

Proyecto Final: Clasificación de vertebrados en el bosque húmedo de Australia

Hendrik Sulbaran

July 5, 2016

La metadata usada corresponde a la **Distribución, especialización de historias de vida y filogenia de vertebrados en el bosque nublado de Australia**. <http://esapubs.org/archive/ecol/E091/181/metadata.htm>
(<http://esapubs.org/archive/ecol/E091/181/metadata.htm>)

Objetivo General:

- Determinar cuales de las variables en la metadata permiten discriminar entre taxones al grupo de vertebrados en el bosque húmedo de Australia usando varios algoritmos de clasificación o regresión del aprendizaje automático .

Descripción de las variables

Ecological ArchivesE091-181-D1

S. E. Williams, J. VanDerWal, J. Isaac, L. P. Shoo, C. Storlie, S. Fox, E. E. Bolitho, C. Moritz, C. J. Hoskin, and Y. M. Williams. 2010. Distributions, life-history specialization, and phylogeny of the rain forest vertebrates in the Austalian Wet. Tropics. *Ecology* 91:2493.

En: <http://esapubs.org/archive/ecol/E091/181/metadata.htm>

Nombre Variable	Definicion de las Variables	Unidad	Tipo	Codigo de la Variable
TAXA	Grupo taxonomico general	N/A	Caracter	BIRD = aves / FROG =ranas /MAMM=mamiferos /REPT = reptile
mass	Masa corporal promedio de individuos adultos de las especies reportadas en la literatura o de la medición directa de la especie por parte de personas afiliadas con el Centro de Biodiversidad Tropical y Cambio Climático	G	Floating point	N/A
mass_male	Igual que maa pero para los individuos de sexo masculino	G	Floating point	N/A
mass_female	Igual que maa pero para los individuos de sexo femenino	G	Floating point	N/A
size	Talle promedio de los adultos de especies reportadas en la literatura o de la medición directa de la especie por parte de personas afiliadas con el Centro de Biodiversidad Tropical y Cambio Climático. Para cada taxón mediciones definiciones son: - aves: longitud total (pico hasta la cola) - Mamíferos: cabeza combinada y la longitud del cuerpo - ranas: hocico-longitud urostyle - reptiles: longitud hocico-cloaca	mm	Floating point	N/A
size_male	Igual que maa pero para los individuos de sexo masculino	mm	Floating point	N/A

size_female	Igual que maa pero para los individuos de sexo femenino	mm	Floating point	N/A
color_dimorphism	Descripcion de si ha sido reportado o no dimorfismo en el color	N/A	Integer	1= dimorfismo en coloracion ha sido reporta en la especie 0= dimorfismo en coloracion no ha sido reportado en la especie
age_1_st_rep	Edad maxima reportada en la literatura de la primera reproduccion	Dias	Integer	N/A
max_lifespan	Esperanza de vida maxima	Años	Integer	N/A
rep_characteristics	Características reproductivas importantes para cada grupo taxonómico general reportado en la literatura	N/A	Integer	RANAS 0 = Desarrollo directa 1 = La etapa larval 2 = Criado en arroyo REPTILES 3 = criadora de embrague 4 = sin embrague melancólicos MAMÍFEROS 5 = iteróparo 6 = semélparos AVES 7 = parasitaria 8 = razas de cada región AWT 9 = cooperativa 10 = mantiene / cuidado biparental vínculo de pareja 11 = cuidado uniparental 12 = sin cuidado parental
clutch_litter_size	Promedio de descendientes producidos informado en un solo evento reproductivo. Para algunas especies	Numero de	Floating	N/A

	poco conocidas, tamaño de puesta se infiere de especies estrechamente relacionadas del mismo género.	individuos	point	
rep_per_year	Numero maximo de eventos reproductivos por año reportado en la literatura por cada especie	Numero de Eventos reproductivos	Integer	N/A
rep_seasonality	Descripción de la estacionalidad de los eventos reproductivos.	N/A	Integer	1 = todos los nacimientos en ≤ 2 meses (muy estacional) 2 = todos los nacimientos en ≤ 6 meses (moderadamente estacional) 3 = nacimientos ocurridos durante > 6 meses (no estacional).
broad_diet	Preferencias dietéticas generales de las especies.	N/A	Integer	1 => 50% hierba 2 = semillas, hierbas, raíces, hojas y frutos 3 = néctar de fruta o con invertebrados 4 => 50% invertebrados o vertebrados
activity_period	Tiempo primario de actividad	N//A	Integer	1 = Diurno 2 = Nocturno 3 = Crepuscular 4 = Activo a cualquier hora
shelter_type	Tipo de vivienda primaria registrada	N/A	Integer	1 = protegida (madriguera, construido nido, árbol hueco, etc.) 2 = intermedio (copas de los árboles, tronco hueco, debajo de cada piedra, anidan en tierra o en la grieta del suelo) 3 = abierto (debajo de los arbustos, en el césped, etc.)
strata_used	Tipo primario del estrato del hábitat usada	N/A	Integer	1 = arbórea 2 = terrestre 3 = agua dulce 4 = Volant 5 = arbórea / terrestre 6 = agua dulce / terrestre
endemic	Estatus endémico regional	N/A	Integer	1 = regionalmente endémica de la zona húmeda tropical 0 = no endémica
rainforest_specialization	Clasificación del grado de especialización de las especies en el	N/A	Integer	0 = no se ha visto en el bosque húmedo 1 = ocasionalmente registró en el bosque

