

Supplement 1

Raster maps of the response and investigated explanatory variables

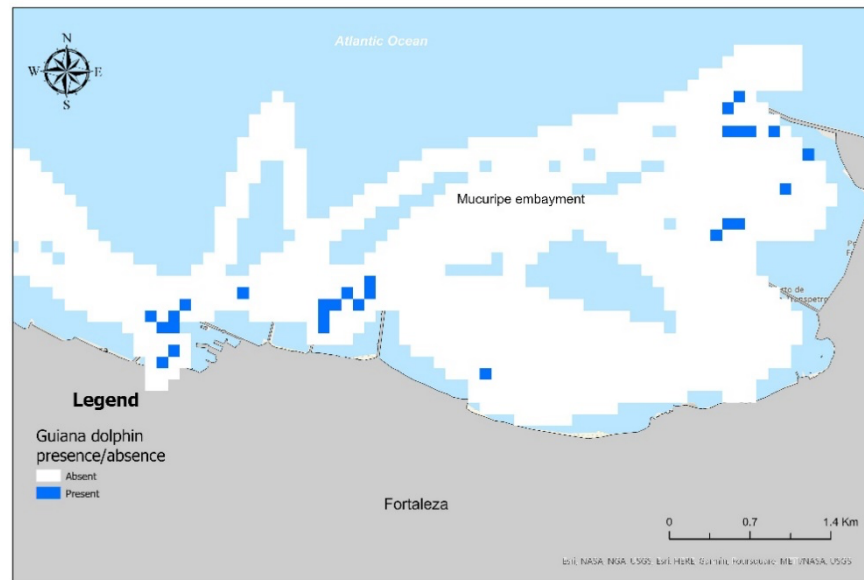


Figure S1. Study area located in Mucuripe embayment, in northeastern Brazil, showing Guiana dolphin presence/absence for each 100 x 100 m grid cell.

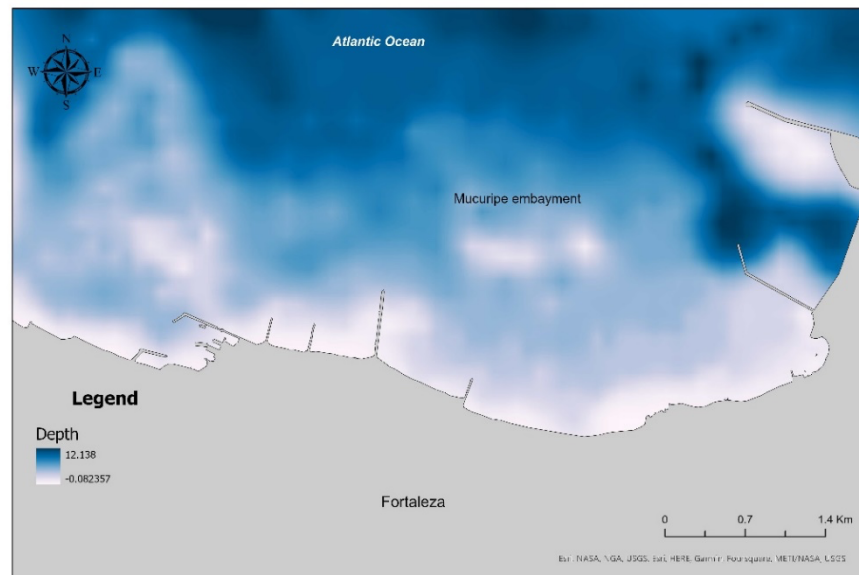


Figure S2. Study area located in Mucuripe embayment, in northeastern Brazil, showing Guiana dolphin depth in meters for each 100 x 100 m grid cell.

