Negative Binomial GAMs, Combined Models, 25 km

Yvonne Barkley

10/4/2020

Load libraries

```
library(tidyverse)
library(mgcv)
library(corrplot)
library(geoR)
library(tidymv)
library(here)
```

Research question:

What environmental variables characterize sperm whale habitat?

Hypothesis: Sperm whales are found in deep, productive offshore waters.

This markdown documents the model selection process and the resulting best-fit models. The first set of models does not include a spatial smoother. Models from this set are compared to a second set of models that include a spatial smoother to account for spatial autocorrelation. The literature suggests using various error distributions as the GAM model family. The full model for each set compares a negative binomial model with a model built using a Tweedie distribution, which are both appropriate for this data set. Overall, the negative binomial models performed better, relatively speaking, and were used for the final models for the model sets with and without spatial smoothers.

Load universal variables

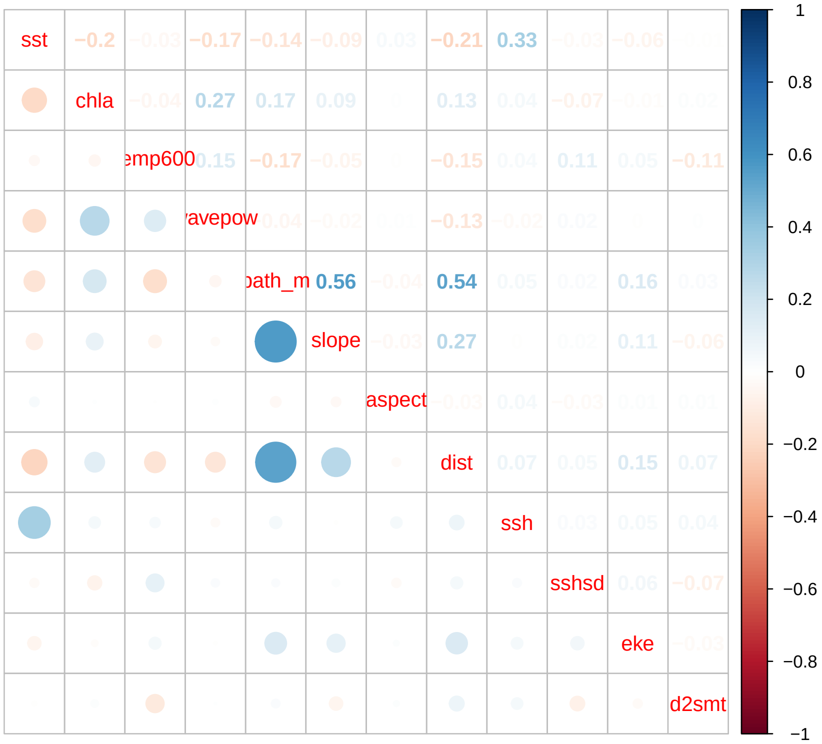
```
# Values used for file and directory names
survey = "AllSurveys"
gridsize = 25
loctype = "Combined"
loctype2 = "Comb"
```

Load data from 'models/data' folder

```
PmScaled <- readRDS(here::here(paste0("output/models/", loctype,
    "/data/", "CompletePm_", gridsize, "km_", loctype2, "_scaled.rda")))
# add column for log effort as offset #
PmScaled$log.effort = log(PmScaled$EffArea)
PmScaled <- subset(PmScaled, chla <= 9)  #some outliers in a handful of absences
PmScaled$distseamt.r = PmScaled$distseamt.r/1000
```

Check correlation of covariates

```
require(corrplot)
corrplot.mixed(cor(PmScaled[, 18:29]), upper = "number", lower = "circle")
```



```
# Are all correlation coefficients < |0.6|?
abs(cor(PmScaled[, 18:29])) <= 0.6
```

```
          sst  chla temp600 wavepow bath_m slope aspect  dist   ssh sshsd   eke
sst     FALSE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
chla     TRUE FALSE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
temp600  TRUE  TRUE   FALSE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
wavepow  TRUE  TRUE    TRUE   FALSE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
bath_m   TRUE  TRUE    TRUE    TRUE  FALSE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
slope    TRUE  TRUE    TRUE    TRUE   TRUE FALSE   TRUE  TRUE  TRUE  TRUE  TRUE
aspect   TRUE  TRUE    TRUE    TRUE   TRUE  TRUE  FALSE  TRUE  TRUE  TRUE  TRUE
dist     TRUE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE FALSE  TRUE  TRUE  TRUE
ssh      TRUE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE FALSE  TRUE  TRUE
sshsd    TRUE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE FALSE  TRUE
eke      TRUE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE FALSE
d2smt    TRUE  TRUE    TRUE    TRUE   TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
        d2smt
sst      TRUE
chla     TRUE
temp600  TRUE
wavepow  TRUE
bath_m   TRUE
slope    TRUE
aspect   TRUE
```

```
dist      TRUE
ssh       TRUE
sshsd     TRUE
eke       TRUE
d2smt    FALSE
```

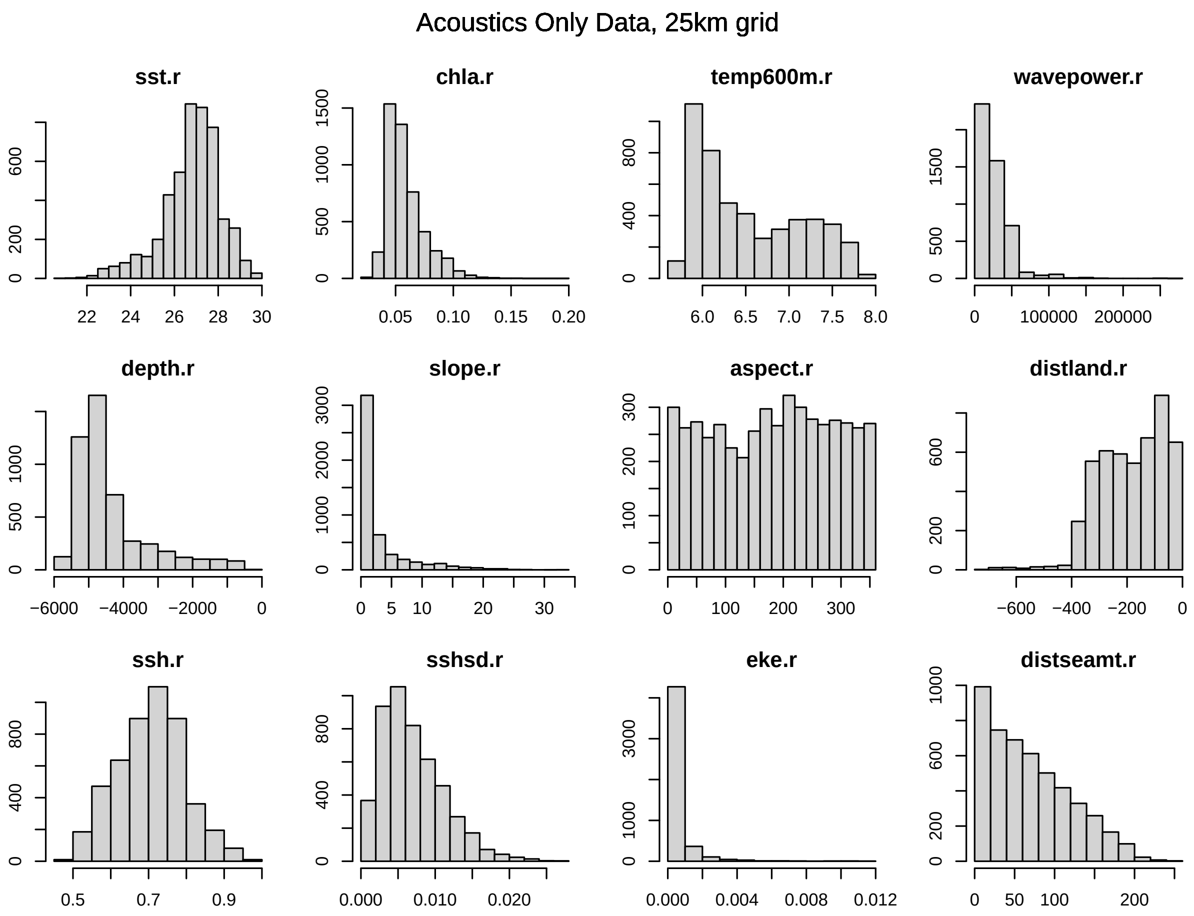
KS tests

I compared the distributions of environmental data between the whales and the absences. Plots are attached in separate powerpoint. In summary, temperature at 600 m, SSH, and chlorophyll were the only variables with significantly different distributions (p-value < 0.05). However, the D statistics were close to zero (D ~ 0.1) for each, indicating that although the distributions were different, they were not that far apart. The plots also show how similar the general shape of the distributions are between where the whales were observed and where they were absent.

Data Visualization

Histograms showing the general distribution of each environmental predictor for the entire dataset.

```r
par(mfrow = c(3, 4), mar = c(3, 3, 2, 1), oma = c(0, 0, 3, 1))

dataSet = PmScaled  #raw values

loopVec <- 30:41  #columns from PmScaled to plot

for (j in loopVec) {

    datPlot <- dataSet[, c(1, j)]

    hist(datPlot[, 2], main = colnames(datPlot)[2], ylab = "frequency",
        xlab = "")
    # plot(datPlot[,2], datPlot[,1], ylab = 'Whales', xlab =
    # colnames(datPlot)[2])
    mtext(paste0("Acoustics Only Data, ", gridsize, "km grid"),
        side = 3, line = 1, outer = TRUE, cex = 1, font = 1)

}
```



```
# dev.off()
```

Data Splitting

Split the data into train and test sets

```
require(dplyr)
splitdf <- function(dataframe, seed = NULL) {
    if (!is.null(seed))
        set.seed(seed)
    index <- 1:nrow(dataframe)
    trainindex <- sample(index, trunc(length(index) * 0.7))
    trainset <- dataframe[trainindex, ]
    testset <- dataframe[-trainindex, ]
    list(trainset = trainset, testset = testset)
}

trainComb = NULL
testComb = NULL
seed = 2

for (s in c(1641, 1303, 1604, 1705, 1706)) {
    trSub <- filter(PmScaled, survey == s)

    # subset for presences and split 70/30
    pres1 <- filter(trSub, pa > 0)  # & loc == 1) #include all presence data/acoustic encounters
    listPres <- splitdf(pres1, seed)  #output is list for train and test

    # subset for absences and split 70/30
    abs0 <- filter(trSub, pa == 0)
    listAbs <- splitdf(abs0, seed)  #output is list for train and test

    # combine train data for presence and absence
    trainAll <- rbind(listPres$trainset, listAbs$trainset)

    # combine test data for presence and absence
    testAll <- rbind(listPres$testset, listAbs$testset)

    trainComb = rbind(trainComb, trainAll)
    testComb = rbind(testComb, testAll)

    # trainAcOnly$log.effort <- log(trainAcOnly$EffArea)
    # testAcOnly$log.effort <- log(testAcOnly$EffArea)
}
saveRDS(trainComb, here::here(paste0("output/models/", loctype,
    "/data/Train_", gridsize, "km_", loctype2, "_Comb.rda")))
saveRDS(testComb, here::here(paste0("output/models/", loctype,
    "/data/Test_", gridsize, "km_", loctype2, "_Comb.rda")))

# nrow(dplyr::filter(trainAcOnly, trainAcOnly$pa >0))
# nrow(dplyr::filter(testAcOnly, testAcOnly$pa >0))
```

Generalized Additive Models

The data are treated as count data, number of sperm whale encounters per cell, and we used the negative binomial distribution to model the response variable for comparison with the Tweedie distribution. We used thin-plate regression splines (the default basis) for the smoothers of the environmental predictors. Each smoother was limited to 3 degrees of freedom (k=3) to reduce overfitting parameters per recommendations from other studies building similar types of cetaceans distribution models.The log of the effort was included as an offset to account for the variation in effort per cell.

