

Tema 9. Proyecto 4

Viacheslav Shalisko

23 de julio de 2019

Modelo correlativo de distribución de una especie biológica

Variante con SVM para casos cuando MaxEnt no funciona

Cargar las bibliotecas requeridas

```
library(sp)
library(raster)
library(dismo)
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:raster':
##
##      interpolate
```

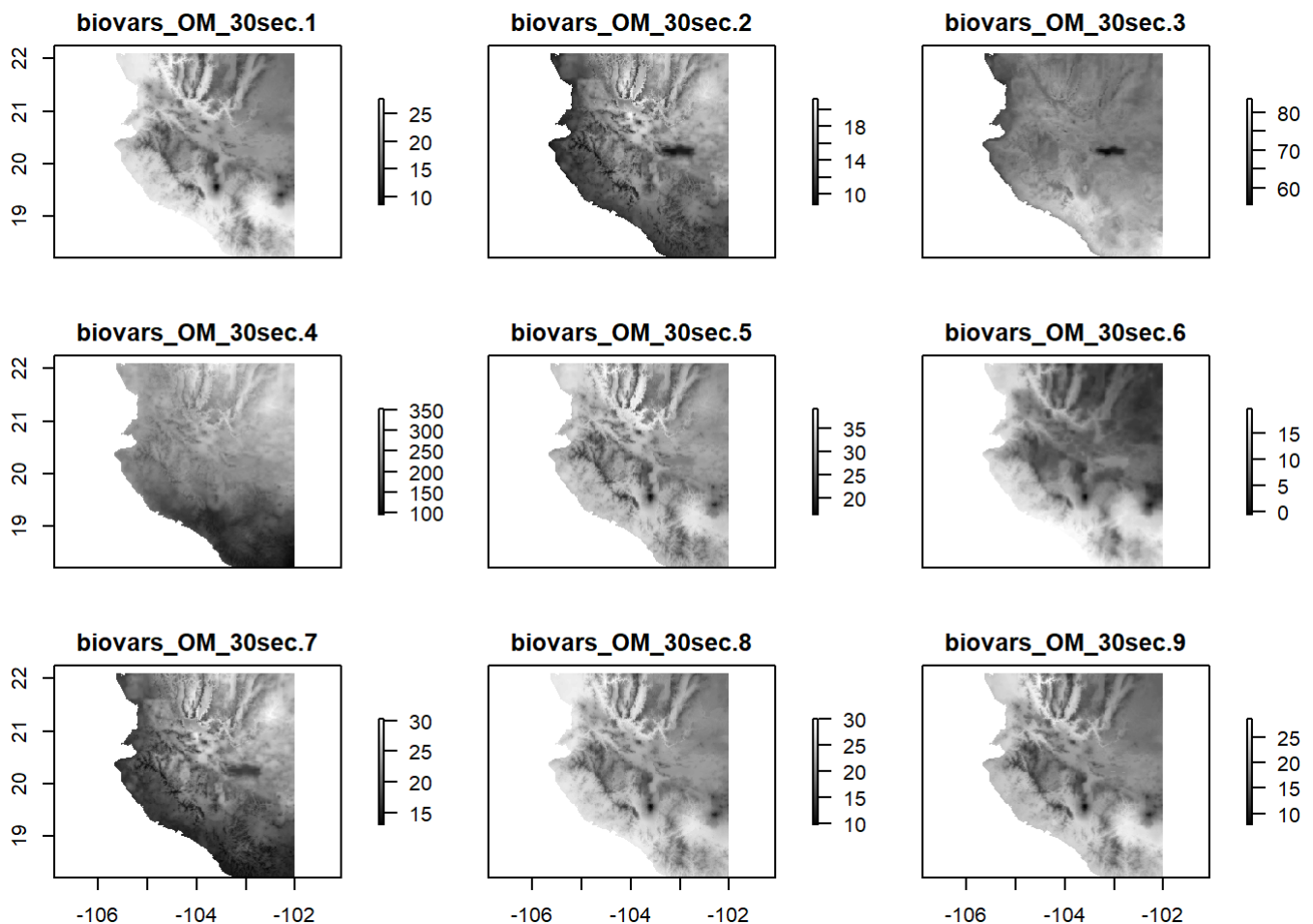
Lectura de los archivos fuente

1) GeoTIFF con 19 capas de variables BIOCLIM (WorldClim 2)