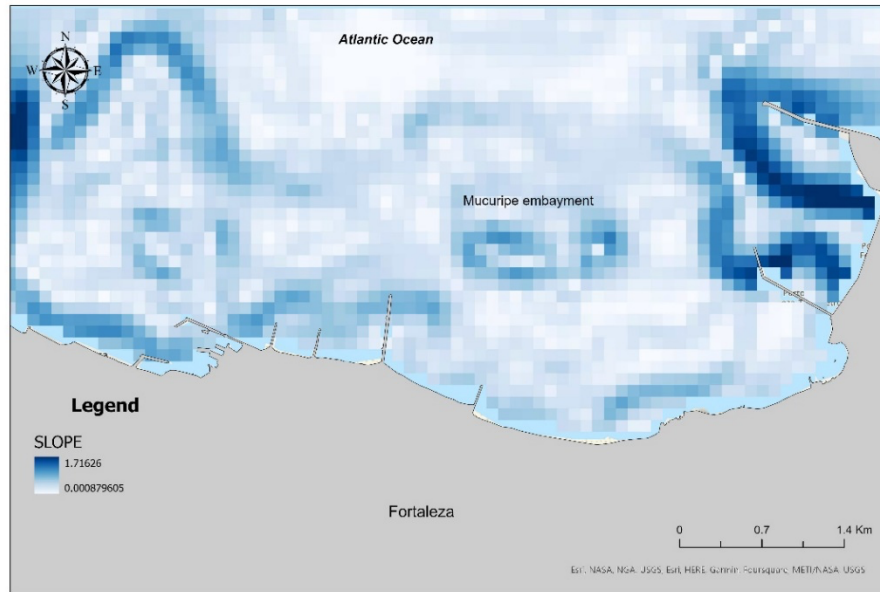


Figure S3. Study area located in Mucuripe embayment, in northeastern Brazil, showing slope, in degrees, for each 100 x 100 m grid cell.

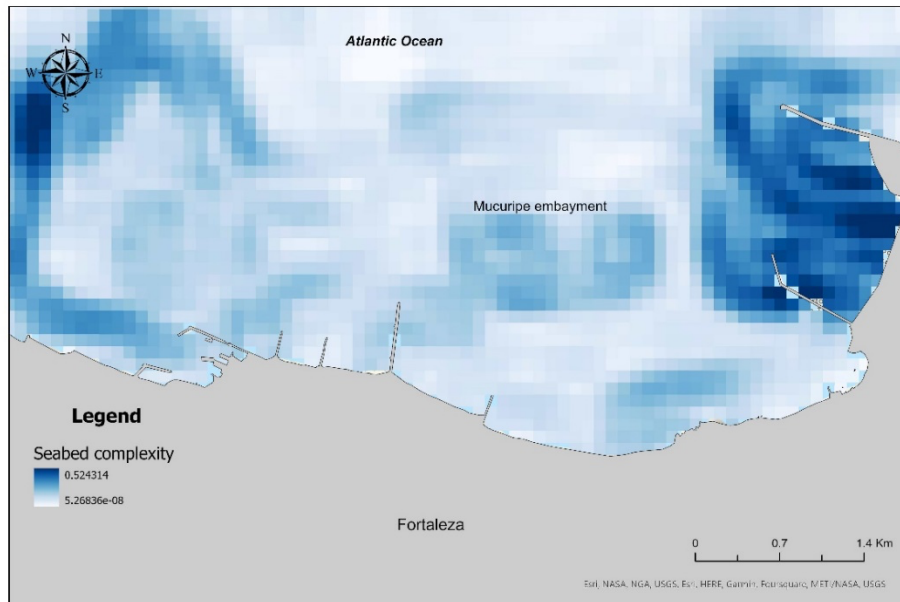


Figure S4. Study area located in Mucuripe embayment, in northeastern Brazil, showing seabed complexity, in degrees, for each 100 x 100 m grid cell.

