

# Ecological niche models with Maxent

*Eugenio Valderrama - ev243@cornell.edu*

*11/13/2019*

## Contents

Install packages	1
Obtain and prepare environmental layers	2
Find and clean records for your species	2
Use Maxent	6
Predict on current and future climate	8
Test your model	12
Check the correlations among variables	13
Exercise	16
Notes	16

## Install packages

This document will guide you through the steps to obtain an ecological niche model using **MaxEnt** with **WorldClim** climatic variables and **GBIF** records. The code is based on several sources including: <https://github.com/danlwarren/ENMTools> <http://spatialecology.weebly.com/r-code--data/category/sdm-maxent> [https://cmerow.github.io/RDataScience/3\\_6\\_Teaching\\_Ecoinformatics.html](https://cmerow.github.io/RDataScience/3_6_Teaching_Ecoinformatics.html)

To begin with you need to install and load some packages. The lines to install and load packages have been annotated in case you already have them, if that is not the case you can delete the ‘#’ symbol at the beginning of the relevant line. If you are using MacOS and get errors when loading the library **rJava** go to the terminal and type ‘sudo R CMD javareconf’ and restart R or RStudio.

```
#install.packages(raster)
#library(raster)
#install.packages('rgdal')
#library(rgdal)
#install.packages('maps')
#library(maps)
#install.packages('mapdata')
#library(mapdata)
#install.packages('dismo')
#library(dismo)
#install.packages('rJava')
#library(rJava)
#install.packages('maptools')
#library(maptools)
#install.packages('jsonlite')
#library(jsonlite)
```

## Obtain and prepare environmental layers

Now download the WorldClim Bioclimatic variables (check for more info <https://www.worldclim.org/bioclim>), in this case with a resolution of 2.5 minutes (c.4.5 km at the equator).

```
currentEnv=getData("worldclim", var="bio", res=2.5)
```

We are going to predict using an scenario (one of the many available; HadGEM2-ES,rcp85) of the Bioclimatic variables for 2070. Download the future climate layers with the same resolution and match the names of current and future climatic variables.

```
futureEnv=getData('CMIP5', var='bio', res=2.5, rcp=85, model='HE', year=70)
names(futureEnv)=names(currentEnv)
```

Also to avoid having too much data we are going to focus on just a few Bioclimatic variables. You can check what they actually are here <https://www.worldclim.org/bioclim> and use the ones you prefer. Notice that the listed variables in the code are the ones we are **not** using.

```
currentEnv=dropLayer(currentEnv, c("bio2", "bio3", "bio4", "bio10", "bio11", "bio13",
                                   "bio14", "bio15"))
futureEnv=dropLayer(futureEnv, c("bio2", "bio3", "bio4", "bio10", "bio11", "bio13",
                                   "bio14", "bio15"))
```

## Find and clean records for your species

Now get the records from GBIF for the Pawpaw tree (*Asimina triloba*, Annonaceae) and store them in an object and get rid of records without geographic coordinates and duplicates. This tree species grows on natural areas on campus, but unless you are really good with bark characters you'll probably have to wait a few months to spot it.

```
atraw=gbif('asimina', 'triloba')
```

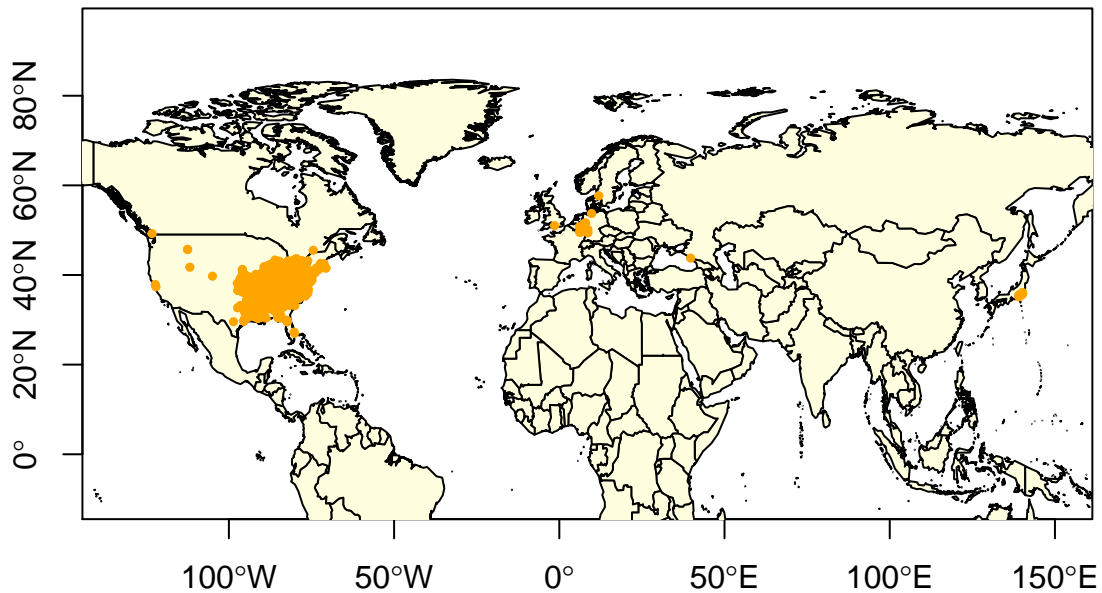
```
## 6129 records found
```

```
## 0-300-600-900-1200-1500-1800-2100-2400-2700-3000-3300-3600-3900-4200-4500-4800-5100-5400-5700-6000-6
```

```
at=subset(atraw, !is.na(lon) & !is.na(lat))
atdups=duplicated(at[, c("lon", "lat")])
at <-at[!atdups, ]
```