	hábitat de bosque húmedo			húmedo 2 = utiliza selva como hábitat sub-óptima 3 = comúnmente registrado en el bosque húmedo, pero no las especies núcleo en el hábitat 4 = Común en bosque húmedo tropical, pero común en otros ambientes forestales 5 = bosque es hábitat central, pero también ocurre en esclerófilo húmedo 6 = Bosque húmedo
potential_dist	Área de la distribución potencial	km ²	Floating point	Área predicha por el modelo
elev_min	Elevación mínima a la que ha aparecido la especie	m.s.n.m	Floating point	N/A
elev_max	Elevación máxima a la que ha aparecido la especie	m.s.n.m	Floating point	N/A
temp_min	Temperatura mínima media anual del hábitat en la que se ha visto la especie	°C	Floating point	N/A
temp_max	Temperatura máxima media anual del hábitat en la que se ha visto la especie	°C	Floating point	N/A
precip_min	Precipitación mínima media anual del hábitat en la que se ha visto la especie	mm	Floating point	N/A
precip_max	Precipitación máxima media anual del hábitat en la que se ha visto la especie	mm	Floating point	N/A

Librerías usadas

```
library(MASS)
library(grid)
library(SnowballC)
library(e1071)
library(RColorBrewer)
library(wordcloud)
library(RWeka)
library(lattice)
library(ggplot2)
library(rpart)
library(rpart.plot)
library(neural)
library(nnet)
library(neuralnet)
library(grid)
library(MASS)
library(neuralnet)
library(C50)
library(caret)
library(rpart.plot)
library(gmodels)
library(randomForest)
```

FALSE randomForest 4.6-12

FALSE Type rfNews() to see new features/changes/bug fixes.

FALSE
FALSE Attaching package: 'randomForest'

FALSE The following object is masked from 'package:ggplot2':
FALSE
FALSE margin

```
library(kernlab)
```

FALSE
FALSE Attaching package: 'kernlab'

FALSE The following object is masked from 'package:ggplot2':
FALSE
FALSE alpha

Datos

```
d<- read.csv("~/Documents/PROYECTO_MACHINE_LEARNING/spp_data_clean1.csv", stringsAsFactors = TRUE)
View(d)
str(d)
```

```
## 'data.frame':    242 obs. of  27 variables:
## $ TAXA              : Factor w/ 5 levels "BIRD","FROG",...: 1 1 1 1 1 1 1 1 1 1 1 1 ...
## $ mass              : num  8.9 11 515 166 33.8 14.5 2330 212 56 12.6 ...
## $ mass_male         : num  0 11.6 350 179 33.4 15 2450 223 61.6 12.7 ...
## $ mass_female       : num  0 10 680 156 34.3 14 2210 201 52.7 12.5 ...
## $ size              : num  111 145 550 280 190 130 700 425 250 155 ...
## $ size_male         : num  0 0 0 0 0 0 0 0 0 0 ...
## $ size_female       : num  0 0 0 0 0 0 0 0 0 0 ...
## $ color_dimorphism  : int  0 1 0 0 0 0 0 1 0 1 ...
## $ age_1st_rep       : int  0 0 365 0 365 365 365 0 365 0 ...
## $ max_lifespan      : num  15 16 20 20 21 21 13 27 0 4 ...
## $ rep_characteristics : int  9 9 9 10 10 10 11 10 10 10 ...
## $ clutch_litter_size : num  2 2 3 2 5.5 5 18 4.5 3 2 ...
## $ rep_per_year      : num  1 5 1 1 0 0 1 1 3 1 ...
## $ rep_seasonality   : int  1 3 1 2 3 2 2 2 3 2 ...
## $ broad_diet        : int  4 4 4 3 4 4 2 3 3 4 ...
## $ activity_period   : int  1 1 1 1 1 1 1 1 1 1 ...
## $ shelter_type      : int  1 1 1 1 2 2 2 1 1 1 ...
## $ strata_used       : int  4 4 4 4 4 4 2 4 4 4 ...
## $ endemic           : int  1 0 0 0 0 0 0 0 0 1 ...
## $ rainforest_specialization: int  6 4 5 5 4 4 4 4 4 5 ...
## $ potential_dist    : num  6259 9388 22812 15288 29724 ...
## $ elev_min          : int  560 200 0 10 2 1 1 20 0 0 ...
## $ elev_max          : int  1540 1540 1437 1527 1080 880 1449 1540 1208 1183 ...
## $ temp_min          : num  15.7 15.7 16.2 15.8 18.2 19.4 16.2 15.7 17.7 17.9 ...
## $ temp_max          : num  20.9 23 24.4 24.5 24.7 24.7 24.5 23.9 24.6 25 ...
## $ precip_min        : int  1309 1237 1124 1186 817 817 877 1202 1239 1202 ...
## $ precip_max        : int  7413 7408 6505 7292 4157 3727 6602 7408 4157 3869 ...
```

```
d<-na.omit(d)#Omite los valores perdidos NA
```