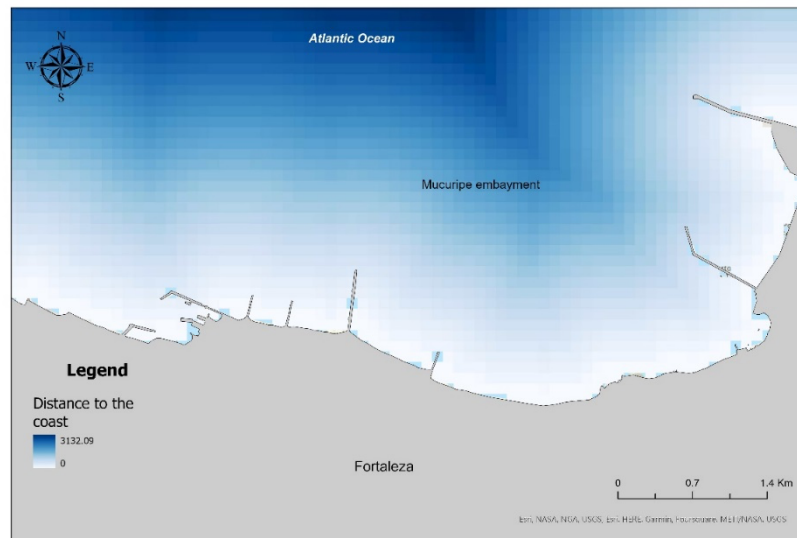


Figure S5. Study area located in Mucuripe embayment, in northeastern Brazil, showing distance to the coast, in meters, for each 100 x 100 m grid cell.

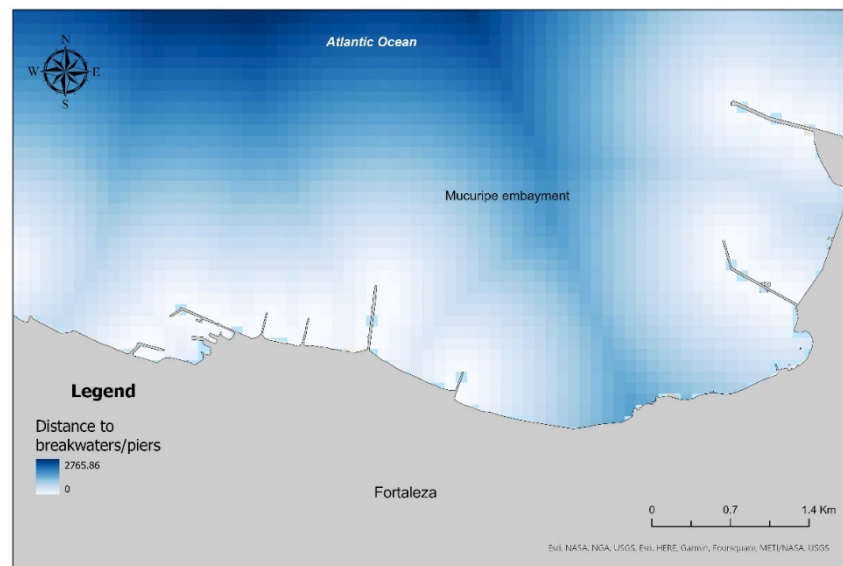


Figure S6. Study area located in Mucuripe embayment, in northeastern Brazil, showing distance to breakwaters and piers, in meters, for each 100 x 100 m grid cell.

