Tutorial for SDMPlay: 2/ Generate SDM outputs

2021-02-01

Species distribution modelling (SDM) has been developed for several years to address conservation issues, to assess the direct impact of human activities on ecosystems and to predict the potential distribution shifts of invasive species (see Elith et al. 2006, Pearson 2007, Elith and Leathwick 2009). SDM relates species occurrences with environmental information and can predict species distribution on their entire occupied space. **This approach has been increasingly applied to Southern Ocean case studies, but requires corrections in such a context, due to the broad scale area, the limited number of presence records available and the spatial and temporal aggregations of these datasets.**

SDMPlay is a pedagogic package that will allow you to compute SDMs, to understand the overall method, and to produce model outputs. The package, along with its associated vignettes, highlights the different steps of model calibration and describes how to choose the best method to generate accurate and relevant outputs. SDMPlay proposes codes to apply a popular machine learning approach, BRT (Boosted Regression Trees) and introduces MaxEnt (Maximum Entropy). It contains occurrences of marine species and environmental descriptor datasets as examples associated with several vignette tutorials.

Objectives of tutorial #2/ Generate SDM outputs

First, basic approaches to generate SDM outputs are provided (prediction maps, contribution percentages of the environmental descriptors, response plots, interactions between variables). Second, classic tools to evaluate model performance are supplied (Area Under the Curve, True Skill Statistics, Biserial Pearson Correlation) and are completed with tools to perform null models (Raes and ter Steege 2007, van Proosdij et al. 2016).

See also. . .

- **Tutorial #1/ Compute Species Distribution Models**
  Focusses on data structure, data preparation and general model computing.
- **Tutorial #3/ Importance of model calibration**
  Highlights the procedure to accurately calibrate your model and proposes some methods to limit the influence of several biases.
- **Tutorial #4/ Spatial cross-validation**
  Cross-validation is a method to validate your model. When working with presence data spatially aggregated, the cross-validation procedure should be adapted. This tutorial provides some elements to apply this method (refering to Guillaumot et al. 2019).
- **Tutorial #5/ Spatial extrapolation**
  Models can extrapolate when projected on broad scale areas. This tutorial provides codes to calculate extrapolation scores and generate extrapolation maps that could be associated to SDM maps (refering to Guillaumot et al. 2020).

Let's start with Tutorial #2 !

After running your model (follow Tutorial #1), several model outputs can be presented and used for model interpretation. First, you can plot the map of your results (Fig. 1):