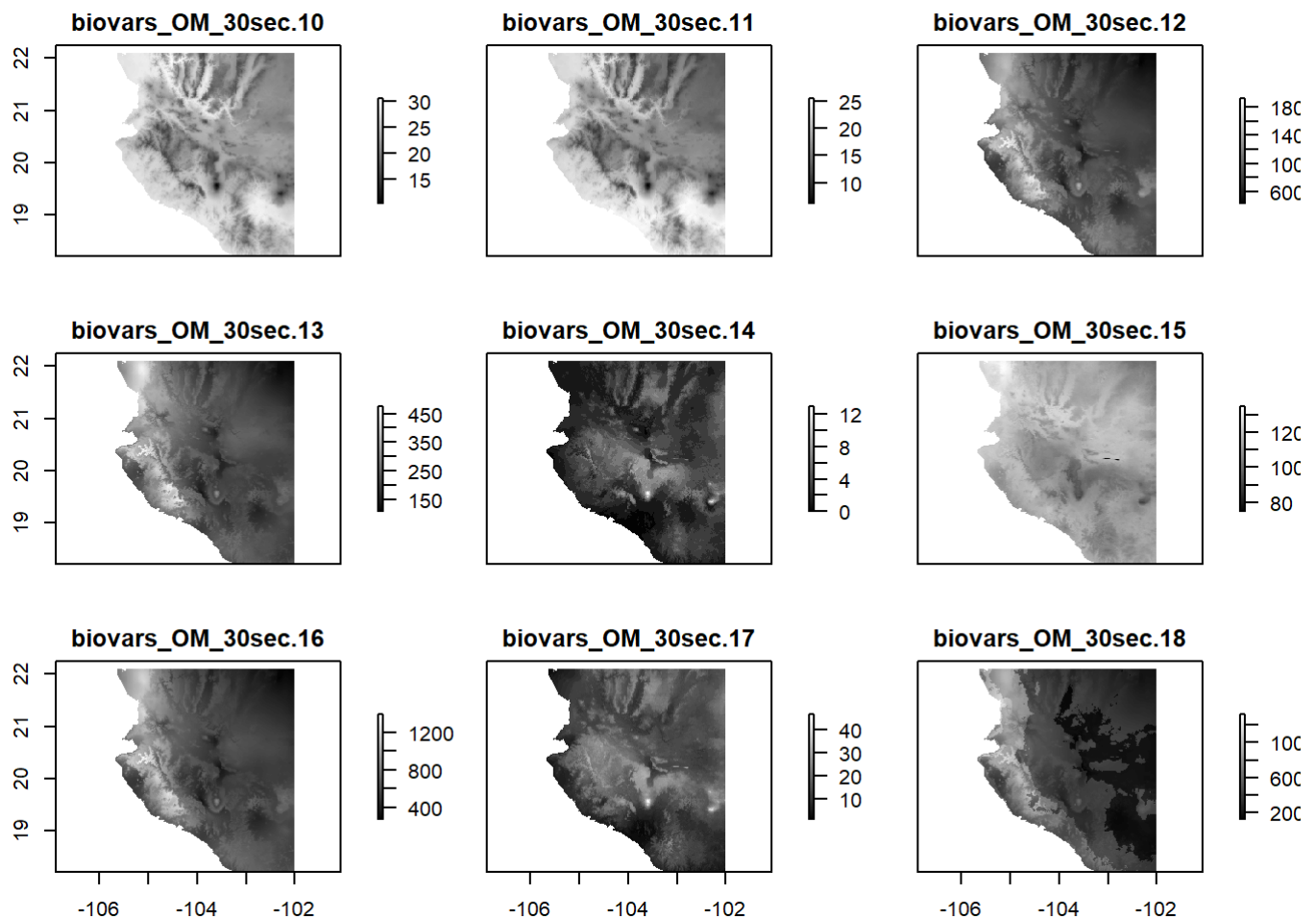
```
# raster con variables BIOCLIM (WorldClim 2)
bio_vars <- stack("datos/biovars_0M_30sec.tif")
bio_vars
```

```
## class      : RasterStack
## dimensions : 484, 469, 226996, 19  (nrow, ncol, ncell, nlayers)
## resolution : 0.008333333, 0.008333333  (x, y)
## extent     : -105.9248, -102.0165, 18.21198, 22.24531  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## names      : biovars_OM_30sec.1, biovars_OM_30sec.2, biovars_OM_30sec.3, biovars_OM_30sec.
4, biovars_OM_30sec.5, biovars_OM_30sec.6, biovars_OM_30sec.7, biovars_OM_30sec.8, biovars_OM
_30sec.9, biovars_OM_30sec.10, biovars_OM_30sec.11, biovars_OM_30sec.12, biovars_OM_30sec.13,
biovars_OM_30sec.14, biovars_OM_30sec.15, ...
## min values :          7.804167,          8.383333,          54.437225,          94.80443
6,          15.700000,          -1.200000,          12.800001,          9.116667,
7.083333,          9.466666,          5.550000,          431.000000,          108.000000,
0.000000,          74.650505, ...
## max values :          27.89167,          21.29167,          83.96118,          353.8048
1,          39.40000,          19.60000,          30.30000,          30.23333,
28.76667,          30.76667,          25.50000,          1936.00000,          478.00000,
14.00000,          135.29018, ...
```

```
plot(bio_vars,c(1:9), col = grey(1:100/100), nc=3)
```



```
plot(bio_vars,c(10:18), col = grey(1:100/100), nc=3)
```



