An example of species distribution modelling with `biomod2`

`biomod2` version : 2.0.4
R version 2.15.2 (2012-10-26)

Damien Georges & Wilfried Thuiller

February 4, 2013

Contents

1 Introduction

This vignette illustrates how to build, evaluate and project a single species distribution model using `biomod2` package. The three main modeling steps, described bellow, are the following :

1. formatting the data
2. computing the models
3. making the projections

The example is deliberately simple (few technicals explanations) to make sure it is easy to transpose to your own data relatively simply.

NOTE 1 :
Several other vignettes will be written soon to help you to go through `biomod2` details and subtleties

2 Formatting the data

In this vignette, we will work (because it is the most common case) with :

- only presences data that we will be extracted from a raster
- environmental raster layers (e.g. Worldclim)

Let's import our data.

R input

```
# load the library
library(biomod2)
# load our species raster
# we consider only the presences of Myocastor coypus species
myResp.ras <- raster( system.file(
                        "external/species/Myocastor_coypus.img",
                        package="biomod2") )
# extract the presences data

# the name
myRespName <- 'Myocastor'
# the XY coordinates of the presence
myRespXY <- xyFromCell(object=myResp.ras,
                       cell=which(myResp.ras[]>0))
# and the presence data
myResp <- extract(x=myResp.ras, y=myRespXY)
# load the environmental raster layers (could be .img, ArcGIS rasters or any supported form

# Environmental variables extracted from Worldclim (bio_3, bio_4,
# bio_7, bio_11 & bio_12)
myExpl = stack( system.file( "external/climat/current/bio3.grd",
                             package="biomod2"),
                system.file( "external/climat/current/bio4.grd",
                             package="biomod2"),
                system.file( "external/climat/current/bio7.grd",
                             package="biomod2"),
                system.file( "external/climat/current/bio11.grd",
                             package="biomod2"),
                system.file( "external/climat/current/bio12.grd",
                             package="biomod2"))
```

NOTE 2 :
You may have community or atlas data for which you have both presence and absence. In this case extract the presences and the absences points and code them by 0/1.

NOTE 3 :

If your environmental data are in matrix/data.frame format, you have to give a species as vector (or a one column Spatial.points.data.frame) having a length that match with the number of rows of your environmental data. That implies to add NA's in all points where you do not have information on species presence/absence.

When your data are correctly loaded, you have to transform them in an appropriate `biomod2` format. This is done using `BIOMOD_FormatingData`. As all models need both presences and absences to run, you may need to add some pseudo-absences (or background data) to your data. That is necessary in the case of presence-only, and may be useful in the case of insufficient absence data. 3 algorithms are now implemented to extract a range of pseudo-absence data: 'random', 'SRE' and 'disk'.
Here, we will create two sets of pseudo-absence data using the random algorithm.

NOTE 4 :
If you have both presence-absence data and a large number of presence (not the case here), it's strongly recommended to split your data.frame into two pieces and to keep a part for evaluating all your models on the same data.set (i.e. eval.xxx args)

NOTE 5 :
The PA.nb.absences arg represents the total number of pseudo-absence extracted for each set of extraction (true absences + selected PA). It must be then higher than the number of true absences (if any). If not, no pseudo-absences are selected.

R input

```
myBiomodData <- BIOMOD_FormatingData(resp.var = myResp,
                                     expl.var = myExpl,
                                     resp.xy = myRespXY,
                                     resp.name = myRespName,
                                     PA.nb.rep = 2,
                                     PA.nb.absences = 200,
                                     PA.strategy = 'random')
```

R output

```
-=-=-=-=-=-=-=-= Myocastor Data Formating -=-=-=-=-=-=-=-=

     ! No data has been set aside for modeling evaluation
   > Pseudo Absences Selection checkings...
```

```
   > random pseudo absences selection
   > Pseudo absences are selected in explanatory variables
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-= Done -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

At this point, check whether the data are correctly formatted by printing and plotting the created object.

R input

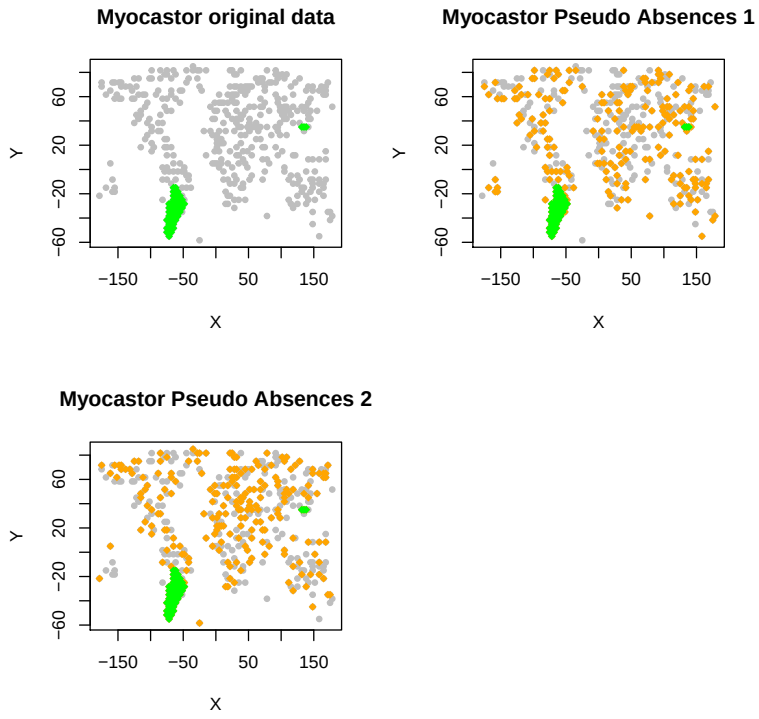
```
myBiomodData
```

R output