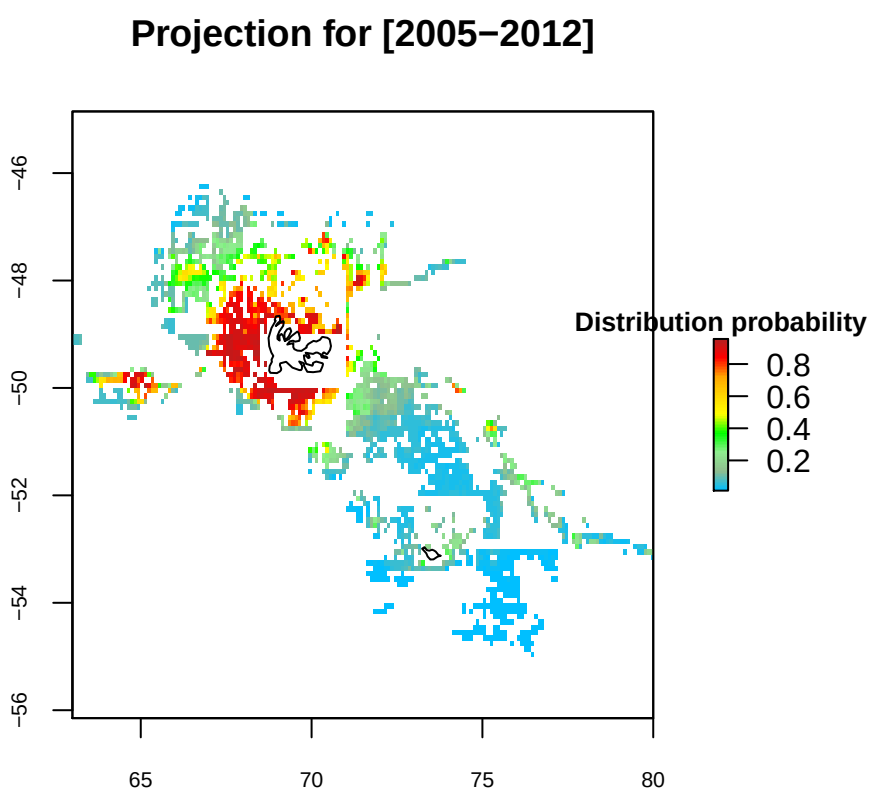

Figure 1: Model predictions for Ctenocidaris nutrix, for [2005-2012], on the Kerguelen Plateau area, with BRT

```
# Plot the map of your results (Fig. 1)
palettecolor <- colorRampPalette(c("deepskyblue", "darkseagreen","lightgreen","green",
                                   "yellow","gold","orange", "red","firebrick"))(100)

plot(Cteno_model_2005_2012$raster.prediction,col=palettecolor, main="Projection for [2005-2012]",
     cex.axis= 0.7,
     legend.width=0.5, legend.shrink=0.25,
     legend.args=list(text='Distribution probability', side=3, font=2, cex=0.8))
points(worldmap, type="l")
```

Contribution of the different environmental descriptors

The '$response' part of the produced model variable also provides several information that you can use to study your model, among which the contribution of each environmental descriptor to the model (Fig. 2).

```
contributions <- Cteno_model_2005_2012$response$contributions
b <- barplot(contributions[,2], ylab="Contribution (%)")
text(b-0.1, par("usr")[3] - 0.025, srt = 45, adj = 1, labels=contributions[,1],cex=0.5,xpd=T)
```

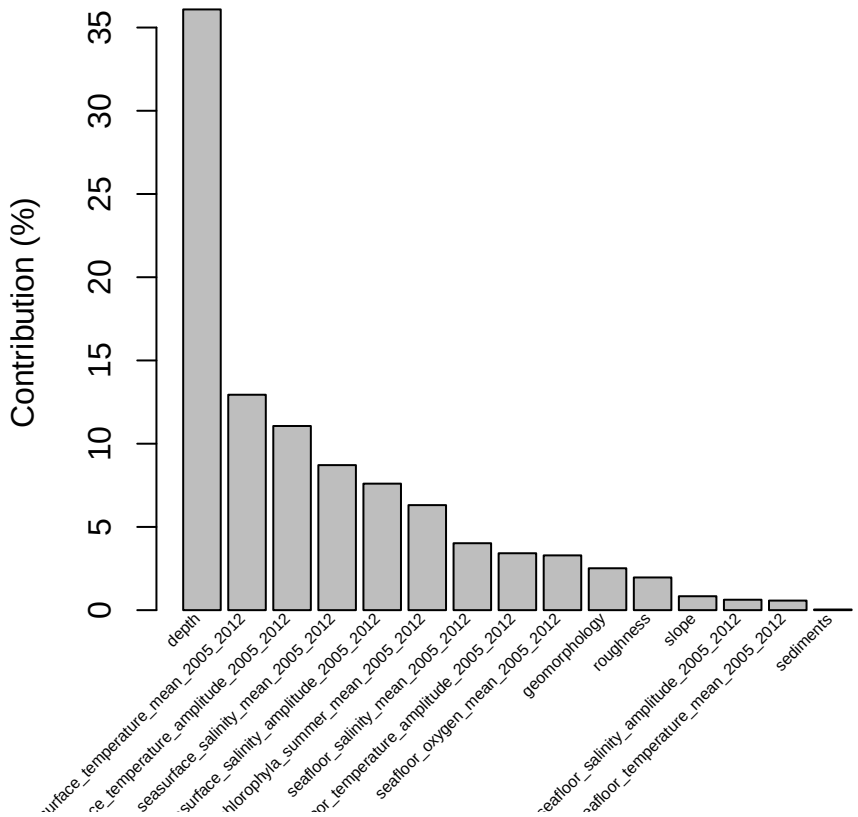

Figure 2: Percentage of contribution of each environmental descriptors to the model

