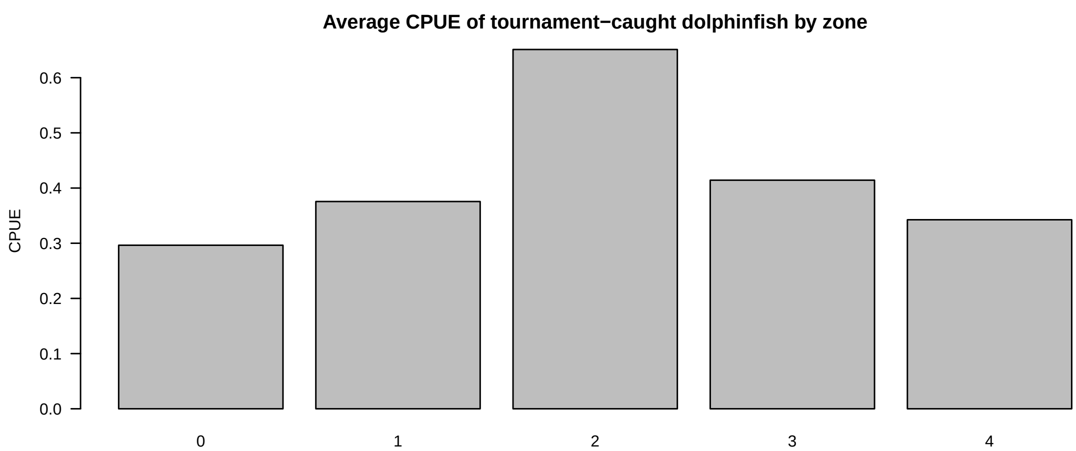

Average weight of tournament−caught dolphinfish by year
(Southern coast)
average weight (kg)
0
2
4
6
8
10
2000 2001 2002 2003 2004 2005 2007 2008 2009 2010 2011 2012 2013 2014 2016 2017 2019


Average weight of tournament−caught dolphinfish by year
(other coasts)
average weight (kg)
0
2
4
6
8
10
2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020

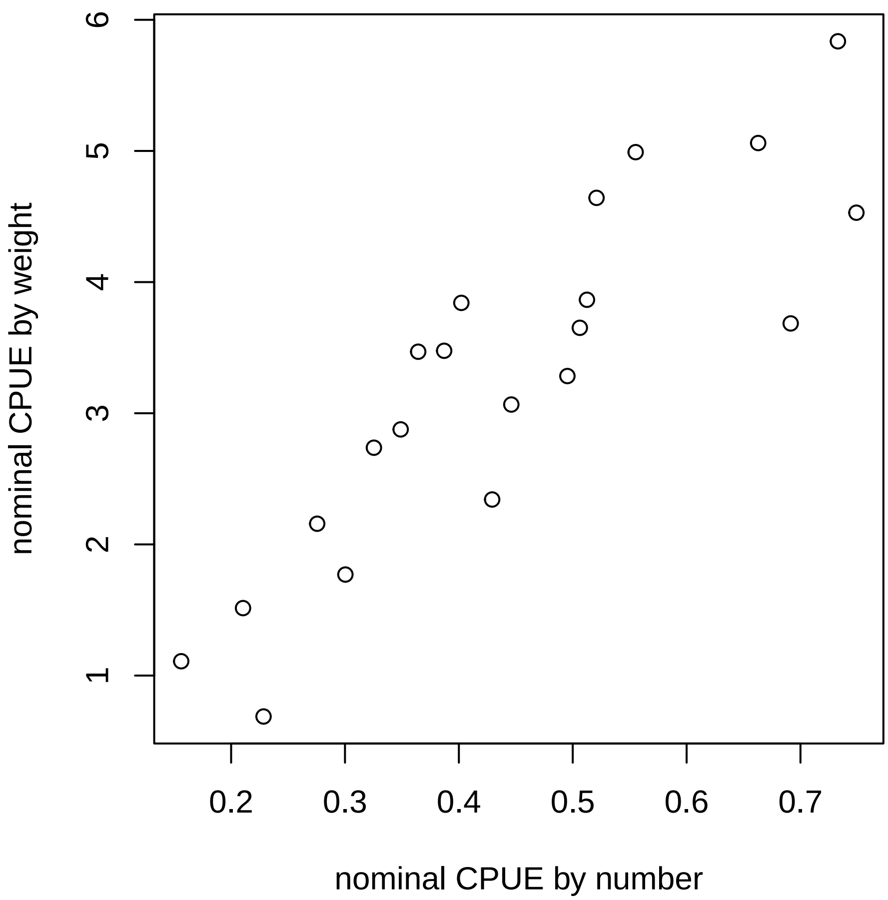
```
# look at nominal trends by year, month and zone
barplot(tapply(dat$cpue, dat$year, mean, na.rm = T), las = 1,
         main = "Average CPUE of tournament-caught dolphinfish by year",
        xlab = "year", ylab = "CPUE")
barplot(tapply(dat$cpue, dat$mon, mean, na.rm = T), las = 1,
        main = "Average CPUE of tournament-caught dolphinfish by month",
        xlab = "month", ylab = "CPUE")
barplot(tapply(dat$cpue, dat$zone, mean, na.rm = T), las = 1,
         main = "Average CPUE of tournament-caught dolphinfish by zone",
        xlab = "zone", ylab = "CPUE")
```