La metadata

TAXA	mass	mass_male	mass_female	size	size_male	size_female	color	dimorphism	age_1st_rep	max_lifespan	rep_characteristics	clutch_litter_size	rep_per_year	rep_seasonality	broad_diet	activity_period	shelter_type	strata_used	endemic	rainforest_specialization	potential_dist	elev_m	
BIRD	8.9	0	0	111	0	0	0	0	0	15	9	2	1	1	4	1	1	4	1	6	6258.8	1	
BIRD	11	11.6	10	145	0	0	0	1	0	16	9	2	5	3	4	1	1	4	0	4	9388.3	1	
BIRD	515	350	680	550	0	0	0	0	365	20	9	3	1	1	4	1	1	4	0	5	22812.1	1	
BIRD	166	179	156	280	0	0	0	0	0	20	10	2	1	2	3	1	1	4	0	5	15288	1	
BIRD	33.8	33.4	34.3	190	0	0	0	0	365	21	10	5.5	0	3	4	1	2	4	0	4	29723.5	1	
BIRD	14.5	15	14	130	0	0	0	0	365	21	10	5	0	2	4	1	2	4	0	4	21331.3	1	
BIRD	2330	2450	2210	700	0	0	0	0	365	13	11	18	1	2	2	1	2	2	0	4	20359.8	1	
BIRD	212	223	201	425	0	0	1	0	27	10	4.5	1	2	3	1	1	1	4	0	4	11661.6	1	
BIRD	56	61.6	52.7	250	0	0	0	0	365	0	10	3	3	3	3	1	1	4	0	4	14537.7	1	
BIRD	12.6	12.7	12.5	155	0	0	1	0	4	10	2	1	2	2	4	1	1	4	1	5	18332	1	
BIRD	41	41.1	40.8	175	0	0	0	0	0	9	9	3.5	3	3	4	1	1	4	0	3	22397.8	1	
BIRD	789.5	815	764	550	0	0	0	0	2000	57	10	2	1	1	2	1	1	4	0	3	36551.2	1	
BIRD	29.5	27	22	220	0	0	0	0	6	7	1	1	0	2	4	1	1	4	0	4	6345.6	1	
BIRD	49.9	49.8	49.9	280	0	0	1	0	6	7	1	1	0	2	4	3	1	4	0	4	24496	1	
BIRD	32.9	33.7	32.1	230	0	0	1	0	6	7	1	1	1	3	4	3	1	4	0	3	35583.8	1	
BIRD	41375	34100	58500	1750	0	0	0	0	1460	40	11	4	0.8	2	3	1	3	2	0	6	18462	1	
BIRD	111	115	107	270	0	0	1	0	12	10	2	0	3	2	1	1	1	4	0	5	20327.3	1	
BIRD	23.9	23.3	24.5	180	0	0	0	0	6	8	1	0	0	2	4	3	1	4	0	4	13422.5	1	
BIRD	18.6	18.6	18.5	170	0	0	0	0	6	8	1	1	0	2	4	3	1	4	0	4	30189	1	
BIRD	18.7	18.5	18.8	160	0	0	1	0	6	7	1	1	0	1	4	3	1	4	0	4	17878	1	
BIRD	15	0	0	120	0	0	0	0	12	10	1	2	3	4	1	2	4	0	5	20632.9	1		
BIRD	39.1	40.4	37.7	210	0	0	0	0	7	10	2	1	2	4	1	1	4	1	6	7286.1	1		
BIRD	33	34.4	31.5	195	0	0	0	0	17	10	2.5	0	3	4	1	1	4	0	4	28546.2	1		
BIRD	420	421	418	410	0	0	0	0	365	15	10	1	0	3	3	1	1	4	0	5	14031.3	1	
BIRD	100	0	0	270	0	0	0	0	9	10	1.64	0	2	3	1	1	1	4	0	4	13350.4	1	
BIRD	16.9	17.3	16.5	155	0	0	1	0	12	10	2.5	1	3	4	1	1	1	4	0	5	11197.7	1	
BIRD	156	160.9	152	380	0	0	0	0	6	10	3	0	2	4	1	1	1	4	0	4	17292.2	1	
BIRD	94.8	91.8	97.7	320	0	0	1	365	13	8	0	0	0	4	1	1	1	4	0	3	26150.5	1	
BIRD	41	41.8	0	145	0	0	1	274	15	10	2.64	3	2	3	2	3	1	1	4	0	5	17473.1	1
BIRD	8.5	8.9	8.1	90	0	0	1	0	9	10	3	2	3	3	3	1	1	4	0	3	33479.8	1	
BIRD	79.6	79.1	80	300	0	0	0	0	8	10	4.5	0	1	4	1	1	1	4	0	3	36509.5	1	
BIRD	450	468.1	444.1	440	0	0	0	0	14	10	1	3	2	3	1	1	1	4	0	5	15231.2	1	
BIRD	13.1	0	0	118	0	0	0	1	244	3	10	4	3	2	2	1	1	4	0	4	9482.4	1	
BIRD	226.5	233.2	219.8	410	0	0	1	0	10	0	1	5	2	3	1	1	1	4	0	3	28237.7	1	
BIRD	7	7.5	6.7	110	0	0	0	0	4	0	2.4	0	3	4	1	1	1	4	0	3	15615.8	1	
BIRD	5.5	5.5	5.5	145	0	0	0	0	9	9	2.5	2	3	4	1	1	1	4	0	6	18070.6	1	
BIRD	7.4	7.7	7.1	108	0	0	1	0	4	0	2	0	3	4	1	1	1	4	0	3	0	1	
BIRD	7.5	7.5	7.5	108	0	0	1	0	4	10	2	0	3	4	1	1	1	4	0	3	32216.4	1	
BIRD	35.6	38.5	32.7	195	0	0	0	0	0	0	1.6	1	2	4	1	1	1	4	0	5	12913.4	1	
BIRD	93.2	97.9	88.5	200	0	0	0	0	20	8	0	0	2	4	1	1	2	4	0	3	28454	1	
BIRD	32.1	32.2	32	195	0	0	1	0	5	10	1	0	2	2	3	1	1	1	4	0	3	26206.1	1
BIRD	36.4	36.8	35.9	482	48.1	48.3	0	0	5	10	2.5	0	2	3	1	1	1	4	1	5	12287.3	1	