Plot your occurrences for diagnostic purposes. Setting the extent of the map to 10 degrees beyond your maximum and minimum values of latitude and longitude.

```
data(wrld_simpl)
lonb<-round(c(min(at$lon)-10,max(at$lon)+10))
latb<-round(c(min(at$lat)-10,max(at$lat)+10))
plot(wrld_simpl, xlim=lonb, ylim=latb, axes=TRUE, col="light yellow")
points(at$lon, at$lat, col="orange", pch=20, cex=0.75)
```

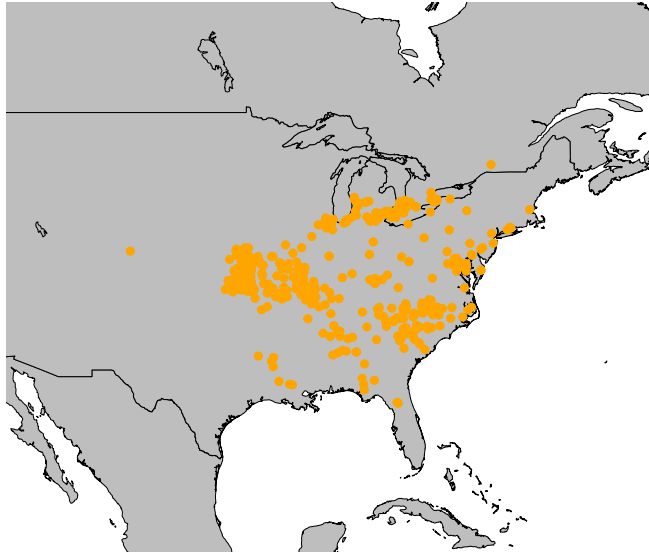


Eliminate occurrences outside the native range, Pawpaw trees are native to the east of the USA, so anything with longitude between 65<sup>a</sup> and 105<sup>a</sup> W and latitude above 25<sup>a</sup>N is plausible in this case (GBIF has occurrences of cultivated plants in other continents, poorly georeferenced occurrences in the middle of the ocean).

```
at <- at[at$lon < -65 & at$lon > -105 & at$lat > 25 &
      at$basisOfRecord == "PRESERVED_SPECIMEN", ]
lonb<-round(c(min(at$lon)-10,max(at$lon)+10))
latb<-round(c(min(at$lat)-10,max(at$lat)+10))
```

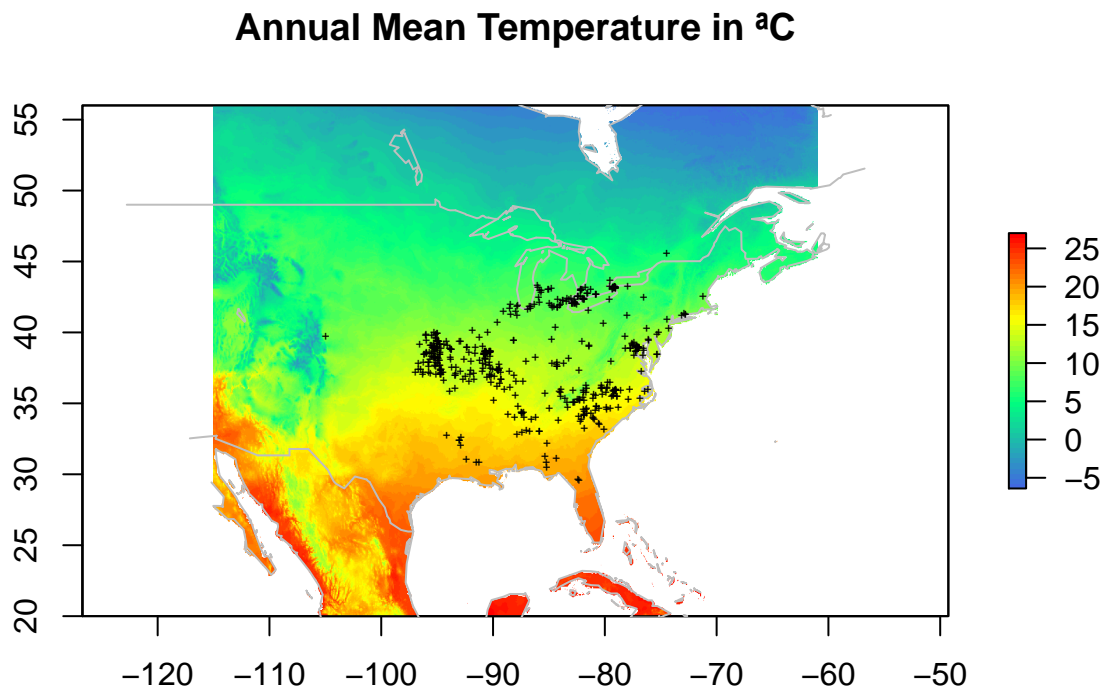
Plot again to check the filtered occurrences.

```
map('worldHires', xlim=lonb, ylim=latb, fill=TRUE, col='grey',border='black',lwd='0.2')
points(at$lon, at$lat, col="orange", pch=20, cex=0.75)
```



Now that you have the presence data you can see some of the climatic variables that you are going to include in the model. Like **BIO1** that is **Annual Mean Temperature in °C** for current climate.