Response plots

Response plots are useful indicators of environmental preferential values for the species. The y axis contains distribution probabilities predicted by the model and response plots associate these values with environmental data ('x' axis) (Fig. 3). You can use the quick and simple function below to generate them, or you can manually create them, with the extract function of the *raster* package: extract the values of your model predictions for each latitude-longitude pixels (these probability values will be your y axis) and also extract the values of the environment at the corresponding pixels (e.g. take the temperature layer, temperature values will be your x axis). Plot y~x and fit a polynom to observe the trend.

```
# Figure 3:
library(dismo)
gbm.plot(Cteno_model_2005_2012$response,n.plots=12,cex.axis=0.6,cex.lab=0.7, smooth=TRUE)
```

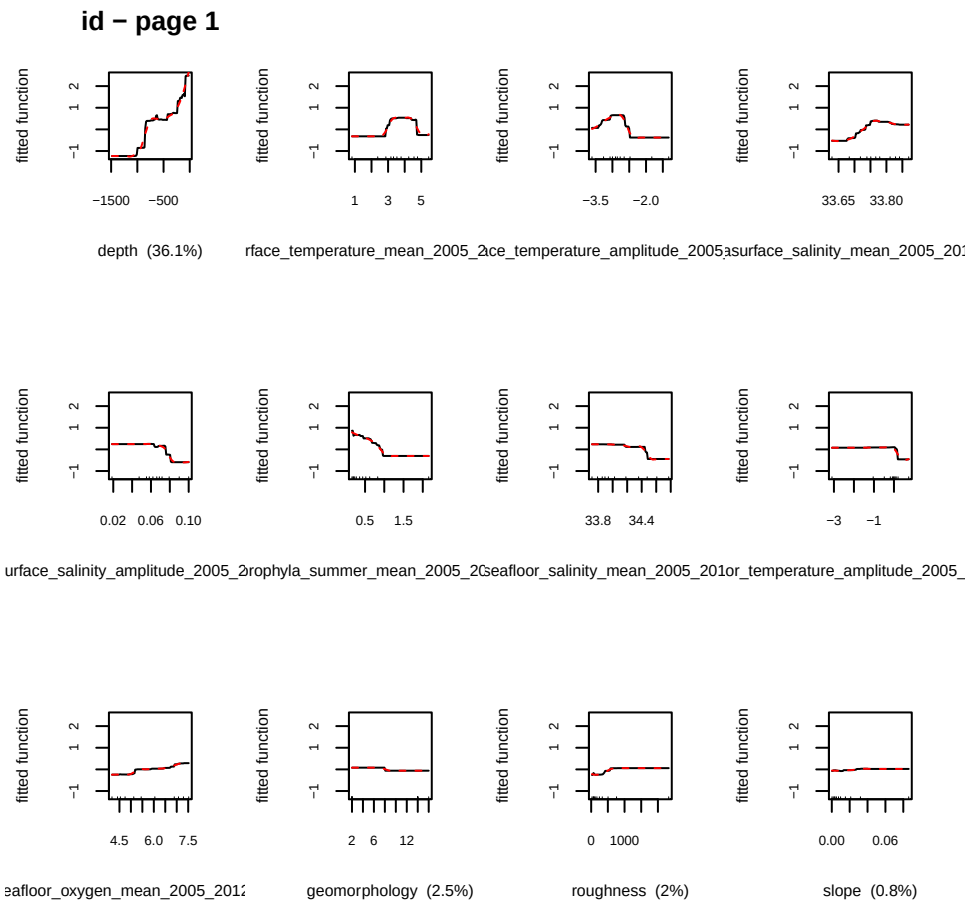

Figure 3: Response plots= Partial dependence plot, that represent how the environmental conditions (x axis) influence the predicted probabilities (y axis)