```
-=-=-=-=-=-=-= 'BIOMOD.formated.data.PA' -=-=-=-=-=-=-=-

sp.name =  Myocastor

         59 presences,  0 true absences and  384
undifined points in dataset


         5 explanatory variables

     bio_3          bio_4           bio_7
 Min.   :11.1   Min.   :  115   Min.   : 54.4
 1st Qu.:22.5   1st Qu.: 2372   1st Qu.:178.9
 Median :41.8   Median : 5684   Median :282.0
 Mean   :42.6   Mean   : 6696   Mean   :291.0
 3rd Qu.:57.5   3rd Qu.:10514   3rd Qu.:394.4
 Max.   :90.6   Max.   :20982   Max.   :673.8
     bio_11           bio_12
 Min.   :-447.3   Min.   :   8
 1st Qu.:-152.4   1st Qu.: 274
 Median :  51.2   Median : 608
 Mean   :  15.7   Mean   : 908
 3rd Qu.: 207.5   3rd Qu.:1261
 Max.   : 274.4   Max.   :4972


 2 Pseudo Absences dataset available ( PA1 PA2 ) with
200 absences in each (true abs + pseudo abs)

-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

R input

```
plot(myBiomodData)
```



The colors for this plot match with...

- Presences
- Absences
- Pseudo Absences
- Remaining Backgroud

3 Modeling

3.1 Building models

This step may be considered as the core of the modeling procedure within `biomod2`. Here you have to choose between 10 different algorithms ('GLM', 'GBM', 'GAM', 'CTA', 'ANN', 'SRE', 'FDA', 'MARS', 'RF', 'MAXENT'). Before running the models, you can customize their set of parameters and options using `BIOMOD_ModelingOptions`. The created object is then given to `BIOMOD_Modeling` in the next step. For the sake of simplicity, we keep all default options.

NOTE 6 :

A vignette on models' parametrization will be available soon

R input
```
# 2. Defining Models Options using default options.
myBiomodOption <- BIOMOD_ModelingOptions()
```

We are now ready for running the set of models on our species. As we do not have evaluation data, we will make 1-fold cross-validation (number controlled by NbRunEval argument) of our models by randomly splitting our data set into 2 subsets : DataSplit % for calibrating and training the models and the remainder for testing them. Each model will be tested (and evaluated if any evaluation data is given) according to models.eval.meth evaluation metrics (chosen into 'KAPPA', 'TSS', 'ROC', 'FAR', 'SR', 'ACCURACY', 'BIAS', 'POD', 'CSI' and 'ETS'). To ensure our models will be comparable in term of scale, we decided to rescale them all with a binomial GLM (rescal.all.models). The VarImport argument corresponds to the number of resampling of each explanatory variable to measure the relative importance of each variable for each selected model.

NOTE 7 :
No weights are given but some will be automatically generated. Indeed, in the particular case of pseudo-absence selection, we make sure the prevalence is kept to 0.5. It means that the presence data have the same weight than the pseudo-absence data, even if a large number of the latter has been extracted.

R input
```
# 3. Computing the models

myBiomodModelOut <- BIOMOD_Modeling(
                            myBiomodData,
                            models = c('SRE','CTA','RF','MARS','FDA'),
                            models.options = myBiomodOption,
                            NbRunEval=1,
                            DataSplit=80,
                            Yweights=NULL,
                            VarImport=3,
                            models.eval.meth = c('TSS','ROC'),
                            SaveObj = TRUE,
                            rescal.all.models = TRUE)
```