Aprendizaje automático

Reglas de Aprendizaje: RIPPER

```
modelripper<-JRip(TAXA~ ., data = d)
modelripper
```

```
## JRIP rules:
## =====
##
## (broad_diet <= 3) and (rep_characteristics <= 6) and (rep_characteristics >= 5) =>
TAXA=MAMM (24.0/0.0)
## (rep_seasonality <= 1) and (rep_characteristics >= 5) and (rep_characteristics <=
6) => TAXA=MAMM (5.0/0.0)
## (rep_characteristics <= 2) and (age_1st_rep >= 365) and (size <= 110) => TAXA=FR0G
(29.0/0.0)
## (rep_characteristics <= 4) => TAXA=REPT (54.0/7.0)
## => TAXA=BIRD (90.0/2.0)
##
## Number of Rules : 5
```

El clasificador **JRip** ha aprendido 5 reglas para clasificar.

Primera regla:

- Si la preferencia dietética de las especies (**broad_diet**)<= 3 es decir; come hierbas, semillas, néctar y frutos y las características reproductivas es (**rep_characteristics**)<=6 y (**rep_characteristics**)>=5 es decir; iteróparo y seméparo, la clasificación general de la especie es **mamífero**

Segunda regla:

- Si la estacionalidad de los eventos reproductivos (**rep_seasonality** <= 1) es decir; todos los nacimientos en menos de 2 meses (muy estacional), y las características reproductivas es (**rep_characteristics**)<=6 y (**rep_characteristics**)>=5 es decir; iteróparo y seméparo, la clasificación general de la especie es **mamífero**

Tercera regla:

- Si las características reproductivas (**rep_characteristics** ≤ 2) es decir; se desarrolla directamente, tiene etapa larval y es criado en arroyo, y la edad máxima (en días) de reproducción es (**age_1st_rep** ≥ 365) y el tamaño promedio de los adultos desde el hocico–final de la columna vertebral (en mm) (**size** ≤ 110) la clasificación general de la especie es **rana**

Cuarta regla:

- Si las características reproductivas (**rep_characteristics** ≤ 4) la clasificación general de la especie es reptil.