Interactions between variables

Environmental variables can be related between each other. You can represent these interactions by 3D plots with gbm (Fig. 4).

```
interactions <- gbm.interactions(Cteno_model_2005_2012$response)
head(interactions$rank.list[,c(5,2,4)])
```

```
##   int.size                                  var1.names
## 1     4.41   seafloor_temperature_amplitude_2005_2012
## 2     4.07          chlorophyla_summer_mean_2005_2012
## 3     2.95    seasurface_salinity_amplitude_2005_2012
## 4     2.66 seasurface_temperature_amplitude_2005_2012
## 5     2.10 seasurface_temperature_amplitude_2005_2012
## 6     1.53                                  roughness
##                              var2.names
## 1                                 depth
## 2                                 depth
## 3                                 depth
## 4                                 depth
## 5 seasurface_temperature_mean_2005_2012
## 6                                 depth
```

```
# Plot interactions in 3D (Fig. 4)
gbm.perspec(Cteno_model_2005_2012$response,interactions$rank.list[1,1],
            interactions$rank.list[1,3], cex.lab=0.6, cex.axis=0.6,par(mar=c(0,0,0,0)))
```

Binarize model predictions

Instead of representing model predictions by a distribution probability contained between 0 and 1, you can threshold these probabilities into suitable/unsuitable areas for species distribution. Several thresholds exist, up to you to choose the best one (Liu et al. 2013). We will give here the example of the MaxSSS (Maximum Sensitivity plus Specificity) threshold that is adapted to work with presence-only datasets (Liu et al. 2013).

```
Cteno_model_2005_2012$eval.stats$maxSSS
```

```
## [1] 0.5154886
```

```
maxSSS <- Cteno_model_2005_2012$eval.stats$maxSSS

# Plot binary map predictions (Fig. 5)
plot(Cteno_model_2005_2012$raster.prediction, col=c("lightblue","red"), breaks=c(0, maxSSS ,1),
     main="Projection for [2005-2012]",
     cex.axis= 0.7,
     legend.width=0.5, legend.shrink=0.25,
     legend.args=list(text='Distribution probability', side=3, font=2, cex=0.7))
points(worldmap, type="l")
```

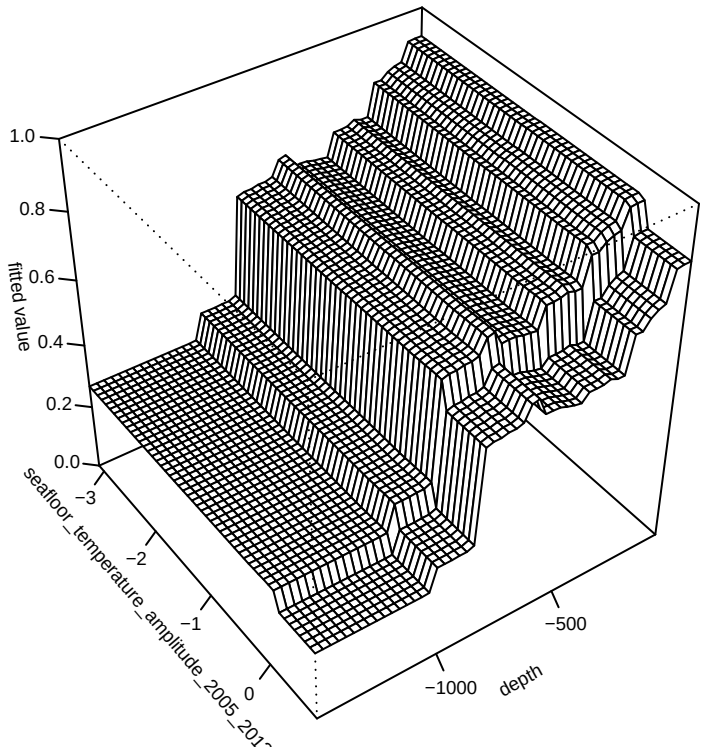

Figure 4: Detail of the interaction between two environmental descriptors: seasurface temperature amplitude and seasurface temperature mean

Model statistics: assess model performance and accuracy

Several statistics enable to evaluate a model. Please refer to literature for further details on each of these statistics (Fielding and Bell 1997, Allouche et al. 2006, Elith et al. 2006).

