Bluefish in the NMFS bottom trawl

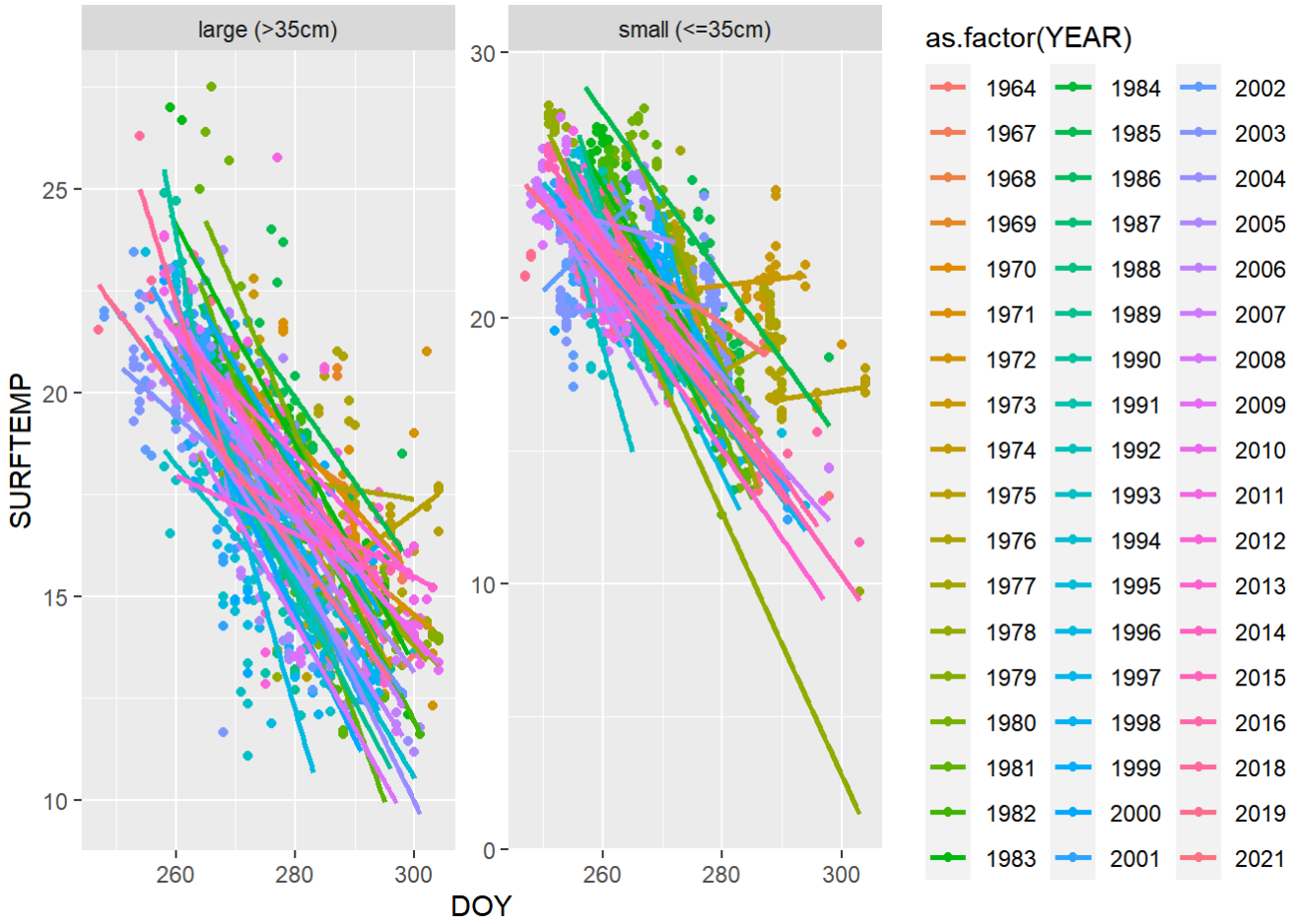
Abigail Tyrell

2022-07-14

0.1 Tow date, latitude, longitude, and surface temperature vary by year

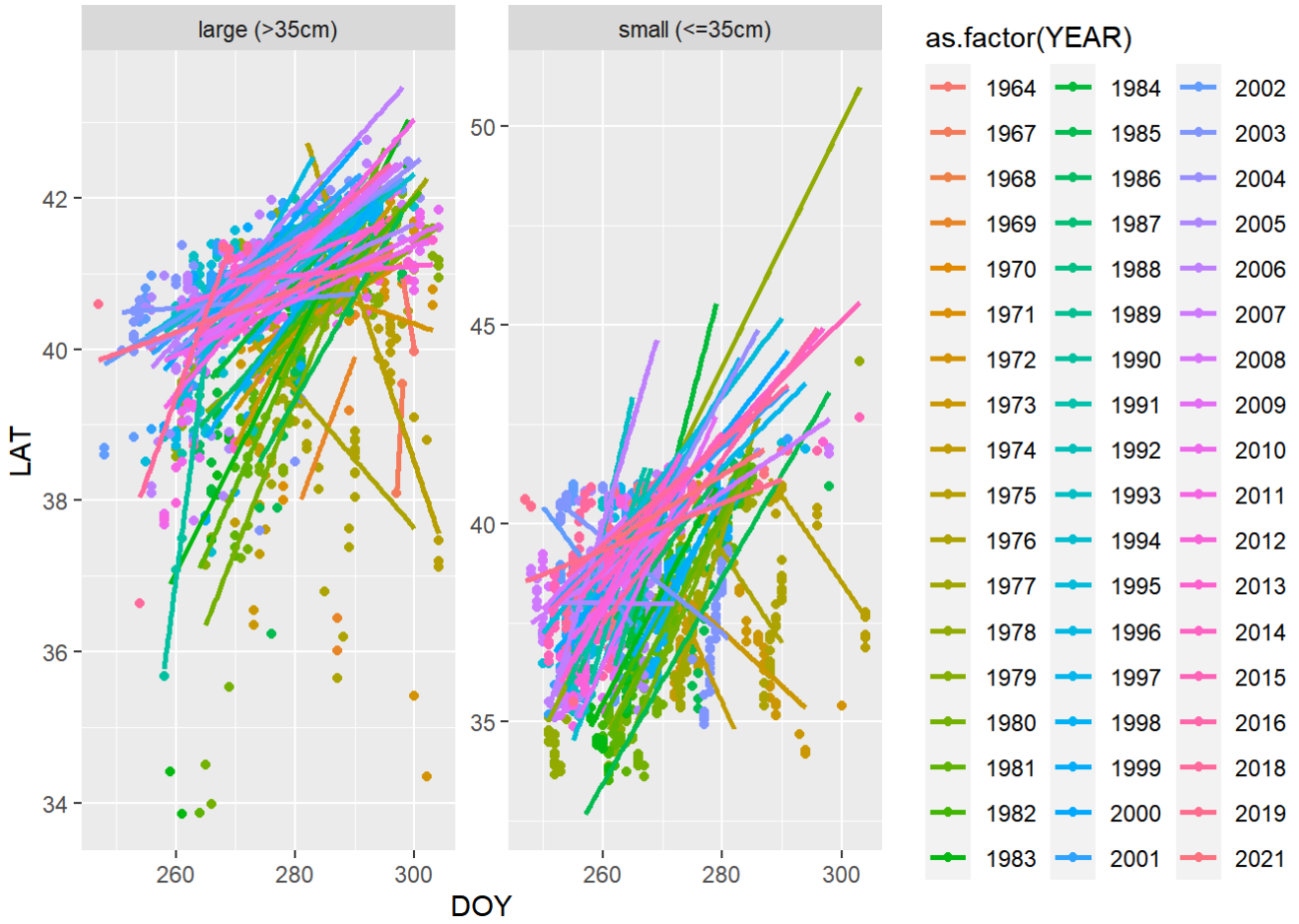
```
dat <- bluefish_nmfs %>%
  dplyr::mutate(DOY = lubridate::yday(EST_TOWDATE),
                size_group = ifelse(LENGTH <= 35,
                                    "small (<=35cm)",
                                    "large (>35cm)")) %>%
  tidyr::drop_na(SURFTEMP) %>%
  dplyr::filter(SURFTEMP > 0,
                SEASON == "FALL",
                DOY < 305) # don't include November

dat %>%
  ggplot2::ggplot(ggplot2::aes(x = DOY,
                               y = SURFTEMP,
                               color = as.factor(YEAR))) +
  ggplot2::geom_point() +
  ggplot2::geom_smooth(method = "lm", se = FALSE) +
  ggplot2::facet_wrap(~size_group, scales = "free")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



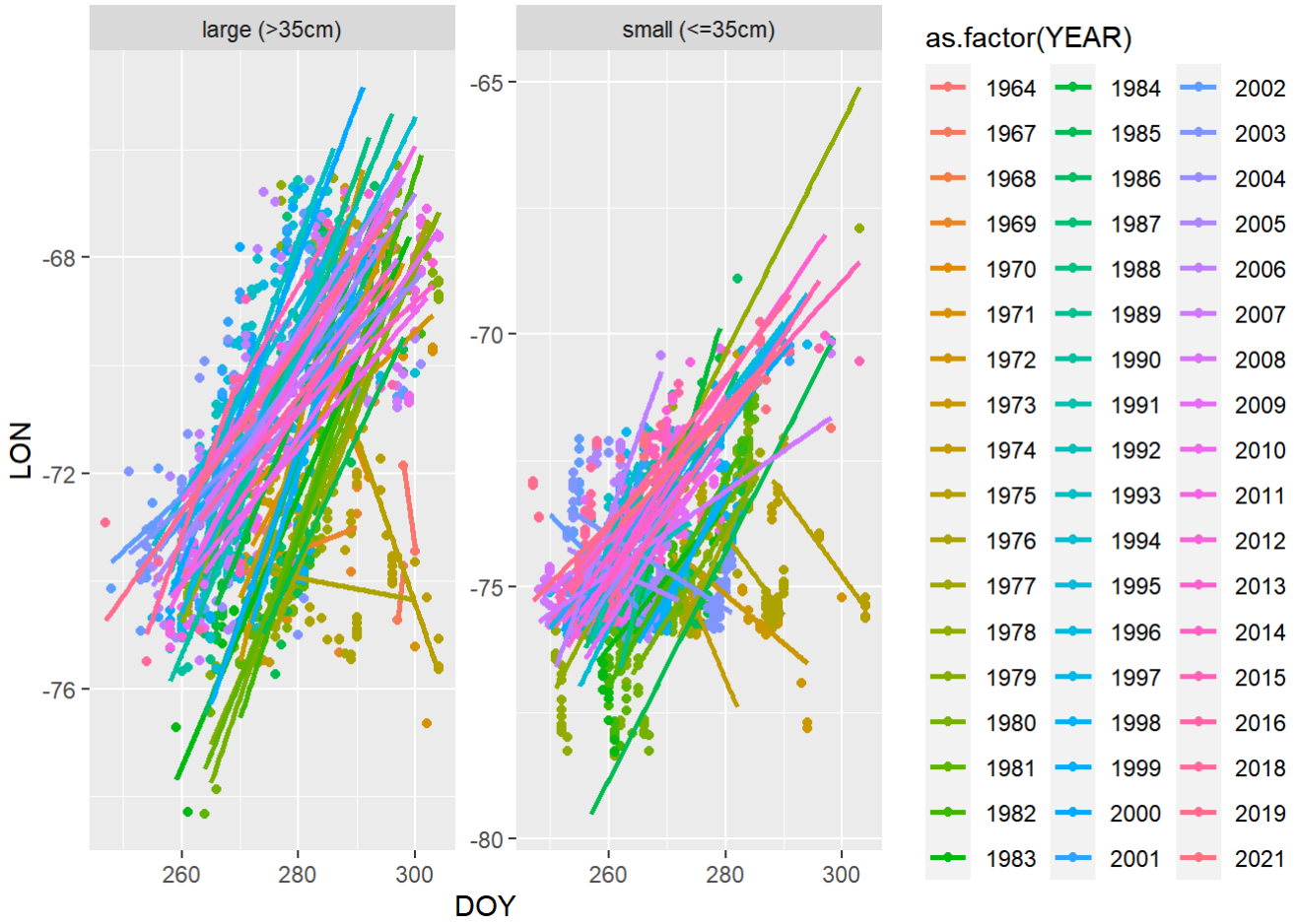

```
dat %>%
  ggplot2::ggplot(ggplot2::aes(x = DOY,
                               y = LAT,
                               color = as.factor(YEAR))) +
  ggplot2::geom_point() +
  ggplot2::geom_smooth(method = "lm", se = FALSE) +
  ggplot2::facet_wrap(~size_group, scales = "free")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
dat %>%
  ggplot2::ggplot(ggplot2::aes(x = DOY,
                               y = LON,
                               color = as.factor(YEAR))) +
  ggplot2::geom_point() +
  ggplot2::geom_smooth(method = "lm", se = FALSE) +
  ggplot2::facet_wrap(~size_group, scales = "free")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
library(mgcv)
```

```
## Loading required package: nlme
```

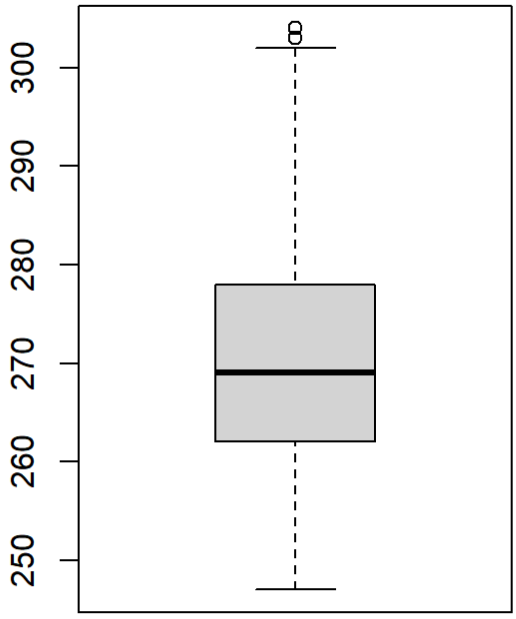
```
##
## Attaching package: 'nlme'
```

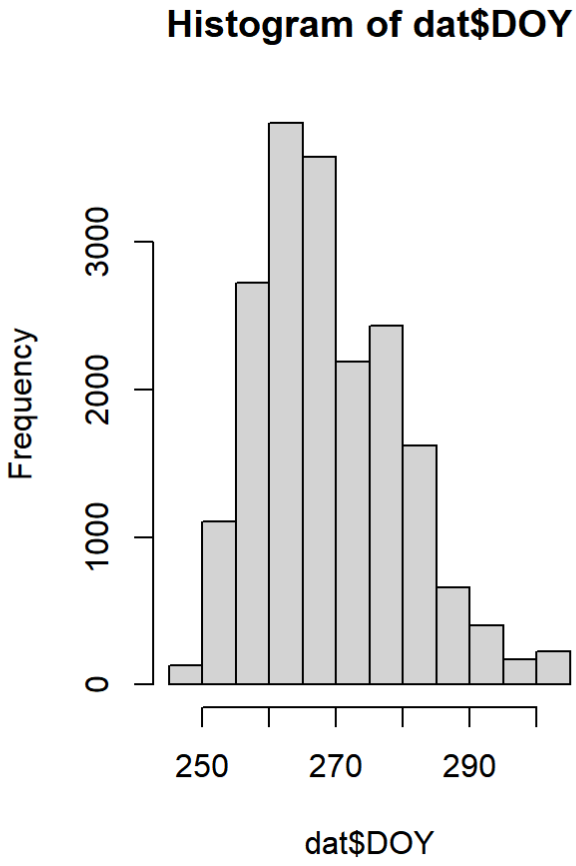
```
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
## This is mgcv 1.8-33. For overview type 'help("mgcv-package")'.
```

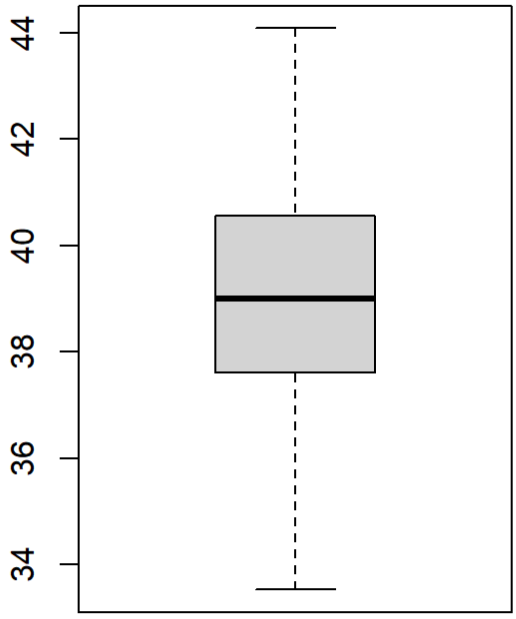
```
correct_temp <- function(data, formula, gam_formula) {