R output
```
Loading required library...

Checking Models arguments...
```

```
Creating suitable Workdir...

                        ! Weights where defined to rise a 0.5 prevalence !

-=-=-=-  Myocastor  Modeling Summary -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
5  environmental variables ( bio_3 bio_4 bio_7 bio_11 bio_12 )
Number of evaluation repetitions : 2
Models selected : SRE CTA RF MARS FDA
Total number of model runs : 20
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-


-=-=-=- Run :  Myocastor_PA1


-=-=-=-=-=-=-=- Myocastor_PA1_RUN1

Model=Surface Range Envelop
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Classification tree
         5 Fold Cross-Validation
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Breiman and Cutler's random forests for classification and regression
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Multiple Adaptive Regression Splines
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Flexible Discriminant Analysis
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...


-=-=-=-=-=-=-=- Myocastor_PA1_Full

Model=Surface Range Envelop
        Evaluating Model stuff...
        Evaluating Predictor Contributions...
```

```
Model=Classification tree
         5 Fold Cross-Validation
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Breiman and Cutler's random forests for classification and regression
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Multiple Adaptive Regression Splines
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Flexible Discriminant Analysis
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...


-=-=-=- Run :  Myocastor_PA2


-=-=-=-=-=-=- Myocastor_PA2_RUN1

Model=Surface Range Envelop
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Classification tree
         5 Fold Cross-Validation
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Breiman and Cutler's random forests for classification and regression
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Multiple Adaptive Regression Splines
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Flexible Discriminant Analysis
        Model scaling...
```

```
        Evaluating Model stuff...
        Evaluating Predictor Contributions...


-=-=-=-=-=-=-=- Myocastor_PA2_Full

Model=Surface Range Envelop
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Classification tree
         5 Fold Cross-Validation
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Breiman and Cutler's random forests for classification and regression
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Multiple Adaptive Regression Splines
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

Model=Flexible Discriminant Analysis
        Model scaling...
        Evaluating Model stuff...
        Evaluating Predictor Contributions...

-=-=-=- Done -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

When this step is over, have a look at some outputs :

- modeling summary

R input

```
myBiomodModelOut
```

R output

```
-=-=-=-=-=-=-=-=-=-=-= 'BIOMOD.models.out -=-=-=-=-=-=-=-=-=-=-=

Specie modelised : Myocastor

Considered variables : bio_3 bio_4 bio_7 bio_11 bio_12


Computed Models :  Myocastor_PA1_RUN1_SRE
```

```
Myocastor_PA1_RUN1_CTA Myocastor_PA1_RUN1_RF
Myocastor_PA1_RUN1_MARS Myocastor_PA1_RUN1_FDA
Myocastor_PA1_Full_SRE Myocastor_PA1_Full_CTA
Myocastor_PA1_Full_RF Myocastor_PA1_Full_MARS
Myocastor_PA1_Full_FDA Myocastor_PA2_RUN1_SRE
Myocastor_PA2_RUN1_CTA Myocastor_PA2_RUN1_RF
Myocastor_PA2_RUN1_MARS Myocastor_PA2_RUN1_FDA
Myocastor_PA2_Full_SRE Myocastor_PA2_Full_CTA
Myocastor_PA2_Full_RF Myocastor_PA2_Full_MARS
Myocastor_PA2_Full_FDA


Failed Models :  none

-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

- models evaluations

R input
```
# get all models evaluation
myBiomodModelEval <- getModelsEvaluations(myBiomodModelOut)
# print the dimnames of this object
dimnames(myBiomodModelEval)
```

R output
```
[[1]]
[1] "TSS" "ROC"

[[2]]
[1] "Testing.data" "Cutoff"       "Sensitivity"
[4] "Specificity"

[[3]]
[1] "SRE"  "CTA"  "RF"   "MARS" "FDA"

[[4]]
[1] "RUN1" "Full"

[[5]]
Myocastor_PA1 Myocastor_PA2
        "PA1"         "PA2"
```

R input
```
# let's print the TSS scores of Random Forest
myBiomodModelEval["TSS","Testing.data","RF",,]
```

R output

```
     PA1   PA2
RUN1 0.9 0.925
Full 1.0 1.000
```

R input

```
# let's print the ROC scores of all selected models
myBiomodModelEval["ROC","Testing.data",,,]
```

R output

```
, , PA1

      RUN1  Full
SRE  0.896 0.811
CTA  0.845 0.972
RF   0.958 1.000
MARS 0.920 0.968
FDA  0.944 0.964

, , PA2

      RUN1  Full
SRE  0.867 0.811
CTA  0.941 0.985
RF   0.985 1.000
MARS 0.961 0.984
FDA  0.996 0.981
```

R input

```
```

- Relative importance of the explanatory variables

R input

```
# print variable importances
getModelsVarImport(myBiomodModelOut)
```