- **Area under the Curve (AUC)**
- **True Skill Statistics (TSS)**
- **Biserial Pearson Correlation (COR)**

You can also have a look at https://ipa-tys.github.io/ROCR/articles/ROCR.html

```
# To obtain these statistics, go into the statistic part of your model
Cteno_model_2005_2012$eval.stats$AUC #AUC
```

```
## [1] 0.959616
```

```
Cteno_model_2005_2012$eval.stats$TSS # TSS
```

```
## [1] 0.6515557
```

```
Cteno_model_2005_2012$eval.stats$COR # COR
```

```
## [1] 0.8299933
```

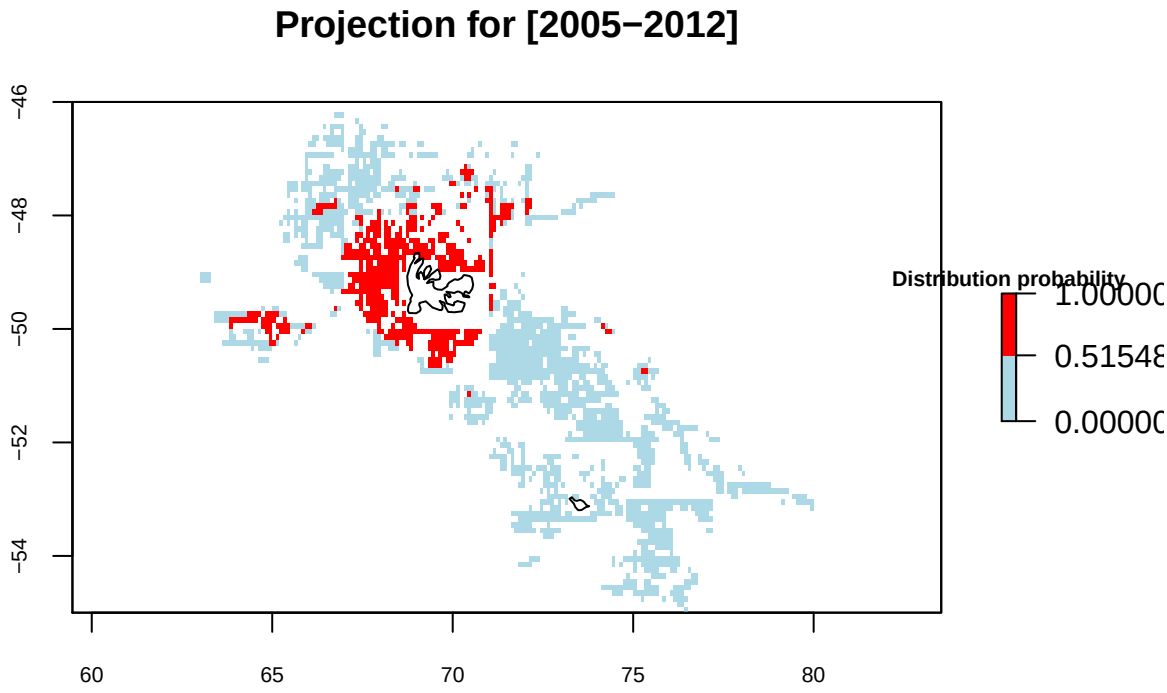

Figure 5: Predicted distribution probabilities for Ctenocidaris nutrix, for [2005-2012], using BRT, represented by suitable (red) or unsuitable (blue) areas.