2) Tabla GBiF con registros de presencia

```
# tabla de registros de ocurrencia
ocurrencias <- read.delim("datos/0016216-190621201848488.csv")
dim(ocurrencias)
```

```
## [1] 4146 45
```

```
str(ocurrencias)
```

```
## 'data.frame': 4146 obs. of 45 variables:
## $ gbifID : num 2.28e+09 2.28e+09 2.27e+09 2.27e+09 2.27e+09 ...
## $ datasetKey : Factor w/ 43 levels "33948acc-bfac-4107-92c8-6464de38782
2",...: 5 5 5 5 5 10 10 38 38 38 ...
## $ occurrenceID : Factor w/ 4135 levels "", "15629d62-f56c-4a37-84a2-ea2d66
cba4c5",...: 59 52 63 62 53 3653 3652 6 15 10 ...
## $ kingdom : Factor w/ 1 level "Plantae": 1 1 1 1 1 1 1 1 1 1 ...
## $ phylum : Factor w/ 1 level "Tracheophyta": 1 1 1 1 1 1 1 1 1 1
...
## $ class : Factor w/ 1 level "Magnoliopsida": 1 1 1 1 1 1 1 1 1 1
...
## $ order : Factor w/ 1 level "Fagales": 1 1 1 1 1 1 1 1 1 1 ...
## $ family : Factor w/ 1 level "Fagaceae": 1 1 1 1 1 1 1 1 1 1 ...
## $ genus : Factor w/ 1 level "Quercus": 1 1 1 1 1 1 1 1 1 1 ...
## $ species : Factor w/ 67 levels "", "Quercus acutifolia",...: 52 52 52
37 65 4 20 1 37 4 ...
## $ infraspecificEpithet : logi NA NA NA NA NA NA ...
## $ taxonRank : Factor w/ 2 levels "GENUS","SPECIES": 2 2 2 2 2 2 2 1 2
2 ...
## $ scientificName : Factor w/ 95 levels "Quercus acapulcensis Trel.",...: 73
73 73 53 92 5 33 46 52 5 ...
## $ countryCode : Factor w/ 1 level "MX": 1 1 1 1 1 1 1 1 1 1 ...
## $ locality : Factor w/ 2362 levels "", "'Cerro La Petaca' a SW outlier
of the Sa de Manantlan; along trail to El durazno, 5-10 Km ESE of Casimiro Casti"| __truncate
d__,...: 1 1 1 1 1 2261 1278 1353 963 1543 ...
## $ publishingOrgKey : Factor w/ 16 levels "174c6da2-e002-42fe-9f56-2f9d670240f
4",...: 5 5 5 5 5 10 10 9 9 9 ...
## $ decimalLatitude : num 20.6 20.7 20.8 20.8 20.6 ...
## $ decimalLongitude : num -103 -104 -103 -103 -104 ...
## $ coordinateUncertaintyInMeters: num NA 15 NA 31 31 NA NA 15000 10000 5000 ...
## $ coordinatePrecision : logi NA NA NA NA NA NA ...
## $ elevation : num NA NA NA NA NA ...
## $ elevationAccuracy : num NA NA NA NA NA NA NA NA NA NA ...
## $ depth : num NA NA NA NA NA NA NA NA NA NA ...
## $ depthAccuracy : num NA NA NA NA NA NA NA NA NA NA ...
## $ eventDate : Factor w/ 1333 levels "", "1838-10-17T00:00:00Z",...: 1326
1321 1331 1332 1322 1288 1301 10 226 812 ...
## $ day : int 26 18 20 20 25 17 3 29 26 19 ...
## $ month : int 4 3 5 5 3 6 11 10 7 2 ...
## $ year : int 2019 2019 2019 2019 2019 2013 2013 1904 1965 1987
...
## $ taxonKey : int 2879646 2879646 2879646 2879847 2879637 2878107 288
0560 2877951 2877219 2878107 ...
## $ speciesKey : int 2879646 2879646 2879646 2879847 2879637 2878107 288
0560 NA 2879847 2878107 ...
## $ basisOfRecord : Factor w/ 4 levels "HUMAN_OBSERVATION",...: 1 1 1 1 1 3 3
3 3 3 ...
## $ institutionCode : Factor w/ 45 levels "ASU","CAS","CETI",...: 21 21 21 21 2
1 29 29 1 1 1 ...
## $ collectionCode : Factor w/ 50 levels "Angiosperms",...: 39 39 39 39 39 33
33 41 41 41 ...
## $ catalogNumber : Factor w/ 2940 levels "00169001","00169024",...: 833 736
931 914 764 224 223 2752 2755 2753 ...
## $ recordNumber : Factor w/ 319 levels "", "1026", "10484",...: 1 1 1 1 1 302
25 19 21 20 ...
## $ identifiedBy : Factor w/ 268 levels "", "A. Alvarez Alvarez",...: 1 1 1 1
1 186 186 1 109 1 ...
```

```
## $ dateIdentified      : Factor w/ 51 levels "", "1913-01-01T00:00:00Z", ...: 51 39
49 48 40 24 24 1 1 1 ...
## $ license             : Factor w/ 3 levels "CC_BY_4_0", "CC_BY_NC_4_0", ...: 2 2 2
2 2 1 1 1 1 1 ...
## $ rightsHolder       : Factor w/ 17 levels "", "Adilson Pinedo-Escatel", ...: 6 10
3 5 11 13 13 1 1 1 ...
## $ recordedBy         : Factor w/ 791 levels "", "A. Alvarez A.", ...: 162 392 106
113 425 396 39 84 485 675 ...
## $ typeStatus         : Factor w/ 6 levels "", "HOLOTYPE", ...: 1 1 1 1 1 1 1 1 1
...
## $ establishmentMeans : logi NA NA NA NA NA NA ...
## $ lastInterpreted    : Factor w/ 3784 levels "2018-12-05T23:22:04.639Z", ...: 378
1 3780 3779 3778 3777 3767 3768 3479 3480 3478 ...
## $ mediaType          : Factor w/ 2 levels "", "STILLIMAGE": 2 2 2 2 1 1 1 2 2 2
...
## $ issue              : Factor w/ 54 levels "COORDINATE_REPROJECTED", ...: 19 20 2
0 20 20 19 19 19 19 ...
```