Quinta regla:

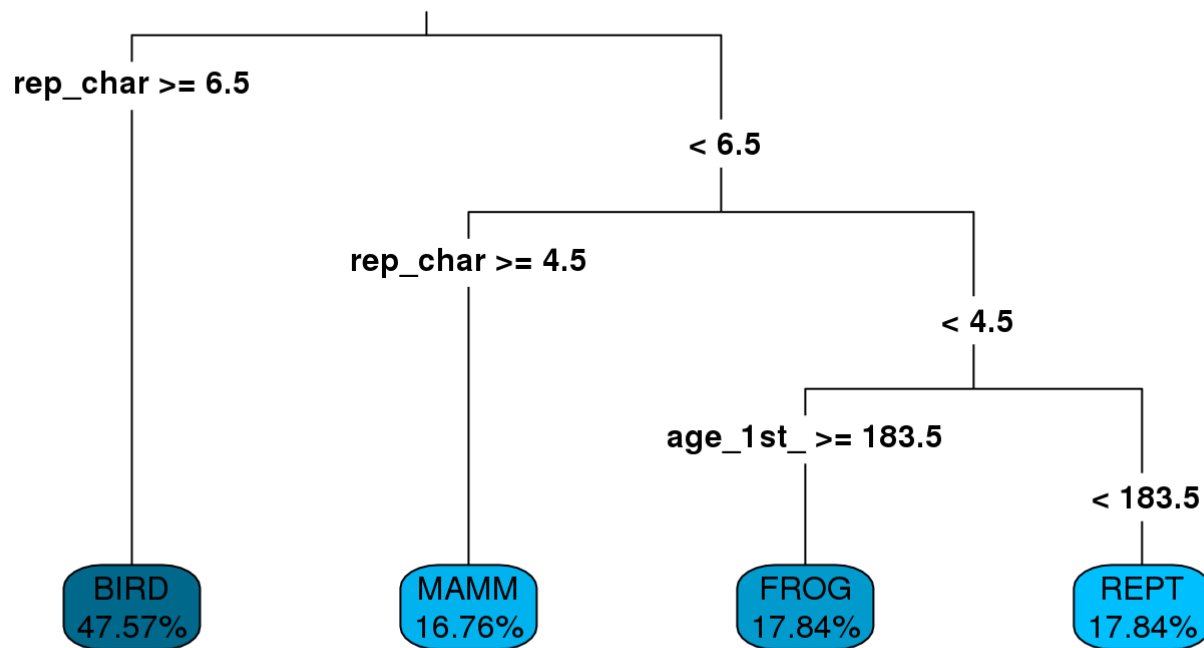
- Por último, la quinta regla implica que cualquier especie que no esta cubierta por las cuatro reglas anteriores indica que la clasificación general de la especie es **ave**.

Árboles de regresión

```
set.seed(300)
d_train<- d[1:185, ]#75%
d_test<- d[186:242, ]#25%
m.rpart <- rpart(TAXA ~ ., data = d_train)
m.rpart
```

```
## n= 185
##
## node), split, n, loss, yval, (yprob)
##      * denotes terminal node
##
## 1) root 185 92 BIRD (0.5 0.17 0.16 0.0054 0.16)
##    2) rep_characteristics>=6.5 88  0 BIRD (1 0 0 0 0) *
##    3) rep_characteristics< 6.5 97 66 FROG (0.052 0.32 0.31 0.01 0.31)
##      6) rep_characteristics>=4.5 31  1 MAMM (0 0 0.97 0.032 0) *
##      7) rep_characteristics< 4.5 66 35 FROG (0.076 0.47 0 0 0.45)
##        14) age_1st_rep>=183.5 33  4 FROG (0.03 0.88 0 0 0.091) *
##        15) age_1st_rep< 183.5 33  6 REPT (0.12 0.061 0 0 0.82) *
```


Árbol de Decisión



Vemos que en el caso del algoritmo de árboles de regresión, las variables que permites discriminar en la clasificación taxonomica general sigue siendo la (**rep_characteristics**>=6.5), y la (**age_1st_rep** >= 183.5)

```

#===== evaluando el rendimiento del modelo =====
p.rpart<-predict(m.rpart, d_test, type = 'class')
MC<- table(d_test[, 'TAXA'], p.rpart)
MC

```

```

##      p.rpart
##      BIRD  FROG  MAMM  MONO  REPT
##  BIRD      0     0     0     0     0
##  FROG      0     0     0     0     0
##  MAMM      0     0     0     0     0
##  MONO      0     0     0     0     0
##  REPT      0     2     0     0    15

```

En la matriz de confusión vemos que sólo 2 individuos de **REP** fueron clasificados erróneamente como **FROG**

```
summary(p.rpart)
```

```

##  BIRD  FROG  MAMM  MONO  REPT
##    0    42    0    0    15