R output

```
, , RUN1, PA1

         SRE   CTA    RF  MARS   FDA
bio_3  0.377 0.812 0.371 0.753 0.903
bio_4  0.428 0.429 0.058 0.450 0.600
bio_7  0.350 0.154 0.051 0.770 0.790
bio_11 0.472 0.147 0.231 0.633 0.612
bio_12 0.147 0.169 0.049 0.383 0.057
```

```
, , Full, PA1

          SRE   CTA    RF  MARS   FDA
bio_3   0.370 0.783 0.349 0.549 0.861
bio_4   0.390 0.247 0.449 0.727 0.690
bio_7   0.429 0.000 0.106 0.090 0.567
bio_11 0.535 0.402 0.424 0.654 0.569
bio_12 0.134 0.277 0.120 0.024 0.060

, , RUN1, PA2

          SRE   CTA    RF  MARS   FDA
bio_3   0.371 0.772 0.541 0.879 0.937
bio_4   0.423 0.000 0.081 0.697 0.561
bio_7   0.396 0.196 0.074 0.213 0.700
bio_11 0.494 0.484 0.304 0.356 0.538
bio_12 0.237 0.426 0.108 0.261 0.133

, , Full, PA2

          SRE   CTA    RF  MARS   FDA
bio_3   0.337 0.830 0.482 0.596 0.876
bio_4   0.383 0.053 0.146 0.451 0.584
bio_7   0.417 0.000 0.111 0.266 0.649
bio_11 0.454 0.548 0.281 0.380 0.595
bio_12 0.226 0.330 0.073 0.259 0.082
```

3.2 Ensemble modeling

Here comes one of the most interesting features of `biomod2`. `BIOMOD_EnsembleModeling` combines individual models to build some kind of meta-model. In the following example, we decide to exclude all models having a TSS score lower than 0.85.

NOTE 8 :
Models are now combined by repetition, other way to combine them (e.g. by Models, all together...) will be available soon

R input

```
myBiomodEM <- BIOMOD_EnsembleModeling(
                      modeling.output = myBiomodModelOut,
                      chosen.models = 'all',
                      eval.metric = c('TSS'),
                      eval.metric.quality.threshold = c(0.85),
                      prob.mean = T,
```

```
                        prob.cv = T,
                        prob.ci = T,
                        prob.ci.alpha = 0.05,
                        prob.median = T,
                        committee.averaging = T,
                        prob.mean.weight = T,
                        prob.mean.weight.decay = 'proportional' )
```

R output

```
-=-=-=-=-=-=-=-=-=-= Build Ensemble Models -=-=-=-=-=-=-=-=-=-

  ! all models available will be included in ensemble.modeling
  > Evaluation & Weighting methods summary :
     TSS over 0.85


 > PA1_RUN1_AllAlgos ensemble modeling
  > TSS
  > models kept :  Myocastor_PA1_RUN1_RF, Myocastor_PA1_RUN1_MARS
  > Mean of probabilities...
  > Coef of variation of probabilities...
  > Median of ptobabilities...
  > Confidence Interval...
     > 2.5 %
     > 97.5 %
  >  Comittee averaging...
  > Prababilities wegthing mean...

 > PA1_Full_AllAlgos ensemble modeling
  > TSS
  > models kept :  Myocastor_PA1_Full_CTA, Myocastor_PA1_Full_RF, Myocastor_PA1_Full_MARS
  > Mean of probabilities...
  > Coef of variation of probabilities...
  > Median of ptobabilities...
  > Confidence Interval...
     > 2.5 %
     > 97.5 %
  >  Comittee averaging...
  > Prababilities wegthing mean...

 > PA2_RUN1_AllAlgos ensemble modeling
  > TSS
  > models kept :  Myocastor_PA2_RUN1_CTA, Myocastor_PA2_RUN1_RF, Myocastor_PA2_RUN1_MARS,
  > Mean of probabilities...
  > Coef of variation of probabilities...
  > Median of ptobabilities...
  > Confidence Interval...
     > 2.5 %
     > 97.5 %
```

```
  >  Comittee averaging...
  > Prababilities wegthing mean...

 > PA2_Full_AllAlgos ensemble modeling
  > TSS
  > models kept :  Myocastor_PA2_Full_CTA, Myocastor_PA2_Full_RF, Myocastor_PA2_Full_MARS,
  > Mean of probabilities...
  > Coef of variation of probabilities...
  > Median of ptobabilities...
  > Confidence Interval...
     > 2.5 %
     > 97.5 %
  >  Comittee averaging...
  > Prababilities wegthing mean...
-=-=-=-=-=-=-=-=-=-=-=-=-=-= Done -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

You can easily access to the data and outputs of `BIOMOD_Modeling` using some specific functions to make your life easier.
Let's see the meta-models evaluation scores.

NOTE 9 :
We decide to evaluate all meta-models produced even the CV (Coefficient of Variation) one which is quite hard to interpret. You may consider it as: higher my score is, more the variation is localised where my species is forecasted as present.

R input
```
# print summary
myBiomodEM
```

R output
```
-=-=-=-=-=-=-=-= 'BIOMOD.EnsembleModeling.out' -=-=-=-=-=-=-=-=

sp.name : Myocastor

expl.var.names : bio_3 bio_4 bio_7 bio_11 bio_12


models computed:
Myocastor_PA1_RUN1_AllAlgos_EMbyTSS, Myocastor_PA1_Full_AllAlgos_EMbyTSS, Myocastor_PA2_RUN1