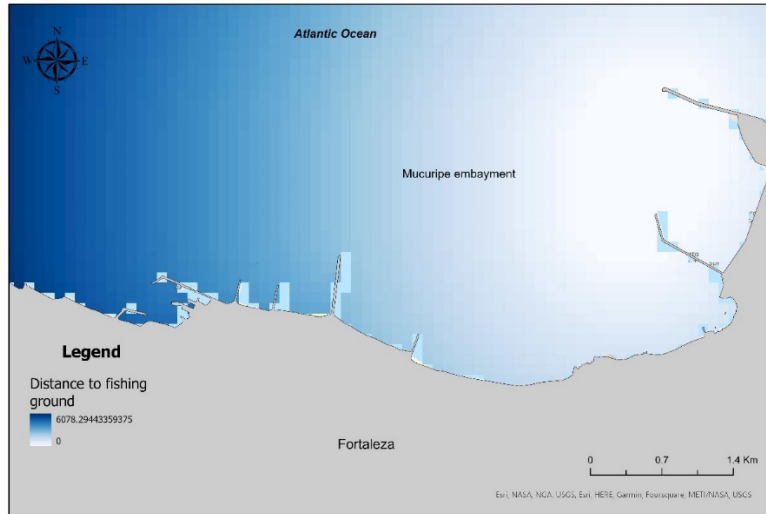


Figure S7. Study area located in Mucuripe embayment, in northeastern Brazil, showing distance to fishing ground, in meters, for each 100 x 100 m grid cell.

Supplement 2

R codes describing the modeling protocol

```
library(maptools)
library(foreign)
library(MuMIn) # for Akaike weight and averaging
library(geoR)
library(DHARMa) #evaluate chosen model
library(mgcv) #perform GAM
library(lattice)# for dotplot
library(DHARMa)# validate the chosen model (overdispersion and residuals)
library(mgcViz)
library(gstat)
library(sp)

MODEL_DATA<-read.dbf("MODEL_DATA.dbf", as.is=FALSE)

colSums(is.na(MODEL_DATA)) #looking for missing values

# INVESTIGATE COLLINEARITY

panel.cor <- function(x, y, digits=2, prefix="", cex.cor, ...)
{
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r <- abs(corr(x, y))
  txt <- format(c(r, 0.123456789), digits=digits)[1]
  txt <- paste(prefix, txt, sep="")
}
```

```
if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
text(0.5, 0.5, txt, cex = cex.cor * r)
}
pairs(~DEPTH_MASK+SLOPE+SD_SLOPE2+DIST_COAST+DIST_BREAK+DIST_FISHG, data=MODEL_DATA,
lower.panel=panel.smooth, upper.panel=panel.cor,
pch=20, main="Environmental Variables Pair Plot Matrix")

# some autocorrelation between dist_coast and dist_espig. Drop one: DIST_COAST

panel.cor <- function(x, y, digits=2, prefix="", cex.cor, ...)
{
usr <- par("usr"); on.exit(par(usr))
par(usr = c(0, 1, 0, 1))
r <- abs(cor(x, y))
txt <- format(c(r, 0.123456789), digits=digits)[1]
txt <- paste(prefix, txt, sep="")
if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
text(0.5, 0.5, txt, cex = cex.cor * r)
}
pairs(~DEPTH_MASK+SLOPE+SD_SLOPE2+DIST_BREAK+DIST_FISHG, data=MODEL_DATA,
lower.panel=panel.smooth, upper.panel=panel.cor,
pch=20, main="Environmental Variables Pair Plot Matrix")

# NO COLLINEARITY

#performing GAM (package mgcv)

# build models with different combinations of the variables, keeping the method to REML
```

```
mod1<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
gam.check(mod1)
```

```
#problem with DIST_FISHG: small p values - GAm-Check, indicate that residuals are not randomly  
distributed)assumption of independence -Has to be investigated
```

```
summary(mod1)
```

```
AIC(mod1)
```

```
summary(mod1)
```

```
mod1.1<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,k=15,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
gam.check(mod1.1)
```

```
#EDF DE SLOPE IS 1, INDICATING LINEAR RELATIONSHIP. REMOVE SMOOTHING
```

```
#problem with DIST_FISHG: small p values - GAm-Check, indicate that residuals are not randomly  
distributed)assumption of independence -Has to be investigated
```

```
summary(mod1.1)
```

```
AIC(mod1.1)
```

```
mod1.2<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,k=20,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
gam.check(mod1.2)
```

```
#problem with DIST_FISHG: small p values - GAm-Check, indicate that residuals are not randomly  
distributed)assumption of independence -Has to be investigated
```

```
summary(mod1.2)
```

```
AIC(mod1.2)
```

```
mod1.3<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ (SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,k=20,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
gam.check(mod1.3)
```

```
#problem with DIST_FISHG: small p values - GAm-Check, indicate that residuals are not randomly  
distributed)assumption of independence -Has to be investigated
```

```
summary(mod1.3)
```

```
AIC(mod1.3)
```