```

```
summary(d$TAXA)
```

```
## BIRD FROG MAMM MONO REPT
##   93   31   30    1   47
```

Random Forest

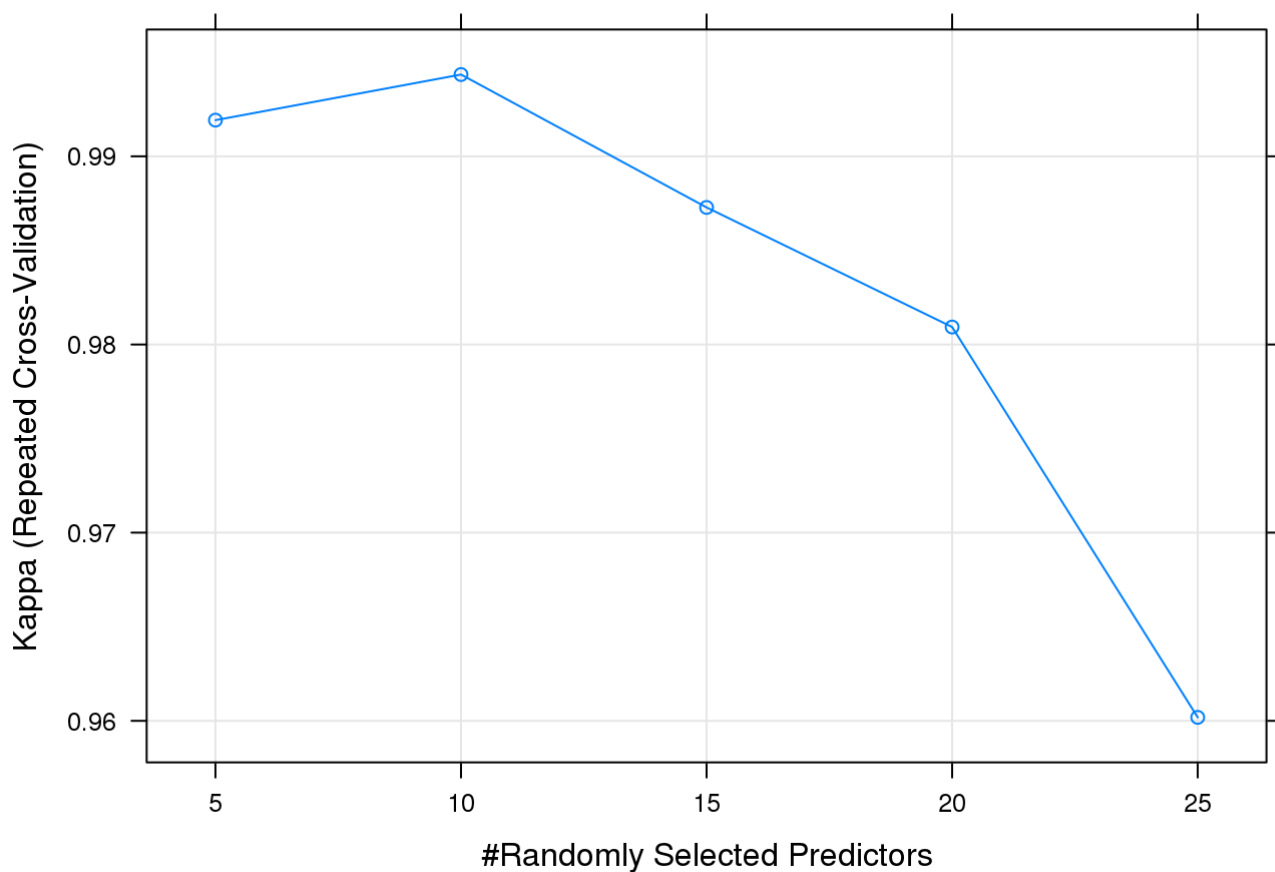
```
set.seed(300)
rf<- randomForest(TAXA~., data = d)
rf
```

```
##
## Call:
## randomForest(formula = TAXA ~ ., data = d)
##           Type of random forest: classification
##           Number of trees: 500
## No. of variables tried at each split: 5
##
##           OOB estimate of  error rate: 1.49%
## Confusion matrix:
##           BIRD FROG MAMM MONO REPT class.error
## BIRD      93    0    0    0    0 0.00000000
## FROG       0   30    0    0    1 0.03225806
## MAMM       1    0   29    0    0 0.03333333
## MONO       0    0    1    0    0 1.00000000
## REPT       0    0    0    0   47 0.00000000
```