Average CPUE of tournament−caught dolphinfish by year
CPUE
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
2020


Average CPUE of tournament−caught dolphinfish by month
CPUE
0.0
0.1
0.2
0.3
0.4
0.5
0.5
1
2
3


Average CPUE of tournament−caught dolphinfish by zone
CPUE
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
1
2
3
4

```
par(mfrow = c(1, 1), mar = c(5, 5, 1, 1))
plot(tapply(dat$cpue, dat$year, mean, na.rm = T), tapply(dat$cpueW, dat$year, mean, na.rm =
     xlab = "nominal CPUE by number", ylab = "nominal CPUE by weight")
```




```
cor(tapply(dat$cpue, dat$year, mean, na.rm = T), tapply(dat$cpueW, dat$year, mean, na.rm =
```

```
[1] 0.8745617
```

There appears to be high variability in the CPUE by year. The highest CPUE appears to occur in December and March. CPUE also varies by zone, the Southern coast having the highest catch rates. CPUE based on number and weight are highly correlated.

16.5 Calculate standardized CPUE trends

Now we will carry out the standardization, using day, month and zone as standardization factors. Since the catch rates are highly skewed, we carry out the standardization based on the log CPUE.

```
# convert variables to factors ----------------------
dat$year <- as.factor(dat$year)
dat$mon <- as.factor(dat$mon)
dat$zone <- as.factor(dat$zone)

# standardization ----------------------------------

out <- glm(log(dat$cpue) ~ dat$year + dat$mon + dat$zone,
                    family = gaussian)
summary(out)
```

```
Call:
glm(formula = log(dat$cpue) ~ dat$year + dat$mon + dat$zone,
    family = gaussian)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.41897    1.00202  -1.416    0.160
dat$year2001  0.29413    0.84273   0.349    0.728
dat$year2002  0.49382    0.85858   0.575    0.567
dat$year2003 -0.51506    0.85809  -0.600    0.550
dat$year2004 -0.13450    0.85593  -0.157    0.875
dat$year2005 -0.17719    0.93153  -0.190    0.850
dat$year2006  0.04658    0.88927   0.052    0.958
dat$year2007  0.84960    0.86556   0.982    0.329
dat$year2008  0.15122    0.85899   0.176    0.861
dat$year2009  0.56933    0.85373   0.667    0.507
dat$year2010 -0.08845    0.86578  -0.102    0.919
dat$year2011  0.47062    0.87309   0.539    0.591
dat$year2012  0.84258    0.86576   0.973    0.333
dat$year2013  0.34104    0.89906   0.379    0.705
dat$year2014  0.76659    0.88318   0.868    0.388
dat$year2015  1.17262    0.88874   1.319    0.190
```

```
dat$year2016 -0.85525    0.85299  -1.003    0.319
dat$year2017 -0.08506    0.85856  -0.099    0.921
dat$year2018  0.57657    1.17003   0.493    0.623
dat$year2019  0.30635    0.88178   0.347    0.729
dat$year2020 -0.65522    0.91964  -0.712    0.478
dat$mon1     -0.32315    0.39481  -0.819    0.415
dat$mon2     -0.06517    0.38303  -0.170    0.865
dat$mon3     -0.13434    0.40835  -0.329    0.743
dat$zone1     0.13014    0.47370   0.275    0.784
dat$zone2     0.70758    0.50349   1.405    0.163
dat$zone3     0.38841    0.56139   0.692    0.491
dat$zone4    -0.13385    0.48181  -0.278    0.782

(Dispersion parameter for gaussian family taken to be 0.6288276)

    Null deviance: 101.331  on 119  degrees of freedom
Residual deviance:  57.852  on  92  degrees of freedom
AIC: 310.99

Number of Fisher Scoring iterations: 2
```