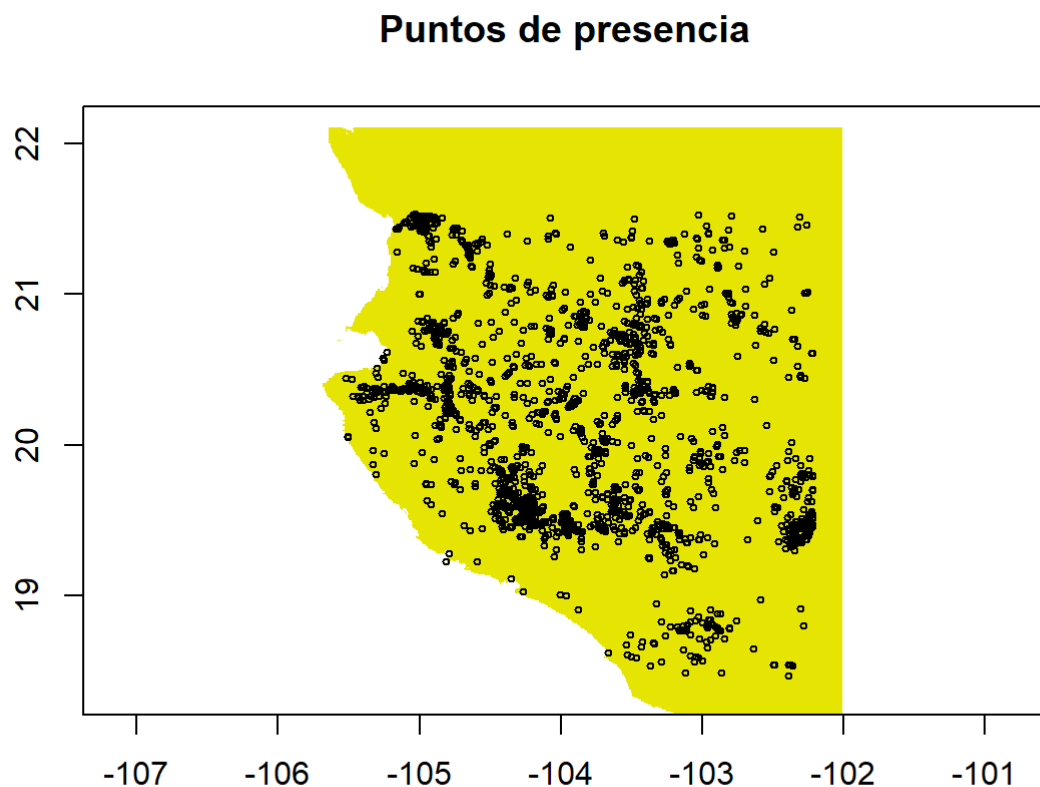
```
levels(ocurrencias$species)
```

```
## [1] "" "Quercus acutifolia"
## [3] "Quercus affinis" "Quercus aristata"
## [5] "Quercus candicans" "Quercus castanea"
## [7] "Quercus chihuahuensis" "Quercus convallata"
## [9] "Quercus cortesii" "Quercus crassifolia"
## [11] "Quercus crassipes" "Quercus cualensis"
## [13] "Quercus deserticola" "Quercus diversifolia"
## [15] "Quercus durifolia" "Quercus dysophylla"
## [17] "Quercus eduardi" "Quercus elliptica"
## [19] "Quercus fulva" "Quercus gentryi"
## [21] "Quercus germana" "Quercus glaucescens"
## [23] "Quercus glaucoides" "Quercus grahamii"
## [25] "Quercus gravesii" "Quercus grisea"
## [27] "Quercus hintonii" "Quercus iltisii"
## [29] "Quercus insignis" "Quercus jonesii"
## [31] "Quercus laeta" "Quercus lancifolia"
## [33] "Quercus laurifolia" "Quercus laurina"
## [35] "Quercus laxa" "Quercus liebmannii"
## [37] "Quercus magnoliifolia" "Quercus martinezii"
## [39] "Quercus mexicana" "Quercus microphylla"
## [41] "Quercus nixoniana" "Quercus oaxacana"
## [43] "Quercus obtusata" "Quercus ocoteifolia"
## [45] "Quercus oocarpa" "Quercus peduncularis"
## [47] "Quercus planipocula" "Quercus potosina"
## [49] "Quercus praeco" "Quercus praineana"
## [51] "Quercus rekonis" "Quercus resinosa"
## [53] "Quercus rugosa" "Quercus salicifolia"
## [55] "Quercus scytophylla" "Quercus sebifera"
## [57] "Quercus sideroxyla" "Quercus splendens"
## [59] "Quercus subspathulata" "Quercus tuberculata"
## [61] "Quercus tuitensis" "Quercus urbani"
## [63] "Quercus uxoris" "Quercus vicentensis"
## [65] "Quercus viminea" "Quercus virginiana"
## [67] "Quercus xalapensis"
```

Generar mascara de tierra firme

```
mask_reclass_table <- matrix(c(-Inf, Inf, 1), ncol=3, byrow=TRUE)
bio_vars_mask <- reclassify(subset(bio_vars,1),mask_reclass_table)

# representar todos registros
plot(bio_vars_mask, legend = FALSE, main = "Puntos de presencia")
points(ocurrencias$decimalLongitude, ocurrencias$decimalLatitude, cex = 0.5)
```



Generar 1000 puntos aleatorios para muestreo de la variabilidad ambiental en zona de tierra firme

```
set.seed(0)
puntos_aleatorios_fondo <- randomPoints(bio_vars_mask, 1000)
```

Seleccionar la especie de interés (solo columnas de nombre de especie y coordenadas)

```
ocurrencias_especie <- ocurrencias[ocurrencias$species == "Quercus candicans",c("species","decimalLongitude","decimalLatitude")]
```

```
dim(ocurrencias_especie)
```

```
## [1] 179 3
```

```
head(ocurrencias_especie)
```

```
##           species decimalLongitude decimalLatitude
## 21  Quercus candicans      -103.4005         20.36783
## 24  Quercus candicans      -103.6285         19.45942
## 106 Quercus candicans      -103.6166         19.72361
## 107 Quercus candicans      -103.6166         19.72361
## 114 Quercus candicans      -104.2367         19.55667
## 200 Quercus candicans      -103.8483         20.78667
```

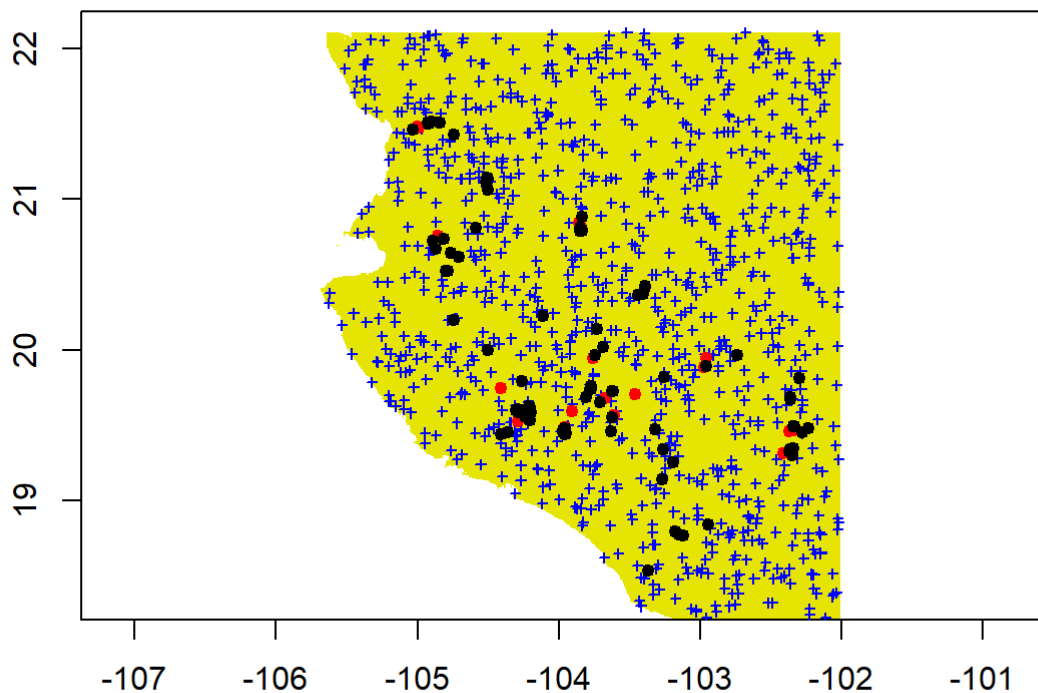
Subdividir datos en dos juegos: entrenamiento y control

```
# separar datos
fold <- kfold(ocurrencias_especie, k = 4)

ocurrencias_especie_entrenamiento <- ocurrencias_especie[ fold != 4, ]
ocurrencias_especie_control <- ocurrencias_especie[ fold == 4, ]

plot(bio_vars_mask, legend = FALSE, main = "Puntos de presencia de la especie y muestreo")
points(puntos_aleatorios_fondo, col = "blue", cex = 0.5, pch = 3)
points(ocurrencias_especie_control[,2], ocurrencias_especie_control[,3],
      col = "red", cex = 0.8, pch = 19)
points(ocurrencias_especie_entrenamiento[,2], ocurrencias_especie_entrenamiento[,3],
      col = "black", cex = 0.8, pch = 19)
```