En la matriz de confusión, vemos 2 individuos mal clasificados: 1 **MAMM** como **BIRD**, y 1 **FROG** como **REPT**

```
ctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 10)
set.seed(5)
grid_rf <- expand.grid(.mtry = c(5,10,15,20,25))
m_rf <- train(TAXA~ ., data = d, method = "rf", metric = "Kappa", trControl = ctrl, tuneGrid = grid_rf)
options(warn=0)
print(m_rf)
```

```
## Random Forest
##
## 202 samples
## 26 predictor
## 5 classes: 'BIRD', 'FROG', 'MAMM', 'MONO', 'REPT'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 10 times)
## Summary of sample sizes: 182, 183, 179, 183, 182, 182, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##   5     0.9944653  0.9919229
##   10    0.9961083  0.9943464
##   15    0.9912114  0.9872834
##   20    0.9867854  0.9809278
##   25    0.9723591  0.9601833
##
## Kappa was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 10.
```



Vemos que el parámetro con métrica = 5, tiene una exactitud del 99%, y un valor de kappa de 0.99

Cambiando la variable TAXA

```
d2<- read.csv("~/Documents/PROYECTO_MACHINE_LEARNING/spp_data_clean2.csv", stringsAsFactors = TRUE)
View(d2)
str(d2)
```

```
## 'data.frame':    242 obs. of  27 variables:
## $ TAXA                : int  1 1 1 1 1 1 1 1 1 1 ...
## $ mass                : num  8.9 11 515 166 33.8 14.5 2330 212 56 12.6 ...
## $ mass_male           : num  0 11.6 350 179 33.4 15 2450 223 61.6 12.7 ...
## $ mass_female         : num  0 10 680 156 34.3 14 2210 201 52.7 12.5 ...
## $ size                : num  111 145 550 280 190 130 700 425 250 155 ...
## $ size_male           : num  0 0 0 0 0 0 0 0 0 0 ...
## $ size_female         : num  0 0 0 0 0 0 0 0 0 0 ...
## $ color_dimorphism    : int  0 1 0 0 0 0 0 1 0 1 ...
## $ age_1st_rep         : int  0 0 365 0 365 365 365 0 365 0 ...
## $ max_lifespan        : num  15 16 20 20 21 21 13 27 0 4 ...
## $ rep_characteristics  : int  9 9 9 10 10 10 11 10 10 10 ...
## $ clutch_litter_size  : num  2 2 3 2 5.5 5 18 4.5 3 2 ...
## $ rep_per_year        : num  1 5 1 1 0 0 1 1 3 1 ...
## $ rep_seasonality     : int  1 3 1 2 3 2 2 2 3 2 ...
## $ broad_diet          : int  4 4 4 3 4 4 2 3 3 4 ...
## $ activity_period     : int  1 1 1 1 1 1 1 1 1 1 ...
## $ shelter_type        : int  1 1 1 1 2 2 2 1 1 1 ...
## $ strata_used         : int  4 4 4 4 4 4 2 4 4 4 ...
## $ endemic             : int  1 0 0 0 0 0 0 0 0 1 ...
## $ rainforest_specialization: int  6 4 5 5 4 4 4 4 4 5 ...
## $ potential_dist      : num  6259 9388 22812 15288 29724 ...
## $ elev_min            : int  560 200 0 10 2 1 1 20 0 0 ...
## $ elev_max            : int  1540 1540 1437 1527 1080 880 1449 1540 1208 1183
## ...
## $ temp_min           : num  15.7 15.7 16.2 15.8 18.2 19.4 16.2 15.7 17.7 17.
9 ...
## $ temp_max           : num  20.9 23 24.4 24.5 24.7 24.7 24.5 23.9 24.6 25
## ...
## $ precip_min         : int  1309 1237 1124 1186 817 817 877 1202 1239 1202
## ...
## $ precip_max         : int  7413 7408 6505 7292 4157 3727 6602 7408 4157 386
9 ...
```