25 km spatial scale

- NEGATIVE BINOMIAL DISTRIBUTION
- Knots contrained to k=3 according to literature on cetacean distribution models.
- Automatic term selection is uses an additional penalty term when determining the smoothness of the function (‘select’ argument = TRUE)..
- We excluded all non-significant variables (alpha=0.05) and refit the models until all variables were significant.
- REML is restricted maximum likelihood used to optimize the estimates for the smoothing parameters of each predictor variable.

Load training and test data

```
# seed 1
trainComb <- readRDS(here::here(paste0("output/models/", loctype,
    "/data/Train_", gridsize, "km_", loctype2, "_Comb.rda")))
testComb <- readRDS(here::here(paste0("output/models/", loctype,
    "/data/Test_", gridsize, "km_", loctype2, "_Comb.rda")))
```

Model Selection

SET 1

Full Models

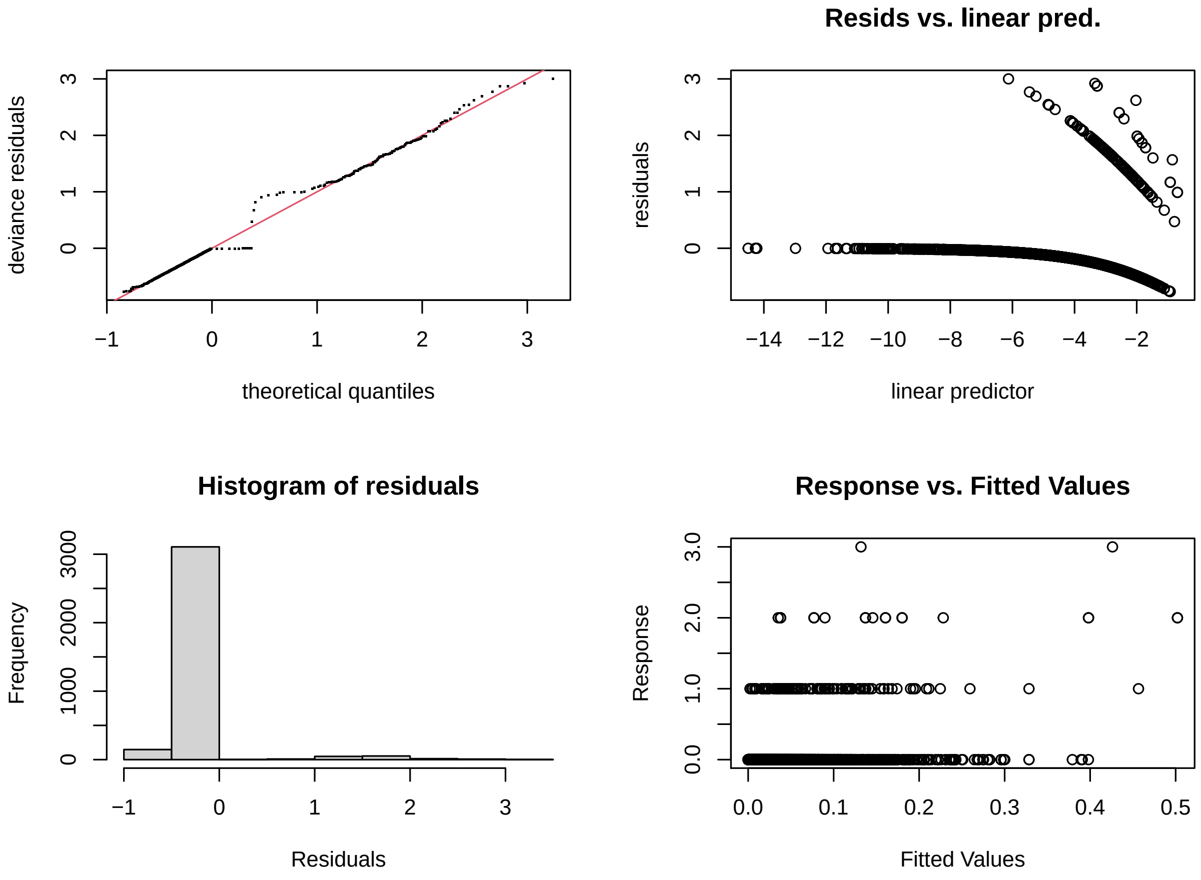
```
+ does not include spatial smoother
+ does not include slope or aspect due to the variation between left and right
```

```
require(mgcv)
nbComb <- gam(pa ~ s(bath_m, k = 3) + s(dist, k = 3) + s(d2smt,
    k = 3) + s(sst, k = 3) + s(chla, k = 3) + s(temp600, k = 3) +
    s(ssh, k = 3) + s(sshsd, k = 3) + s(eke, k = 3) + s(wavepow,
    k = 3) + offset(log.effort), data = trainComb, family = nb,
    link = "log", select = TRUE, method = "REML")
summary(nbComb)
```

```
Family: Negative Binomial(0.519)
Link function: log

Formula:
pa ~ s(bath_m, k = 3) + s(dist, k = 3) + s(d2smt, k = 3) + s(sst,
    k = 3) + s(chla, k = 3) + s(temp600, k = 3) + s(ssh, k = 3) +
    s(sshsd, k = 3) + s(eke, k = 3) + s(wavepow, k = 3) + offset(log.effort)

Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -22.34079    0.09616  -232.3   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                 edf Ref.df Chi.sq  p-value
s(bath_m)  1.853e-03      2  0.001 0.388384
s(dist)    2.001e-04      2  0.000 0.546821
s(d2smt)   2.423e-03      2  0.002 0.319005
s(sst)     1.641e+00      2  9.187 0.003074 **
s(chla)    8.951e-01      2  8.151 0.002209 **
s(temp600) 1.713e+00      2 24.268 8.06e-07 ***
s(ssh)     8.280e-01      2  4.732 0.016055 *
s(sshsd)   1.354e+00      2 11.055 0.000566 ***
s(eke)     3.096e-01      2  0.393 0.258038
s(wavepow) 5.179e-05      2  0.000 0.869499
---
```

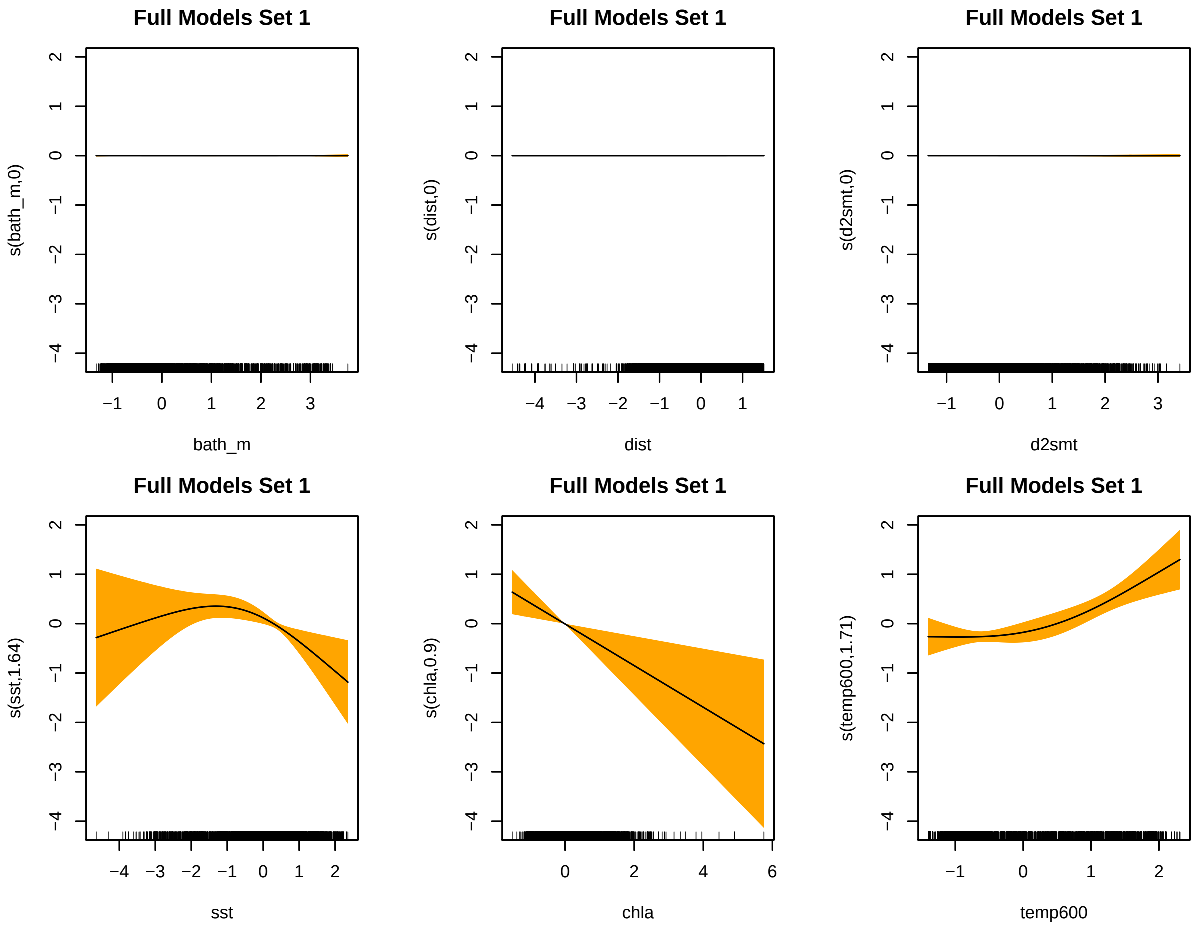
```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.0705   Deviance explained = 9.33%
-REML =  563.8  Scale est. = 1         n = 3387
```