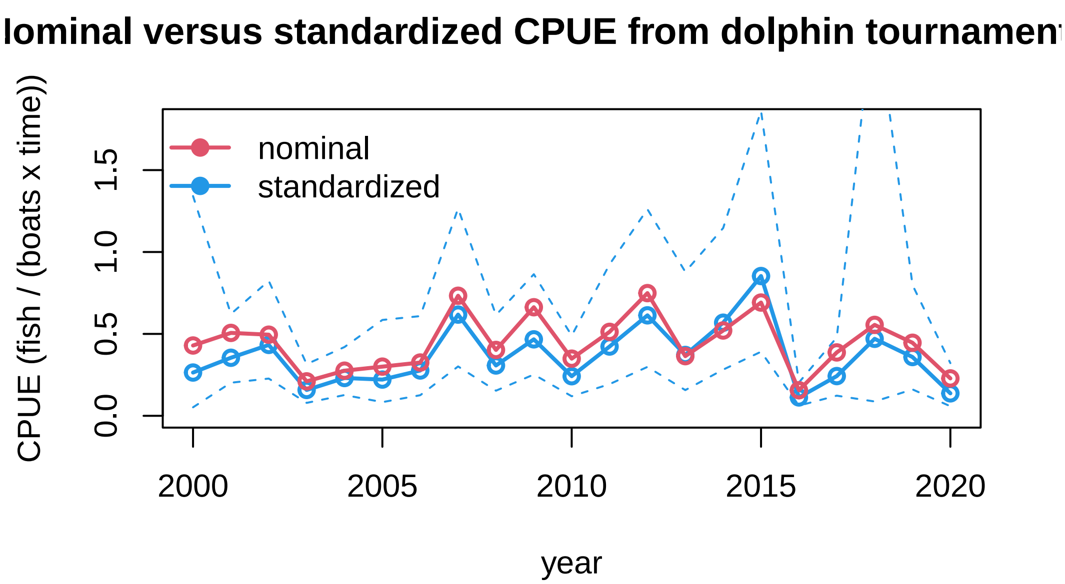
```
anova(out)
```

```
Analysis of Deviance Table

Model: gaussian, link: identity

Response: log(dat$cpue)

Terms added sequentially (first to last)

         Df Deviance Resid. Df Resid. Dev      F   Pr(>F)
NULL                       119    101.331
dat$year 20   32.476        99     68.855 2.5823 0.001182 **
dat$mon   3    2.382        96     66.473 1.2628 0.291846
dat$zone  4    8.620        92     57.852 3.4272 0.011697 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# estimated coefficients from standardization
ind <- emmeans(out, "year", type = "response") %>% as.data.frame()
names(ind)[1] <- "Year"
```

Finally, we compare the standardized CPUE with the nominal CPUE. The two time series are somewhat closely related, indicating similar years of low and high relative abundance.

```
# look at nominal vs standardized index ------------------------
plot(yrs, ind$response, type = "l", lwd = 2, col = 4, main = "Nominal versus standardized C
     xlab = "year", ylab = "CPUE (fish / (boats x time))", ylim = c(0, 1.8))
points(yrs, ind$response, pch = 1, lwd = 2, col = 4)
lines(yrs, ind$lower.CL, col = 4, lty = 2)
lines(yrs, ind$upper.CL, col = 4, lty = 2)
lines(yrs, tapply(dat$cpue, dat$year, mean, na.rm = T) *1.0, col = 2, lwd = 2)
points(yrs, tapply(dat$cpue, dat$year, mean, na.rm = T) *1.0, col = 2, lwd = 2, pch = 1)
legend("topleft", c("nominal", "standardized"), lwd = 2, pch = 19, col = c(2, 4), bty = "n"
```



```
cor(ind$response, tapply(dat$cpue, dat$year, mean, na.rm = T))
```

```
[1] 0.9064402
```