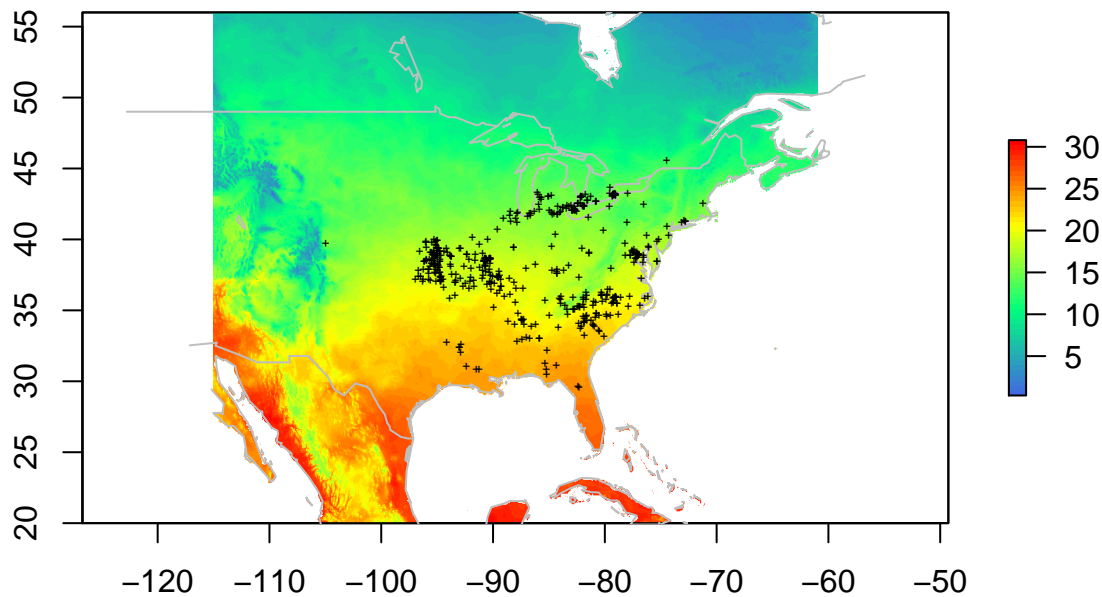
```
colfunc<-colorRampPalette(c("royalblue","springgreen","yellow","red"))
plot(currentEnv[["bio1"]]/10,xlim=lonb, ylim=latb, main="Annual Mean Temperature in °C",
      col=colfunc(50))
map('worldHires',xlim=lonb,ylim=latb, fill=FALSE, add=TRUE,col='grey')
points(at$lon, at$lat, pch="+", cex=0.4)
```



Or the same variable in the scenario of climate change for 2070.

```
plot(futureEnv[["bio1"]]/10,main="Future Annual Mean temperature in °C",
     col=colfunc(50),xlim=lonb,ylim=latb)
map('worldHires',xlim=lonb, ylim=latb, fill=FALSE, add=TRUE,col='grey')
points(at$lon, at$lat, pch="+", cex=0.4)
```

## Future Annual Mean temperature in °C



## Use Maxent

Crop the environmental layers of current and future climate to the area where Pawpaws are native.

```
model.extent<-extent(c(lonb,latb))
modelEnv=crop(currentEnv,model.extent)
modelFutureEnv=crop(futureEnv, model.extent)
```

Now randomly split your dataset in some occurrences for training the model (the ones you use to create the ENM) and some for testing how well the model performs (some occurrences that you didn't include in the algorithm, but where a good model should be predicting suitable climate). In this case we are keeping 20% for testing because we have many occurrences.

```
atocc=cbind.data.frame(at$lon,at$lat)
fold <- kfold(atocc, k=5)
attest <- atocc[fold == 1, ]
attrain <- atocc[fold != 1, ]
```

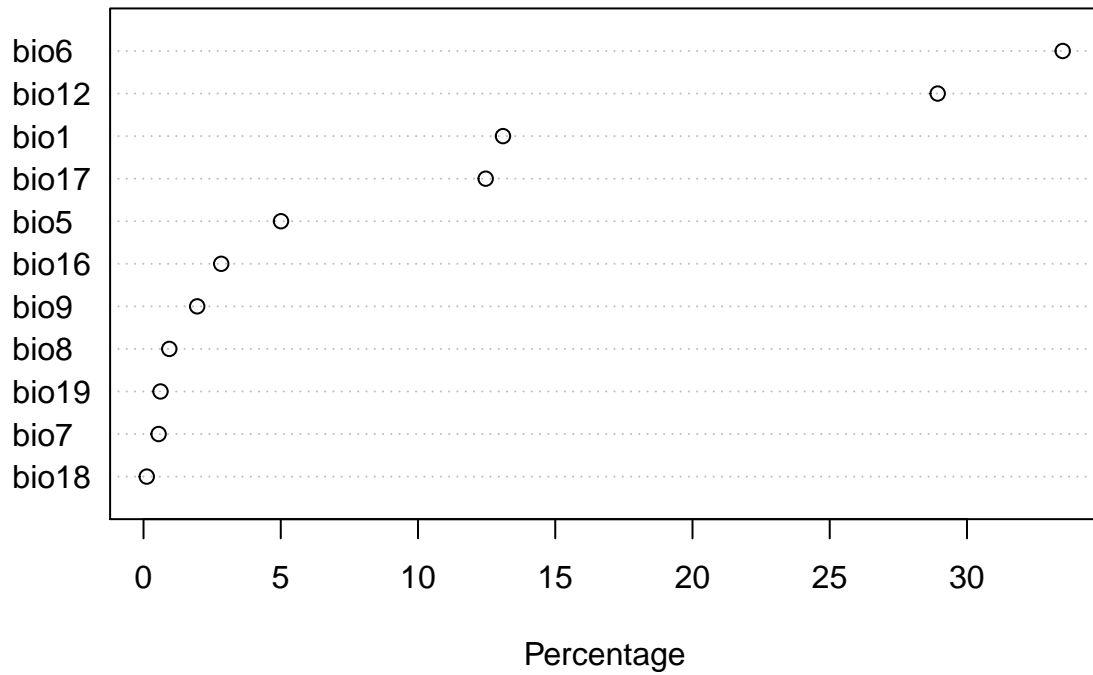
Now run Maxent with the training occurrences and the current environmental variables. You will get an error if you don't have maxent installed, download it from the website specified in the error and save it in the directory that is also specified in the error.

```
at.me <- maxent(modelEnv, attrain)
```

Check the variables that helped the algorithm the most.