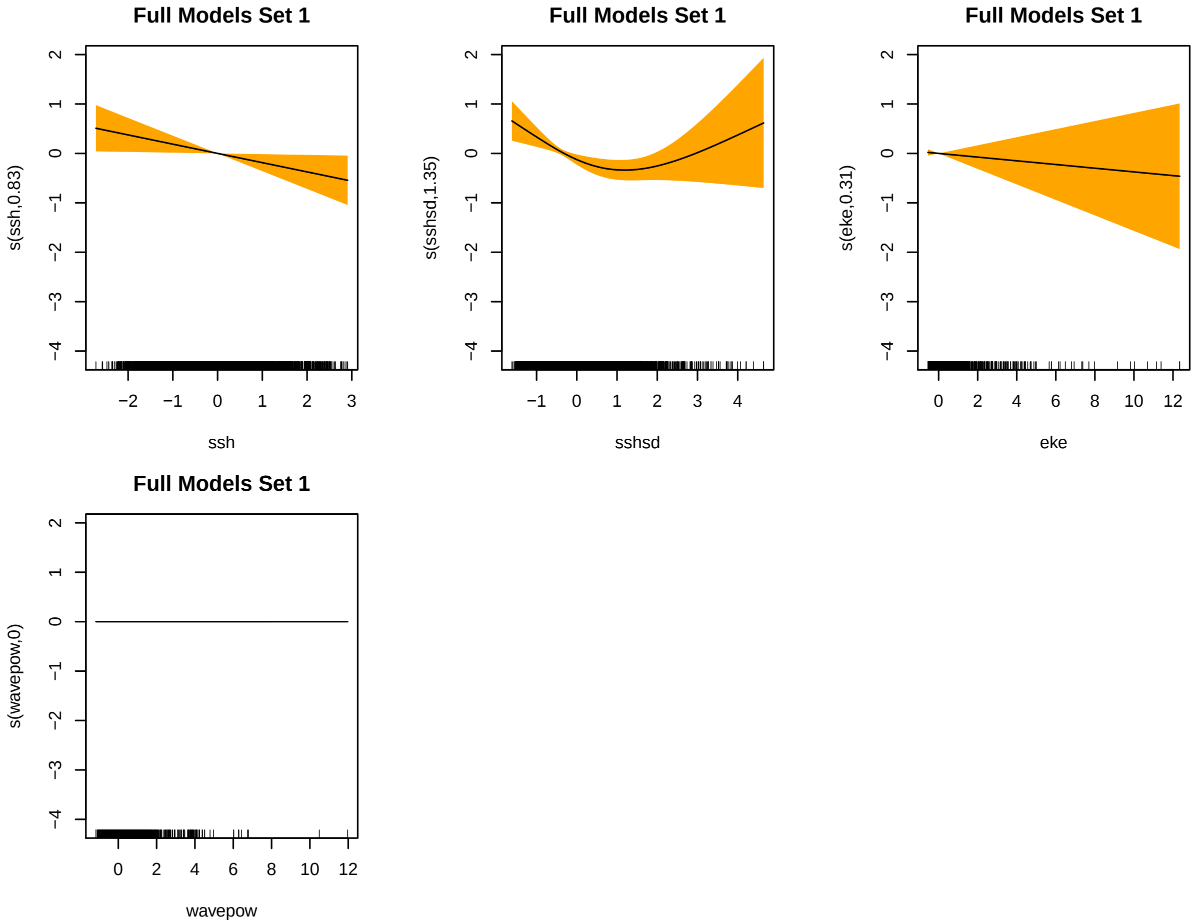
```
# model diagnostics
par(mfrow = c(2, 2))
gam.check(nbComb)
```



```
Method: REML   Optimizer: outer newton
full convergence after 15 iterations.
Gradient range [-0.0002370071,6.913598e-05]
(score 563.8002 & scale 1).
eigenvalue range [-1.369955e-05,6.541124].
Model rank =  21 / 21

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(bath_m)  2.00e+00 1.85e-03    0.79  <2e-16 ***
s(dist)    2.00e+00 2.00e-04    0.81   0.005 **
```

```
s(d2smt)   2.00e+00 2.42e-03    0.81  <2e-16 ***
s(sst)     2.00e+00 1.64e+00    0.79  <2e-16 ***
s(chla)    2.00e+00 8.95e-01    0.80  <2e-16 ***
s(temp600) 2.00e+00 1.71e+00    0.70  <2e-16 ***
s(ssh)     2.00e+00 8.28e-01    0.77  <2e-16 ***
s(sshsd)   2.00e+00 1.35e+00    0.80   0.005 **
s(eke)     2.00e+00 3.10e-01    0.78  <2e-16 ***
s(wavepow) 2.00e+00 5.18e-05    0.78  <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```





Trying the Tweedie for comparison

```
twComb <- gam(pa ~ s(bath_m, k = 3) + s(dist, k = 3) + s(d2smt,
    k = 3) + s(sst, k = 3) + s(chla, k = 3) + s(temp600, k = 3) +
    s(ssh, k = 3) + s(sshsd, k = 3) + s(eke, k = 3) + s(wavepow,
    k = 3) + offset(log.effort), data = trainComb, family = tw,
    link = "log", select = TRUE, method = "REML")
summary(twComb)
```

```
Family: Tweedie(p=1.01)
Link function: log

Formula:
pa ~ s(bath_m, k = 3) + s(dist, k = 3) + s(d2smt, k = 3) + s(sst,
    k = 3) + s(chla, k = 3) + s(temp600, k = 3) + s(ssh, k = 3) +
    s(sshsd, k = 3) + s(eke, k = 3) + s(wavepow, k = 3) + offset(log.effort)

Parametric coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -22.35497    0.09245  -241.8   <2e-16 ***
```

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

Approximate significance of smooth terms:
                 edf Ref.df      F  p-value
s(bath_m)  1.441e-04      2  0.000 0.383560
s(dist)    5.370e-05      2  0.000 0.640297
s(d2smt)   1.574e-01      2  0.093 0.277147
s(sst)     1.708e+00      2  5.904 0.000816 ***
s(chla)    9.103e-01      2  4.891 0.000919 ***
s(temp600) 1.786e+00      2 16.931 6.63e-09 ***
s(ssh)     8.647e-01      2  3.162 0.006592 **
s(sshsd)   1.519e+00      2  8.785 1.71e-05 ***
s(eke)     3.866e-01      2  0.278 0.228618
s(wavepow) 4.553e-05      2  0.000 0.854695
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.0765   Deviance explained = 9.06%
-REML = 396.15  Scale est. = 1.0359    n = 3387
```

Table 1: Model Comparison Metrics

Full Models	ExpDev	AIC
Full Tweedie-twComb	9.06%	3323.01
Full Neg Bin-nbComb	9.33%	1122.42

Reduced Models

- Negative Binomial -> higher explained deviance, lower AIC than Tweedie
- Removed non-significant variables:
  - depth
  - distance to land
  - distance to seamount
  - eke
  - wave power
- Keep:
  - SST
  - Chl a
  - Temp at 600 m
  - SSH
  - SSHsd

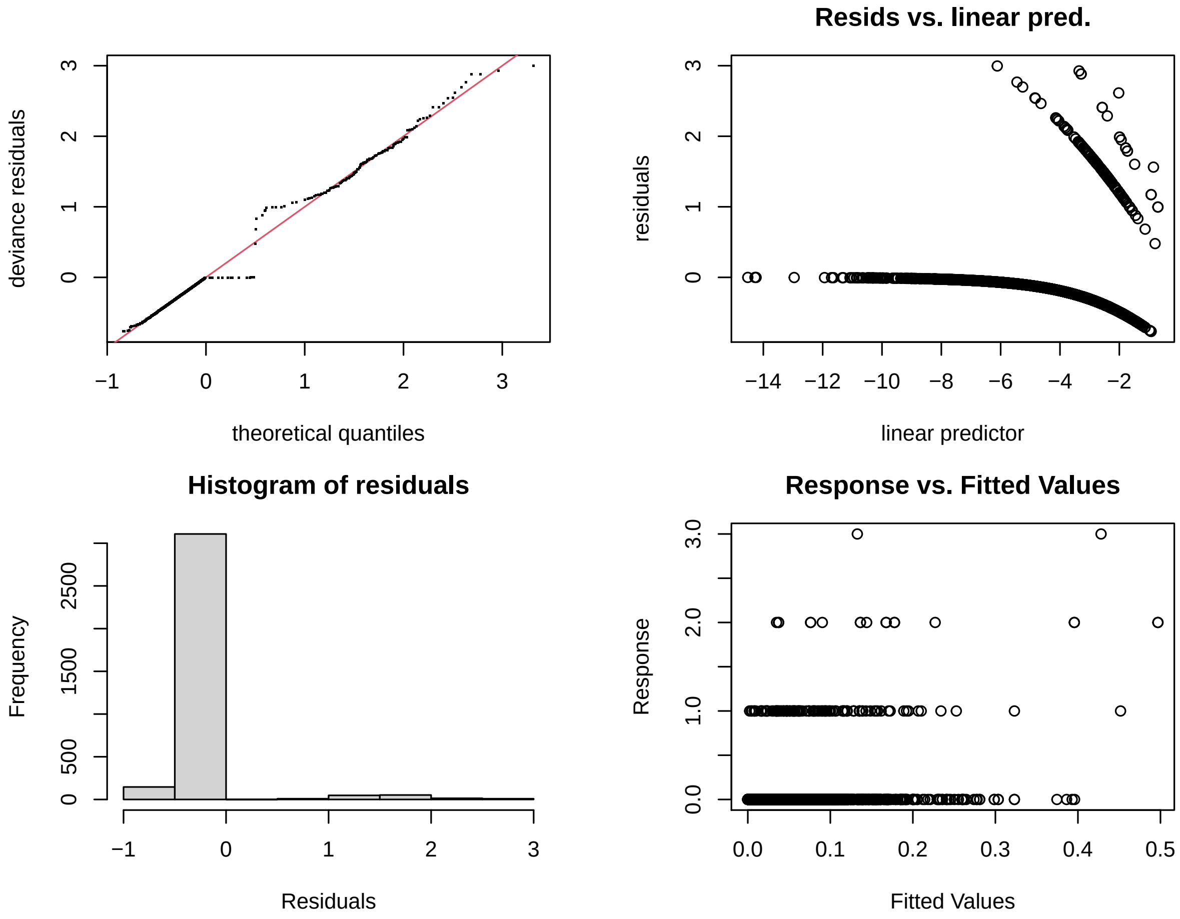
```
Family: Negative Binomial(0.518)
Link function: log

Formula:
pa ~ s(sst, k = 3) + s(chla, k = 3) + s(temp600, k = 3) + s(ssh,
    k = 3) + s(sshsd, k = 3) + offset(log.effort)
```

```
Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -22.34000    0.09613  -232.4   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Approximate significance of smooth terms:
              edf Ref.df Chi.sq  p-value
s(sst)     1.6428      2  9.128 0.003294 **
s(chla)    0.8953      2  8.173 0.002187 **
s(temp600) 1.7118      2 24.248 8.15e-07 ***
s(ssh)     0.8334      2  4.919 0.014480 *
s(sshsd)   1.3487      2 11.136 0.000538 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.0702   Deviance explained = 9.23%
-REML = 563.82  Scale est. = 1         n = 3387
```