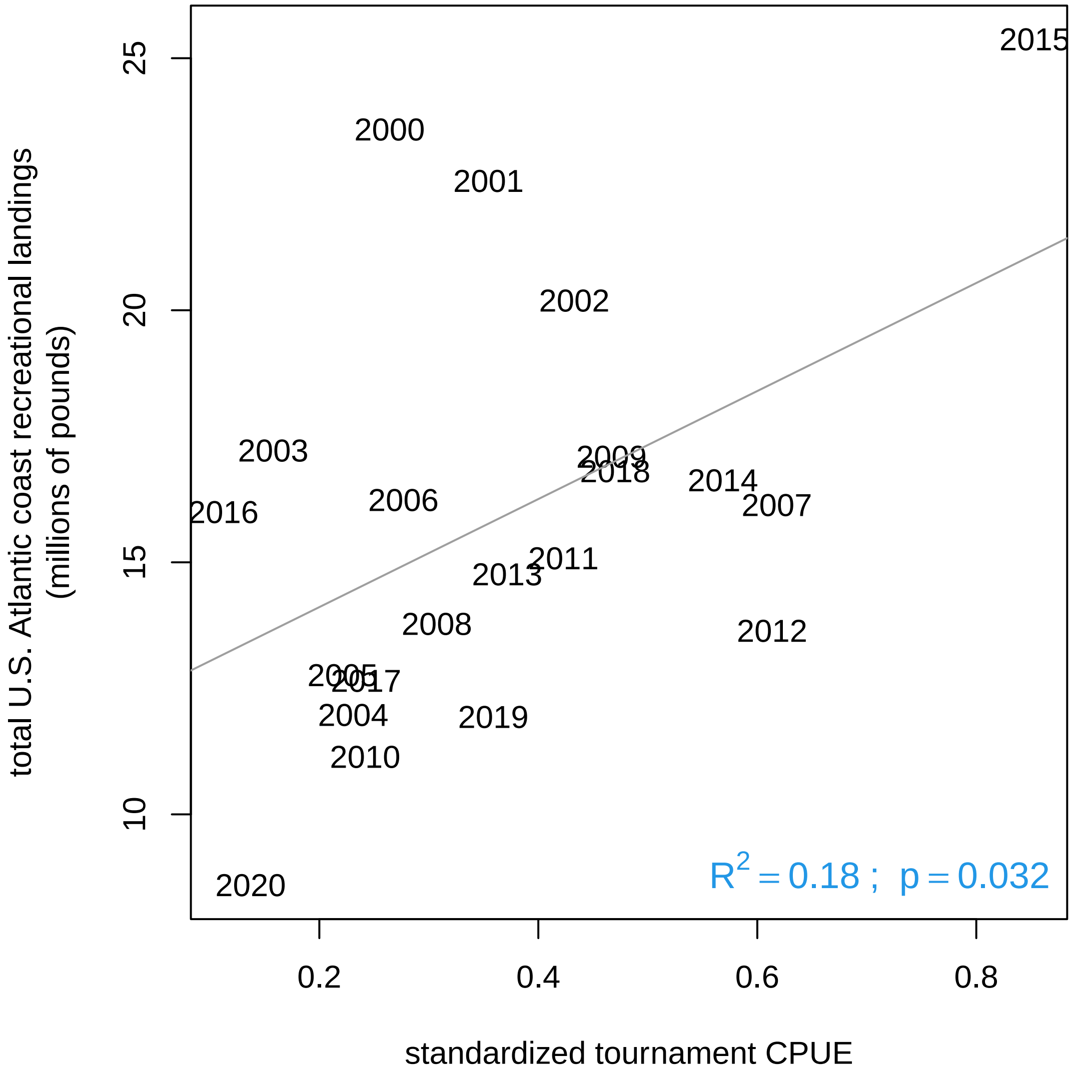
Finally, let's look at how our standardized index compares to the South Atlantic landings. We experimented with CPUE in both units of abundance and weight, and for different combinations of winter months (e.g., Dec - Feb, Jan - Apr). The highest correlation with South Atlantic landings occurs with tournament data from December to March with CPUE based on abundance. In this case the tournament data describes 18% of the variation in South Atlantic landings, with 2015 as a well above-average outlier in both the tournament CPUE and the landings.

```
rec <- read.csv("data/recLandings.csv")
names(rec)[1] <- "Year"
rec$ATL <- rowSums(rec[3:6], na.rm = T)

d <- merge(rec, ind, by = "Year")

par(mar = c(4, 6, 2, 1))
plot(d$response, d$ATL/10^6, col = 0, xlab = "standardized tournament CPUE",
     ylab = "total U.S. Atlantic coast recreational landings\n(millions of pounds)")
text(d$response, d$ATL/10^6, d$Year, col = 1)
out <- lm(d$ATL/10^6 ~ d$response)
abline(out, col = 8)
summary(out)
```

```
Call:
lm(formula = d$ATL/10^6 ~ d$response)

Residuals:
   Min     1Q Median     3Q    Max
-4.904 -2.455 -1.423  2.822  8.784

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   11.968      1.892   6.326 4.52e-06 ***
d$response    10.716      4.618   2.320   0.0316 *
---
```