- **Correctly classified test data (%)**
  Before running your model you can split your dataset into two subsets: one for training your model and the remaining data, that you independently keep, to test it. Once your model has run, you can extract the distribution probabilities predicted by the model at the location of your test data. You can then evaluate the proportion of test data for which the distribution probability is contained into the suitability threshold you have chosen (i.e. MaxSSS here). You can compile this with the following lines:

```
# Run your model with a subset of your presence-only records
# There are several methods to split your dataset into several subsets, the example of
# a simple random splitting (70% training/30% test) will be given here.
# But see Tutorial #4 for more examples.

library(raster)
library(dismo)
library(SDMPlay)
data("ctenocidaris.nutrix")
ctenocidaris.nutrix.occ <- ctenocidaris.nutrix[,c(7,8)] # longitude (first column),
                                                        # latitude (second column)
# Split your dataset into test and training subsets
idx <- sample.int(nrow(ctenocidaris.nutrix.occ),
                  size= round(nrow(ctenocidaris.nutrix.occ)*70/100), replace=F )
presence_data <- ctenocidaris.nutrix.occ[idx,]
test_data <- ctenocidaris.nutrix.occ[-idx,]

# plot training and test data on top of the bathymetry layer (Fig. 6)
data(predictors2005_2012)
data("worldmap")

library(RColorBrewer)
my.palette.blue <- rev(brewer.pal(n = 9, name = "Blues"))

plot(subset(predictors2005_2012,1), col=my.palette.blue)
points(worldmap, type="l")
points(presence_data, pch=20, col="pink")
points(test_data, pch=20, col="green")
legend("bottomright", pch=20, col=c("pink", "green"), legend=c("training","test"), bg="white")
```

```
# Build your model as precedently, except that you will use only your subset of presence data
# (i.e. your training subset)
predictors2005_2012_1500m <- SDMPlay:::delim.area(predictors2005_2012, longmin=62,
                                                  longmax=80, latmin=-55, latmax=-45,
                                                  interval=c(0,-1500))
SDMtable_ctenocidaris_1500 <- SDMPlay:::SDMtab(xydata=presence_data,
       predictors=predictors2005_2012_1500m,
       unique.data=FALSE,
       same=TRUE)
background.occ_1500 <- subset(SDMtable_ctenocidaris_1500,SDMtable_ctenocidaris_1500$id==0)[,c(2,3)]

Cteno_model_2005_2012 <- SDMPlay:::compute.brt(x=SDMtable_ctenocidaris_1500,
                                               proj.predictors=predictors2005_2012_1500m,
                                               tc = 2, lr = 0.001, bf = 0.75, n.trees = 500)
```