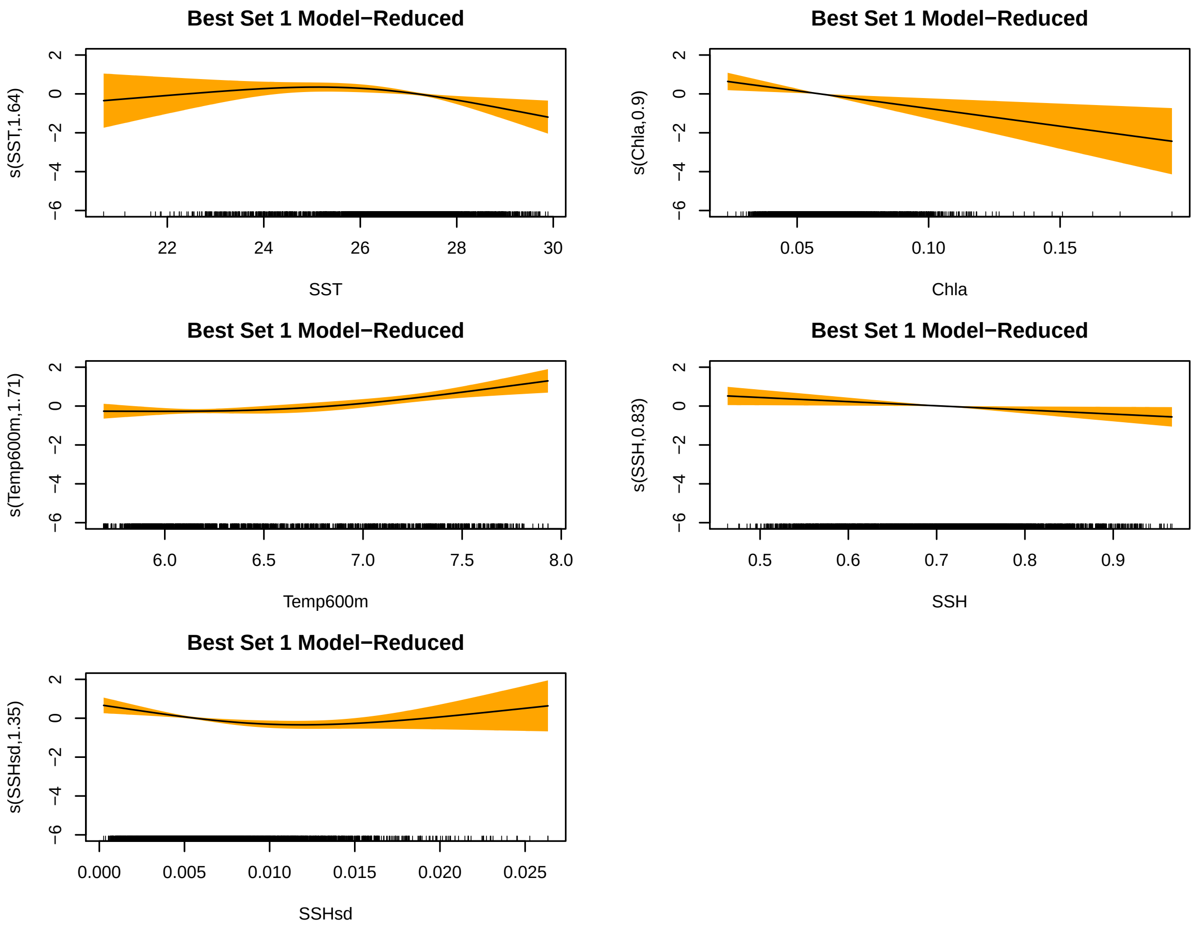
```r
nbCombb <- gam(pa ~ s(SST, k = 3) + s(Chla, k = 3) + s(Temp600m,
    k = 3) + s(SSH, k = 3) + s(SSHsd, k = 3) + offset(log.effort),
    data = trainComb, family = nb, link = "log", select = TRUE,
    method = "REML")
summary(nbCombb)
```

```
Family: Negative Binomial(0.518)
Link function: log

Formula:
pa ~ s(SST, k = 3) + s(Chla, k = 3) + s(Temp600m, k = 3) + s(SSH,
    k = 3) + s(SSHsd, k = 3) + offset(log.effort)

Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -22.34000    0.09613  -232.4   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Approximate significance of smooth terms:
               edf Ref.df Chi.sq  p-value
s(SST)      1.6428      2  9.128 0.003294 **
s(Chla)     0.8953      2  8.173 0.002187 **
s(Temp600m) 1.7118      2 24.248 8.15e-07 ***
s(SSH)      0.8334      2  4.919 0.014480 *
s(SSHsd)    1.3487      2 11.136 0.000538 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.0702   Deviance explained = 9.23%
-REML = 563.82  Scale est. = 1         n = 3387
```

```
# Model diagnostics
par(mar = c(4, 4, 3, 3), mfrow = c(2, 2))
gam.check(nbCombb)
```



```
Method: REML   Optimizer: outer newton
full convergence after 13 iterations.
Gradient range [-1.836518e-05,1.098891e-05]
(score 563.8187 & scale 1).
Hessian positive definite, eigenvalue range [1.663515e-05,6.55339].
Model rank =  11 / 11

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(SST)      2.000 1.643    0.79  <2e-16 ***
s(Chla)     2.000 0.895    0.80  <2e-16 ***
s(Temp600m) 2.000 1.712    0.70  <2e-16 ***
s(SSH)      2.000 0.833    0.77  <2e-16 ***
s(SSHsd)    2.000 1.349    0.80  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



SET 2

Full Models: Includes s(Longitude,Latitude)

Includes 2D Lat-Lon smoother to account for spatial structure in the data and fit the spatial variation not explained by the other predictors
* temperature at 600m is STILL significant compared to the previous models, including the Acoustics Only models
* SST, Chlorophyll and SSHsd remain significant

```
nbCombLL <- gam(pa ~ s(Longitude, Latitude) + s(bath_m, k = 3) +
    s(dist, k = 3) + s(d2smt, k = 3) + s(sst, k = 3) + s(chla,
    k = 3) + s(temp600, k = 3) + s(ssh, k = 3) + s(sshsd, k = 3) +
    s(eke, k = 3) + s(wavepow, k = 3) + offset(log.effort), data = trainComb,
    family = nb, link = "log", select = TRUE, method = "REML")
summary(nbCombLL)
```

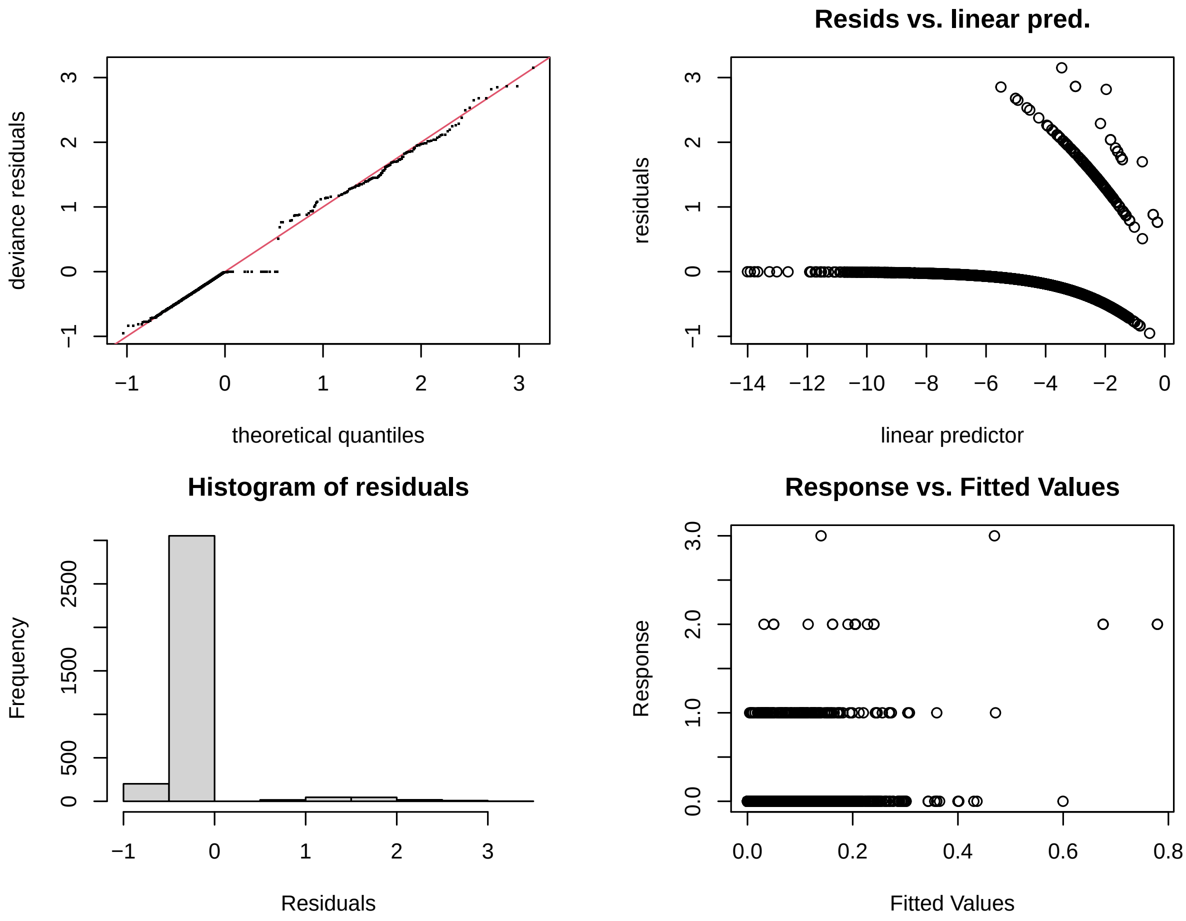
```
Family: Negative Binomial(0.85)
Link function: log