  print("glm")
  model <- lm(as.formula(formula),
              data = data)
  par(mfrow = c(2,2))
  plot(model)
  par(mfrow = c(1,1))
  pred <- ggeffects::ggpredict(model, terms = "YEAR")
  print(pred)
  cat("\n\n")
  # plot(pred) %>% print()
  # cat("\n\n")

  print("gam")
  model2 <- gam(as.formula(gam_formula),
              data = data)
  gam.check(model2)
    par(mfrow = c(2,2))
  plot(model2, all.terms = TRUE, rug = TRUE) %>%
    try()
  par(mfrow = c(1,1))
  pred2 <- ggeffects::ggpredict(model2, terms = "YEAR")
  print(pred2)
  cat("\n\n")
  # plot(pred2) %>% print()
  # cat("\n\n")

  compare_data <- rbind(pred %>% dplyr::mutate(type = "GLM"),
                        pred2 %>% dplyr::mutate(type = "GAM"))

  plt <- compare_data %>%
    ggplot2::ggplot(ggplot2::aes(x = x %>%
                                   as.character() %>%
                                   as.numeric(),
                                 y = predicted,
                                 color = type,
                                 group = type)) +
    ggplot2::geom_line() +
    ggplot2::geom_point() +
    ggplot2::theme_bw()

  print(plt)

  # plot(pred$x, pred$predicted, type = "b")
  # points(pred2$x, pred2$predicted, col = "red")
  # lines(pred2$x, pred2$predicted, col = "red")

  AIC(model, model2)
}

# need to expand data based on counts
```

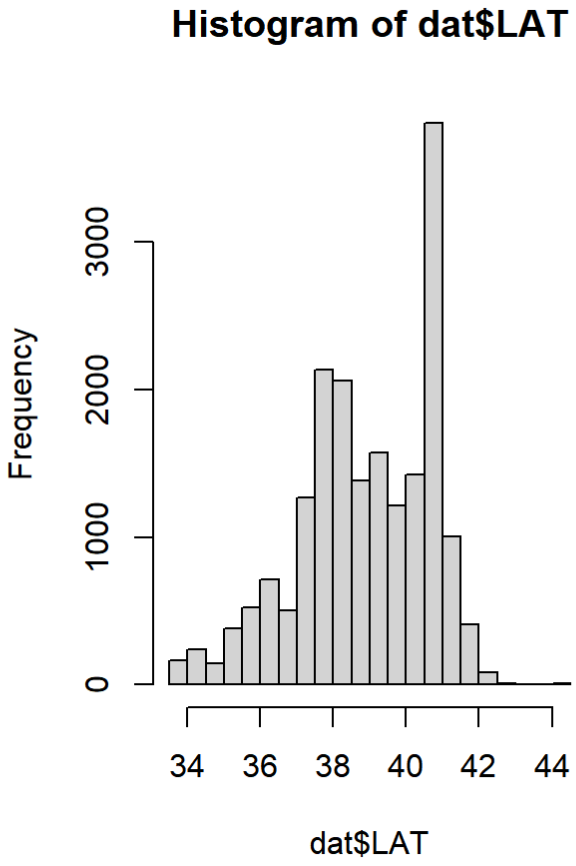
```
# dat <- bluefish_nmfs %>%
#   tidyr::drop_na(LENGTH, SURFTEMP) %>%
#   dplyr::mutate(size_group = ifelse(LENGTH <= 35,
#                                     "small (<=35cm)",
#                                     "large (>35cm)"),
#                 DOY = lubridate::yday(EST_TOWDATE)) %>%
#   tidyr::drop_na(SURFTEMP) %>%
#   dplyr::filter(SURFTEMP > 0,
#                 SEASON == "FALL",
#                 DOY < 305) %>% # don't include November
#   dplyr::group_by(YEAR, SEASON, size_group) %>%
#   dplyr::mutate(n_fish = sum(NUMLEN),
#                 n_event = dplyr::n()) %>%
#   dplyr::filter(n_fish > 9, n_event > 9) %>%
#   bluefishLifeHistory::expand_n(col = "NUMLEN")
#
# saveRDS(dat, file = here::here("data-raw/nmfs_survey_expanded_9.RDS"))

dat <- readRDS(here::here("data-raw/nmfs_survey_expanded_9.RDS"))

dat <- dat %>%
    # make sure each year has enough of a spread of data
    dplyr::group_by(YEAR, size_group) %>%
    dplyr::mutate(range_doy = max(DOY) - min(DOY)) %>%
    dplyr::filter(range_doy > 19)

## check for normality, outliers

par(mfrow = c(1,2))

boxplot(dat$DOY)
hist(dat$DOY)
```



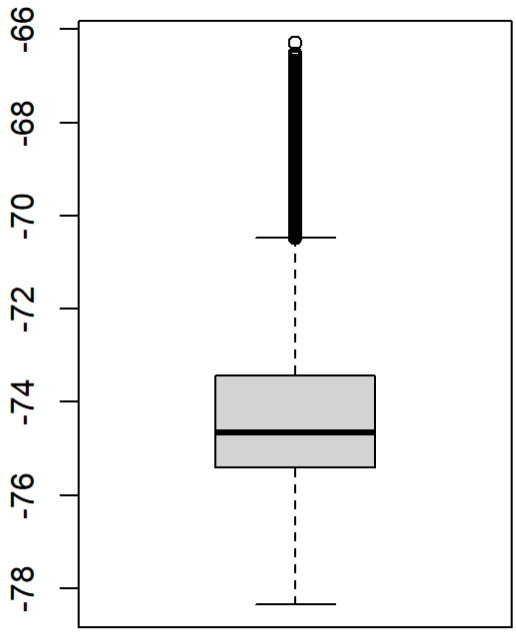


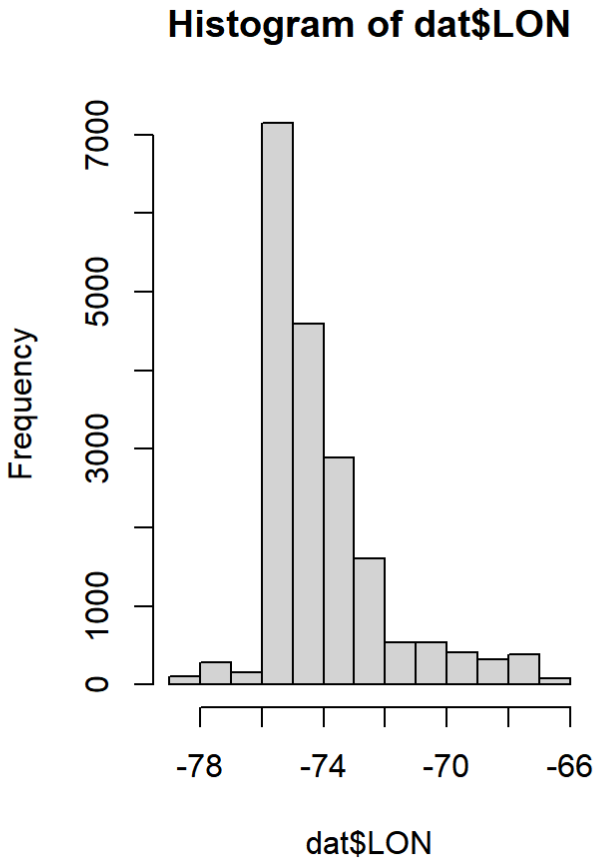

```
boxplot(dat$LAT)
hist(dat$LAT)
```



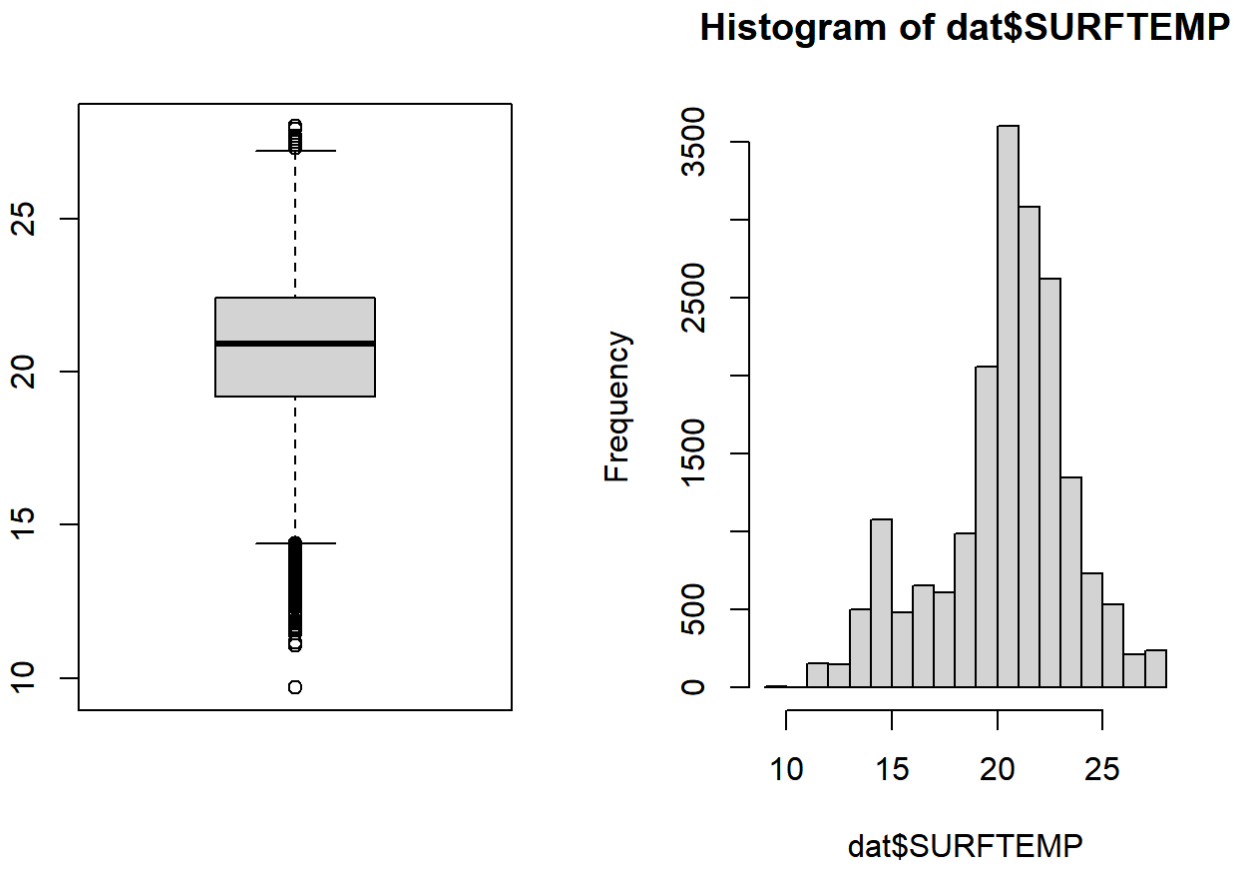



```
boxplot(dat$LON)
hist(dat$LON)
```






```
boxplot(dat$SURFTEMP)
hist(dat$SURFTEMP)
```




```
  par(mfrow = c(1,1))


# remove outlier LAT > 44
dat <- dat %>%
  dplyr::filter(LAT < 44)
```

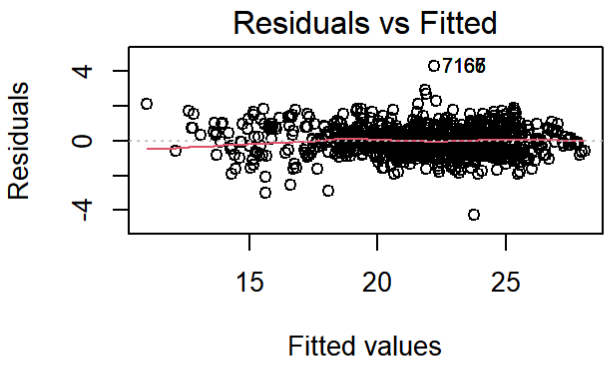
0.2 Surface temperature

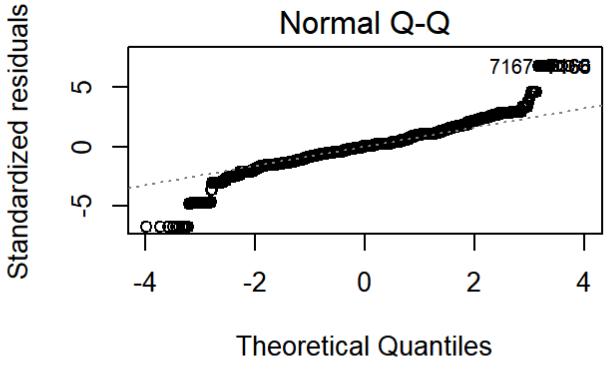
Try to account for annual variation in latitude, longitude, and day of year when looking at how surface temperature preferences change over time

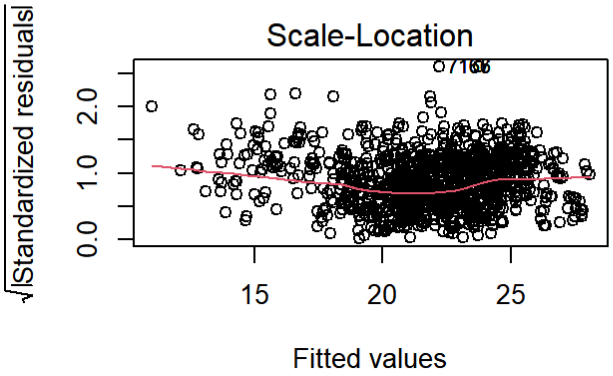
0.2.1 Small bluefish

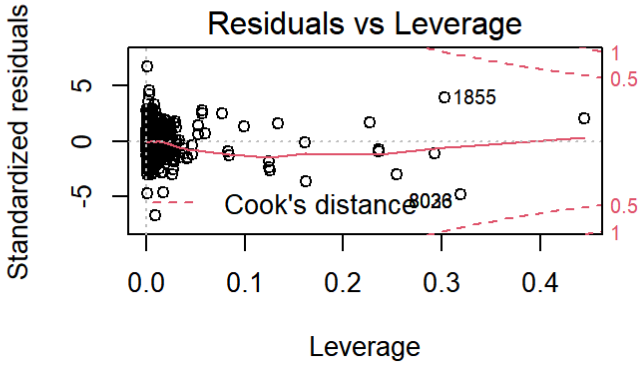
```
correct_temp(data = dat %>%
                  dplyr::filter(size_group == "small (<=35cm)"),
             formula = "SURFTEMP ~ LAT*LON + as.factor(YEAR)*DOY",
             gam_formula = "SURFTEMP ~ s(LAT) + s(LON) + te(LAT, LON, k = 7) + as.factor(Y
EAR)*DOY")
```