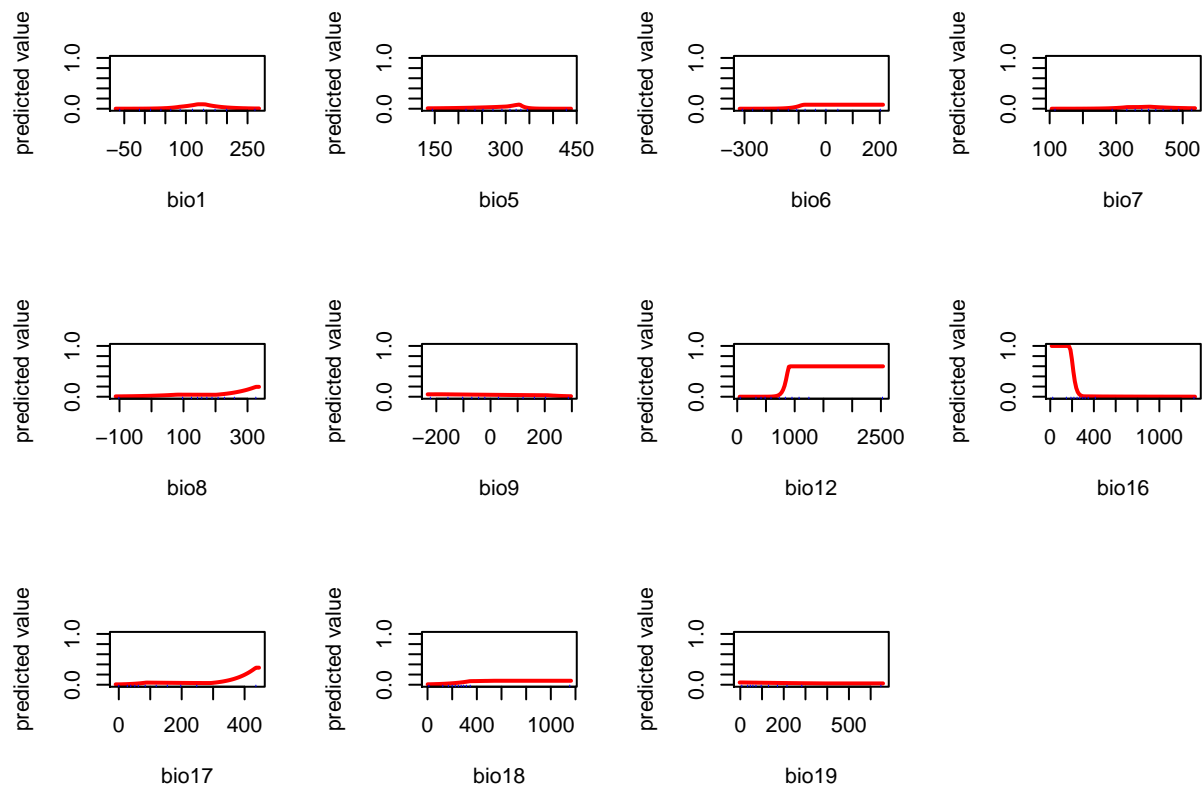
```
plot(at.me)
```

## Variable contribution



And the response curves of each variable for you dataset.

```
response(at.me)
```



## Predict on current and future climate

Now that you created a model of the climatic niche of the Pawpaw trees you can predict if each cell in the whole study area has suitable climate for Pawpaws according to your model with current climate...

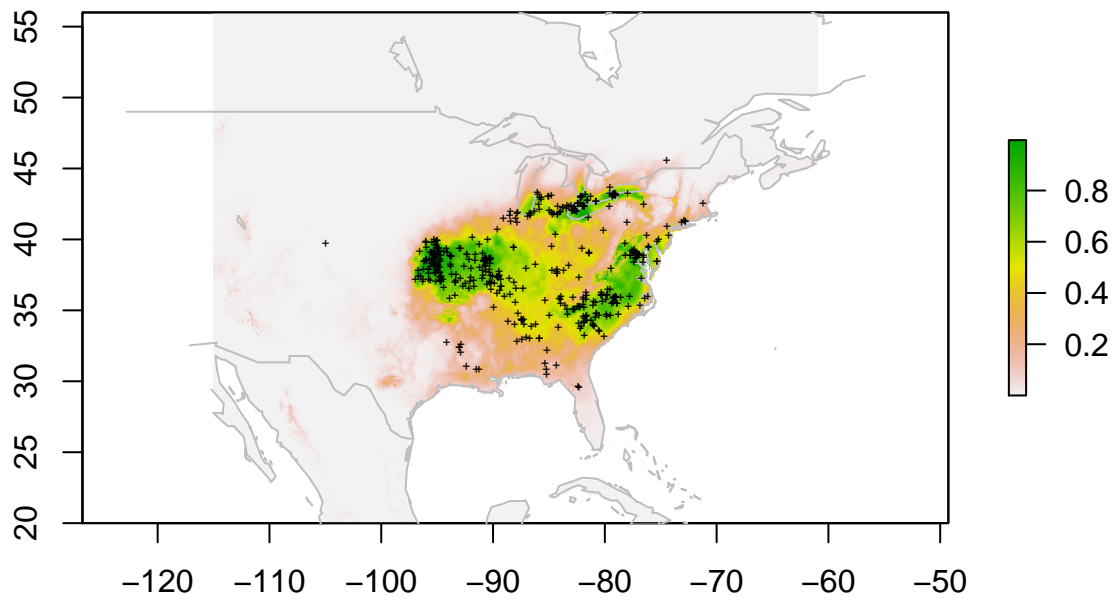
```
at.pred <- predict(at.me, modelEnv)
```

and see it on a map

```
plot(at.pred, main="Predicted Suitability")
map('worldHires', fill=FALSE, add=TRUE,col='grey',xlim=lonb,ylim=latb)
points(at$lon, at$lat, pch="+", cex=0.4)
```