Formula:
```

```
pa ~ s(Longitude, Latitude) + s(bath_m, k = 3) + s(dist, k = 3) +
    s(d2smt, k = 3) + s(sst, k = 3) + s(chla, k = 3) + s(temp600,
    k = 3) + s(ssh, k = 3) + s(sshsd, k = 3) + s(eke, k = 3) +
    s(wavepow, k = 3) + offset(log.effort)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -22.5653     0.1131  -199.5   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Approximate significance of smooth terms:
                            edf Ref.df Chi.sq  p-value
s(Longitude,Latitude) 1.167e+01     29 64.179 1.20e-12 ***
s(bath_m)             4.707e-01      2  0.908  0.15623
s(dist)               2.719e-01      2  0.291  0.18619
s(d2smt)              6.139e-01      2  1.568  0.09329 .
s(sst)                1.226e+00      2  3.356  0.05181 .
s(chla)               8.896e-01      2  7.621  0.00238 **
s(temp600)            8.369e-01      2  5.034  0.00664 **
s(ssh)                2.045e-04      2  0.000  1.00000
s(sshsd)              1.631e+00      2 19.623 6.29e-06 ***
s(eke)                7.248e-04      2  0.001  0.34291
s(wavepow)            1.875e-04      2  0.000  0.65732
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.108   Deviance explained = 17.9%
-REML =  547.9  Scale est. = 1         n = 3387
```

```r
AIC(nbCombLL)
```

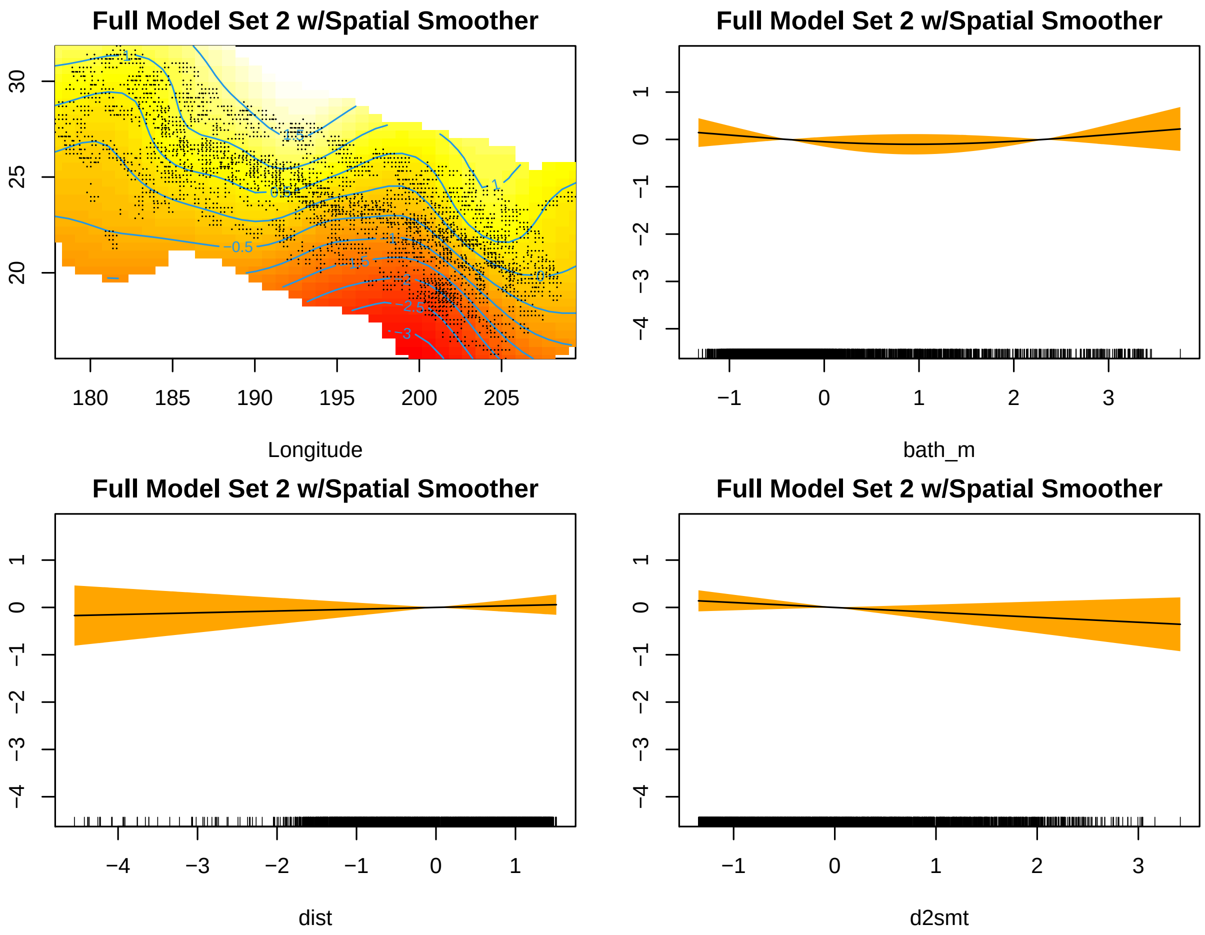
```
[1] 1087.018
```

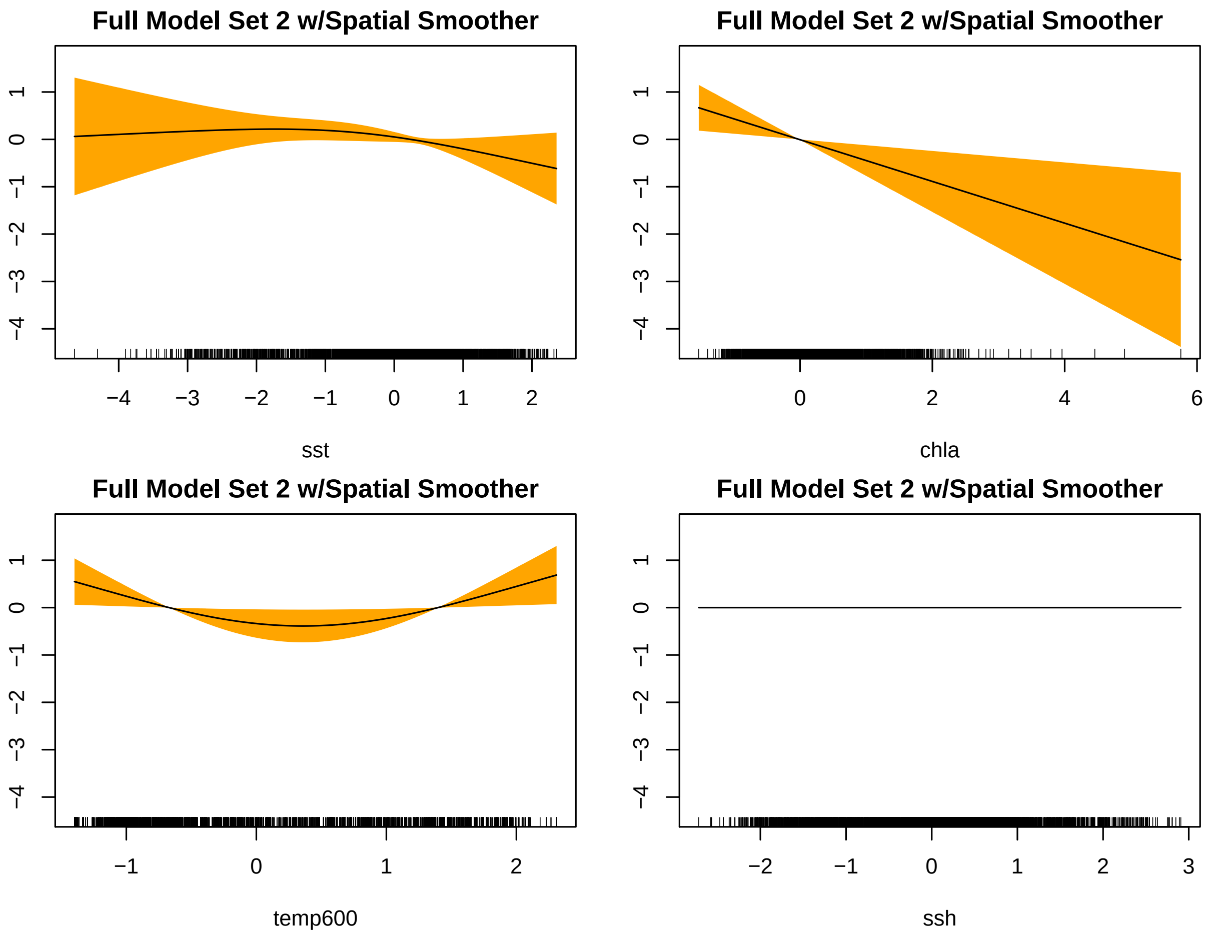
```r
# model diagnostics
par(mar = c(4, 4, 3, 3), mfrow = c(2, 2))
gam.check(nbCombLL)
```

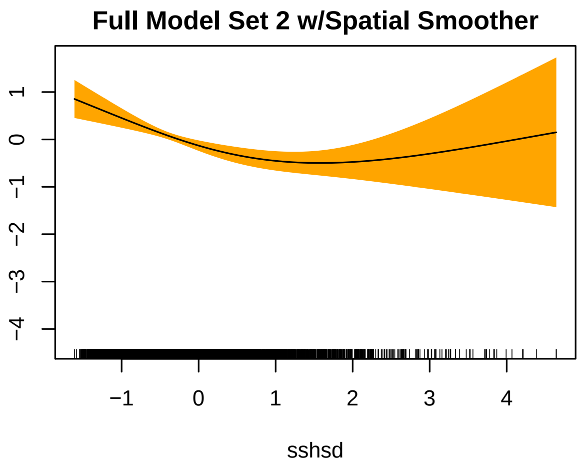


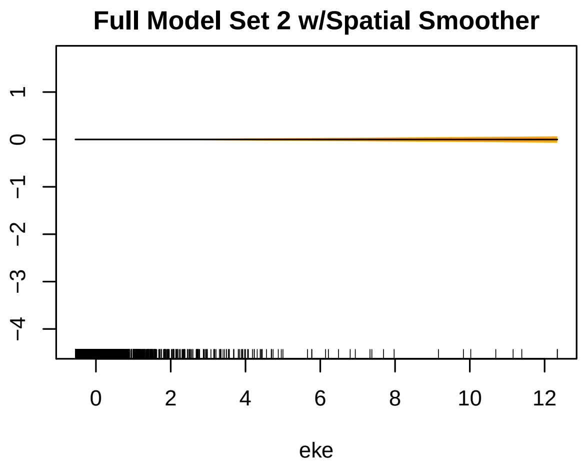
```
Method: REML   Optimizer: outer newton
full convergence after 17 iterations.
Gradient range [-0.0002350384,7.293134e-05]
(score 547.9022 & scale 1).
Hessian positive definite, eigenvalue range [1.861338e-06,4.241927].
Model rank =  50 / 50