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

Residual standard error: 3.819 on 19 degrees of freedom
Multiple R-squared:  0.2208,    Adjusted R-squared:  0.1798
F-statistic: 5.385 on 1 and 19 DF,  p-value: 0.0316
```

```
r2 <- summary(out)$adj.r.squared
p_val <- summary(out)$coefficients[2, 4]
p_display <- ifelse(p_val < 0.001, "p < 0.001", paste("p =", round(p_val, 3)))
legend("bottomright",
       legend = bquote(R^2 == .(round(r2, 2)) ~ "; " ~ p == .(round(p_val, 3))),
       bty = "n", cex = 1.2, text.col = 4)
```




```
# output the index
write.csv(ind, file = "indices/PRtournament_index.csv", row.names = F)
```

17 Wintertime pelagic longline index

A substantial portion of the U.S. pelagic longline fishery operates in the Caribbean region, and we can analyze the catch rates in this region as a potential predictive index of dolphin abundance in the South Atlantic later in the year. The logbook data are confidential and cannot be shown here, but we show the results of the standardization process. The code for processing the pelagic longline logbook data can be seen in the file "PLL_index.R" and the steps are outlined below.

17.1 Analysis of logbook data

The goal of the analysis is to create a standardized index of relative abundance for dolphin, using wintertime catch rates from the pelagic longline (PLL) fleet. The standardization process removes the potential influence of nuisance factors from the catch data, such as species targeting or differences in gear deployment, so that the index represents the underlying abundance of the fish.

The steps are as follows:

1. Read in the pelagic longline data which contains set level entries including location of fishing, target species, gear used, other details of the fishing trip and number of dolphin caught, discarded alive, discarded dead, and the total pounds of dolphin kept.
2. Format the longitude and latitude of the sets into decimal degrees, and find the points that fall in the Caribbean region (we use 76°W - 60°W and 12°N - 23°N as boundaries). Check that subsetting was done correctly, and extract the Caribbean data from the data set.
3. Inspect the dolphin catch columns (number of dolphin kept, discarded alive and dead). Replace NA values with zeros. Calculate the total dolphin catch as the total kept and discarded. Remove data for which the number of hooks is not reported. Calculate the catch rate as total dolphin caught divided by 1,000 hooks.

4. Standardize date formatting and add identifiers for month. Index December so that it is included with the winter months of the following year. Subset the data set so that it includes only the winter months (December - February).

5. Explore the different factors to include in the standardization model, relying on previous literature and best practices for standardizing PLL logbook data. Analyze the number of observations for different gear configurations and target species. Factors considered were: temperature, month, number of hooks between floats, bait type, number of lights between hooks, and species targeted. Temperature was binned in 5-degree increments. Month and hooks between floats were converted to factors. There was little resolution in the data for bait type and number of lights per hook, with similar values across all observations. The only major reported species targeted were dolphin, swordfish and mixed; targeting for other species were only listed in a small number of sets.

6. Use the delta-lognormal method to create an index of abundance for dolphin, estimated using a generalized linear model approach. The delta model fits separately the: 1) proportion of positive sets, assuming a binomial error distribution, and 2) the mean catch rate of sets where at least one dolphin was caught, assuming a lognormal error distribution. Use a step-wise regression process was used to determine the set of factors that significantly explain the observed variability. Model selection of fixed factors was based on significance with alpha=0.05. Once the final model is determined, calculate the least-squares means from the linear models for the year predictor estimates. The index of abundance is the standardized proportion positive by year multiplied by the standardized abundance when present.

7. Calculate the deviance explained by each model, the standard errors, and the combined variance for the standardized index.

17.2 Model results

Here we upload the model results and show the standardized index.