Puntos de presencia de la especie y muestreo



Inferir el modelo SVM (Support Vector Machines)

Búsqueda del modelo óptimo puede tardar algunos minutos

```
# generar vector de presencias (1) y valores del fondo (0) para ariabl independiente
y <- c(rep(1, dim(ocurrencias_especie_entrenamiento)[1]),
      rep(0, dim(puntos_aleatorios_fondo)[1]))

names(bio_vars) <- paste('b',1:19,sep='')

# generar dataframe de variables independientes para presencias y ausencias
x_presencia <- extract(bio_vars, ocurrencias_especie_entrenamiento[,2:3])
x_fondo <- extract(bio_vars, puntos_aleatorios_fondo)
# acomodar valores de muestreo de variables independientes en una sola tabla
x <- rbind(x_presencia, x_fondo)

specie_data <- as.data.frame(cbind(x,y))
dim(specie_data)
```

```
## [1] 1134    20
```

```
str(specie_data)
```

```
## 'data.frame':    1134 obs. of  20 variables:
## $ b1 : num  14.7 17.1 14.5 14.5 16.2 ...
## $ b2 : num  12.4 13.2 12.1 12.1 10.8 ...
## $ b3 : num  68.1 70.3 69.5 69.5 69.7 ...
## $ b4 : num  205 188 208 208 191 ...
## $ b5 : num  23.7 26.1 22.8 22.8 23.7 ...
## $ b6 : num  5.5 7.4 5.4 5.4 8.2 ...
## $ b7 : num  18.2 18.7 17.4 17.4 15.5 ...
## $ b8 : num  16.3 18.7 16.5 16.5 18 ...
## $ b9 : num  14.2 16.1 13.5 13.5 14.7 ...
## $ b10: num  17 19.1 16.7 16.7 18.1 ...
## $ b11: num  12 14.6 11.7 11.7 13.7 ...
## $ b12: num  1002 1005 971 971 1198 ...
## $ b13: num  240 214 218 218 271 274 285 267 248 194 ...
## $ b14: num  7 5 7 7 5 7 3 6 7 4 ...
## $ b15: num  104.3 97.5 97.3 97.3 103.4 ...
## $ b16: num  616 583 568 568 745 696 763 678 646 523 ...
## $ b17: num  26 21 29 29 24 28 24 26 27 17 ...
## $ b18: num  224 411 428 428 693 495 514 485 473 371 ...
## $ b19: num  47 61 59 59 71 59 58 54 61 52 ...
## $ y : num  1 1 1 1 1 1 1 1 1 1 ...
```

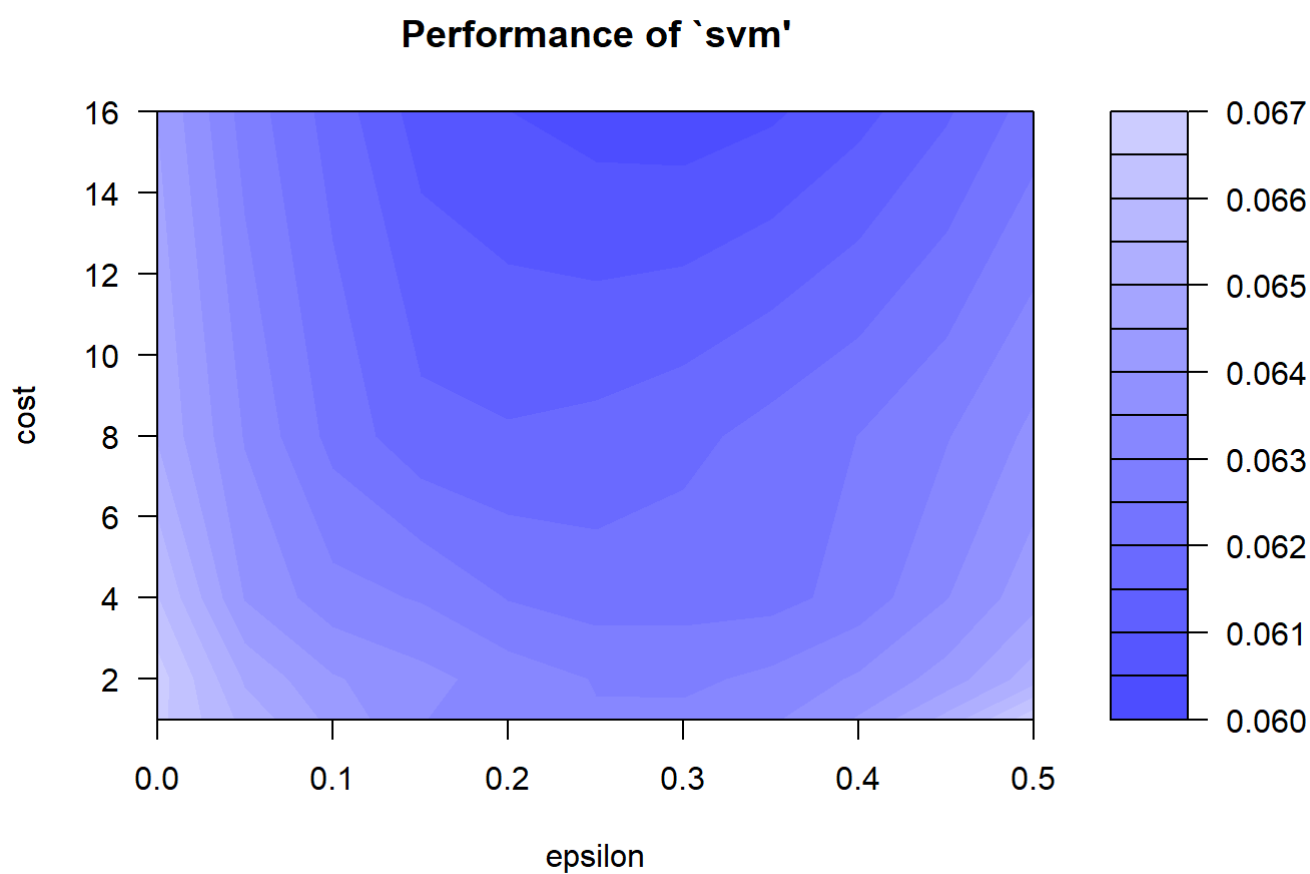
```
# inferir el modelo SVM (un solo modelo, si búsqueda de parámetros óptimos)
#modelo_svm <- svm(y ~ ., data = specie_data, cross = 5)

# buscar Los parámetros óptimos del modelo SVM
tuneResult <- tune(svm, y ~ ., data = specie_data, cross = 5,
                  ranges = list(epsilon = seq(0,0.5,0.05), cost = 2^(0:4)))
print(tuneResult)
```



```
##  
## Parameter tuning of 'svm':  
##  
## - sampling method: 10-fold cross validation  
##  
## - best parameters:  
##   epsilon cost  
##     0.3   16  
##  
## - best performance: 0.06023184
```

```
plot(tuneResult)
```



```
modelo_svm <- tuneResult$best.model
```