AIC(mod1, mod1.1, mod1.2, mod1.3) # AIC is not so different. mod1.2 looks better when analyzing gam check results. Remove s from slope and add k=20 in DIST_FISH

#Building all possible combinations

```
mod1<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,k=20,fx=F), family=binomial,data=MODEL_DATA, method="REML", na.action = "na.fail")
```

```
gam.check(mod1)
```

#problem with DIST_FISHG: small p values - GAM-Check, indicate that residuals are not randomly distributed)assumption of independence -Has to be investigated

```
summary(mod1)
```

```
AIC(mod1)
```

```
summary(mod1)
```

```
mod2<-gam(grid_code~(SLOPE)+s(SD_SLOPE2)+s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod2)
```

```
summary(mod2)
```

```
mod3<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod3)
```

```
summary(mod3)
```

```
plot(mod3, all.terms=TRUE, page=1)
```

```
mod4<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod4)
```

```
summary(mod4)
```

```
mod5<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(SD_SLOPE2)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```


AIC(mod5)

summary(mod5)

```
mod6<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod6)

summary(mod6)

```
mod7<-gam(grid_code~s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod7)

gam.check(mod7)

summary(mod7)

```
mod8<-gam(grid_code~(SLOPE)+ s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod8)

summary(mod8)

```
mod9<-gam(grid_code~(SLOPE)+ s(SD_SLOPE2)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod9)

summary(mod9)

```
mod10<-gam(grid_code~(SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod10)

summary(mod10)

```
mod11<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

AIC(mod11)

```
summary(mod11)
```

```
mod12<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SD_SLOPE2)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod12)
```

```
summary(mod12)
```

```
mod13<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod13)
```

```
summary(mod13)
```

```
mod14<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(DIST_FISHG,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod14)
```

```
summary(mod14)
```

```
mod15<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(DIST_BREAK,fx=F),  
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod15)
```

```
summary(mod15)
```

```
mod16<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE)+ s(SD_SLOPE2)+ s(DIST_BREAK,fx=F)+  
s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod16)
```

```
summary(mod16)
```

```
mod17<-gam(grid_code~s(DEPTH_MASK,fx=F)+ (SLOPE), family=binomial,data=MODEL_DATA,  
method="REML")
```

```
AIC(mod17)
```

```
summary(mod17)
```

```
mod18<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SD_SLOPE2), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod18)
```

```
summary(mod18)
```

```
mod19<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(DIST_BREAK,fx=F),
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod19)
```

```
summary(mod19)
```

```
mod20<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(DIST_FISHG,fx=F),
family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod20)
```

```
summary(mod20)
```

```
mod21<-gam(grid_code~(SLOPE)+ s(SD_SLOPE2), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod21)
```

```
summary(mod21)
```

```
mod22<-gam(grid_code~(SLOPE)+ s(DIST_BREAK,fx=F), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod22)
```

```
summary(mod22)
```

```
mod23<-gam(grid_code~(SLOPE)+ s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod23)
```

```
summary(mod23)
```

```
mod24<-gam(grid_code~s(SD_SLOPE2)+ s(DIST_BREAK,fx=F), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod24)
```

```
summary(mod24)
```

```
mod25<-gam(grid_code~s(SD_SLOPE2)+ s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod25)
```

```
summary(mod25)
```

```
mod26<-gam(grid_code~s(DIST_BREAK,fx=F)+ s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA,
method="REML")
```

```
AIC(mod26)
```

```
summary(mod26)
```

```
mod27<-gam(grid_code~s(DEPTH_MASK,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod27)
```