```r
load("data/linear_model_outputs.RData")

# Print results
cat("Deviance Explained (Binomial): ", round(d2_bin * 100, 2), "%\n")
```

```
Deviance Explained (Binomial):  7.75 %
```

```r
cat("Deviance Explained (Lognormal):", round(d2_log * 100, 2), "%\n")
```

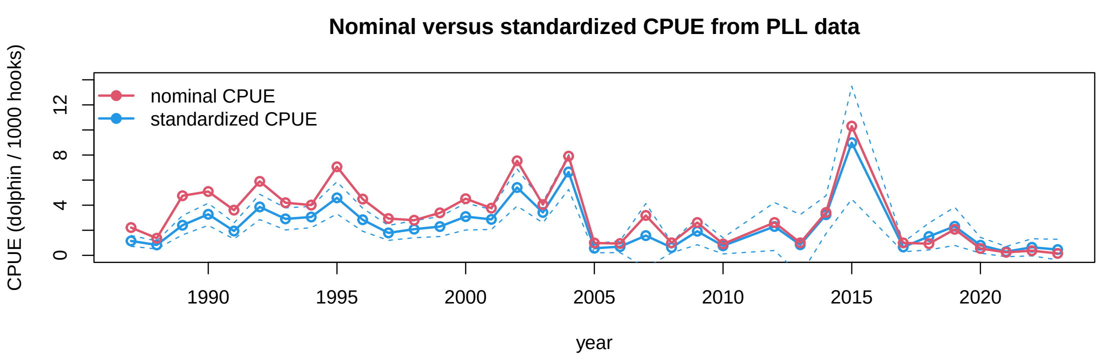
```
Deviance Explained (Lognormal): 17.28 %
```

```r
cat("Total Delta-Lognormal D^2: ", round(total_d2 * 100, 2), "%\n")
```

```
Total Delta-Lognormal D^2:  9.63 %
```

The deviance explained by the model is relatively low, and thus the standardized index is similar to the nominal CPUE index. The primary impact of standardization is to lower slightly the index of relative abundance in the earlier years of the time series.

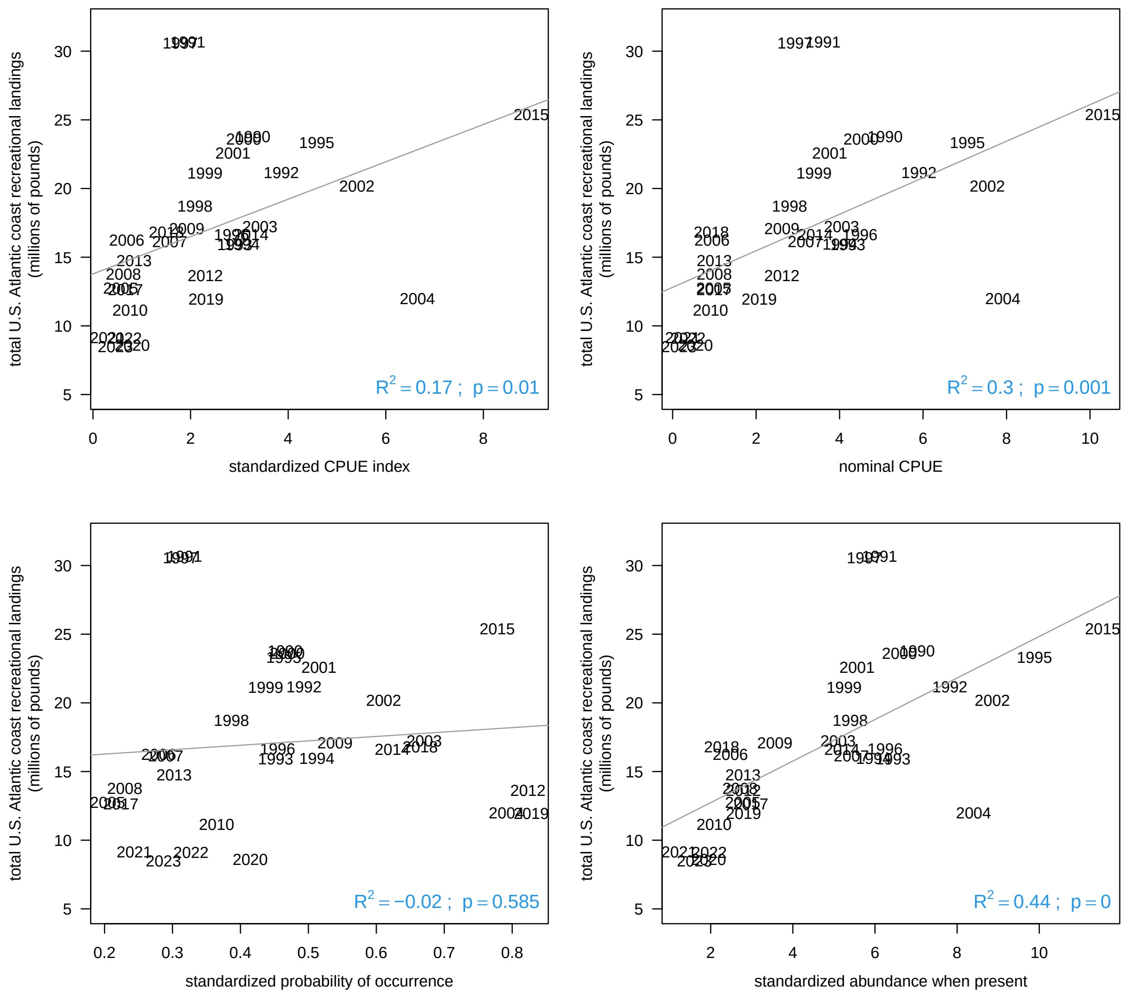
```r
ind <- read.csv("indices/PLL_index.csv")