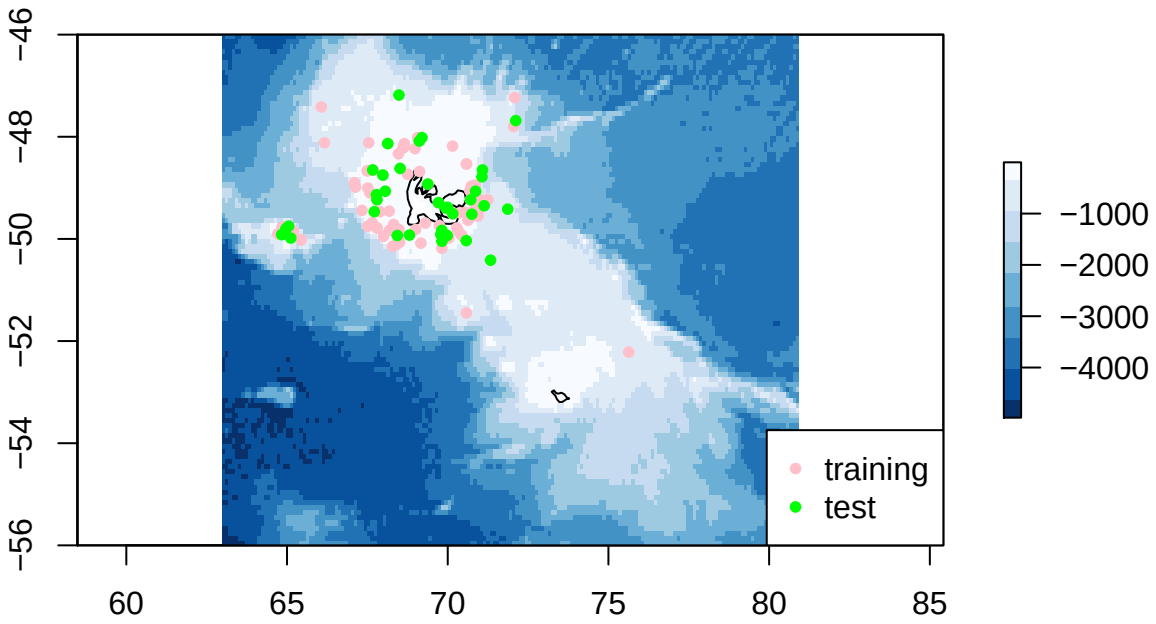

Figure 6: Plot created subsets of training and test data on top of the bathymetry layer

```
# you can then plot your predictions and the test data subset on top of that (Fig.7)

palettecolor <- colorRampPalette(c("deepskyblue", "darkseagreen","lightgreen","green",
                                   "yellow","gold","orange", "red","firebrick"))(100)
plot(Cteno_model_2005_2012$raster.prediction,col=palettecolor, main="Projection for [2005-2012]",
     cex.axis= 0.7,
     legend.width=0.5, legend.shrink=0.25,
     legend.args=list(text='Distribution probability', side=3, font=2, cex=0.8))
points(worldmap, type="l")
points(test_data, pch=20, col="black")
```