```
## [1] "glm"
```

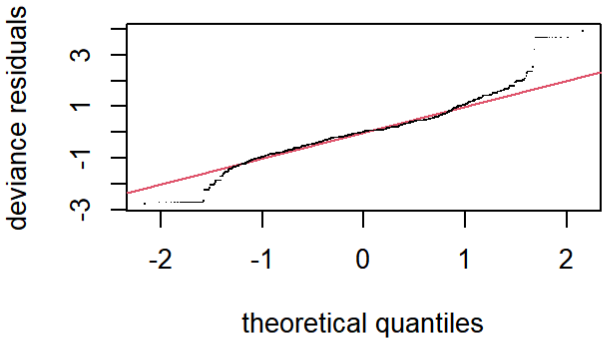


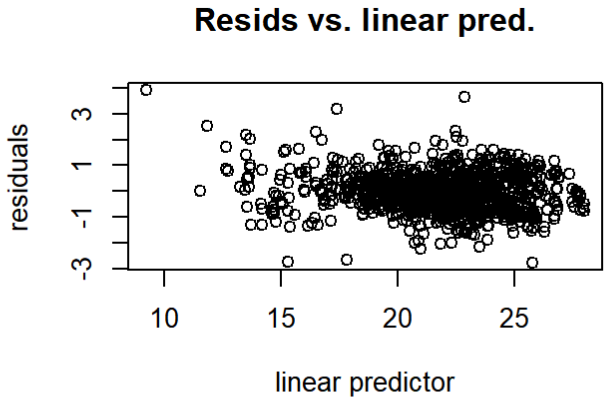


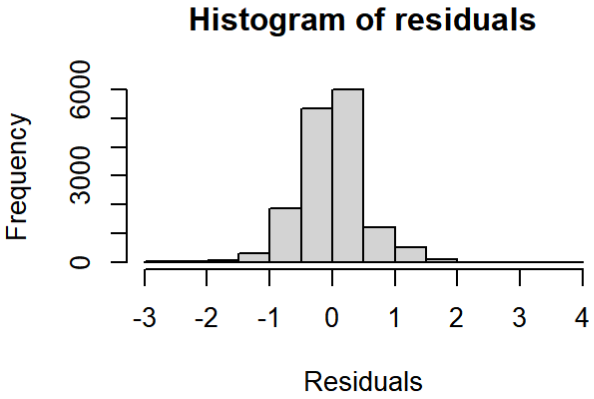


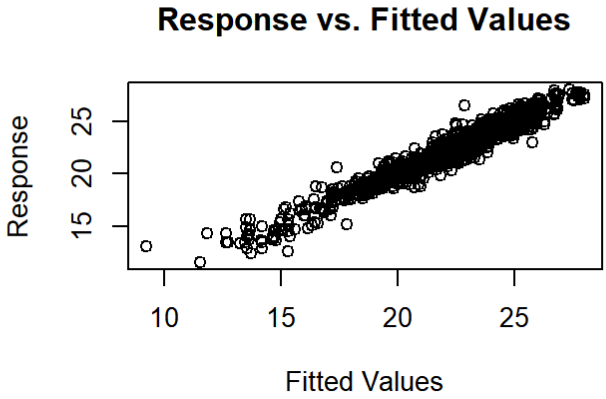



```
## # Predicted values of SURFTEMP
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1973 |     21.79 | [21.56, 22.03]
## 1980 |     22.71 | [22.63, 22.78]
## 1985 |     22.51 | [22.19, 22.84]
## 1996 |     21.06 | [20.95, 21.17]
## 2001 |     21.37 | [21.30, 21.45]
## 2007 |     20.94 | [20.88, 21.00]
## 2012 |     22.16 | [22.02, 22.30]
## 2021 |     23.10 | [22.91, 23.30]
##
## Adjusted for:
## * LAT =  38.51
## * LON = -74.64
## * DOY = 267.77
##
##
## [1] "gam"
```

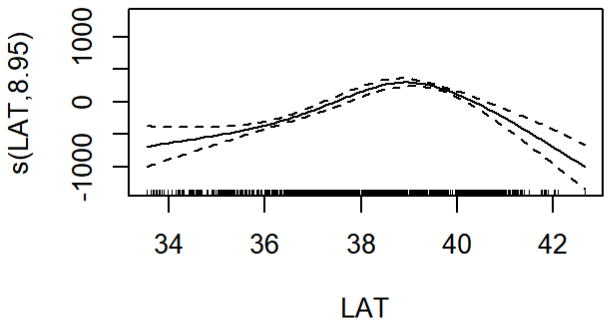


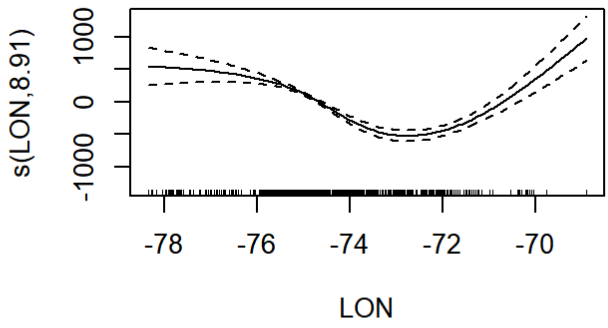


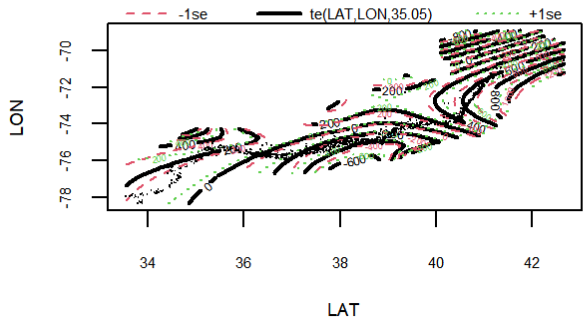


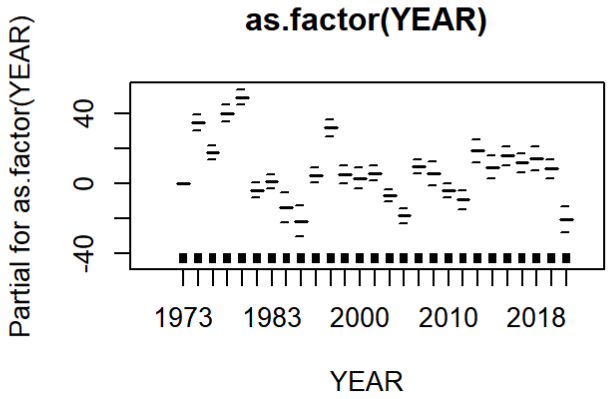



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 40 iterations by steepest
## descent step failure.
## The RMS GCV score gradient at convergence was 1.209951e-07 .
## The Hessian was not positive definite.
## Model rank =  107 / 110
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                k'   edf k-index p-value
## s(LAT)       9.00  8.95    0.36  <2e-16 ***
## s(LON)       9.00  8.91    0.48  <2e-16 ***
## te(LAT,LON) 38.00 35.05    0.17  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

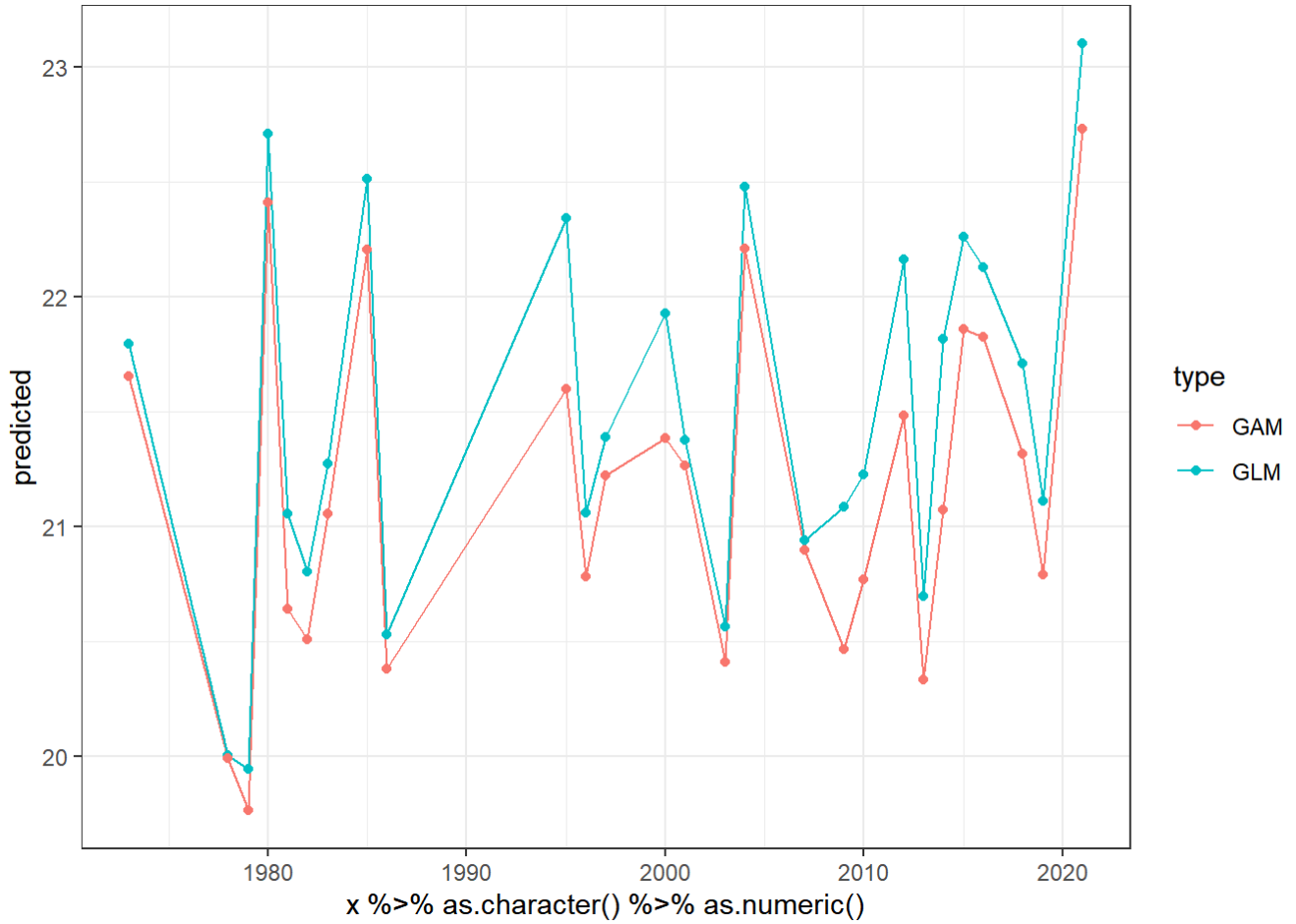









```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of SURFTEMP
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1973 |     21.65 | [21.38, 21.93]
## 1980 |     22.41 | [22.23, 22.59]
## 1985 |     22.20 | [21.87, 22.54]
## 1996 |     20.78 | [20.58, 20.98]
## 2001 |     21.26 | [21.08, 21.45]
## 2007 |     20.90 | [20.72, 21.07]
## 2012 |     21.48 | [21.27, 21.69]
## 2021 |     22.73 | [22.49, 22.96]
##
## Adjusted for:
## * LAT =  38.51
## * LON = -74.64
## * DOY = 267.77
```

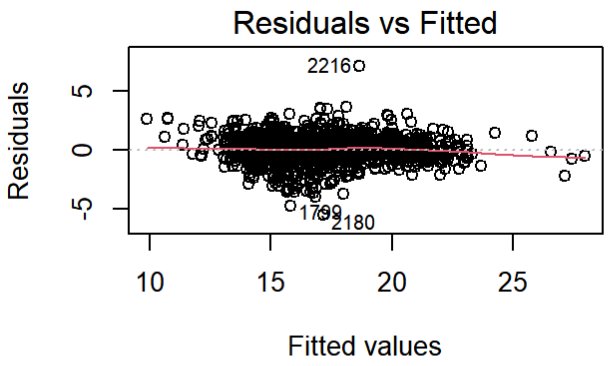


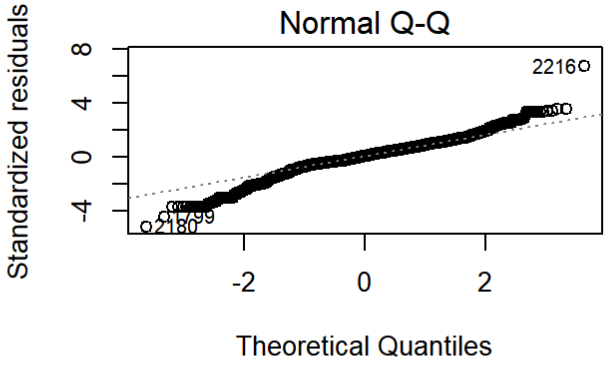

```
##              df      AIC
## model   58.0000 30009.20
## model2 107.9056 25000.59
```

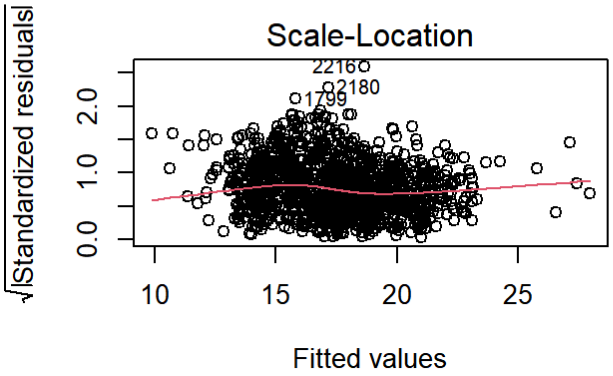
0.2.2 Large bluefish

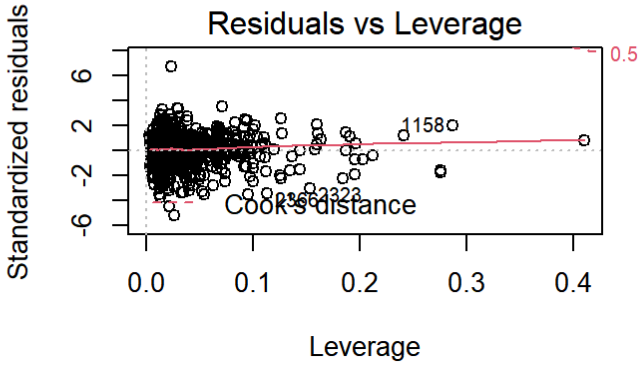
```
correct_temp(data = dat %>%
                  dplyr::filter(size_group == "large (>35cm)"),
             formula = "SURFTEMP ~ LAT*LON + as.factor(YEAR)*DOY",
             gam_formula = "SURFTEMP ~ s(LAT) + s(LON) + te(LAT, LON, k = 7) + as.factor(Y
EAR)*DOY")
```