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(Longitude,Latitude) 2.90e+01 1.17e+01    0.81   0.005 **
s(bath_m)             2.00e+00 4.71e-01    0.81   0.005 **
s(dist)               2.00e+00 2.72e-01    0.83   0.005 **
s(d2smt)              2.00e+00 6.14e-01    0.83   0.010 **
s(sst)                2.00e+00 1.23e+00    0.80  <2e-16 ***
s(chla)               2.00e+00 8.90e-01    0.82   0.005 **
s(temp600)            2.00e+00 8.37e-01    0.72  <2e-16 ***
s(ssh)                2.00e+00 2.05e-04    0.79  <2e-16 ***
s(sshsd)              2.00e+00 1.63e+00    0.81  <2e-16 ***
s(eke)                2.00e+00 7.25e-04    0.81  <2e-16 ***
s(wavepow)            2.00e+00 1.87e-04    0.80  <2e-16 ***
---
```

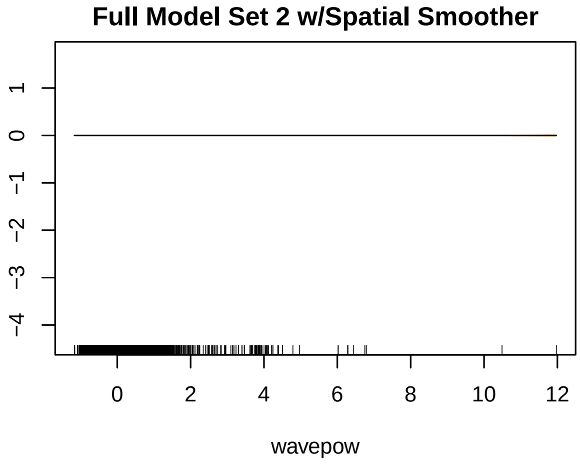
```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```




Full Model Set 2 w/Spatial Smoother
1
0
−1
−2
−3
−4
−4
−3
−2
−1
0
1
2
sst
Full Model Set 2 w/Spatial Smoother
1
0
−1
−2
−3
−4
0
2
4
6
chla
Full Model Set 2 w/Spatial Smoother
1
0
−1
−2
−3
−4
−1
0
1
2
temp600
Full Model Set 2 w/Spatial Smoother
1
0
−1
−2
−3
−4
−2
−1
0
1
2
3
ssh







Checking Tweedie for comparison

Same full model set-up, results in lower explained deviance and higher AIC.

```
twCombLL <- gam(pa ~ s(Longitude, Latitude) + s(bath_m, k = 3) +
    s(dist, k = 3) + s(d2smt, k = 3) + s(sst, k = 3) + s(chla,
    k = 3) + s(temp600, k = 3) + s(ssh, k = 3) + s(sshsd, k = 3) +
    s(eke, k = 3) + s(wavepow, k = 3) + offset(log.effort), data = trainComb,
    family = tw, link = "log", select = TRUE, method = "REML")
summary(twCombLL)
```

```
Family: Tweedie(p=1.01)
Link function: log

Formula:
pa ~ s(Longitude, Latitude) + s(bath_m, k = 3) + s(dist, k = 3) +
    s(d2smt, k = 3) + s(sst, k = 3) + s(chla, k = 3) + s(temp600,
    k = 3) + s(ssh, k = 3) + s(sshsd, k = 3) + s(eke, k = 3) +
    s(wavepow, k = 3) + offset(log.effort)

Parametric coefficients:
```

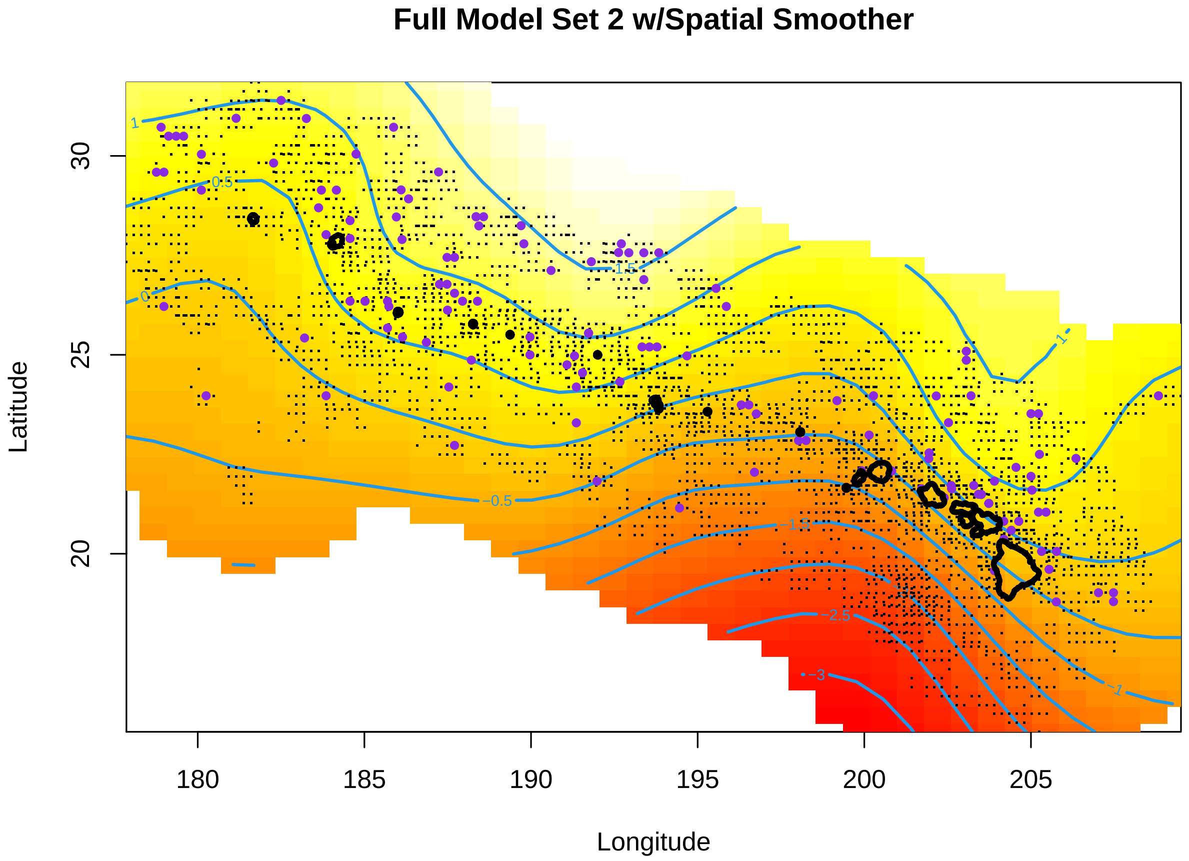


Figure 1: Purple dots represent acoustically detected encounters. Black dots are all data points(grid centroids)