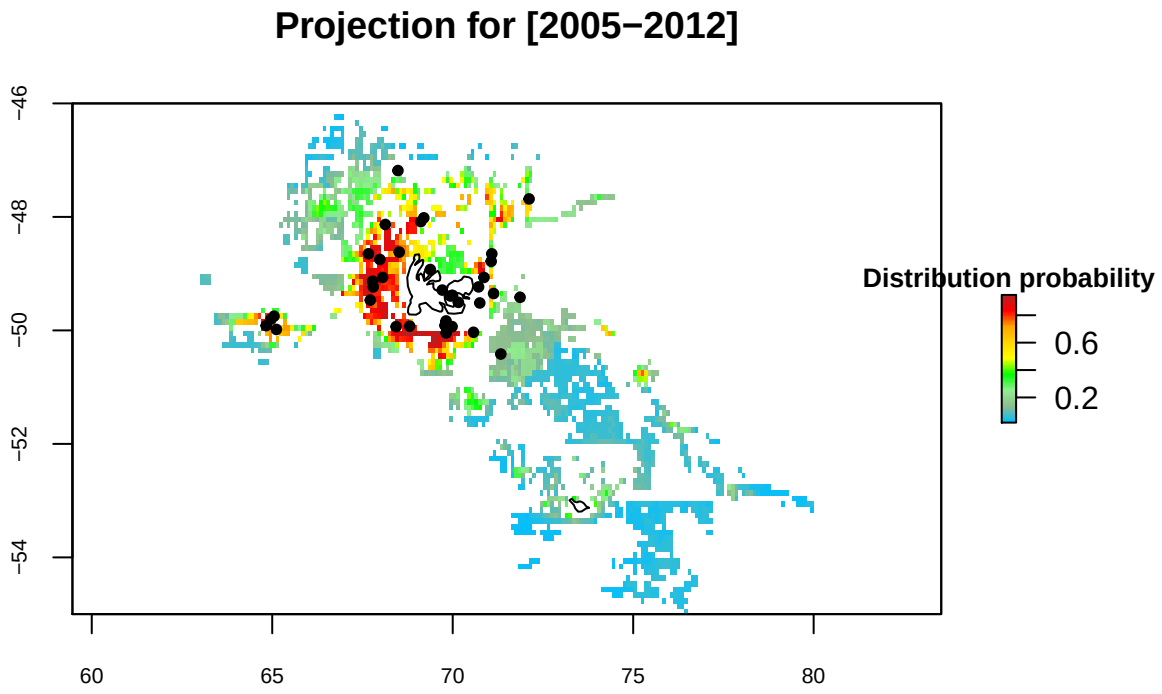

Figure 7: Similar model predictions as for Figure 1, except that only your subset of presence data is used to generate the model and test data (black dots) are presented on top of predictions

```
# Measure the proportion of test data that correctly fall into "suitable areas"
# (suitability defined by the MaxSSS threshold)

values_test_pres <- extract(Cteno_model_2005_2012$raster.prediction,test_data)
maxSSS <- Cteno_model_2005_2012$eval.stats$maxSSS
100*length(which(values_test_pres>maxSSS))/(length(values_test_pres)
                                            -length(which(is.na(values_test_pres))))
```

```
## [1] 66.66667
```

```
# -> you actually assess the proportion of test data that fall on a predicted "suitable"
#    probability, over the total number of test data that do not fall on a NA pixel
```

Null model comparisons

Another way of interpreting your model results are to compare them with results from a null model. Null models are defined in (Raes and ter Steege 2007, van Proosdij et al. 2016) and also applied in Guillaumot et al. (2018). The principle is to generate (case #1) a model that simulates probability distribution when following the presence record sampling bias or (case #2) a model that simulates presence data sampled at random over the entire study area.

A null model #2 is expected to produce distribution maps of equal suitability over the entire study area. If sampling is spatially biased, it is expected that the null model would deviate from the predictions of the standard model (Raes and ter Steege 2007) and of the null model #1. Have a look at the 'null.model' function contained in this package for more details and examples.

References

Allouche, O., Tsoar, A. & Kadmon, R. (2006). Assessing the accuracy of species distribution models: prevalence, kappa and the true skill statistic (TSS). *Journal of Applied Ecology*, 43(6), 1223-1232.

Elith, J., Anderson, R., Dudík, M., Ferrier, S., Guisan, A., Hijmans, R., Huettmann, F., ... & A Loiselle, B. (2006). Novel methods improve prediction of species' distributions from occurrence data. *Ecography*, 29(2), 129-151.

Elith, J. & Leathwick, J.R. (2009). Species distribution models: ecological explanation and prediction across space and time. *Annual Review of Ecology, Evolution, and Systematics*, 40, 677-697.

Fielding, A.H. & Bell, J.F. (1997). A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental Conservation*, 38-49.

Guillaumot, C., Martin, A., Eléaume, M. & Saucède, T. (2018). Methods for improving species distribution models in data-poor areas: example of sub-Antarctic benthic species on the Kerguelen Plateau. *Marine Ecology Progress Series*, 594, 149-164.

Guillaumot, C., Artois, J., Saucède, T., Demoustier, L., Moreau, C., Eléaume, M., ... & Danis, B. (2019). Broad-scale species distribution models applied to data-poor areas. *Progress in Oceanography*, 175, 198-207.

Guillaumot, C., Moreau, C., Danis, B. & Saucède, T. (2020). Extrapolation in species distribution modelling. Application to Southern Ocean marine species. *Progress in Oceanography*, 188, 102438.

Liu, C., White, M. & Newell, G. (2013). Selecting thresholds for the prediction of species occurrence with presence-only data. *Journal of Biogeography*, 40(4), 778-789.

Pearson, R.G. (2007). Species' distribution modeling for conservation educators and practitioners. *Synthesis. American Museum of Natural History*, 50.

Raes, N. & ter Steege, H. (2007). A null-model for significance testing of presence-only species distribution models. *Ecography*, 30(5), 727-736.

van Proosdij, A.S., Sosef, M.S., Wieringa, J.J. & Raes, N. (2016). Minimum required number of specimen records to develop accurate species distribution models. *Ecography*, 39(6), 542-552.