```
## [1] "glm"
```

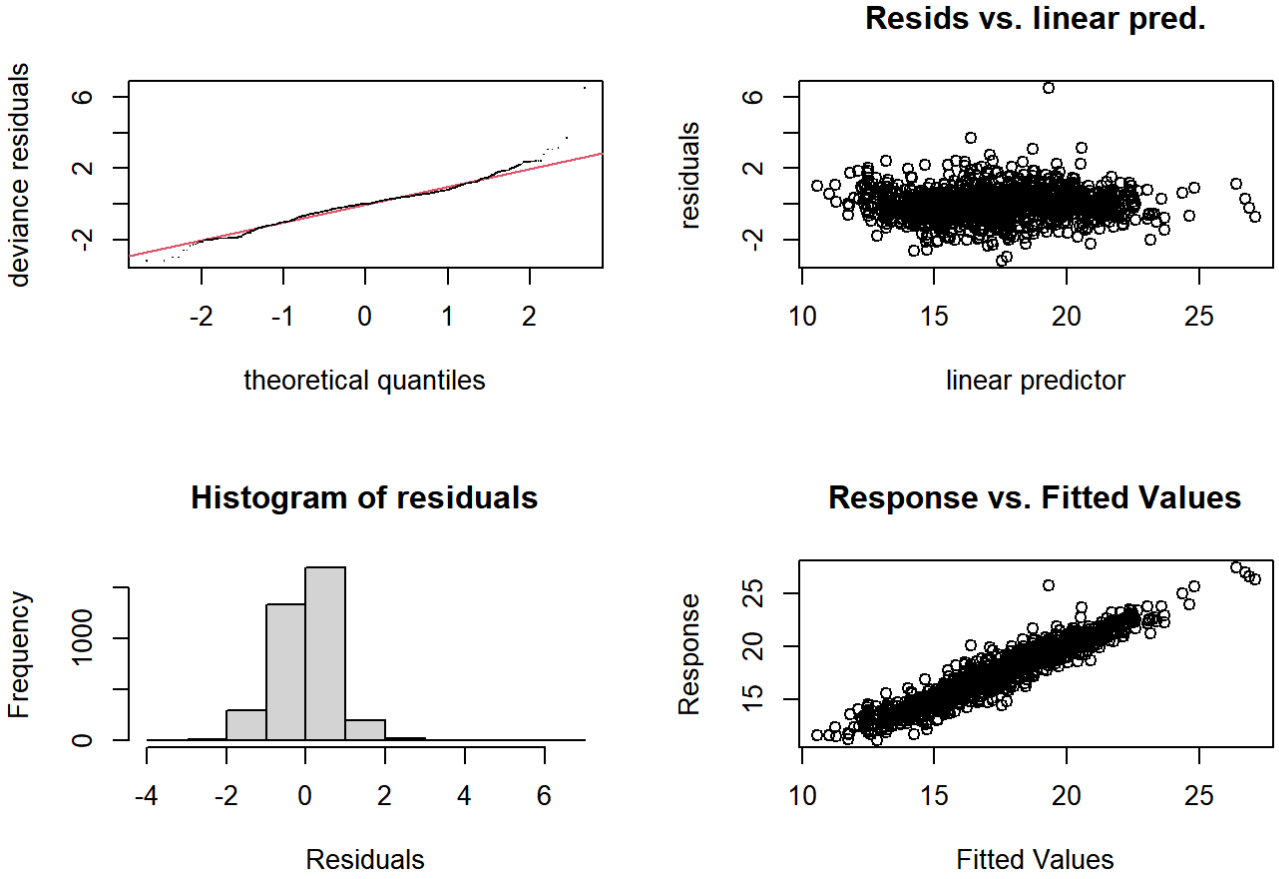




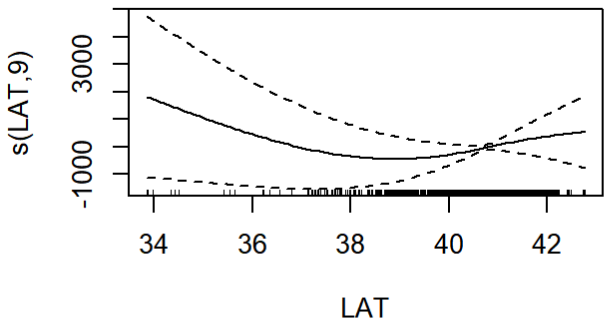


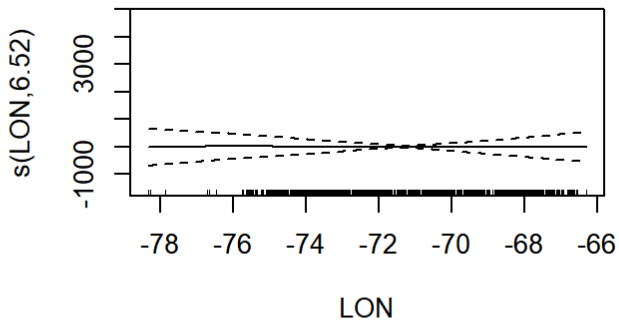


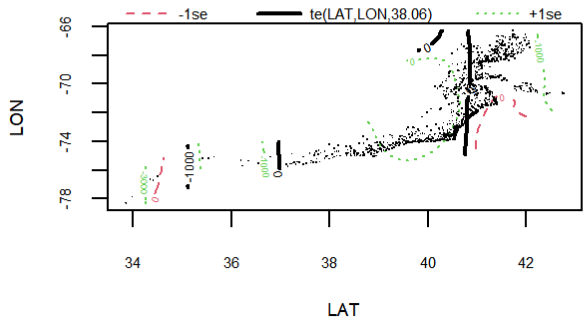

```
## # Predicted values of SURFTEMP
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1971 |     18.67 | [18.31, 19.02]
## 1976 |     16.36 | [16.09, 16.63]
## 1982 |     16.96 | [16.66, 17.27]
## 1987 |     16.38 | [15.98, 16.77]
## 1993 |     16.51 | [16.03, 16.99]
## 1999 |     18.09 | [17.82, 18.35]
## 2004 |     16.72 | [16.55, 16.89]
## 2019 |     17.02 | [16.66, 17.39]
##
## Adjusted for:
## * LAT =  40.55
## * LON = -71.38
## * DOY = 278.83
##
##
## [1] "gam"
```

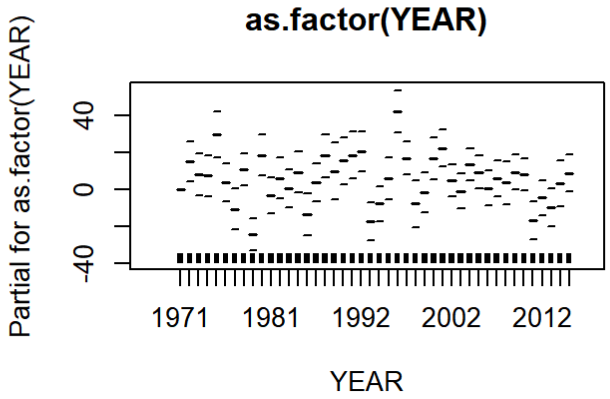



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 26 iterations by steepest
## descent step failure.
## The RMS GCV score gradient at convergence was 0.0001005988 .
## The Hessian was not positive definite.
## Model rank =  147 / 147
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                k'   edf k-index p-value
## s(LAT)       9.00  9.00    0.48  <2e-16 ***
## s(LON)       9.00  6.52    0.46  <2e-16 ***
## te(LAT,LON) 41.00 38.06    0.41  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

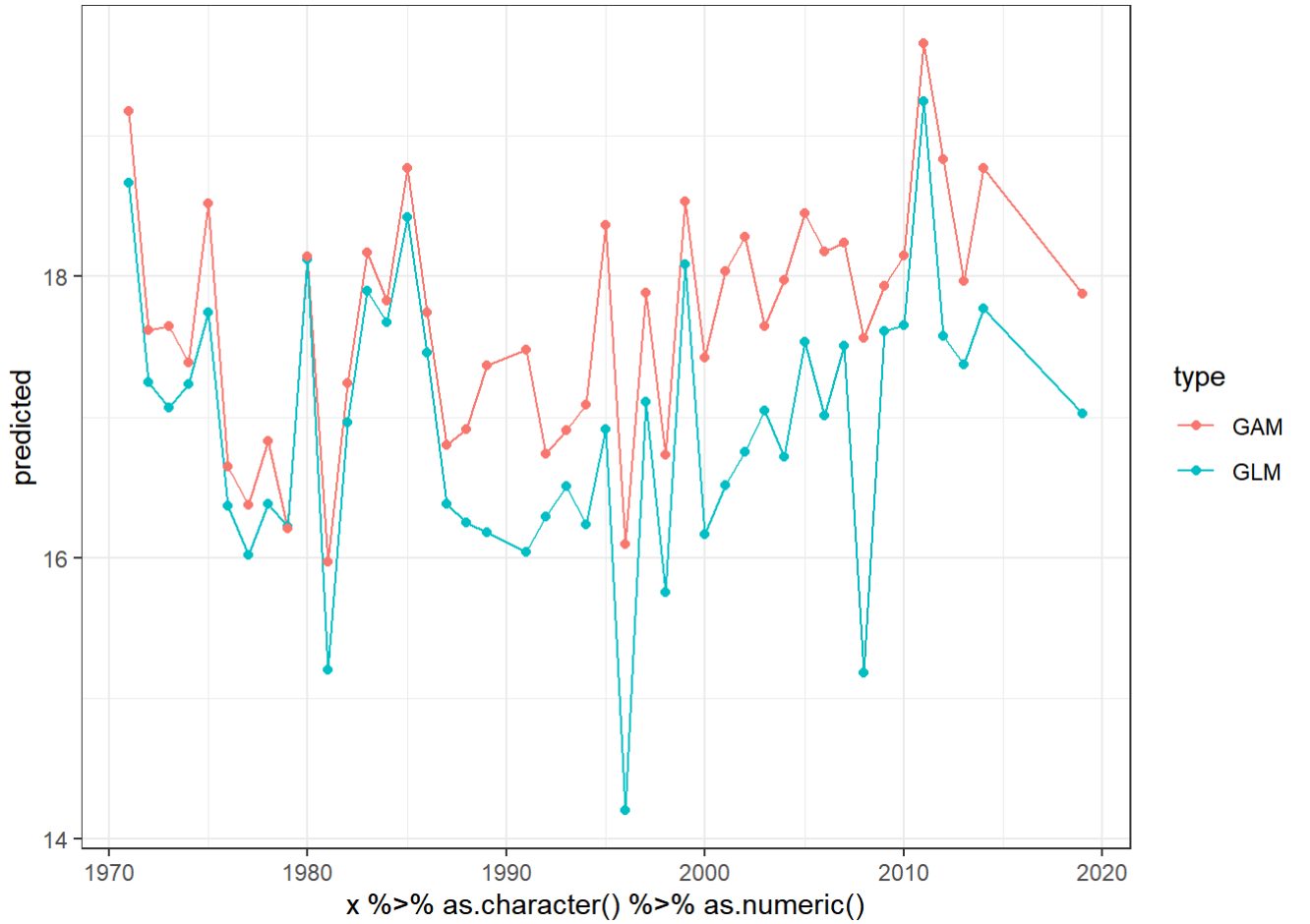









```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of SURFTEMP
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1971 |     19.18 | [18.80, 19.56]
## 1976 |     16.64 | [16.31, 16.97]
## 1982 |     17.24 | [16.90, 17.59]
## 1987 |     16.80 | [16.41, 17.19]
## 1993 |     16.90 | [16.47, 17.34]
## 1999 |     18.53 | [18.20, 18.86]
## 2004 |     17.98 | [17.67, 18.29]
## 2019 |     17.88 | [17.47, 18.28]
##
## Adjusted for:
## * LAT =  40.55
## * LON = -71.38
## * DOY = 278.83
```




```
##              df      AIC
## model   92.0000 10668.57
## model2 142.5806  8042.71
```

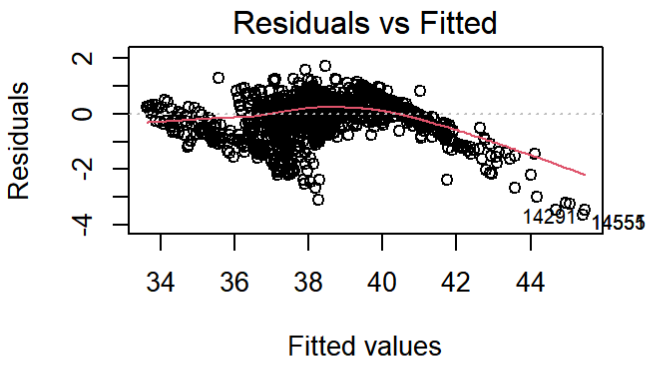
0.3 Latitude

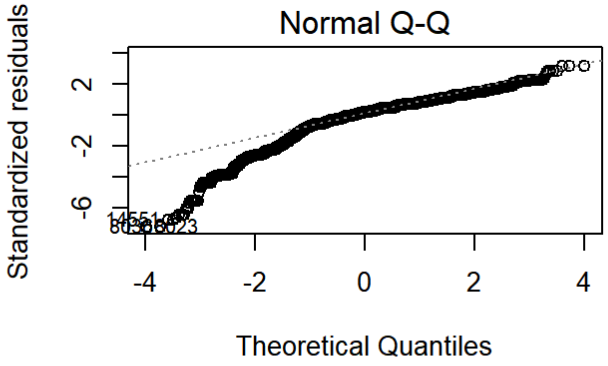
Try to account for annual variation in longitude and day of year when looking at how latitude preferences change over time

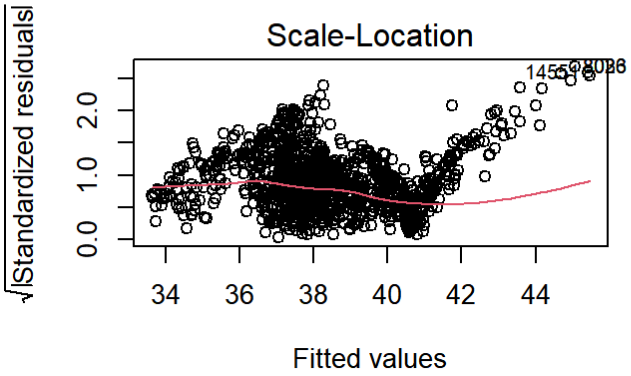
0.3.1 Small bluefish

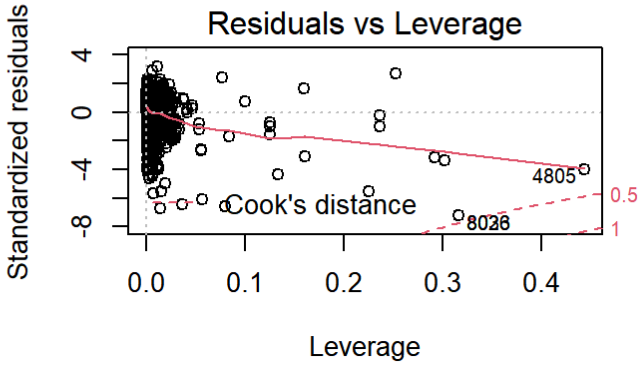
```
correct_temp(data = dat %>%
                  dplyr::filter(size_group == "small (<=35cm)"),
             formula = "LAT ~ LON + as.factor(YEAR)*DOY",
             gam_formula = "LAT ~ s(LON) + as.factor(YEAR)*DOY")
```