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -22.5899     0.1126  -200.7   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                            edf Ref.df      F  p-value
s(Longitude,Latitude) 1.349e+01     29  2.642 2.94e-14 ***
s(bath_m)             7.093e-01      2  0.742  0.12067
s(dist)               7.220e-05      2  0.000  0.36112
s(d2smt)              6.579e-01      2  0.964  0.07211 .
s(sst)                1.268e+00      2  2.012  0.03345 *
s(chla)               9.006e-01      2  4.356  0.00125 **
s(temp600)            8.766e-01      2  3.521  0.00173 **
s(ssh)                2.914e-05      2  0.000  1.00000
s(sshsd)              1.690e+00      2 13.271 1.58e-07 ***
s(eke)                2.951e-04      2  0.000  0.34842
s(wavepow)            4.927e-05      2  0.000  0.58068
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.118   Deviance explained = 17.6%
-REML = 376.76  Scale est. = 1.0337    n = 3387
```

Table 2: Model Comparison Metrics

Full Models	ExpDev	AIC
Full Tweedie w/ s(Lon,Lat)-twCombLL	17.59%	3784.56
Full Neg Bin w/ s(Lon,Lat)-nbCombLL	17.9%	1087.02

Reduced Models

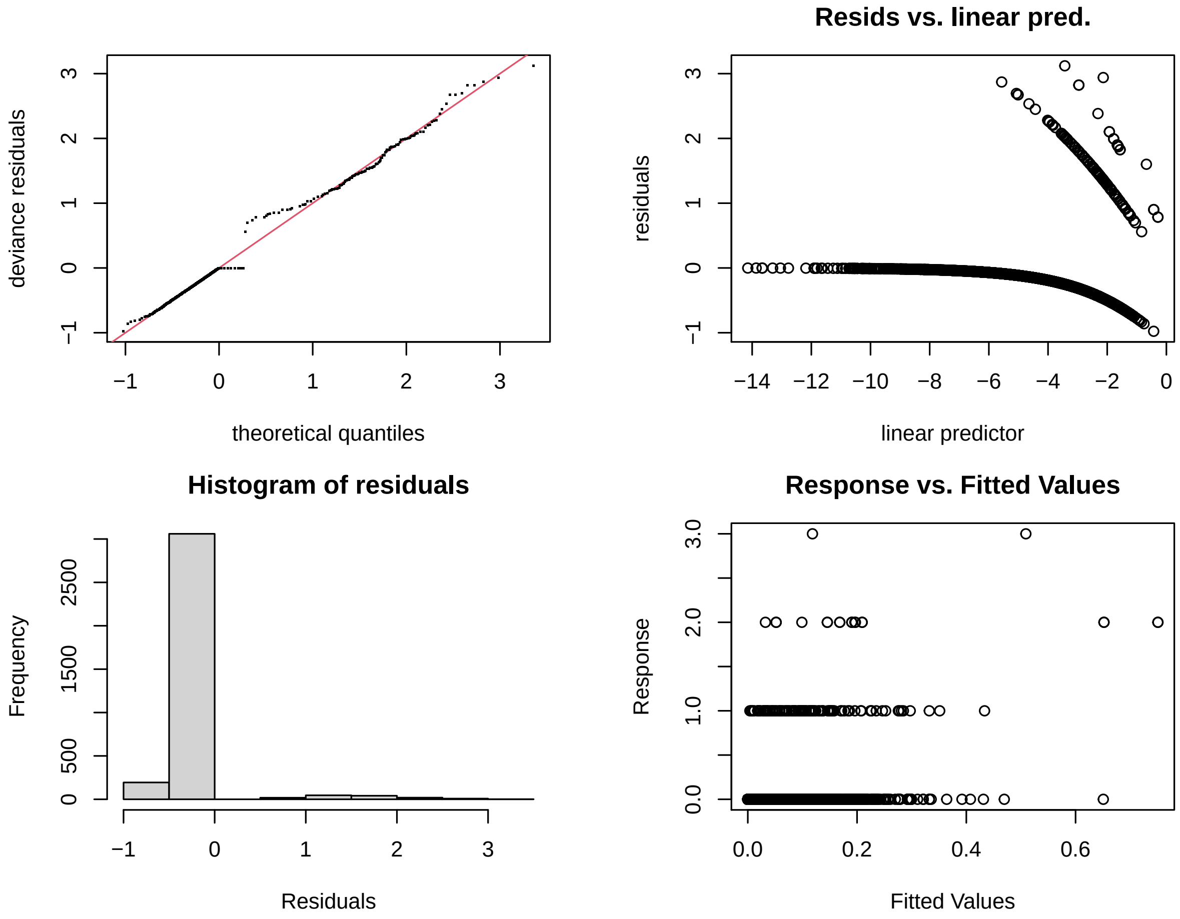
- Negative Binomial: higher explained deviance, lower AIC than Tweedie
- Removed non-significant variables:
  - depth
  - distance to land
  - distance to seamount
  - SST
  - SSH
  - eke
  - wave power
- Keep:
  - Lon, Lat
  - chlorophyll
  - temp at 600 m
  - SSHsd

```
nbCombLLb2 <- gam(pa ~ s(Longitude, Latitude) + s(Chla, k = 3) +
    s(Temp600m, k = 3) + s(SSHsd, k = 3) + offset(log.effort),
    data = trainComb, family = nb, link = "log", select = TRUE,
```

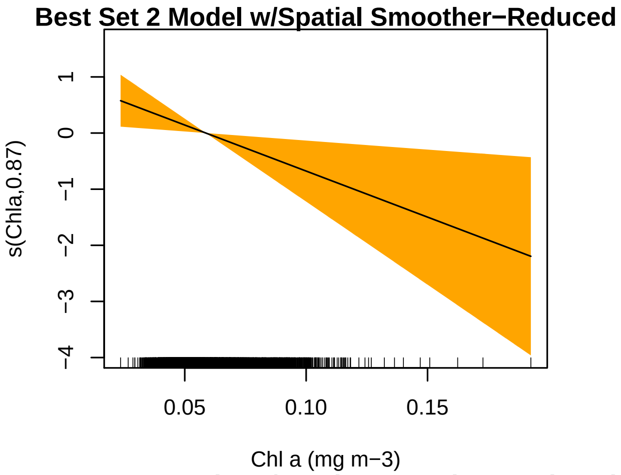
```r
    method = "REML")
summary(nbCombLLb2)
```

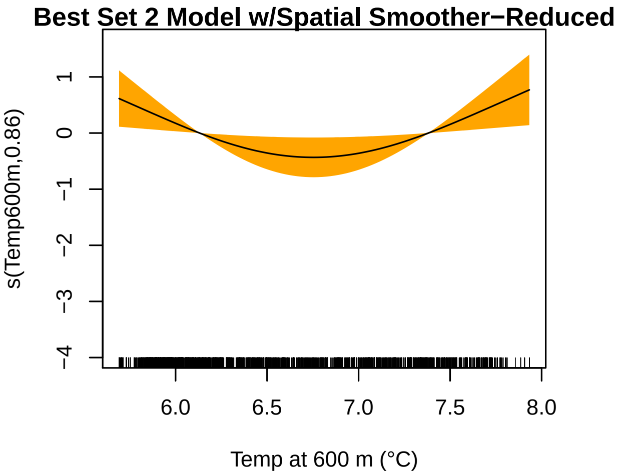
```
Family: Negative Binomial(0.811)
Link function: log

Formula:
pa ~ s(Longitude, Latitude) + s(Chla, k = 3) + s(Temp600m, k = 3) +
    s(SSHsd, k = 3) + offset(log.effort)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -22.5671     0.1139  -198.2   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Approximate significance of smooth terms:
                          edf Ref.df Chi.sq  p-value
s(Longitude,Latitude) 13.2573     29 73.108 2.17e-13 ***
s(Chla)                0.8682      2  6.216  0.00622 **
s(Temp600m)            0.8597      2  5.998  0.00400 **
s(SSHsd)               1.6368      2 19.931 5.66e-06 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.105   Deviance explained = 17.5%
-REML = 548.91  Scale est. = 1         n = 3387
```

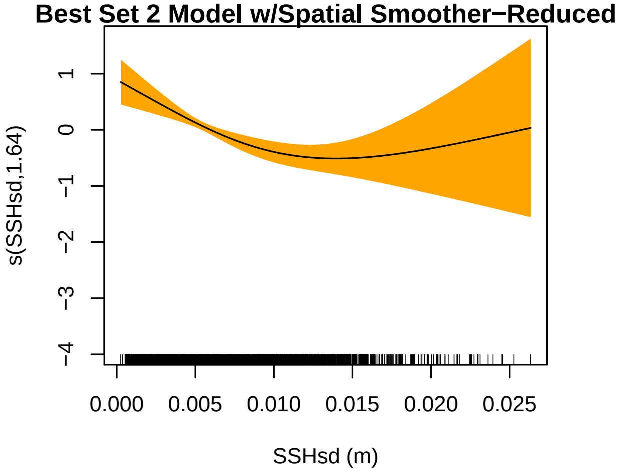
```r
# Model Diagnostics
par(mar = c(4, 4, 3, 3), mfrow = c(2, 2))
gam.check(nbCombLLb2)
```

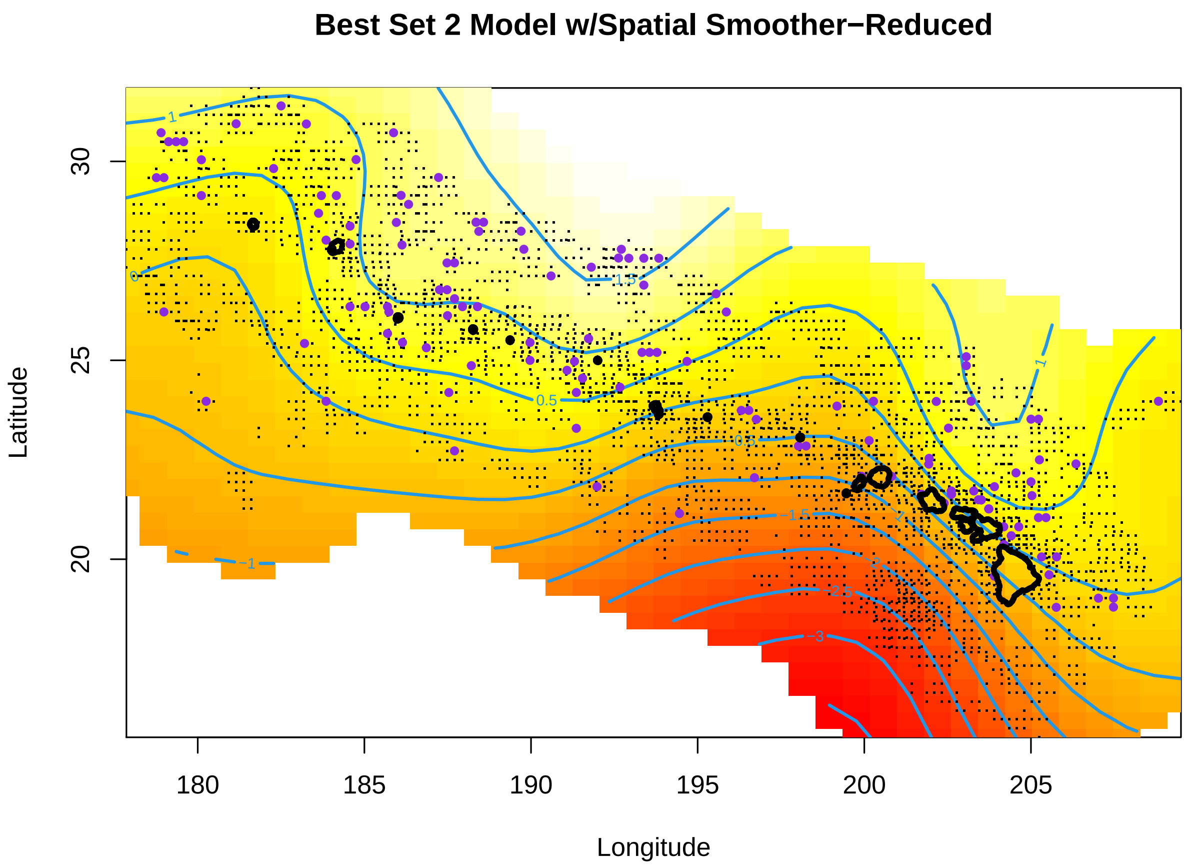