```
summary(mod27)
```

```
mod28<-gam(grid_code~(SLOPE), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod28)
```

```
summary(mod28)
```

```
mod29<-gam(grid_code~s(SD_SLOPE2), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod29)
```

```
summary(mod29)
```

```
mod30<-gam(grid_code~s(DIST_BREAK,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod30)
```

```
summary(mod30)
```

```
mod31<-gam(grid_code~s(DIST_FISHG,fx=F), family=binomial,data=MODEL_DATA, method="REML")
```

```
AIC(mod31)
```

```
summary(mod31)
```

```
##### CALCULATING AUC/ROC
```

```
library(gam)
```

```
library(ROCR)
```

```
#get the predicted probabilities for each sample
```

```
gampred <- predict(mod3, type="response")
```

```
rp <- prediction(as.numeric(gampred), as.numeric(MODEL_DATA$grid_code))
```

```
#now calculate AUC
```

```
auc <- performance( rp, "auc")@y.values[[1]]
```

```
auc
```

```
#plot ROC curve
```

```
roc <- performance( rp, "tpr", "fpr")
```

```
plot( roc )
```

```
#mod7
```

```
gampred2 <- predict(mod7, type="response")
```

```
rp2 <- prediction(as.numeric(gampred2), as.numeric(MODEL_DATA$grid_code)) #kyphosis is the  
absent/present information (PA_RASTER)
```

```
aucmod7 <- performance( rp2, "auc")@y.values[[1]]
```

```
aucmod7 #0.942
```

```
#mod1
```

```
gampred3 <- predict(mod1, type="response")
```

```
rp3 <- prediction(as.numeric(gampred3), as.numeric(MODEL_DATA$grid_code)) #kyphosis is the  
absent/present information (PA_RASTER)
```

```
aucmod4 <- performance( rp3, "auc")@y.values[[1]]
```

```
aucmod4#0.949
```

```
#mod2
```

```
gampred3 <- predict(mod2, type="response")
```

```
rp3 <- prediction(as.numeric(gampred3), as.numeric(MODEL_DATA$grid_code)) #kyphosis is the  
absent/present information (PA_RASTER)
```

```
aucmod2 <- performance( rp3, "auc")@y.values[[1]]
```

```
aucmod2#0.942
```

```
#mod26
```


```
gampred4 <- predict(mod26, type="response")
```

```
rp4 <- prediction(as.numeric(gampred4), as.numeric(MODEL_DATA$grid_code)) #kyphosis is the  
absent/present information (PA_RASTER)
```

```
aucmod26 <- performance( rp4, "auc")@y.values[[1]]
```

```
aucmod26#0.932
```

```
##### MODEL 3 WAS CHOSEN AS THE 'BEST MODEL'
```

```
#Like a logistic regression, the output is on the log scale  we can convert to the probability scale using  
the plogis() function
```

```
plot(mod3, pages = 1, trans = plogis, shift = coef(mod1.1)[1], seWithMean = TRUE)
```

It seems that there is some interaction between dist-break + dist_fishg would be interesting. Check this.

```
mod32<-gam(grid_code~s(DEPTH_MASK,fx=F)+ s(SD_SLOPE2, fx=F)+ s(DIST_BREAK,fx=F, k=20)+  
s(DIST_FISHG,fx=F, k=20)+s(DIST_BREAK,DIST_FISHG, fx=F, k=20), family=binomial,data=MODEL_DATA,  
method="REML")
```

```
AIC(mod32)
```

```
gam.check(mod32)
```

```
summary(mod32)# DE=40.8%
```

```
plot(mod32, pages=1)
```

```
#AUC
```

```
gampred5 <- predict(mod32, type="response")
```

```
rp5 <- prediction(as.numeric(gampred5), as.numeric(MODEL_DATA$grid_code)) #kyphosis is the  
absent/present information (PA_RASTER)
```