```
## [1] "glm"
```

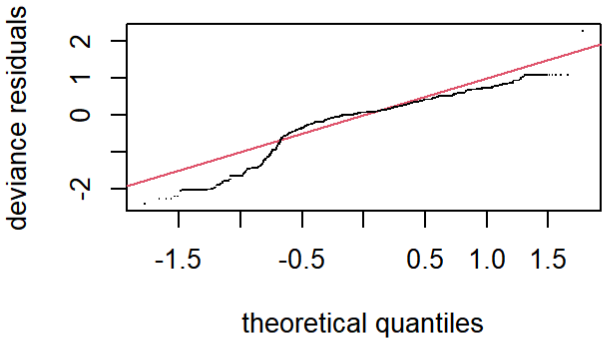


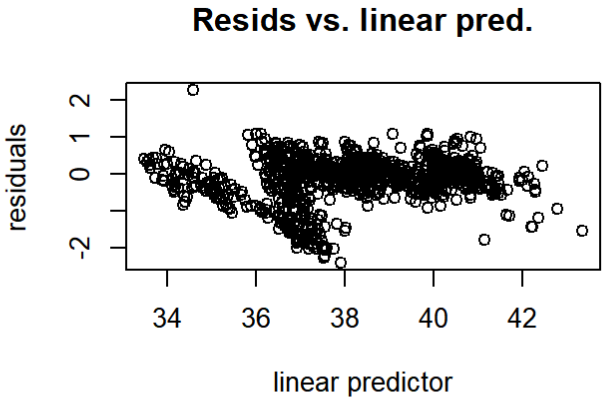


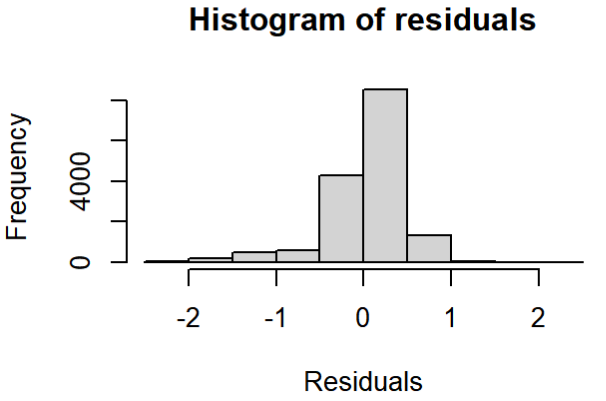


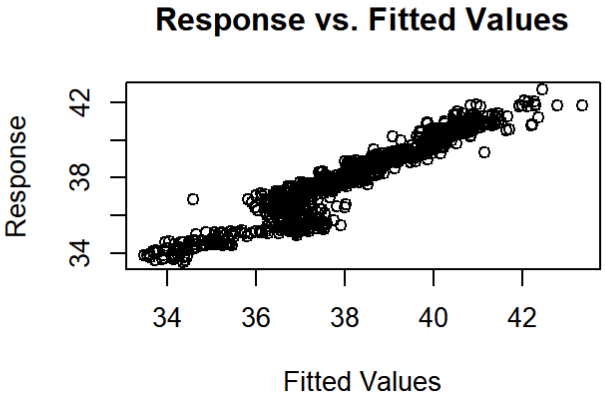



```
## # Predicted values of LAT
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1973 |     38.52 | [38.32, 38.72]
## 1980 |     36.94 | [36.88, 37.00]
## 1985 |     37.50 | [37.22, 37.78]
## 1996 |     39.45 | [39.36, 39.55]
## 2001 |     39.11 | [39.05, 39.17]
## 2007 |     38.85 | [38.79, 38.90]
## 2012 |     39.19 | [39.07, 39.31]
## 2021 |     38.68 | [38.51, 38.84]
##
## Adjusted for:
## * LON = -74.64
## * DOY = 267.77
##
##
## [1] "gam"
```

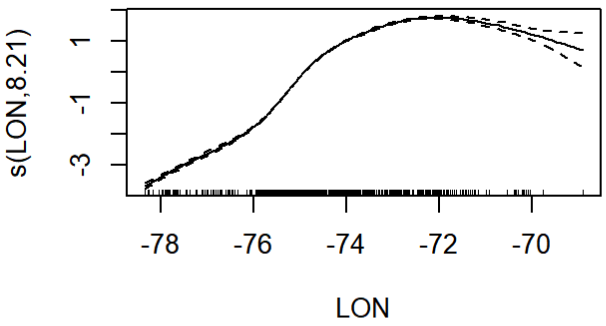


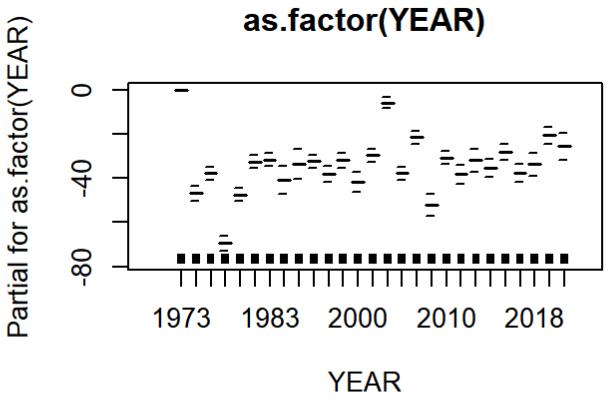







```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 10 iterations.
## The RMS GCV score gradient at convergence was 3.771239e-07 .
## The Hessian was positive definite.
## Model rank =  63 / 63
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'  edf k-index p-value
## s(LON) 9.00 8.21    0.43  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```





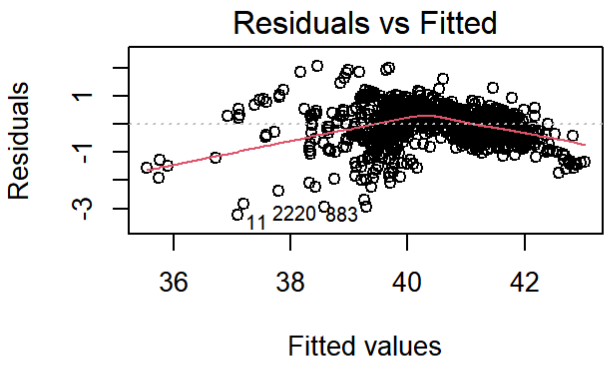

```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of LAT
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1973 |     38.95 | [38.78, 39.12]
## 1980 |     37.62 | [37.57, 37.68]
## 1985 |     37.72 | [37.49, 37.95]
## 1996 |     39.58 | [39.50, 39.66]
## 2001 |     39.62 | [39.57, 39.68]
## 2007 |     39.24 | [39.20, 39.29]
## 2012 |     39.38 | [39.28, 39.48]
## 2021 |     39.33 | [39.19, 39.47]
##
## Adjusted for:
## * LON = -74.64
## * DOY = 267.77
```

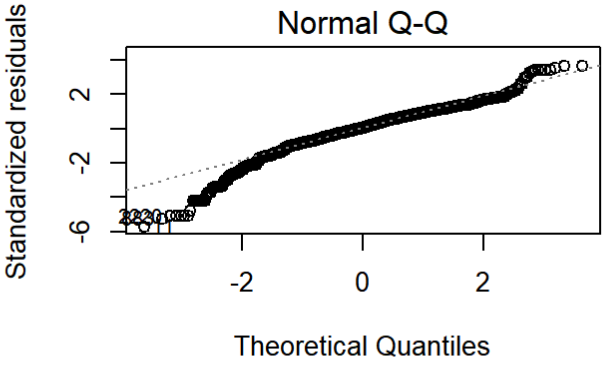

```
##              df      AIC
## model  56.00000 25158.04
## model2 63.21436 18977.21
```

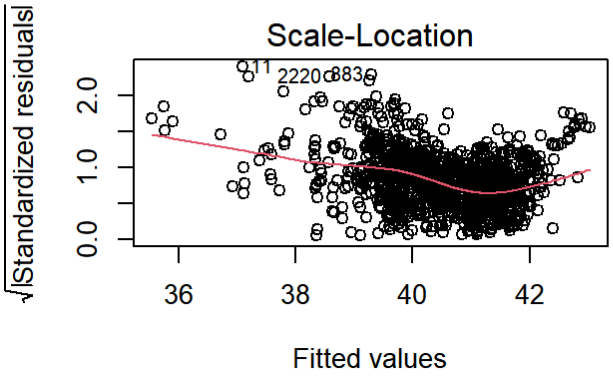
0.3.2 Large bluefish

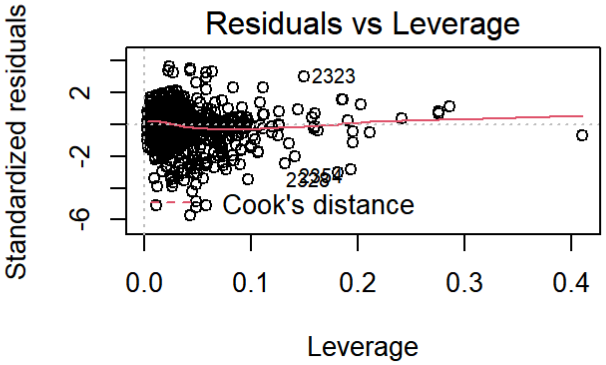
```
correct_temp(data = dat %>%
                   dplyr::filter(size_group == "large (>35cm)"),
             formula = "LAT ~ LON + as.factor(YEAR)*DOY",
             gam_formula = "LAT ~ s(LON) + as.factor(YEAR)*DOY")
```

```
## [1] "glm"
```

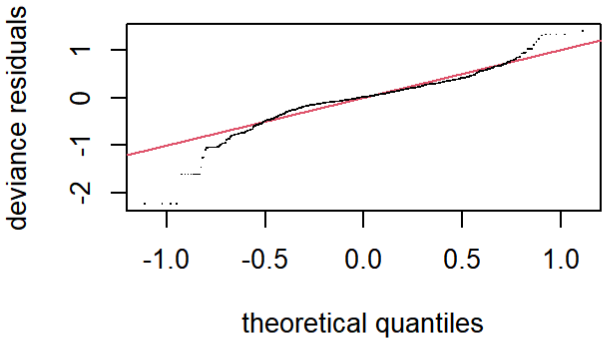


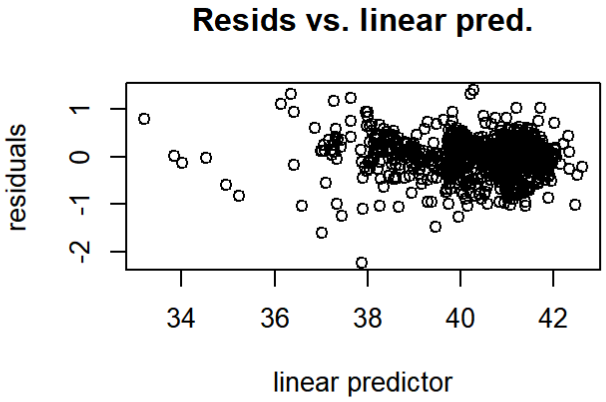


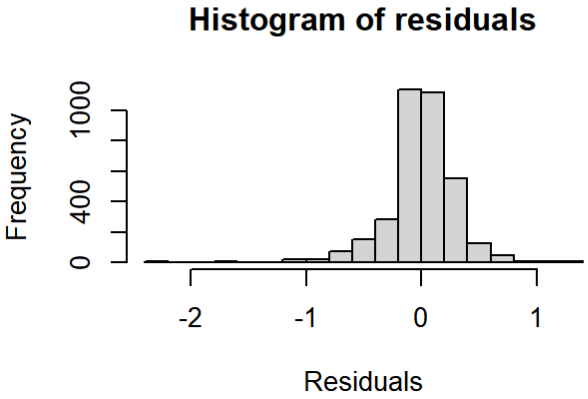


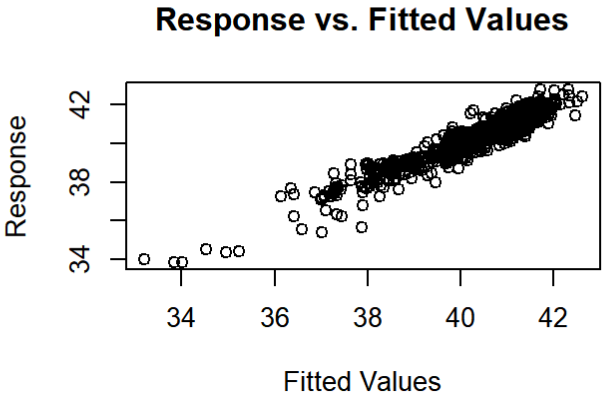



```
## # Predicted values of LAT
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1971 |     40.60 | [40.41, 40.79]
## 1976 |     40.28 | [40.14, 40.42]
## 1982 |     40.59 | [40.43, 40.75]
## 1987 |     40.64 | [40.43, 40.85]
## 1993 |     40.43 | [40.17, 40.68]
## 1999 |     40.29 | [40.15, 40.43]
## 2004 |     41.03 | [40.95, 41.12]
## 2019 |     40.25 | [40.06, 40.45]
##
## Adjusted for:
## * LON = -71.38
## * DOY = 278.83
##
##
## [1] "gam"
```