```
Method: REML   Optimizer: outer newton
full convergence after 10 iterations.
Gradient range [-0.0002756909,0.0001076932]
(score 548.9084 & scale 1).
eigenvalue range [-1.43239e-05,4.480024].
Model rank =  36 / 36

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(Longitude,Latitude) 29.000 13.257    0.81  <2e-16 ***
s(Chla)                2.000  0.868    0.82   0.005 **
s(Temp600m)            2.000  0.860    0.72  <2e-16 ***
s(SSHsd)               2.000  1.637    0.81  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


Best Set 2 Model w/Spatial Smoother−Reduced
s(Chla,0.87)
1
0
−1
−2
−3
−4
0.05
0.10
0.15
Chl a (mg m−3)


Best Set 2 Model w/Spatial Smoother−Reduced
s(Temp600m,0.86)
1
0
−1
−2
−3
−4
6.0
6.5
7.0
7.5
8.0
Temp at 600 m (°C)


Best Set 2 Model w/Spatial Smoother−Reduced
s(SSHsd,1.64)
1
0
−1
−2
−3
−4
0.000
0.005
0.010
0.015
0.020
0.025
SSHsd (m)



Assess Trained Models using Test Data

```
require(magrittr)
require(dplyr)

#### For twCombc, no spatial smoother ####
nbTrainFinal <- trainComb %>% mutate(resid = resid(nbCombb),
    predict = predict(nbCombb))
predTrain <- predict.gam(nbCombb, type = "response", se.fit = TRUE)  #calculate MSE for these
# to compare with test set. If they're super different,
# speaks to the genrality of the model
nbTrainFinal$fit <- predTrain$fit
nbTrainFinal$se.fit <- predTrain$se.fit
# using scale of 0,1,2 makes this hard to interpret Calculate
# MSE AFTER transforming the predictions back to the same
# scale as the observed data
nbMSEtrain <- mean((nbTrainFinal$pa - nbTrainFinal$fit)^2)  #MSE
# mean(abs((nbTrainFinal$pa - nbTrainFinal$fit))) #Mean
# absolute error

nbPred <- predict.gam(nbCombb, newdata = testComb, type = "response",
    se.fit = TRUE)
nbTestFinal <- data.frame(testComb, fit = nbPred$fit, se.fit = nbPred$se.fit)
nbMSEtest <- mean((nbTestFinal$pa - nbTestFinal$fit)^2)  #MSE
# mean(abs((testFinal$pa - testFinal$fit))) #Mean absolute
# error

#### For nbCombcLL, with spatial smoother #### pulling the
#### prediction and residual data from the model
nbTrainLL <- trainComb %>% mutate(resid = resid(nbCombLLb2),
    predict = predict(nbCombLLb2))
predTrainLL <- predict.gam(nbCombLLb2, type = "response")  #calculate MSE for these to compare with tes
nbTrainLL$fit <- predTrainLL

# using scale of 0,1,2 makes this hard to interpret
nbMSEtrainLL <- mean((nbTrainLL$pa - nbTrainLL$fit)^2)  #MSE
# mean(abs((nbTrainFinal$pa - nbTrainFinal$fit))) #Mean
# absolute error Calculate MSE AFTER transforming the
# predictions back to the same scale as the observed data
colnames(testComb)[31] <- "Chla"
colnames(testComb)[32] <- "Temp600m"
colnames(testComb)[39] <- "SSHsd"
nbPredLL <- predict.gam(nbCombLLb2, newdata = testComb, type = "response",
    se.fit = TRUE)
nbTestLL <- data.frame(testComb, fit = nbPredLL$fit, se.fit = nbPredLL$se.fit)
nbMSEtestLL <- mean((nbTestLL$pa - nbTestLL$fit)^2)  #MSE

# mean(abs((testFinal$pa - testFinal$fit))) #Mean absolute
# error

# AIC
nbAIC <- AIC(nbCombb)
nbAICLL <- AIC(nbCombLLb2)

# Explained Deviance
nbExpDev = round(((nbCombb$null.deviance - nbCombb$deviance)/nbCombb$null.deviance) *
    100, 2)
nbExpDevLL = round(((nbCombLLb2$null.deviance - nbCombLLb2$deviance)/nbCombLLb2$null.deviance) *
    100, 2)
```

```
# make summary table of metrics

table = matrix(NA, nrow = 2, ncol = 5)
colnames(table) = c("Best Models", "ExpDev", "AIC", "MSEtrain",
    "MSEtest")

# enter info by row
table[1, ] <- c("nbCombb", paste0(nbExpDev, "%"), round(nbAIC,
    2), round(nbMSEtrain, 3), round(nbMSEtest, 3))
table[2, ] <- c("nbCombLLb2 (w/ s(Lon,Lat))", paste0(nbExpDevLL,
    "%"), round(nbAICLL, 2), round(nbMSEtrainLL, 3), round(nbMSEtestLL,
    3))
require(knitr)
kable(table, caption = "Negative Binomial Model Summary Metrics")
```

Table 3: Negative Binomial Model Summary Metrics

Best Models	ExpDev	AIC	MSEtrain	MSEtest
nbCombb	9.23%	1122	0.054	0.058
nbCombLLb2 (w/ s(Lon,Lat))	17.47%	1083.67	0.052	0.057

Conclusions

The final negative binomial 'Combined' model using the combined data set with the spatial smoother yields Chl a, Temp at 600 m, SSHsd and the spatial smoother as significant in explaining the variation in sperm whale encounter. These differ slightly from the significant variables in the models without the spatial smoother, which included SST, chlorophyll, temperature at 600 m, SSH, and SSHsd. Model results showed a gradual increase in sperm whale encounters as SST increased, peaking at 26°C, followed by a gradual decrease as SST increased to 30°C.

Both sets of Combined models showed sperm whale encounters to have a negative linear relationship with chlorophyll, with encounter rate declining as chlorophyll increases. There are a few outlying values that may be driving this relationship, but it's hard to say for sure. A decline in encounters occurred for values of SSHsd between 0 to approximately 0.015 m before leveling off with a slight increase as SSHsd approached 0.025 m.

Chlorophyll and SSHsd are two variables that are proxies for primary productivity, the former relating to density of phytoplankton and the latter represents variation in sea surface height, with higher values indicating areas of potential upwelling of nutrients. The negative relationship between sperm whale encounter rate and chlorophyll is somewhat counter-intuitive, but the range of values for this variable is so small (most values concentrated between 0.05 and 0.1 mg m-3) that it's not necessarily very informative. The SSHsd ranges between ~0-2.5 cm, with the highest predictions of sperm whale encounters occurring when SSHsd is 0 cm. This indicates that the surrounding SSH (within an 8 km neighborhood) is similar to the location of the whales. Perhaps the stability in SSH represents areas that are less dynamic, which doesn't necessarily mean less productive.

Temperatures at 600 m showed a similar relationship, where encounter rate decreased as the temperature became warmer (from 6 - 6.8°C) and then increased as temperatures reached 7°C and above. This variable also has a spatial relationship within the study area, so it's interesting that it was still significant even after the spatial smoother was included. The temperatures at 600 m appear to increase in a westerly pattern across the study area. This variable relates to the temperature within the depth range of many prey species for sperm whales (primarily squids), so it could represent a gradient within the prey's habitat that is driving prey distribution and hence, sperm whale occurrence.

```
plot(PmScaled$Longitude, PmScaled$temp600m.r, xlab = "Longitude",
    ylab = "Temp at 600 m °C")
```

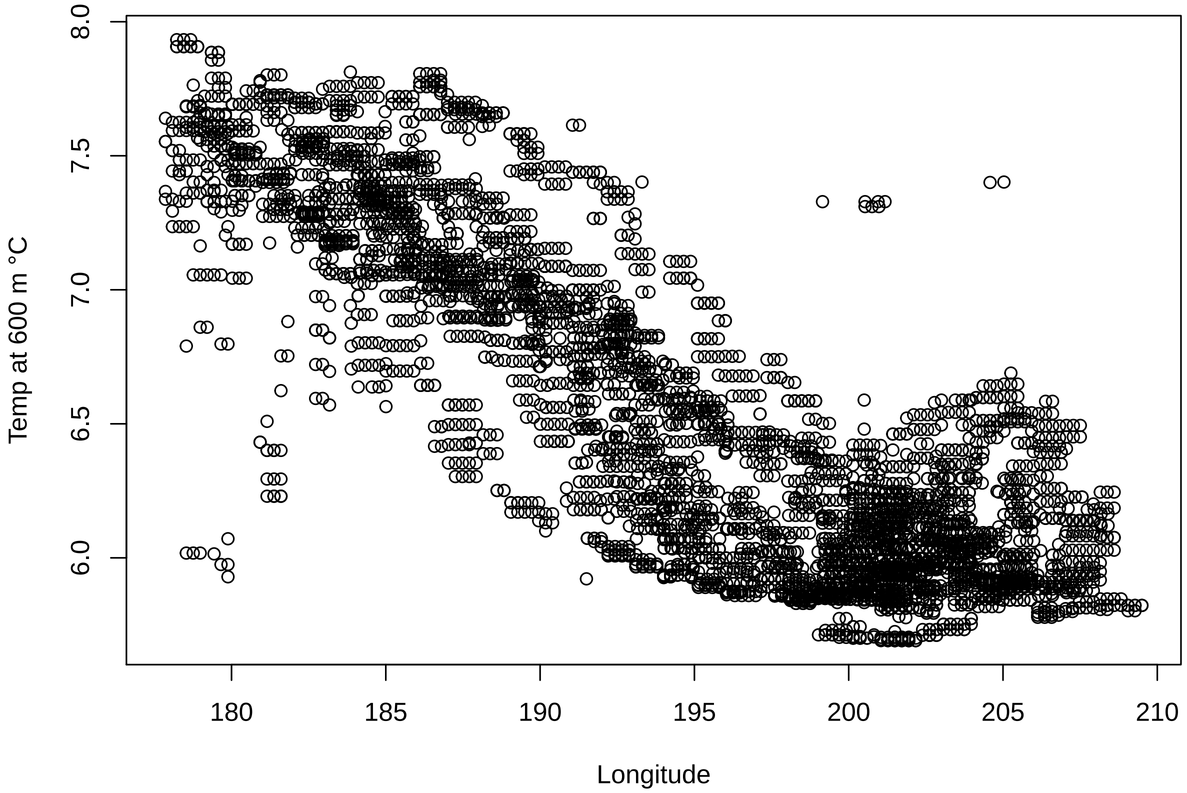


Figure 2: Temperature at 600 m as a function of longitude. Temperatures increase towards the western portion of the study area