plot(ind$Year, ind$index, type = "l", lwd = 2, col = 4, main = "Nominal versus standardized
     xlab = "year", ylab = "CPUE (dolphin / 1000 hooks)", ylim = c(0, 14))
points(ind$Year, ind$index, pch = 1, lwd = 2, col = 4)
lines(ind$Year, ind$index - 1.96*ind$SE, col = 4, lty = 2)
lines(ind$Year, ind$index + 1.96*ind$SE, col = 4, lty = 2)
lines(ind$Year, ind$nominal, col = 2, lwd = 2)
points(ind$Year, ind$nominal, col = 2, lwd = 2, pch = 1)
legend("topleft", c("nominal CPUE", "standardized CPUE"), lwd = 2, pch = 19, col = c(2, 4),
```



Now we compare the nominal and the standardized index with the Atlantic recreational landings. We experimented with altering the forecasting time period (e.g., Dec - Feb versus Jan - Mar) and also with the latitudinal boundaries (e.g., 70°W versus 76°W).

A smaller bounding box of 70°W resulted in very small sample sizes (<10) for some individual years; extending the bounding box to 76°W to cover fishing activity between Cuba and Hispanola resulted in much higher sample sizes and improved correlations. Including only the winter months (Dec - Feb) also resulted in higher correlations than when spring months were included.

```r
rec <- read.csv("data/recLandings.csv")
names(rec)[1] <- "Year"
rec$ATL <- rowSums(rec[3:6], na.rm = T)
rec <- rec[which(rec$Year >= 1990), ]

d1 <- merge(rec, ind, by = "Year")

lis <- c("index", "nominal", "predpos", "Npres")
labs <- c("standardized CPUE index", "nominal CPUE",
          "standardized probability of occurrence",
          "standardized abundance when present")

par(mfrow = c(2, 2), mar = c(4, 5, 2, 1), mgp = c(2.5, 1, 0))

for (i in 1:4) {
  j <- which(names(d1) == lis[i])
  plot(d1[, j], d1$ATL/10^6, col = 0, xlab = labs[i], ylim = c(5, 32), las = 1,
       ylab = "total U.S. Atlantic coast recreational landings\n(millions of pounds)")
  text(d1[, j], d1$ATL/10^6, substr(d1$Year, 1, 4), col = 1)
  out <- lm(d1$ATL/10^6 ~ d1[, j ])
  abline(out, col = 8)
  summary(out)

  r2 <- summary(out)$adj.r.squared
  p_val <- summary(out)$coefficients[2, 4]
  p_display <- ifelse(p_val < 0.001, "p < 0.001", paste("p =", round(p_val, 3)))
  legend("bottomright",
       legend = bquote(R^2 == .(round(r2, 2)) ~ "; " ~ p == .(round(p_val, 3))),
       bty = "n", cex = 1.2, text.col = 4)
}
```



Interestingly, the nominal CPUE is more highly correlated with Atlantic recreational landings than the standardized index. However when the standardized index is broken into its two components, we can see that the probability of occurrence is not significantly correlated with the Atlantic landings, while the abundance when present has a highly significant correlation and explains 44% of the variation in the Atlantic landings.