## Predicted Suitability



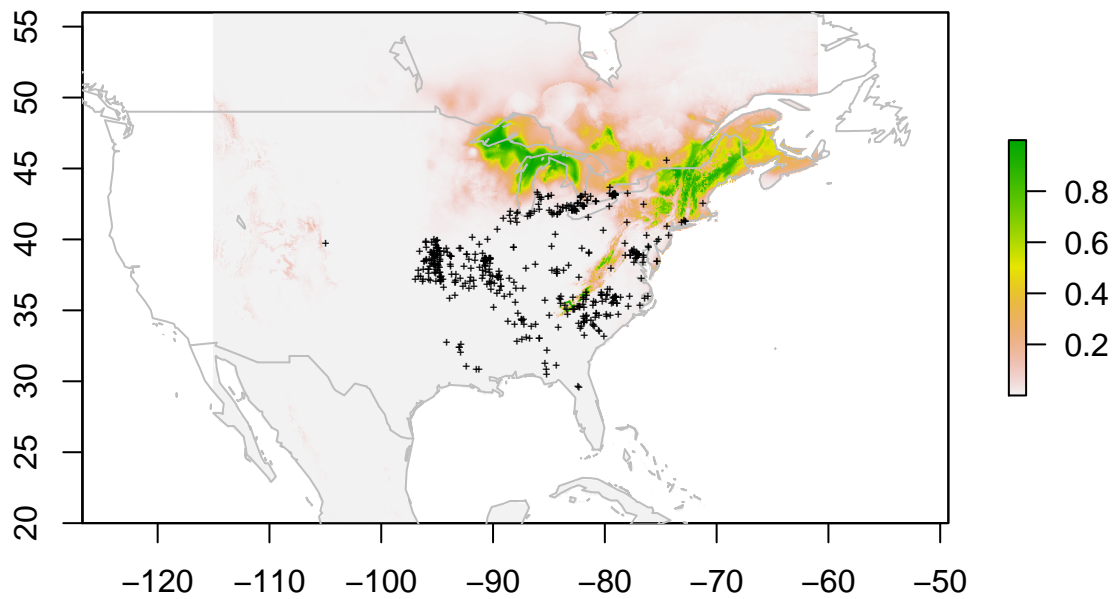
Furthermore you can project the model to the layers with the future climate scenario...

```
at.2070 = predict(at.me, modelFutureEnv)
```

and plot it too.

```
plot(at.2070, main="Predicted Future Suitability")
map('worldHires', fill=FALSE, add=TRUE,col='grey')
points(at$lon, at$lat, pch="+", cex=0.4)
```

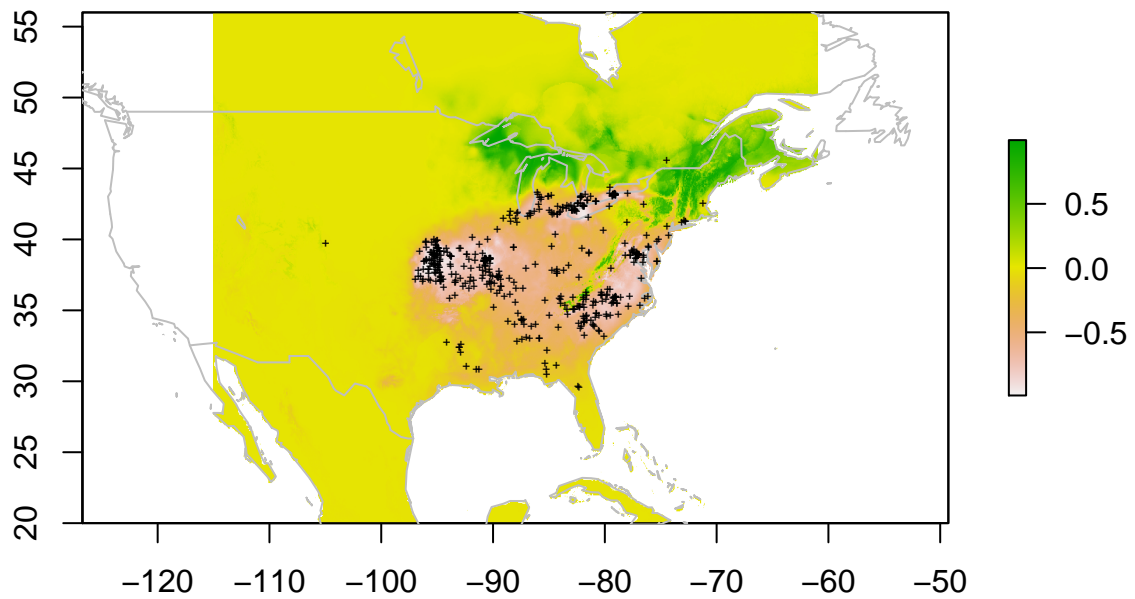
## Predicted Future Suitability



A good way to compare is to visualize the difference for the values on each cell for current vs future climate.