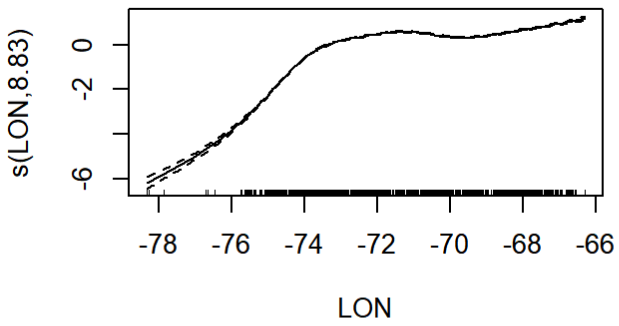


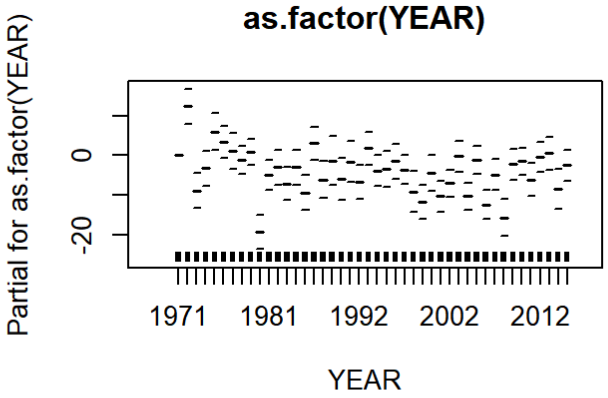




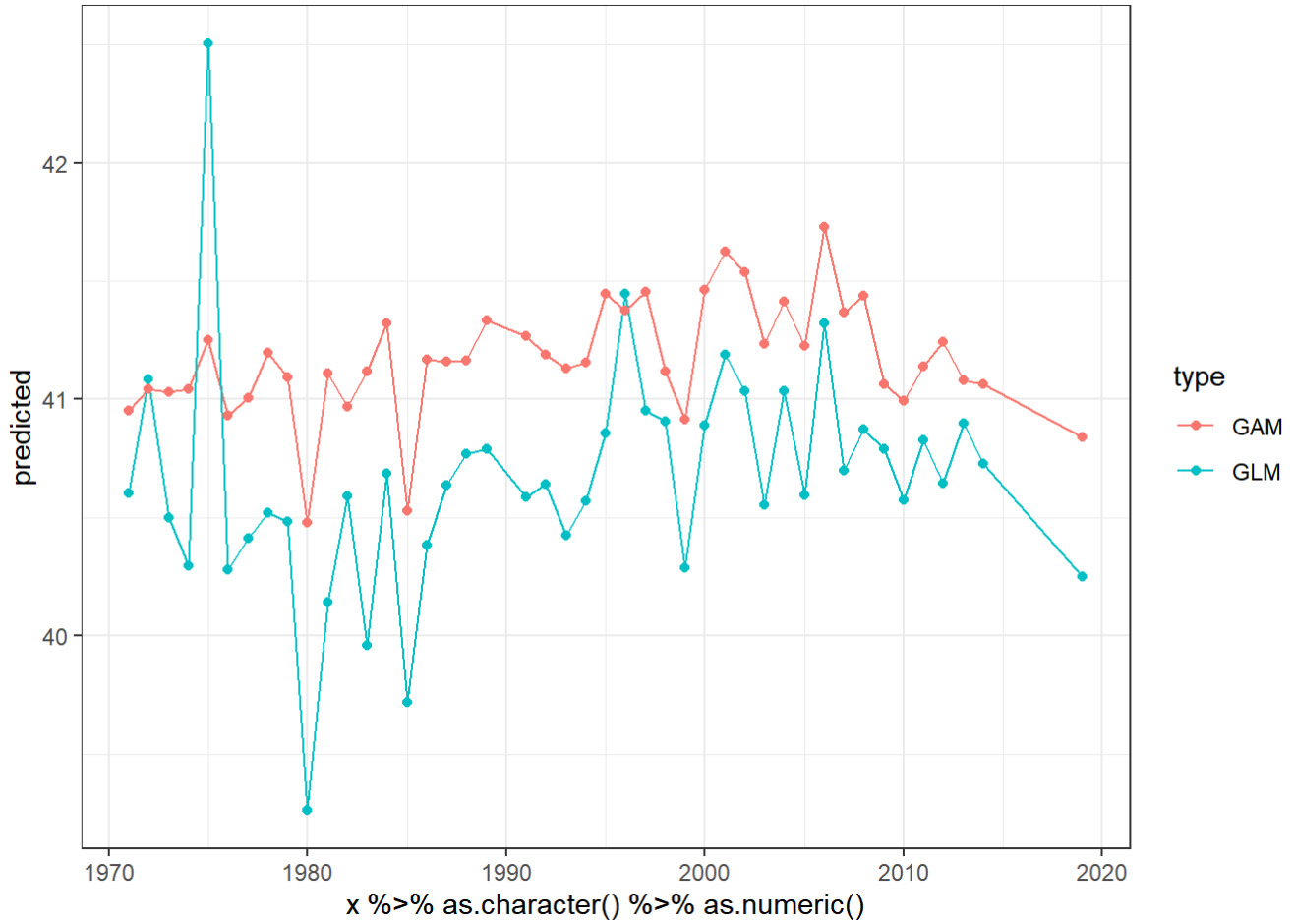



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 11 iterations.
## The RMS GCV score gradient at convergence was 5.079412e-07 .
## The Hessian was positive definite.
## Model rank =  97 / 97
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'  edf k-index p-value
## s(LON) 9.00 8.83    0.41  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```






```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of LAT
##
## YEAR | Predicted |         95% CI
## --------------------------------
## 1971 |     40.95 | [40.85, 41.06]
## 1976 |     40.93 | [40.85, 41.02]
## 1982 |     40.97 | [40.88, 41.06]
## 1987 |     41.16 | [41.04, 41.28]
## 1993 |     41.13 | [40.99, 41.27]
## 1999 |     40.91 | [40.83, 41.00]
## 2004 |     41.41 | [41.36, 41.47]
## 2019 |     40.84 | [40.73, 40.95]
##
## Adjusted for:
## * LON = -71.38
## * DOY = 278.83
```




```
##              df      AIC
## model  90.00000 6264.630
## model2 97.83051 1773.841
```

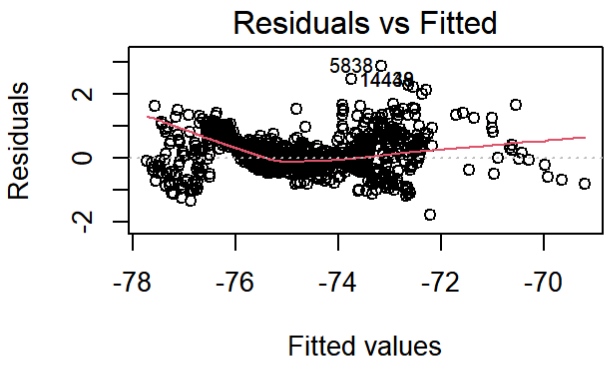
0.4 Longitude

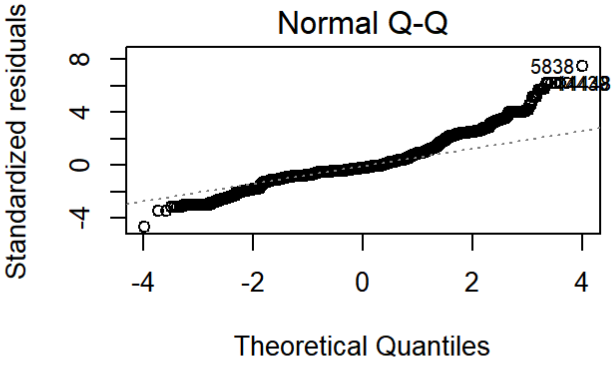
Try to account for annual variation in latitude and day of year when looking at how longitude preferences change over time

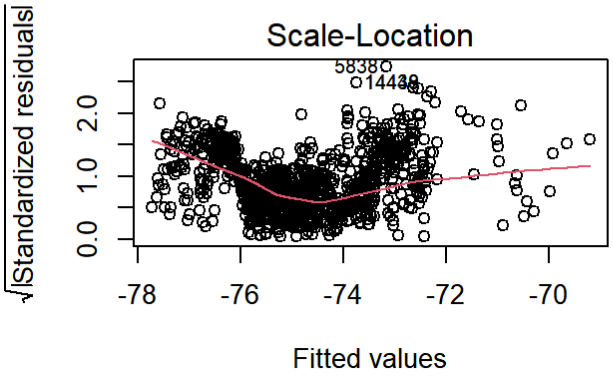
0.4.1 Small bluefish

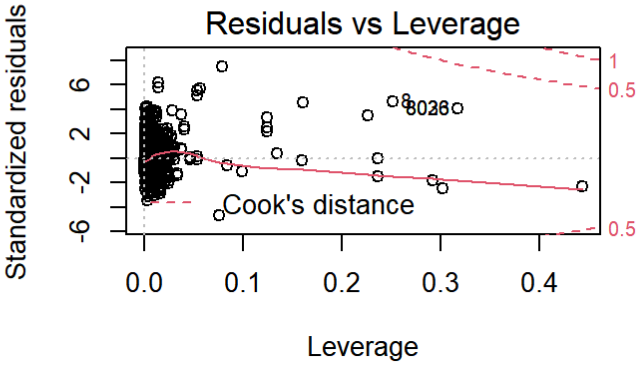
```
correct_temp(data = dat %>%
                  dplyr::filter(size_group == "small (<=35cm)"),
             formula = "LON ~ LAT + as.factor(YEAR)*DOY",
             gam_formula = "LON ~ s(LAT) + as.factor(YEAR)*DOY")
```

```
## [1] "glm"
```

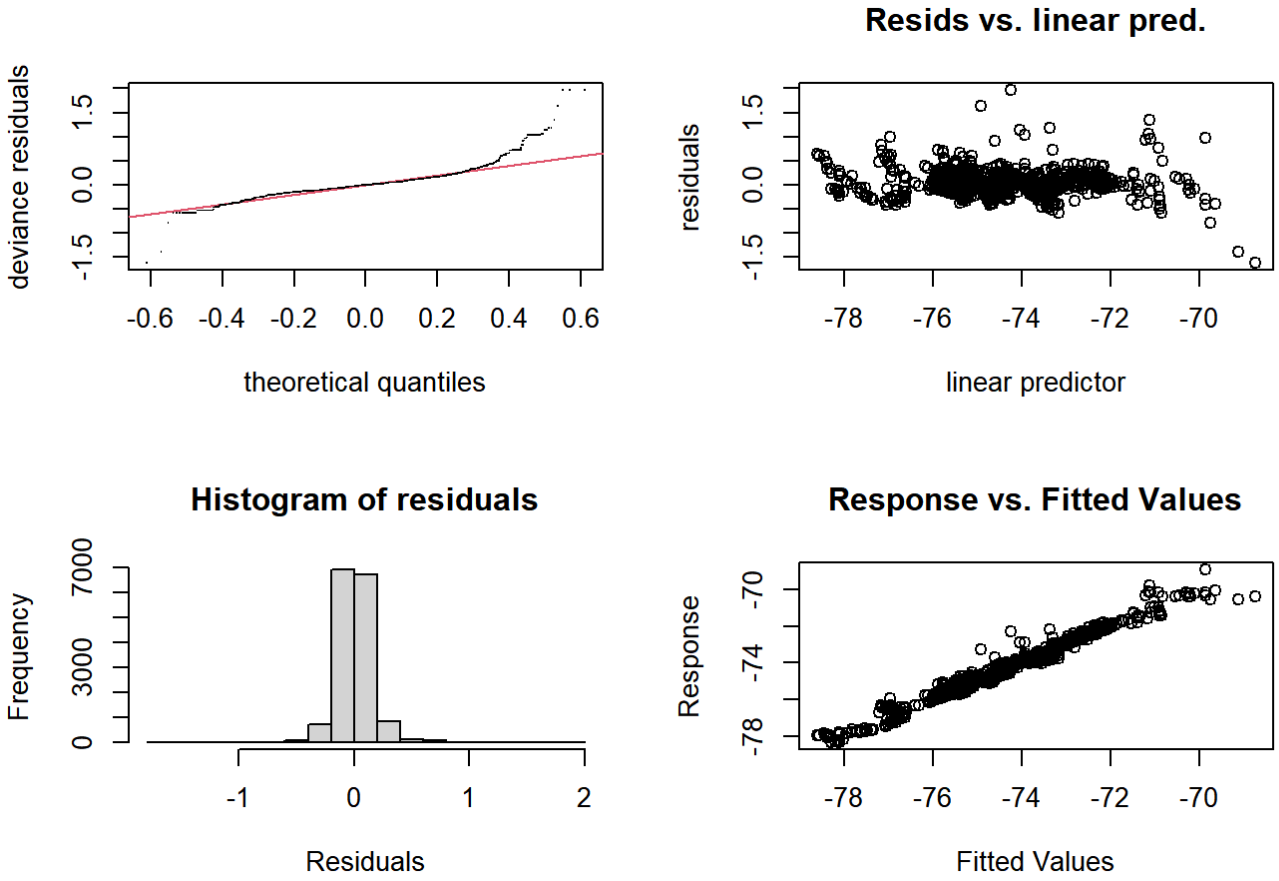




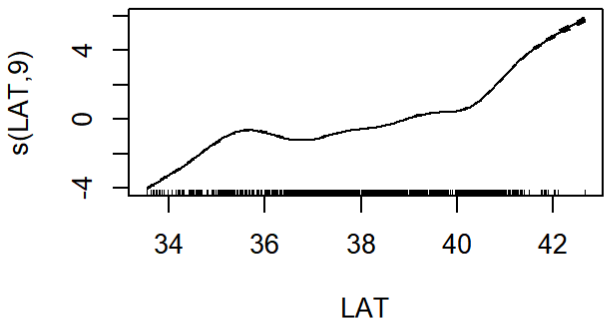


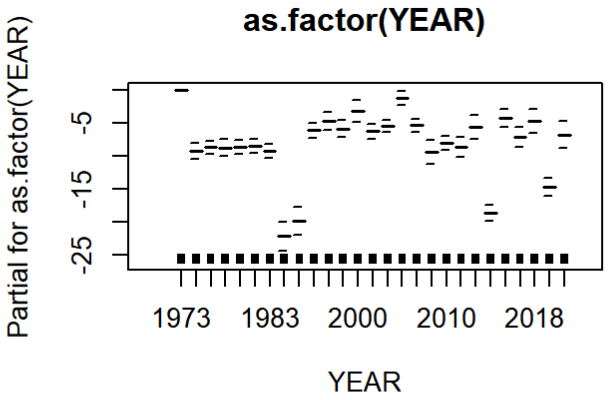



```
## # Predicted values of LON
##
## YEAR | Predicted |           95% CI
## ----------------------------------
## 1973 |    -73.47 | [-73.62, -73.33]
## 1980 |    -74.74 | [-74.79, -74.69]
## 1985 |    -75.67 | [-75.87, -75.47]
## 1996 |    -74.60 | [-74.67, -74.53]
## 2001 |    -74.15 | [-74.19, -74.11]
## 2007 |    -74.77 | [-74.80, -74.73]
## 2012 |    -74.03 | [-74.11, -73.94]
## 2021 |    -73.92 | [-74.04, -73.80]
##
## Adjusted for:
## * LAT =  38.51
## * DOY = 267.77
##
##
## [1] "gam"
```