Revisar modelo

```
str(modelo_svm)
```

```
## List of 34
## $ call          : language best.tune(method = svm, train.x = y ~ ., data = specie_data,
ranges = list(epsilon = seq(0,      0.5, 0.05), cost| __truncated__
## $ type          : num 3
## $ kernel        : num 2
## $ cost          : num 16
## $ degree        : num 3
## $ gamma         : num 0.0526
## $ coef0         : num 0
## $ nu            : num 0.5
## $ epsilon       : num 0.3
## $ sparse        : logi FALSE
## $ scaled        : logi [1:19] TRUE TRUE TRUE TRUE TRUE TRUE ...
## $ x.scale       :List of 2
## ..$ scaled:center: Named num [1:19] 20.5 14.4 70.5 225.1 30.4 ...
## .. ..- attr(*, "names")= chr [1:19] "b1" "b2" "b3" "b4" ...
## ..$ scaled:scale : Named num [1:19] 3.54 2.31 3.59 49.44 3.68 ...
## .. ..- attr(*, "names")= chr [1:19] "b1" "b2" "b3" "b4" ...
## $ y.scale       :List of 2
## ..$ scaled:center: num 0.118
## ..$ scaled:scale : num 0.323
## $ nclasses      : int 2
## $ levels        : num [1:2] 2.731 -0.366
## $ tot.nSV       : int 288
## $ nSV           : int [1:2] 0 0
## $ labels        : int [1:2] 0 0
## $ SV            : num [1:288, 1:19] -1.643 -0.946 -1.693 -1.693 -0.258 ...
## ..- attr(*, "dimnames")=List of 2
## .. ..$ : chr [1:288] "1" "2" "3" "4" ...
## .. ..$ : chr [1:19] "b1" "b2" "b3" "b4" ...
## $ index         : int [1:288] 1 2 3 4 7 8 9 10 11 12 ...
## $ rho           : num -0.471
## $ compprob      : logi FALSE
## $ probA         : NULL
## $ probB         : NULL
## $ sigma         : NULL
## $ coefs         : num [1:288, 1] 16 16 16 16 16 ...
## $ na.action     : NULL
## $ MSE           : num [1:5] 0.0747 0.0481 0.0569 0.0482 0.0763
## $ tot.MSE       : num [1, 1] 0.0608
## $ scorrcoeff    : num 0.428
## $ fitted        : Named num [1:1134] 0.872 0.772 0.771 0.771 0.984 ...
## ..- attr(*, "names")= chr [1:1134] "1" "2" "3" "4" ...
## $ decision.values: num [1:1134, 1] 2.33 2.02 2.02 2.02 2.68 ...
## ..- attr(*, "dimnames")=List of 2
## .. ..$ : chr [1:1134] "1" "2" "3" "4" ...
## .. ..$ : chr "/"
## $ residuals     : Named num [1:1134] 0.1278 0.2284 0.2291 0.2291 0.0156 ...
## ..- attr(*, "names")= chr [1:1134] "1" "2" "3" "4" ...
## $ terms         :Classes 'terms', 'formula' language y ~ b1 + b2 + b3 + b4 + b5 + b6 +
b7 + b8 + b9 + b10 + b11 + b12 +      b13 + b14 + b15 + b16 + b17 + b18 + b19
## .. ..- attr(*, "variables")= language list(y, b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b
11, b12, b13, b14,      b15, b16, b17, b18, b19)
## .. ..- attr(*, "factors")= int [1:20, 1:19] 0 1 0 0 0 0 0 0 0 0 ...
## .. .. ..- attr(*, "dimnames")=List of 2
## .. .. ..$ : chr [1:20] "y" "b1" "b2" "b3" ...
## .. .. ..$ : chr [1:19] "b1" "b2" "b3" "b4" ...
```

```
## .. ... attr(*, "term.labels")= chr [1:19] "b1" "b2" "b3" "b4" ...
## .. ... attr(*, "order")= int [1:19] 1 1 1 1 1 1 1 1 1 1 ...
## .. ... attr(*, "intercept")= num 0
## .. ... attr(*, "response")= int 1
## .. ... attr(*, ".Environment")=<environment: R_GlobalEnv>
## .. ... attr(*, "predvars")= language list(y, b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b1
1, b12, b13, b14,      b15, b16, b17, b18, b19)
## .. ... attr(*, "dataClasses")= Named chr [1:20] "numeric" "numeric" "numeric" "numeric"
...
## .. ... attr(*, "names")= chr [1:20] "y" "b1" "b2" "b3" ...
## - attr(*, "class")= chr [1:2] "svm.formula" "svm"
```

```
summary(modelo_svm)
```