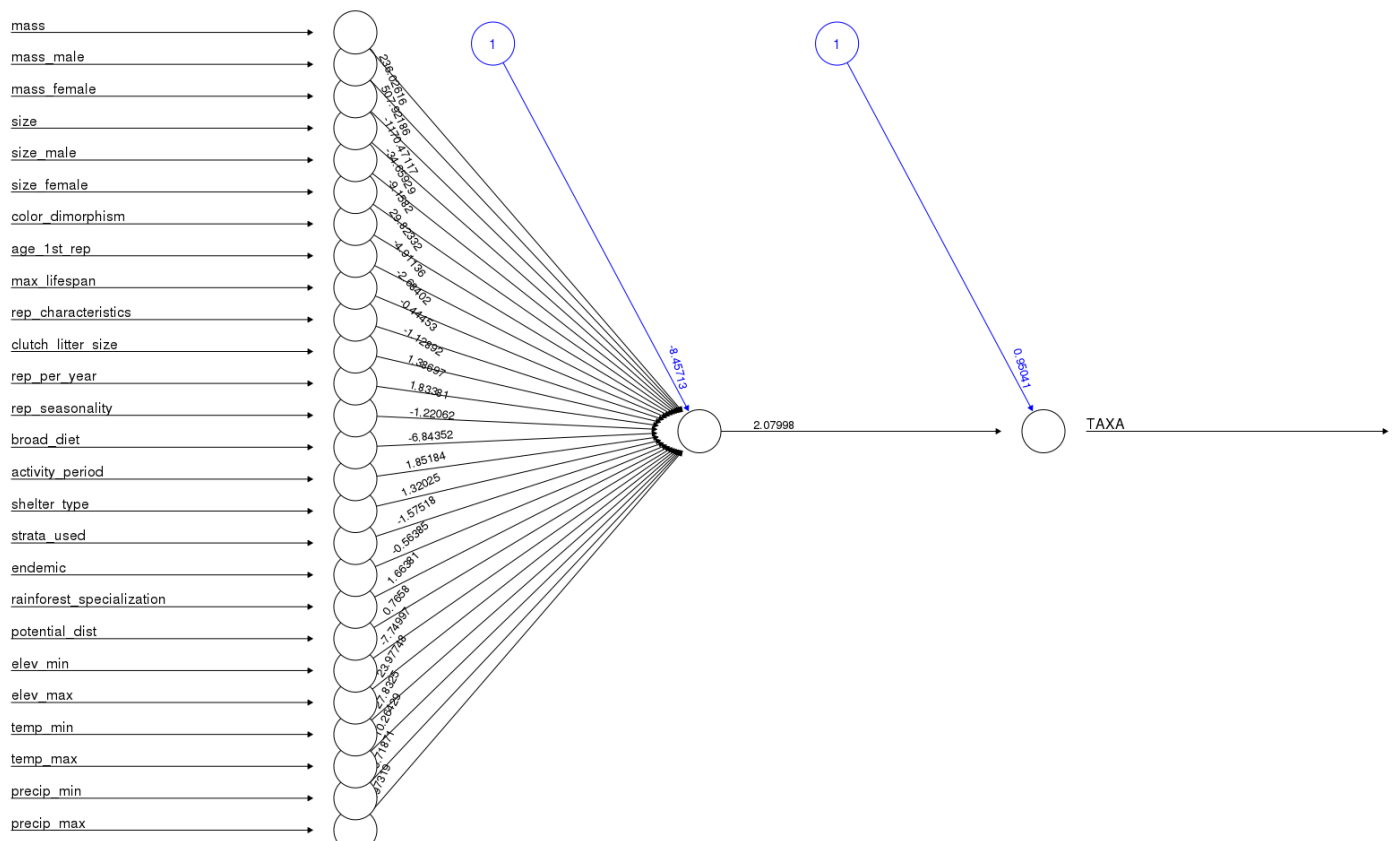
```
d2<-na.omit(d2)
#=== Normalizamos las variables =====
d2<- Normalize(d2)
```

Redes Neurales

Para aplicar el algoritmo de **neuralnet** hemos convertido la variable TAXA categoriaca como factor, a una variable categorica numérica.

```
#==== neuralnet =====
d2$TAXA<-as.numeric(d2$TAXA)
set.seed(300)
d2_nor_train<- d2[1:152, ]#75%
d2_nor_test<- d2[153:202, ]#25%
set.seed(12345)
m1<- neuralnet(TAXA ~ mass + mass_male + mass_female +
               size + size_male + size_female +
               color_dimorphism + age_1st_rep +
               max_lifespan + rep_characteristics +
               clutch_litter_size + rep_per_year +
               rep_seasonality + broad_diet + activity_period +
               shelter_type + strata_used + endemic +
               rainforest_specialization +
               potential_dist + elev_min +
               elev_max + temp_min + temp_max +
               precip_min + precip_max, data = d2_nor_train, hidden = 1)

plot(m1)
```



```
summary(m1)
```

```
##              Length Class      Mode
## call          4   -none-    call
## response     152   -none-   numeric
## covariate    3952   -none-   numeric
## model.list     2   -none-    list
## err.fct        1   -none-   function
## act.fct        1   -none-   function
## linear.output  1   -none-   logical
## data          27 data.frame list
## net.result     1   -none-    list
## weights       1   -none-    list
## startweights  1   -none-    list
## generalized.weights 1 -none-    list
## result.matrix 32   -none-   numeric
```

```
results1<-compute(m1,d2_nor_test[1:26])
predict_TAXA<- results1$net.result
cor(predict_TAXA, d2_nor_test$TAXA)
```

```
##              [,1]
## [1,] -0.2467117677
```

El modelo no funciona puesto que la correlación es negativa