-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

R input
```
# get evaluation scores
getEMeval(myBiomodEM)
```

R output

```
$Myocastor_PA1_RUN1_AllAlgos_EMbyTSS
, , em.mean

    Testing.data Cutoff Sensitivity Specificity
TSS        0.941  599.7       96.61        97.5
ROC        0.994  566.0       96.61        96.5

, , em.cv

    Testing.data Cutoff Sensitivity Specificity
TSS       -0.217  0.000      100.00         0.0
ROC        0.033  0.369       10.17         9.5

, , em.ci.inf

    Testing.data Cutoff Sensitivity Specificity
TSS        0.890    115       91.53        97.5
ROC        0.951      1       91.53        92.5

, , em.ci.sup

    Testing.data Cutoff Sensitivity Specificity
TSS        0.863    744       98.31          88
ROC        0.954    999       93.22          92

, , em.median

    Testing.data Cutoff Sensitivity Specificity
TSS        0.941  599.7       96.61        97.5
ROC        0.994  566.0       96.61        96.5

, , em.ca

    Testing.data Cutoff Sensitivity Specificity
TSS        0.933  747.4       98.31          95
ROC        0.974 1000.0       98.31          95

, , em.pmw

    Testing.data Cutoff Sensitivity Specificity
TSS        0.941  599.7       96.61        97.5
ROC        0.994  561.4       96.61        96.5


$Myocastor_PA1_Full_AllAlgos_EMbyTSS
, , em.mean

    Testing.data Cutoff Sensitivity Specificity
```

```
TSS        0.983  653.0       98.31       100.0
ROC        0.999  638.7       98.31        98.5

, , em.cv

    Testing.data Cutoff Sensitivity Specificity
TSS        0.000  1.000       1.695        49.0
ROC        0.011  0.839       1.695         1.5

, , em.ci.inf

    Testing.data Cutoff Sensitivity Specificity
TSS        0.983  23.33       98.31         100
ROC        0.992  47.79       98.31         100

, , em.ci.sup

    Testing.data Cutoff Sensitivity Specificity
TSS        0.930    960         100          93
ROC        0.965    999         100          93

, , em.median

    Testing.data Cutoff Sensitivity Specificity
TSS        0.950  312.6      100.00        95.0
ROC        0.994  712.0       98.31        96.5

, , em.ca

    Testing.data Cutoff Sensitivity Specificity
TSS        0.983  828.3       98.31         100
ROC        0.999 1000.0       98.31         100

, , em.pmw

    Testing.data Cutoff Sensitivity Specificity
TSS        0.983  643.6       98.31       100.0
ROC        0.999  612.3       98.31        98.5


$Myocastor_PA2_RUN1_AllAlgos_EMbyTSS
, , em.mean

    Testing.data Cutoff Sensitivity Specificity
TSS        0.961  642.5       96.61        99.5
ROC        0.998  576.6       96.61        96.5

, , em.cv
```

```
    Testing.data Cutoff Sensitivity Specificity
TSS        0.000  1.000        0.00        10.5
ROC        0.002  0.769        3.39         3.5

, , em.ci.inf

    Testing.data  Cutoff Sensitivity Specificity
TSS        0.956 141.000       96.61        99.0
ROC        0.990   6.741       96.61        96.5

, , em.ci.sup

    Testing.data Cutoff Sensitivity Specificity
TSS        0.950  944.0      100.00        95.0
ROC        0.979  997.2       94.92        95.5

, , em.median

    Testing.data Cutoff Sensitivity Specificity
TSS        0.955  448.0      100.00        95.0
ROC        0.997  626.3       96.61        96.5

, , em.ca

    Testing.data Cutoff Sensitivity Specificity
TSS        0.931  626.2       96.61        96.5
ROC        0.996  750.0       96.61        96.5

, , em.pmw

    Testing.data Cutoff Sensitivity Specificity
TSS        0.961  634.0       96.61        99.5
ROC        0.998  572.2       96.61        96.5


$Myocastor_PA2_Full_AllAlgos_EMbyTSS
, , em.mean

    Testing.data Cutoff Sensitivity Specificity
TSS        0.980  503.0      100.00        98.0
ROC        0.999  575.5       98.31        98.5

, , em.cv

    Testing.data Cutoff Sensitivity Specificity
TSS            0  1.000       0.000         3.5
ROC            0  0.798       1.695         1.5

, , em.ci.inf
```

```
    Testing.data Cutoff Sensitivity Specificity
TSS        0.983 134.00       98.31       100.0
ROC        0.991  38.34       98.31        98.5

, , em.ci.sup

    Testing.data Cutoff Sensitivity Specificity
TSS        0.955  967.8      100.00        95.5
ROC        0.976  998.2       96.61        95.5

, , em.median

    Testing.data Cutoff Sensitivity Specificity
TSS        0.975    467      100.00        97.5
ROC        0.997    621       98.31        98.0

, , em.ca

    Testing.data Cutoff Sensitivity Specificity
TSS        0.970  626.2         100          97
ROC        0.999  750.0         100          97

, , em.pmw

    Testing.data Cutoff Sensitivity Specificity
TSS        0.980  497.7      100.00        98.0
ROC        0.999  563.4       98.31        98.5
```