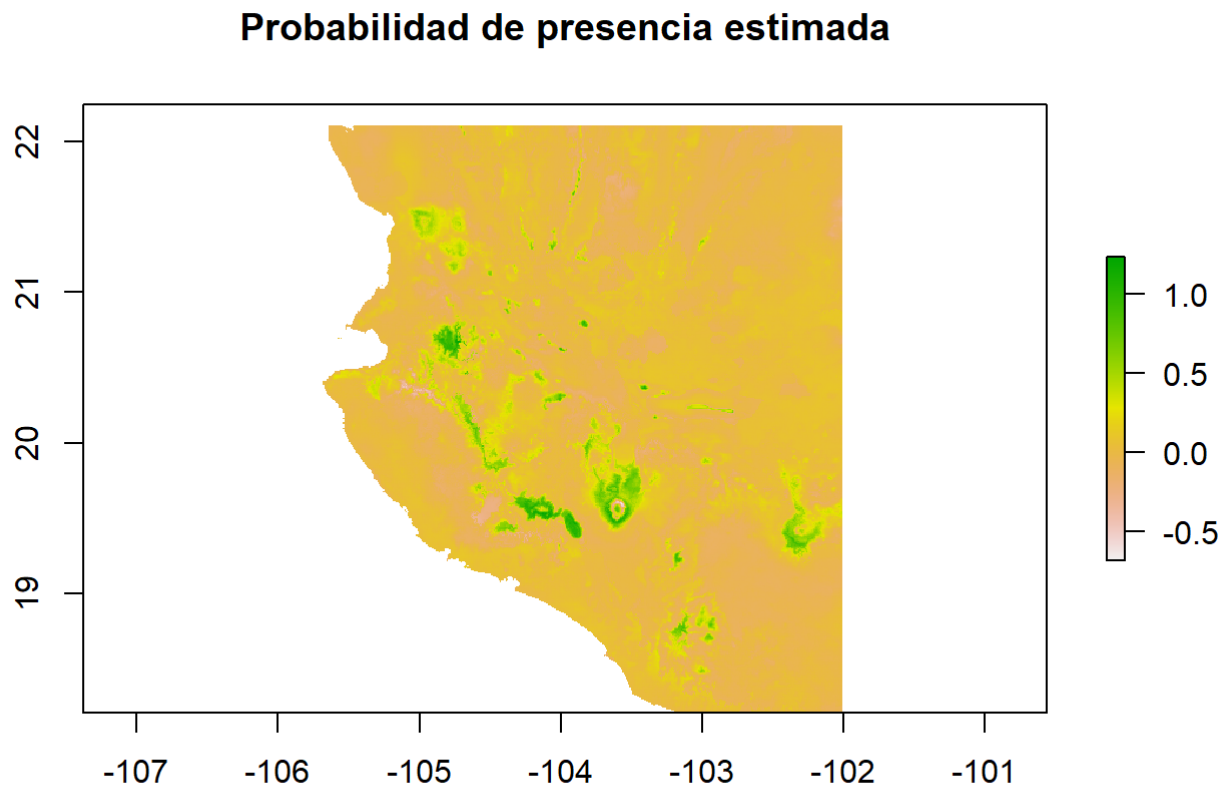
```
##
## Call:
## best.tune(method = svm, train.x = y ~ ., data = specie_data,
##   ranges = list(epsilon = seq(0, 0.5, 0.05), cost = 2^(0:4)),
##   cross = 5)
##
##
## Parameters:
##   SVM-Type:  eps-regression
##   SVM-Kernel: radial
##     cost:   16
##   gamma:    0.05263158
##   epsilon:  0.3
##
##
## Number of Support Vectors:  288
##
##
##
## 5-fold cross-validation on training data:
##
## Total Mean Squared Error: 0.06080961
## Squared Correlation Coefficient: 0.4279483
## Mean Squared Errors:
##  0.0746695 0.04812576 0.0568505 0.04818688 0.07627648
```

Predicción

```
prediccion <- predict(bio_vars, modelo_svm)
prediccion
```

```
## class      : RasterLayer
## dimensions : 484, 469, 226996  (nrow, ncol, ncell)
## resolution : 0.008333333, 0.008333333  (x, y)
## extent     : -105.9248, -102.0165, 18.21198, 22.24531  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
## source     : memory
## names      : layer
## values     : -0.6859468, 1.23517  (min, max)
```

```
plot(prediccion, main = "Probabilidad de presencia estimada")
```