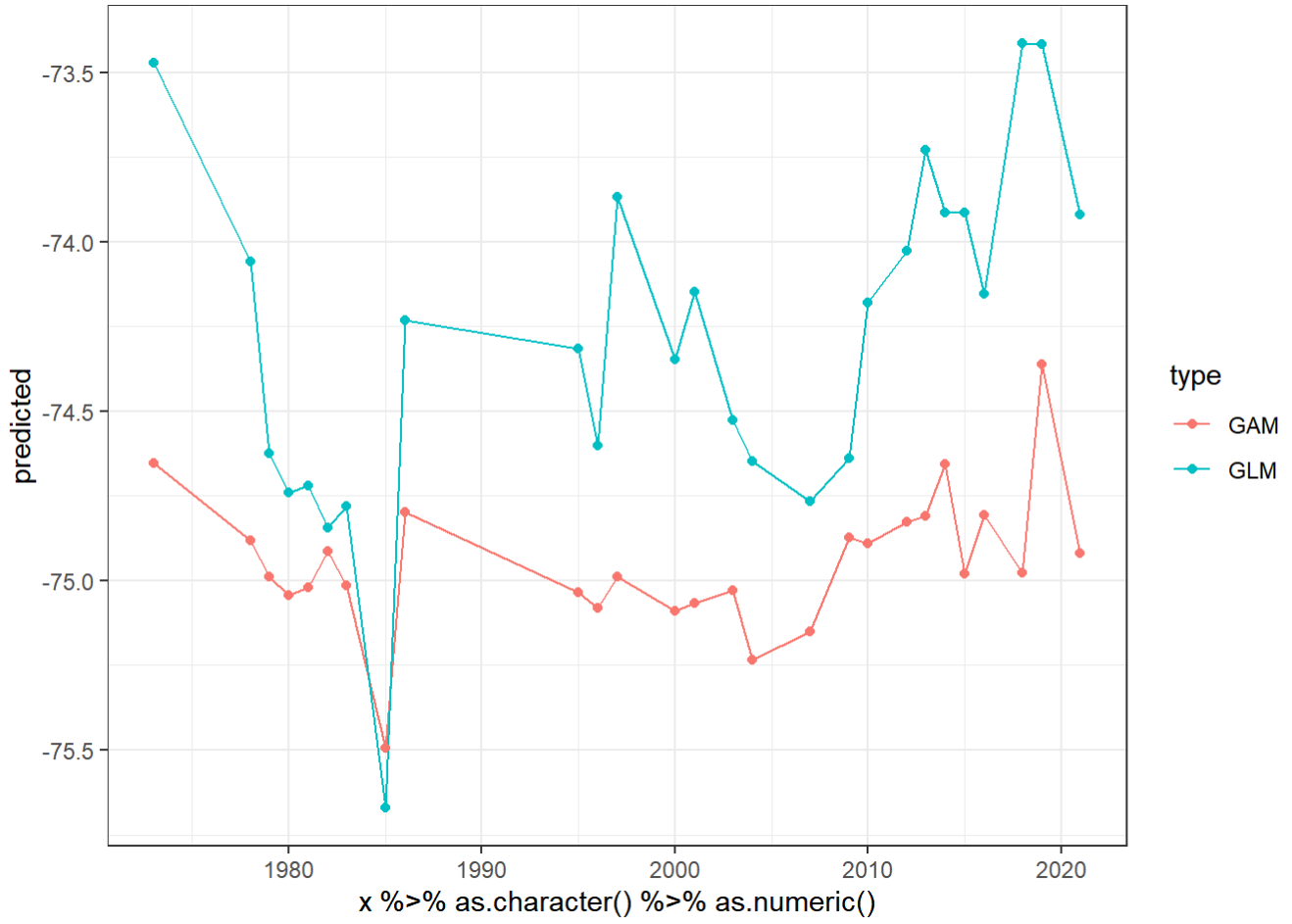



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 12 iterations.
## The RMS GCV score gradient at convergence was 2.053598e-07 .
## The Hessian was positive definite.
## Model rank =  63 / 63
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##        k' edf k-index p-value
## s(LAT)  9   9    0.35  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```






```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of LON
##
## YEAR | Predicted |           95% CI
## ----------------------------------
## 1973 |    -74.65 | [-74.71, -74.59]
## 1980 |    -75.04 | [-75.06, -75.02]
## 1985 |    -75.50 | [-75.57, -75.42]
## 1996 |    -75.08 | [-75.11, -75.05]
## 2001 |    -75.07 | [-75.09, -75.04]
## 2007 |    -75.15 | [-75.17, -75.13]
## 2012 |    -74.83 | [-74.86, -74.79]
## 2021 |    -74.92 | [-74.97, -74.87]
##
## Adjusted for:
## * LAT =  38.51
## * DOY = 267.77
```

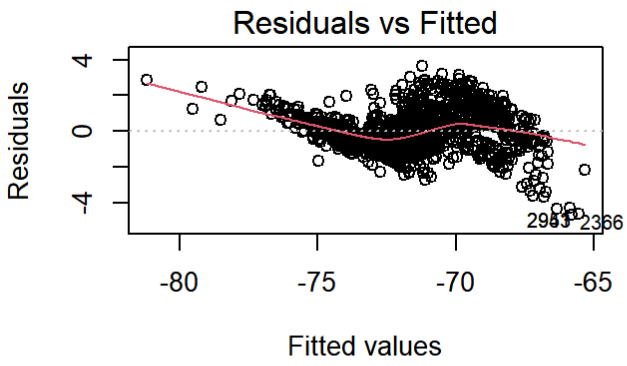


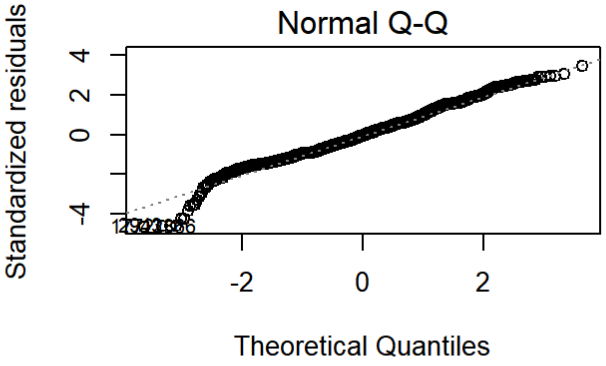

```
##              df       AIC
## model  56.00000  15563.95
## model2 63.99912 -14105.66
```

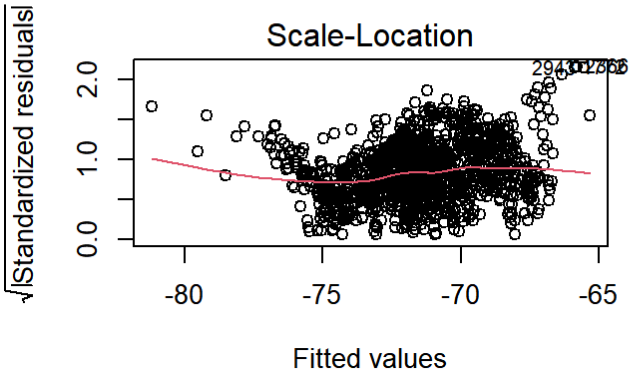
0.4.2 Large bluefish

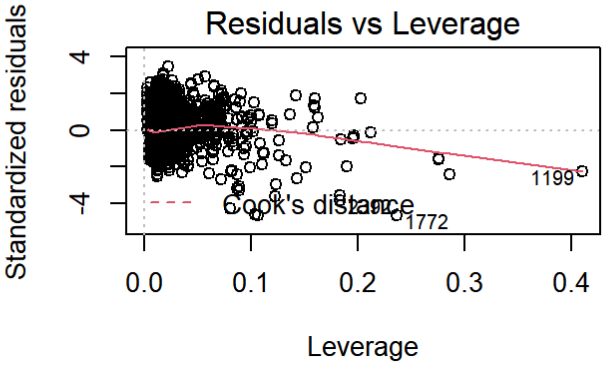
```
correct_temp(data = dat %>%
                   dplyr::filter(size_group == "large (>35cm)"),
             formula = "LON ~ LAT + as.factor(YEAR)*DOY",
             gam_formula = "LON ~ s(LAT) + as.factor(YEAR)*DOY")
```

```
## [1] "glm"
```

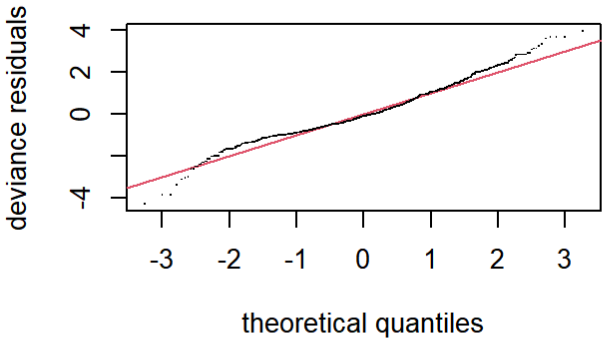


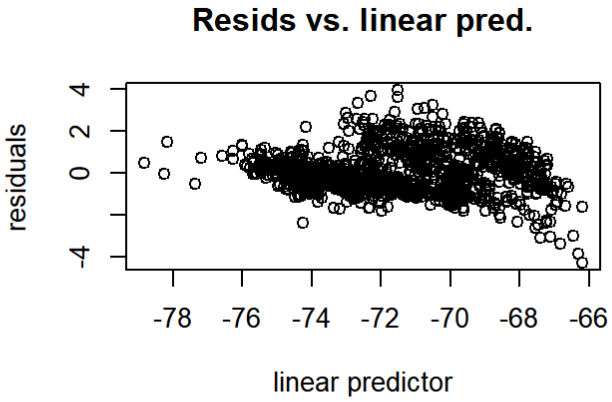


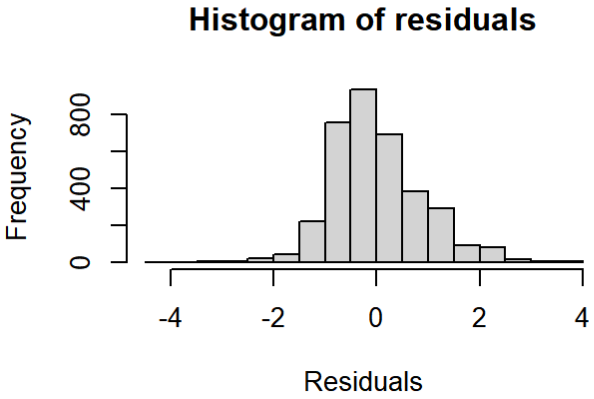


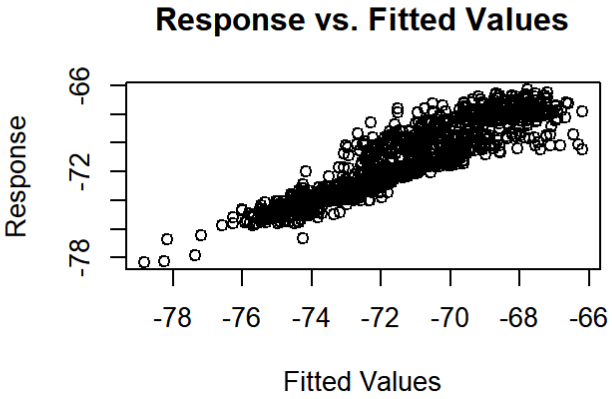



```
## # Predicted values of LON
##
## YEAR | Predicted |           95% CI
## ----------------------------------
## 1971 |    -71.44 | [-71.80, -71.09]
## 1976 |    -72.96 | [-73.22, -72.70]
## 1982 |    -72.76 | [-73.06, -72.46]
## 1987 |    -70.15 | [-70.54, -69.76]
## 1993 |    -68.62 | [-69.08, -68.15]
## 1999 |    -71.52 | [-71.78, -71.26]
## 2004 |    -71.83 | [-71.98, -71.67]
## 2019 |    -70.12 | [-70.48, -69.76]
##
## Adjusted for:
## * LAT =  40.55
## * DOY = 278.83
##
##
## [1] "gam"
```

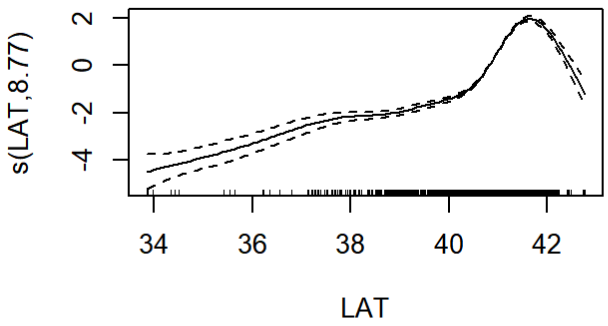


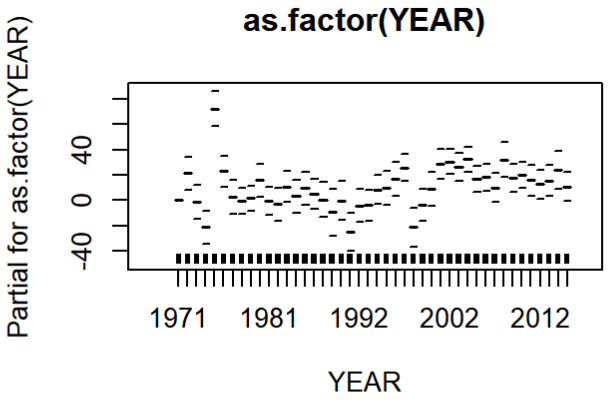




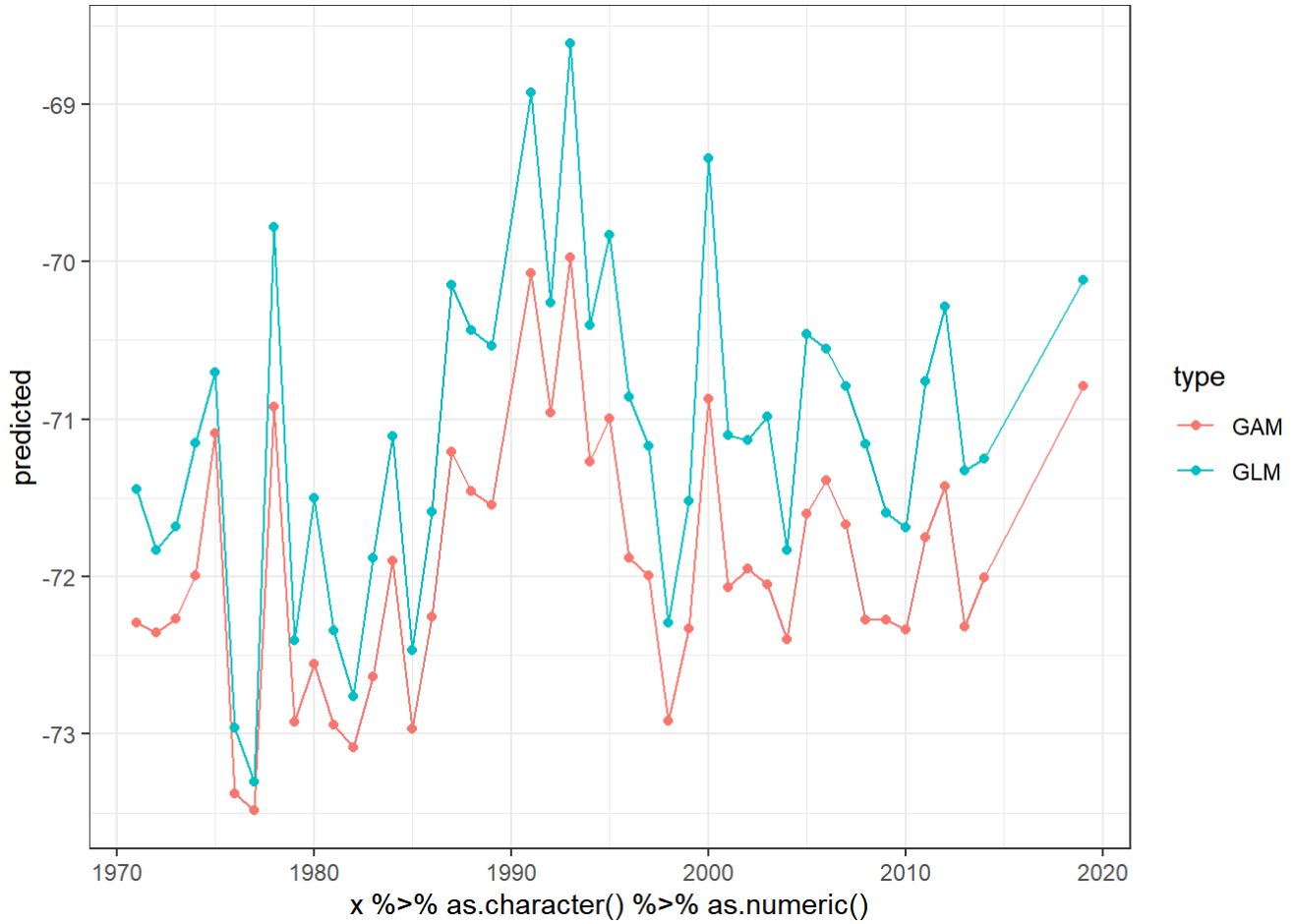



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 6 iterations.
## The RMS GCV score gradient at convergence was 2.795283e-06 .
## The Hessian was positive definite.
## Model rank =  97 / 97
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'  edf k-index p-value
## s(LAT) 9.00 8.77    0.45  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