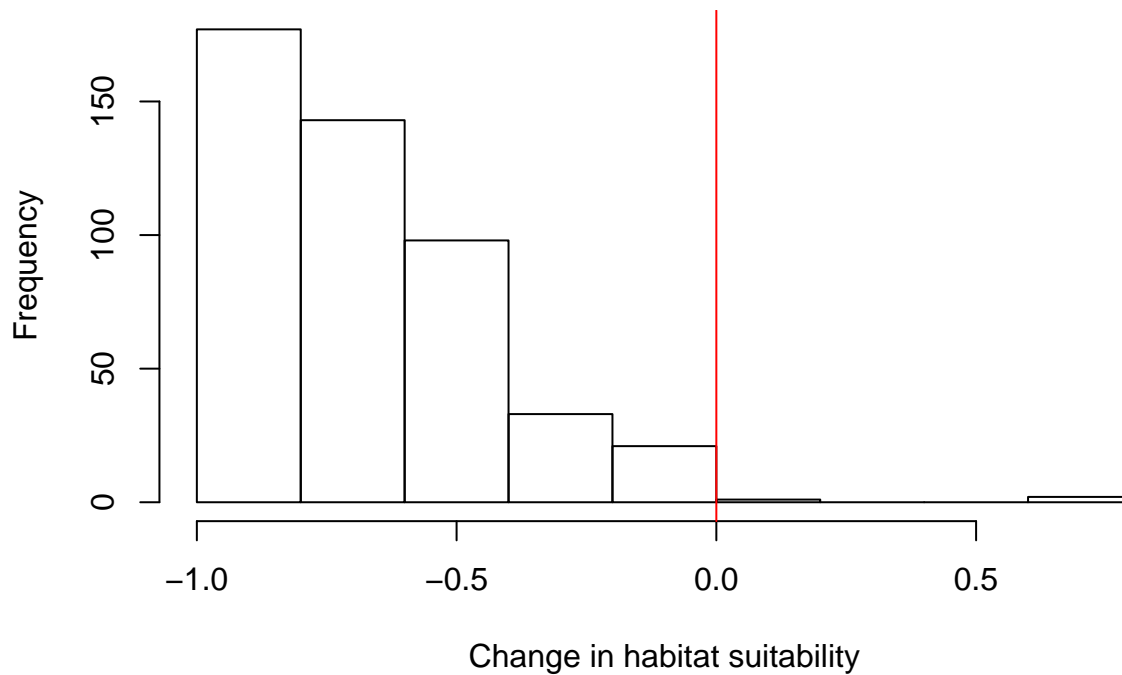
```
at.change=at.2070-at.pred
plot(at.change, main="Predicted Change in Suitability")
map('worldHires', fill=FALSE, add=TRUE,col='grey')
points(at$lon, at$lat, pch="+", cex=0.4)
```

## Predicted Change in Suitability



Or building an histogram of the values for the difference of suitability on each cell that we used as occurrence data.

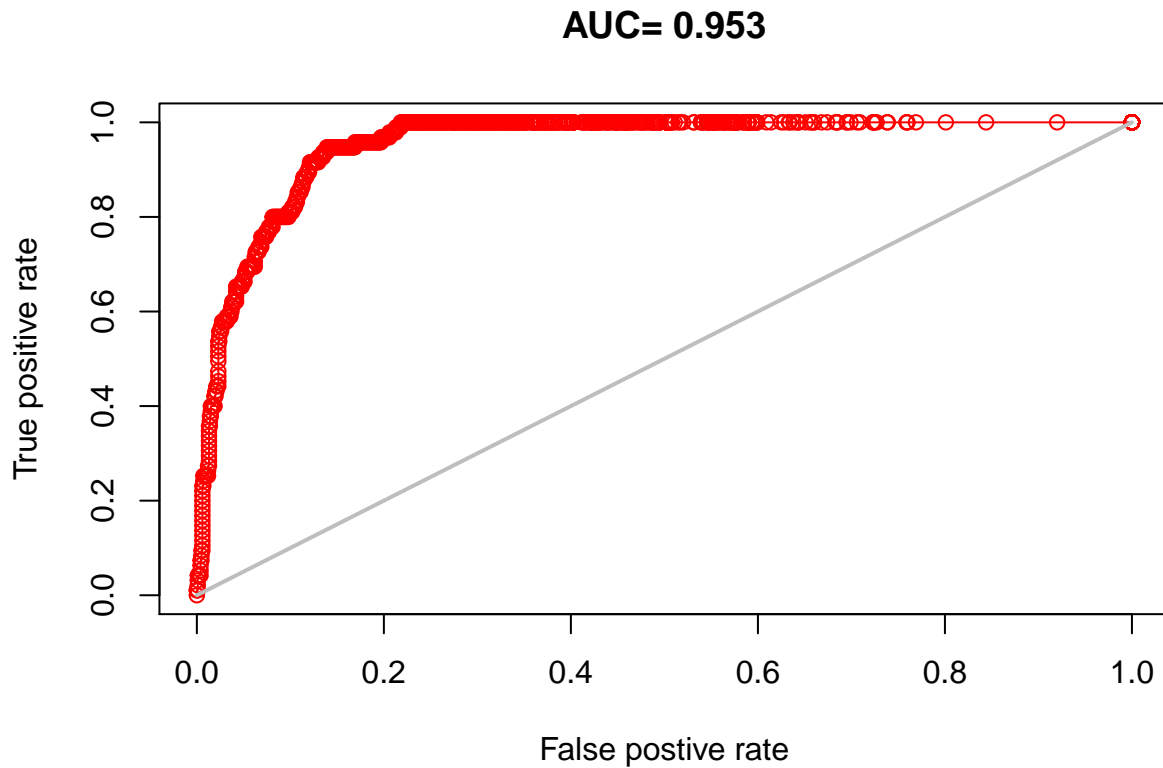
```
atChangePoints = extract(at.change, atocc)
hist(atChangePoints, main="", xlab="Change in habitat suitability")
abline(v=0, col="red")
```