4 Projection

Once the models are calibrated and evaluated, we might want to project the potential distribution of the species over space and time. This is made using `BIOMOD_Projection`

NOTE 10 :
All projections are stored directly on your hard drive

First let's project the individual models on our current conditions (the globe) to visualize them.

R input

```
# projection over the globe under current conditions
myBiomomodProj <- BIOMOD_Projection(
                          modeling.output = myBiomodModelOut,
                          new.env = myExpl,
                          proj.name = 'current',
```

```
                          selected.models = 'all',
                          binary.meth = 'ROC',
                          compress = 'xz',
                          clamping.mask = F)
```

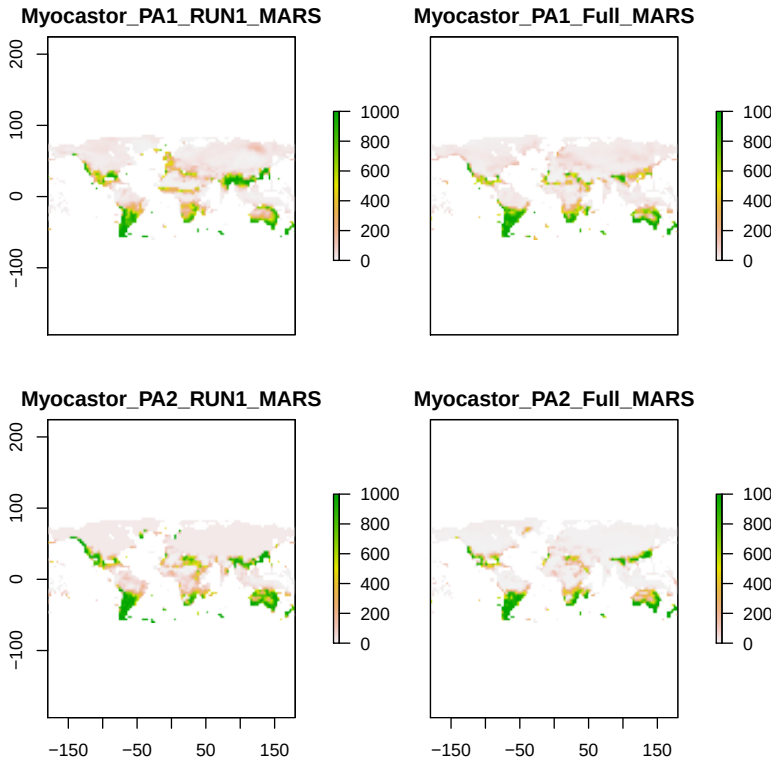
R output

```
-=-=-=-=-=-=-=-=-=-= Do Models Projections -=-=-=-=-=-=-=-=-=-=

        > Building clamping mask

        > Projecting Myocastor_PA1_RUN1_SRE ...
        > Projecting Myocastor_PA1_RUN1_CTA ...
        > Projecting Myocastor_PA1_RUN1_RF ...
        > Projecting Myocastor_PA1_RUN1_MARS ...
        > Projecting Myocastor_PA1_RUN1_FDA ...
        > Projecting Myocastor_PA1_Full_SRE ...
        > Projecting Myocastor_PA1_Full_CTA ...
        > Projecting Myocastor_PA1_Full_RF ...
        > Projecting Myocastor_PA1_Full_MARS ...
        > Projecting Myocastor_PA1_Full_FDA ...
        > Projecting Myocastor_PA2_RUN1_SRE ...
        > Projecting Myocastor_PA2_RUN1_CTA ...
        > Projecting Myocastor_PA2_RUN1_RF ...
        > Projecting Myocastor_PA2_RUN1_MARS ...
        > Projecting Myocastor_PA2_RUN1_FDA ...
        > Projecting Myocastor_PA2_Full_SRE ...
        > Projecting Myocastor_PA2_Full_CTA ...
        > Projecting Myocastor_PA2_Full_RF ...
        > Projecting Myocastor_PA2_Full_MARS ...
        > Projecting Myocastor_PA2_Full_FDA ...

        > Building ROC binaries
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-= Done -=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

R input

R input