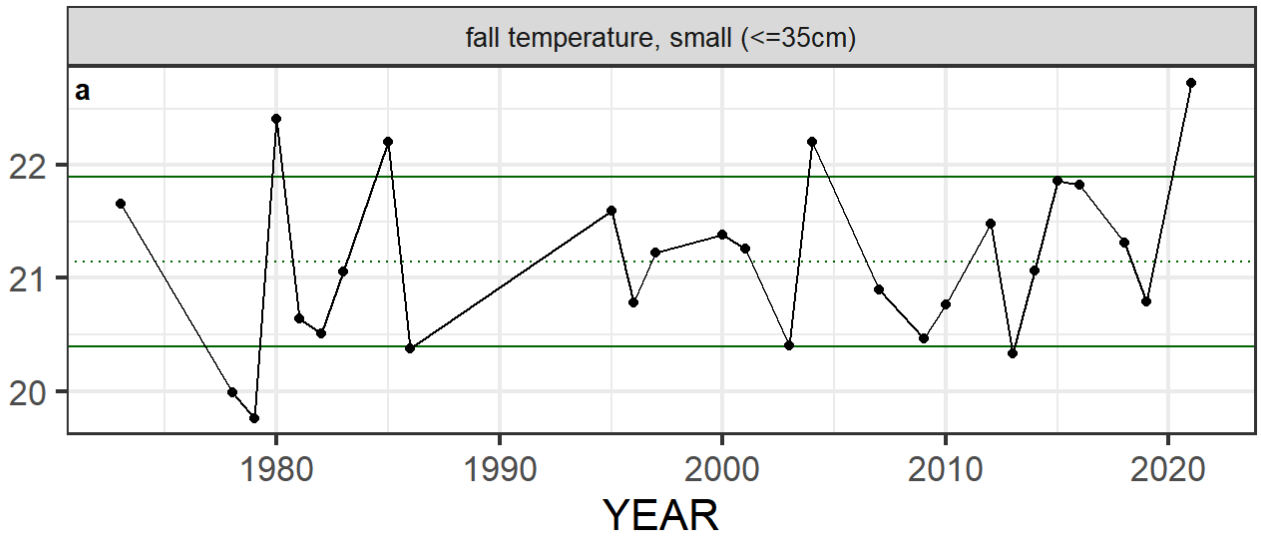


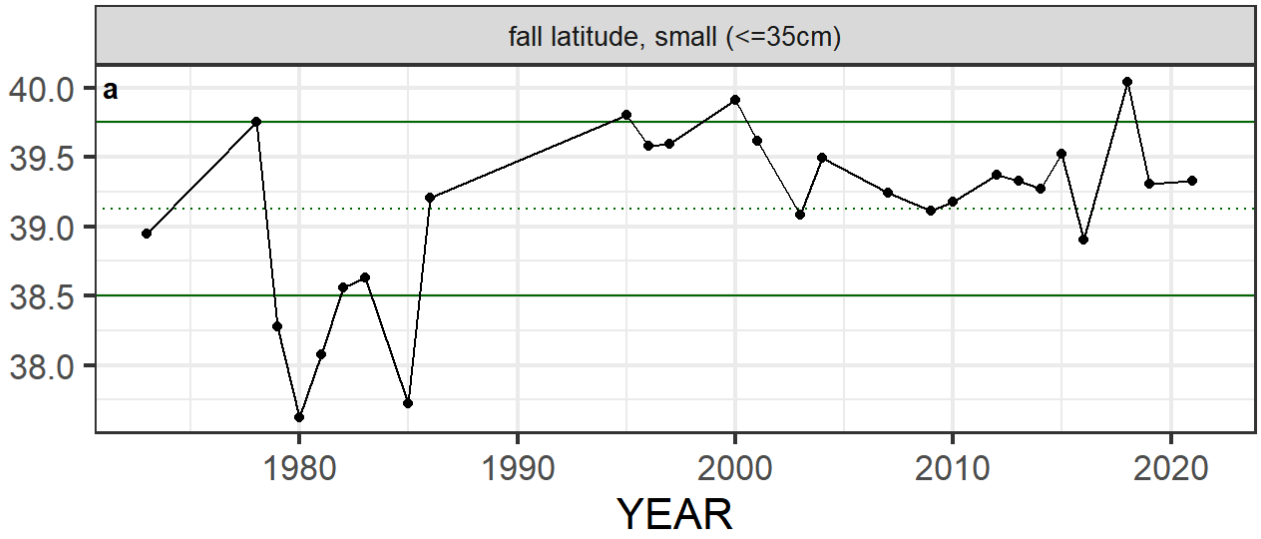

```
## Error in eval(term, data, enclos = pf) :
##   invalid 'envir' argument of type 'closure'
## # Predicted values of LON
##
## YEAR | Predicted |           95% CI
## ----------------------------------
## 1971 |    -72.29 | [-72.60, -71.98]
## 1976 |    -73.37 | [-73.60, -73.15]
## 1982 |    -73.08 | [-73.34, -72.83]
## 1987 |    -71.21 | [-71.54, -70.87]
## 1993 |    -69.97 | [-70.38, -69.57]
## 1999 |    -72.33 | [-72.56, -72.10]
## 2004 |    -72.40 | [-72.55, -72.25]
## 2019 |    -70.79 | [-71.10, -70.48]
##
## Adjusted for:
## * LAT =  40.55
## * DOY = 278.83
```

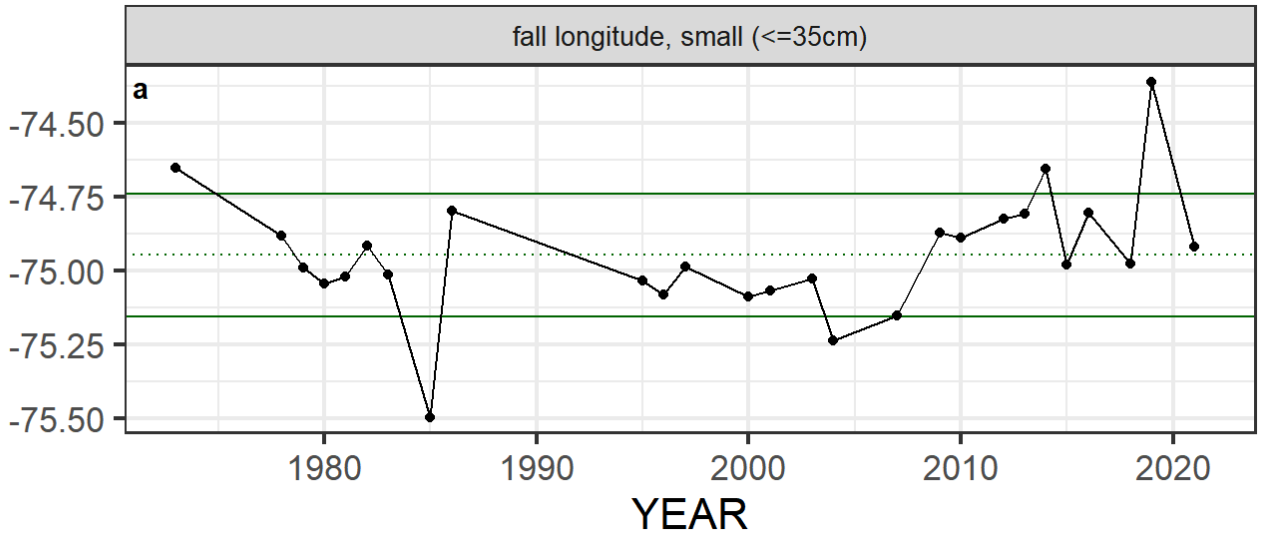


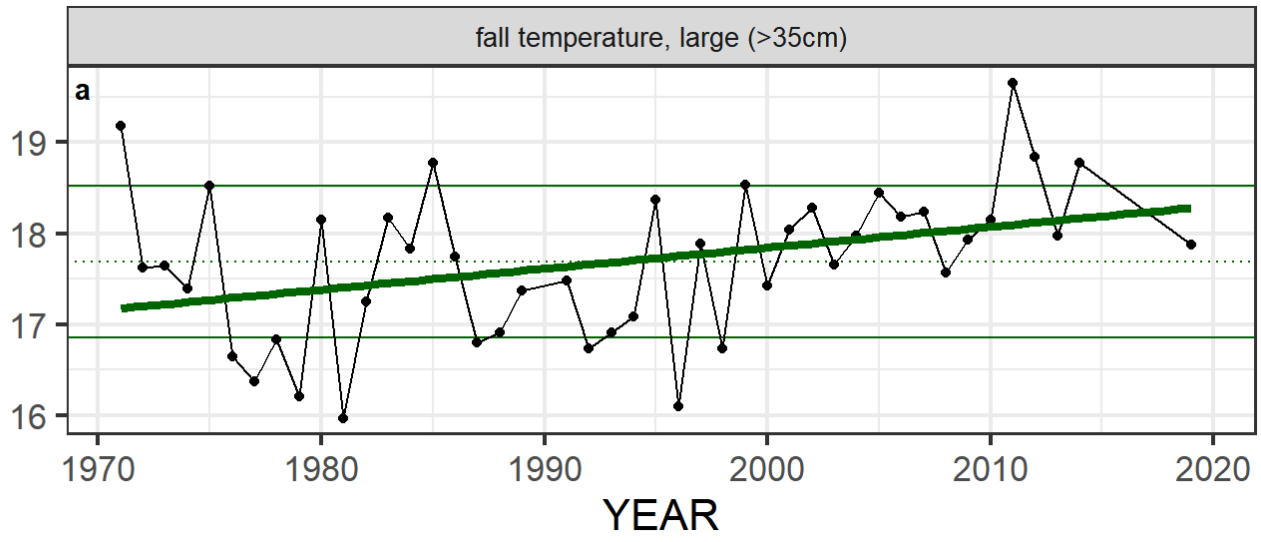

```
##              df       AIC
## model  90.00000 10623.950
## model2 97.76973  9421.225
```

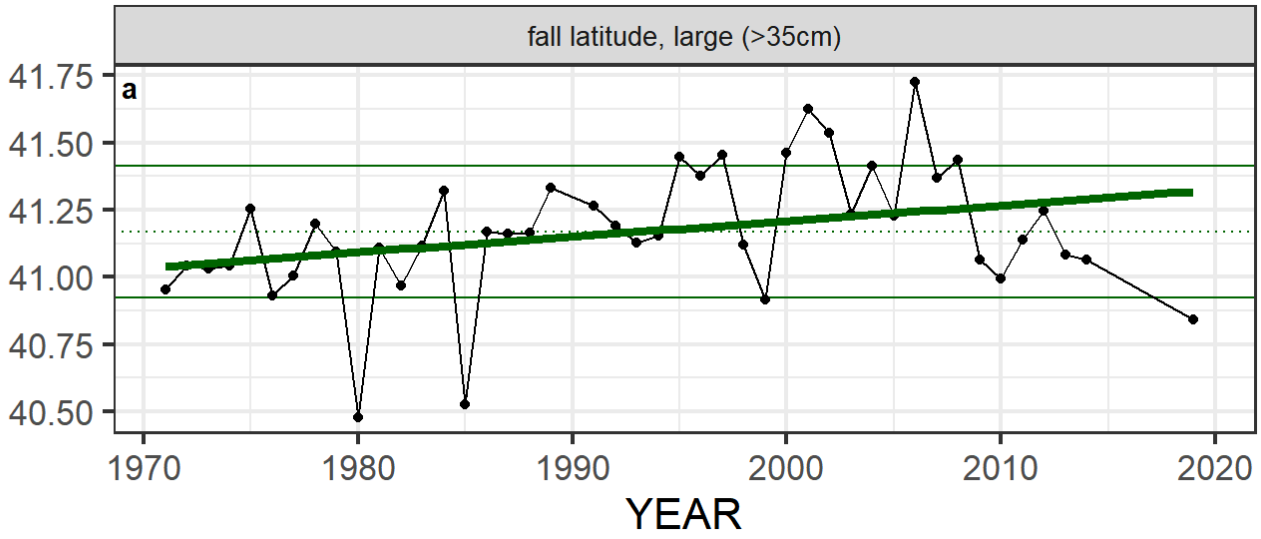
0.5 ESP figures

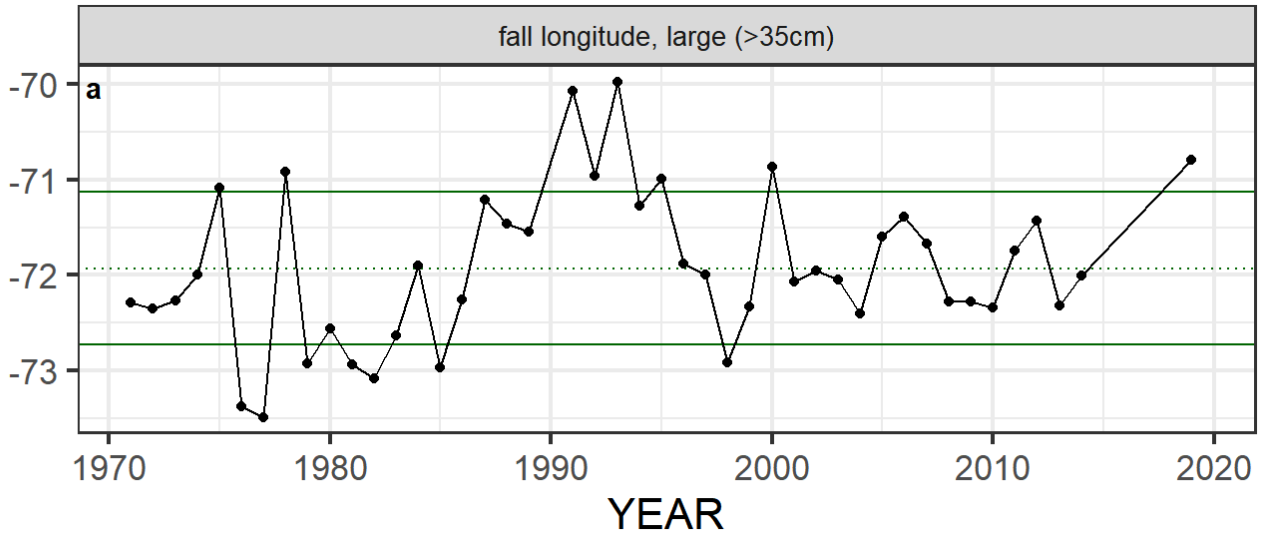

fall temperature, small (<=35cm)
a
22
21
20
1980
1990
2000
2010
2020
YEAR


fall latitude, small (<=35cm)
a
40.0
39.5
39.0
38.5
38.0
1980
1990
2000
2010
2020
YEAR


fall longitude, small (<=35cm)
a
-74.50
-74.75
-75.00
-75.25
-75.50
1980
1990
2000
2010
2020
YEAR


fall temperature, large (>35cm)
a
19
18
17
16
1970
1980
1990
2000
2010
2020
YEAR


fall latitude, large (>35cm)
a
41.75
41.50
41.25
41.00
40.75
40.50
1970
1980
1990
2000
2010
2020
YEAR


fall longitude, large (>35cm)
a
-70
-71
-72
-73
1970
1980
1990
2000
2010
2020
YEAR