```
aucmod32 <- performance( rp5, "auc")@y.values[[1]]
```

```
aucmod32#0.95
```

```
summary(mod32)
```

```
plot(mod32,all.terms=TRUE, pages=1,trans = plogis, shift = coef(mod32x)[1], seWithMean = TRUE)
```

```
#####VALIDATION OF THE MODEL CHECKING IF IT MEETS THE ASSUMPTIONS  
(HOMOGENEITY, INDEPENDENCE, SPATIAL AUTO-CORRELATION RESIDUALS
```

```
library(DHARMa)
```

```
r<-simulateResiduals(mod32, n = 1000, plot = TRUE) # it seems that we do not have overdispersion.
```

```
#investigate spatial auto-correlation using Moran's I IN DHARMa
```

```
MODEL_DATA$resid <- residuals(mod32)
```

```
testData = MODEL_DATA
```

```
fittedModel <- mod32
simulationOutput <- simulateResiduals(fittedModel = fittedModel)
testSpatialAutocorrelation(simulationOutput = simulationOutput, x = testData$POINT_X, y=
testData$POINT_Y)
#the null hypothesis: is that there is no spatial autocorrelation
#P>0.05, I.E. i That is, you fail to reject the null hypothesis that there is no spatial auto-correlation.
#THERE IS NO SPATIAL AUTOCORRELATION IN THE RESIDUALS.

plot(mod32, shade=TRUE,all.terms=TRUE, pages=1,shade.col=8, trans = plogis)

##### VISUALISATION ##### MODEL32

VISUALISATION_DATA<-read.dbf("VISUALISATION_DATA_F.dbf", as.is=FALSE)
MODEL_VISUALISATION<-predict.gam(mod32x, VISUALISATION_DATA, type="response")
write.table(MODEL_VISUALISATION, file=" MODEL_VISUALISATION_F.CSV", sep="," ,dec=".")

#####validation test verify model accuracy)

MODEL_DATA_VAL<-read.dbf("VALIDATION_TEST32XF.dbf", as.is=FALSE)
str(MODEL_DATA_VAL)

GAM_VAL<-gam(grid_code~s(MODEL32X_P),family=binomial, data=MODEL_DATA_VAL)

gam.check(GAM_VAL)

plot(MODEL_DATA_VAL$grid_code,resid(GAM_VAL))

summary(GAM_VAL)
```



```
plot(GAM_VAL, shade=TRUE, shade.col=8, xlab= "Prediction", xlim=c(0, 0.20))

r<-simulateResiduals(GAM_VAL, n = 1000, plot = TRUE) # no problems with residuals distribution, no
overdispersion

MODEL_DATA_VAL$resid <- residuals(GAM_VAL)
str(MODEL_DATA_VAL)

testData_val = MODEL_DATA_VAL
fittedModel_val <- GAM_VAL
simulationOutput <- simulateResiduals(fittedModel = fittedModel)

testSpatialAutocorrelation(simulationOutput = simulationOutput, x = testData$POINT_X, y=
testData$POINT_Y)

#data: simulationOutput
#observed = -0.00312227, expected = -0.00091743, sd = 0.00169453, p-value = 0.1932
#alternative hypothesis: Distance-based autocorrelation

#summary results indicate there is a significant relationship between the likelihood
#of occurrence of Guiana dolphins predicted from the SDM created and the presence and absence of
dolphins
#in the validation dataset

###Model assessment using other method: AUC ROC
library(gam)
library(ROCR)

str(MODEL_DATA_VAL)
#get the predicted probabilities for each sample
gampred7 <- predict(GAM_VAL,MODEL_DATA_VAL, type="response")
```

```
#make a ROCR prediction object using the predicted values from
# our model and the true values from the real data
rp7 <- prediction(as.numeric(gampred7), as.numeric(MODEL_DATA_VAL$grid_code))
#now calculate AUC
auc <- performance( rp7, "auc")@y.values[[1]]
auc
```