```
# make some plots sub-selected by str.grep argument
plot(myBiomomodProj, str.grep = 'MARS')
```



R input

```
# if you want to make custom plots, you can also get the projected map
myCurrentProj <- getProjection(myBiomomodProj)
myCurrentProj
```

R output

```
class       : RasterStack
dimensions  : 45, 108, 4860, 20  (nrow, ncol, ncell, nlayers)
resolution  : 3.333, 3.333  (x, y)
extent      : -180, 180, -60, 90  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs +towgs84=0,0,0
names       : Myocastor_PA1_RUN1_SRE, Myocastor_PA1_RUN1_CTA, Myocastor_PA1_RUN1_RF, Myocast
min values  : 0, 52, 1, 0, 138, 0, 12, 0, 0, 127, 0, 13, 0, 0, 82, ...
max values  : 1000, 922, 1000, 1000, 997, 1000, 949, 1000, 1000, 996, 1000, 958, 1000, 1000,
```

Then we can project the potential distribution of the species over time, i.e. into the future.

R input

```
# load environmental variables for the future.
myExpl2050 = stack( system.file( "external/climat/future/bio3.grd",
                                 package="biomod2"),
                    system.file( "external/climat/future/bio4.grd",
```

```
                                package="biomod2"),
                    system.file( "external/climat/future/bio7.grd",
                                package="biomod2"),
                    system.file( "external/climat/future/bio11.grd",
                                package="biomod2"),
                    system.file( "external/climat/future/bio12.grd",
                                package="biomod2"))
myBiomomodProj2050 <- BIOMOD_Projection(
                                modeling.output = myBiomodModelOut,
                                new.env = stack(myExpl2050),
                                proj.name = 't2050',
                                selected.models = 'all',
                                binary.meth = 'ROC',
                                compress = 'xz',
                                clamping.mask = T)
```

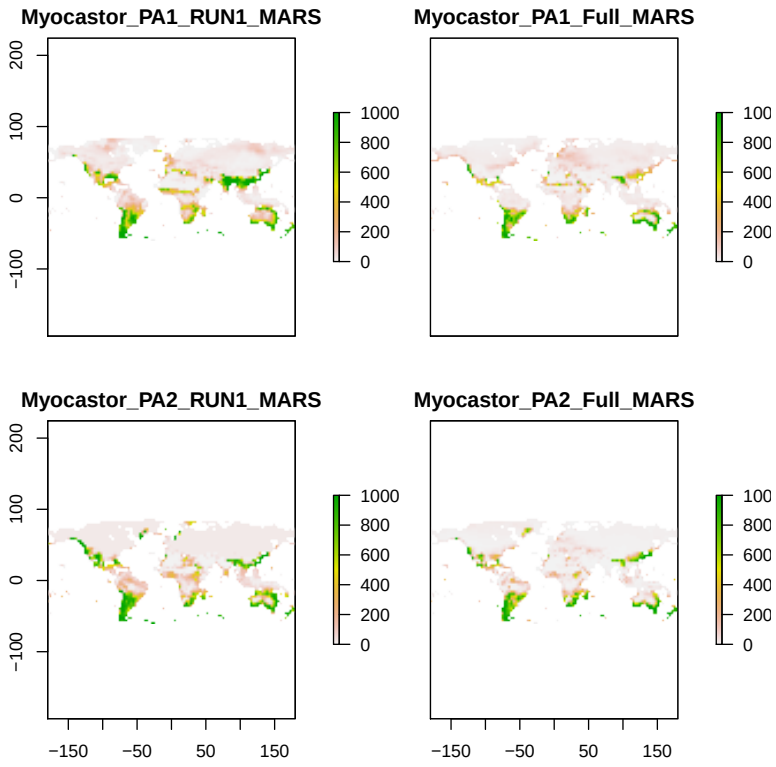
R output

```
-=-=-=-=-=-=-=-=-=-= Do Models Projections -=-=-=-=-=-=-=-=-=-

        > Building clamping mask

        > Projecting Myocastor_PA1_RUN1_SRE ...
        > Projecting Myocastor_PA1_RUN1_CTA ...
        > Projecting Myocastor_PA1_RUN1_RF ...
        > Projecting Myocastor_PA1_RUN1_MARS ...
        > Projecting Myocastor_PA1_RUN1_FDA ...
        > Projecting Myocastor_PA1_Full_SRE ...
        > Projecting Myocastor_PA1_Full_CTA ...
        > Projecting Myocastor_PA1_Full_RF ...
        > Projecting Myocastor_PA1_Full_MARS ...
        > Projecting Myocastor_PA1_Full_FDA ...
        > Projecting Myocastor_PA2_RUN1_SRE ...
        > Projecting Myocastor_PA2_RUN1_CTA ...
        > Projecting Myocastor_PA2_RUN1_RF ...
        > Projecting Myocastor_PA2_RUN1_MARS ...
        > Projecting Myocastor_PA2_RUN1_FDA ...
        > Projecting Myocastor_PA2_Full_SRE ...
        > Projecting Myocastor_PA2_Full_CTA ...
        > Projecting Myocastor_PA2_Full_RF ...
        > Projecting Myocastor_PA2_Full_MARS ...
        > Projecting Myocastor_PA2_Full_FDA ...