## Test your model

We did not check how well our model performs with the test dataset. Evaluate your model with your test occurrences and 1000 background pseudoabsences with the receiver operating characteristic curve (**ROC** curve) and its area under the curve (**AUC**).

```
bg <- randomPoints(modelEnv, 1000) #background "pseudoabsences"  
e1 <- evaluate(at.me, p=attest, a=bg, x=modelEnv)  
plot(e1, 'ROC')
```



## Check the correlations among variables

Checking the variables for correlation is important. Download the following packages to obtain the package ENMTools.

```
#install.packages("devtools")
#library(devtools)
#install_github("danlwarren/ENMTools")
#library(ENMTools)
```

Now estimate the correlations coefficients among the variables you used for the model

```
raster.cor.matrix(modelEnv)
```

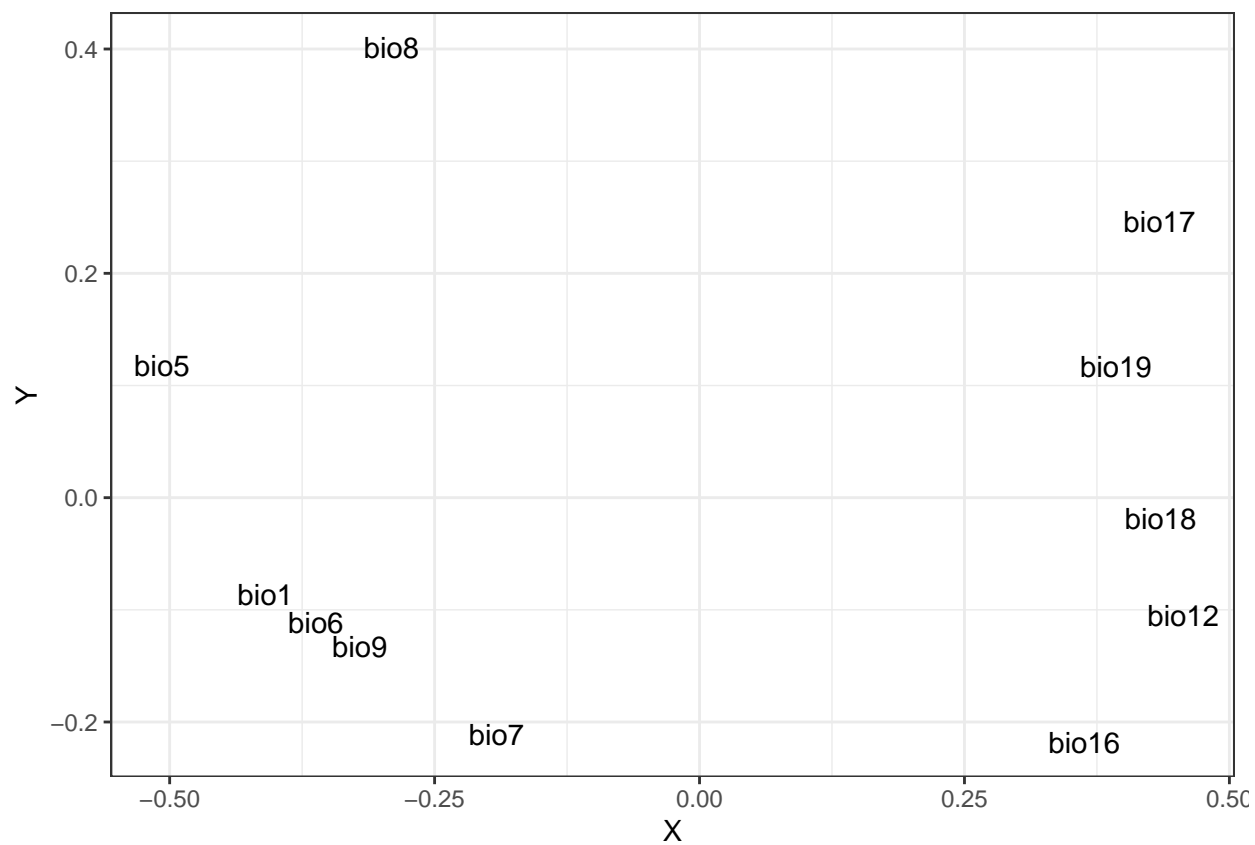
```
##          bio1          bio5          bio6          bio7          bio8
## bio1  1.00000000  0.91575353  0.9793219 -0.8424893  0.64102246
## bio5  0.91575353  1.00000000  0.8432893 -0.5760960  0.65053953
## bio6  0.97932194  0.84328935  1.0000000 -0.9251256  0.54912582
## bio7 -0.84248930 -0.57609597 -0.9251256  1.0000000 -0.37558344
## bio8  0.64102246  0.65053953  0.5491258 -0.3755834  1.00000000
## bio9  0.88012791  0.75401109  0.9081366 -0.8484835  0.35377009
## bio12 0.18965003 -0.02975637  0.2302545 -0.3711966 -0.07802963
## bio16 0.28526801  0.03454428  0.3149690 -0.4546106  0.12655735
## bio17 0.06238398 -0.10104669  0.1136095 -0.2441594 -0.26119042
## bio18 0.13441294 -0.09374053  0.1521556 -0.2976201  0.14824762
## bio19 0.14911803 -0.02467040  0.2026153 -0.3255694 -0.26445049
##          bio9          bio12          bio16          bio17          bio18
```

```
## bio1 0.88012791 0.18965003 0.28526801 0.06238398 0.13441294
## bio5 0.75401109 -0.02975637 0.03454428 -0.10104669 -0.09374053
## bio6 0.90813655 0.23025454 0.31496904 0.11360952 0.15215556
## bio7 -0.84848353 -0.37119661 -0.45461057 -0.24415942 -0.29762012
## bio8 0.35377009 -0.07802963 0.12655735 -0.26119042 0.14824762
## bio9 1.00000000 0.23637252 0.26571668 0.18825552 0.07688325
## bio12 0.23637252 1.00000000 0.84243736 0.89448306 0.81544036
## bio16 0.26571668 0.84243736 1.00000000 0.54579806 0.93386843
## bio17 0.18825552 0.89448306 0.54579806 1.00000000 0.54205948
## bio18 0.07688325 0.81544036 0.93386843 0.54205948 1.00000000
## bio19 0.32584432 0.87859451 0.56783967 0.96101717 0.52568135
##      bio19
## bio1 0.1491180
## bio5 -0.0246704
## bio6 0.2026153
## bio7 -0.3255694
## bio8 -0.2644505
## bio9 0.3258443
## bio12 0.8785945
## bio16 0.5678397
## bio17 0.9610172
## bio18 0.5256814
## bio19 1.0000000
```

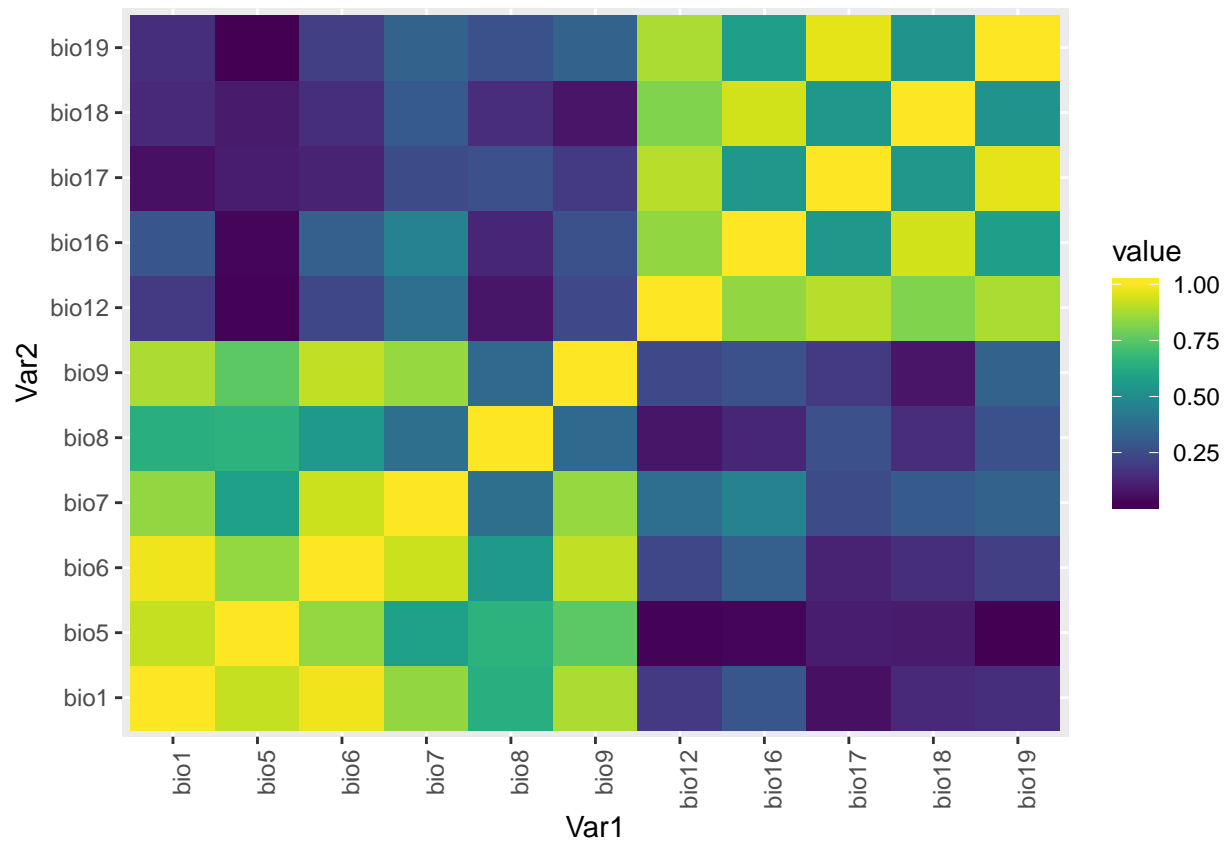
And plot them with multidimensional scaling (**MDS**) and a heatmap

```
raster.cor.plot(modelEnv)
```

```
## $cor.mds.plot
```



```
##  
## $cor.heatmap
```



## Exercise

Create a script that do the following and send it to [ev243@cornell.edu](mailto:ev243@cornell.edu):

- Find records for a species you are interested in
- Filter the records to the native range of your species
- Pick uncorrelated and interesting variables and comment using `#` why you picked them
- Create a model with current climate
- Test your model and comment if it performs well
- Use your ENM to predict to the future or past climate and comment your result

## Notes

If you can not get it to work you can try the Wallace app ([https://wallaceecomod.github.io/vignettes/wallace\\_vignette.html](https://wallaceecomod.github.io/vignettes/wallace_vignette.html)), with the following commands that will open a window of your internet browser with the app. Also find (Mann 510) or email me and ask for help if errors persist.

```
#install.packages('wallace')
#library(wallace)
#run_wallace()
```