        > Building ROC binaries
-=-=-=-=-=-=-=-=-=-=-=-=-=-= Done -=-=-=-=-=-=-=-=-=-=-=-=-=-
```

R input

R input

```
# make some plots, sub-selected by str.grep argument
plot(myBiomomodProj2050, str.grep = 'MARS')
```



The last step of this vignette is to make Ensemble Forcasting, that means to project the meta-models you have created with `BIOMOD_EnsembleModeling`. `BIOMOD_EnsembleForecasting` required the output of `BIOMOD_EnsembleModeling` and `BIOMOD_Projection`. It will combine the projections made according to models ensemble rules defined at the ensemble modelling step.

R input

```
myBiomodEF <- BIOMOD_EnsembleForecasting(
                        projection.output = myBiomomodProj2050,
                        EM.output = myBiomodEM )
```

R output

```
-=-=-=-=-=-=-= Do Ensemble Models Projections -=-=-=-=-=-=-=


        > Projecting Myocastor_PA1_RUN1_AllAlgos_EMbyTSS ...
                > em.mean
                > em.cv
                > em.ci.inf
```

```
            > em.ci.sup
            > em.median
            > em.ca
            > em.pmw
            > Writing proj_t2050_Myocastor_PA1_RUN1_AllAlgos_EMbyTSS.grd on hard drive..

    > Projecting Myocastor_PA1_Full_AllAlgos_EMbyTSS ...
            > em.mean
            > em.cv
            > em.ci.inf
            > em.ci.sup
            > em.median
            > em.ca
            > em.pmw
            > Writing proj_t2050_Myocastor_PA1_Full_AllAlgos_EMbyTSS.grd on hard drive..

    > Projecting Myocastor_PA2_RUN1_AllAlgos_EMbyTSS ...
            > em.mean
            > em.cv
            > em.ci.inf
            > em.ci.sup
            > em.median
            > em.ca
            > em.pmw
            > Writing proj_t2050_Myocastor_PA2_RUN1_AllAlgos_EMbyTSS.grd on hard drive..

    > Projecting Myocastor_PA2_Full_AllAlgos_EMbyTSS ...
            > em.mean
            > em.cv
            > em.ci.inf
            > em.ci.sup
            > em.median
            > em.ca
            > em.pmw
            > Writing proj_t2050_Myocastor_PA2_Full_AllAlgos_EMbyTSS.grd on hard drive..

Nothing is returned but you can access created projections by loading them with 'load(...)'

Available files are :
'Myocastor/proj_t2050/proj_t2050_Myocastor_PA1_RUN1_AllAlgos_EMbyTSS.grd'
'Myocastor/proj_t2050/proj_t2050_Myocastor_PA1_Full_AllAlgos_EMbyTSS.grd'
'Myocastor/proj_t2050/proj_t2050_Myocastor_PA2_RUN1_AllAlgos_EMbyTSS.grd'
'Myocastor/proj_t2050/proj_t2050_Myocastor_PA2_Full_AllAlgos_EMbyTSS.grd'

-=-=-=-=-=-=-=-=-=-=-=-=-=-= Done -=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

Nothing is returned but some additional files have been created in your projection folder (RasterStack or array depending on your projection type). This file contains your meta-models projections.

R input

```
proj_t2050_Myocastor_PA1_Full_AllAlgos_EMbyTSS <- stack("Myocastor/proj_t2050/proj_t2050_My
proj_t2050_Myocastor_PA1_Full_AllAlgos_EMbyTSS
```

R output

```
class       : RasterStack
dimensions  : 45, 108, 4860, 7  (nrow, ncol, ncell, nlayers)
resolution  : 3.333, 3.333  (x, y)
extent      : -180, 180, -60, 90  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs +towgs84=0,0,0
names       : Myocastor_PA1_Full_AllAlgos_EMbyTSS_ef.mean, Myocastor_PA1_Full_AllAlgos_EMby
min values  : 4.0, 1.9, 0.0, 15.0, 0.0, 0.0, 4.0
max values  : 983, 173, 936, 1000, 1000, 1000, 983
```

R input

```
plot(proj_t2050_Myocastor_PA1_Full_AllAlgos_EMbyTSS)
```