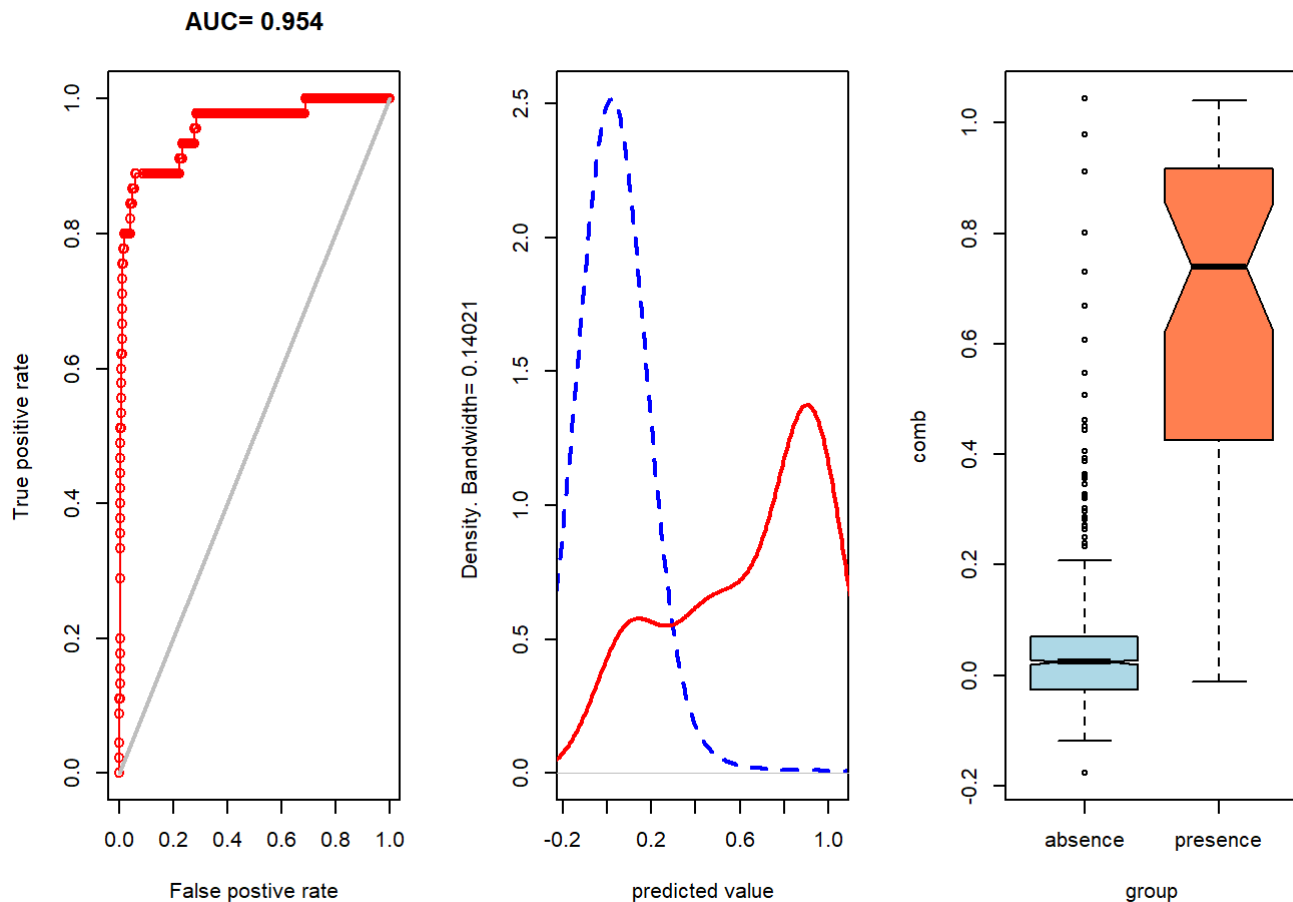


Evaluación

```
evaluacion <- evaluate(p = ocurrencias_especie_control[,2:3],
  a = puntos_aleatorios_fondo,
  x = bio_vars,
  model = modelo_svm)
evaluacion
```

```
## class      : ModelEvaluation
## n presences : 45
## n absences  : 1000
## AUC         : 0.9537556
## cor         : 0.7017058
## max TPR+TNR at : 0.1261488
```

```
par(mfrow=c(1, 3))
plot(evaluacion, 'ROC')
density(evaluacion)
boxplot(evaluacion, col=c('lightblue','coral'), notch=TRUE)
```



```
umbrales <- threshold(evaluacion)
umbrales
```

```
##           kappa spec_sens no_omission prevalence equal_sens_spec
## thresholds 0.4253354 0.1261488 -0.01060254 0.04322124      0.09689703
##           sensitivity
## thresholds 0.07702737
```

```
umbrales$spec_sens
```

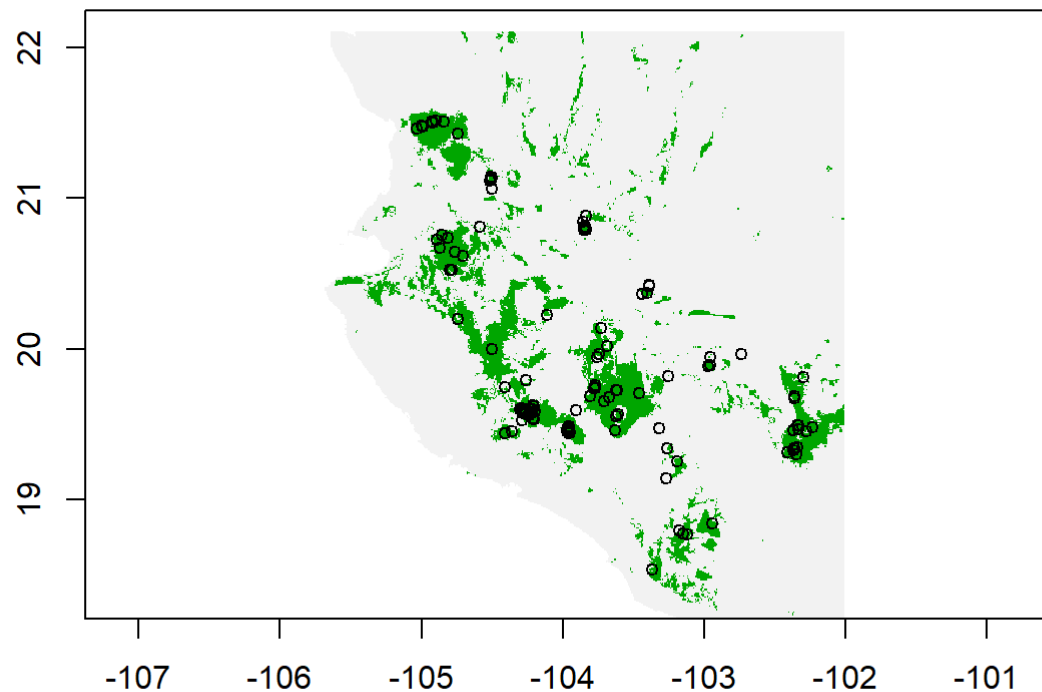
```
## [1] 0.1261488
```

Reclasificación de acuerdo con el umbral de igualdad entre la especificidad y la sensibilidad

```
especie_reclass_table <- matrix(c(-Inf, umbrales$spec_sens, 0,
                                umbrales$spec_sens, Inf, 1),
                                ncol=3, byrow=TRUE)
prediccion_presencia <- reclassify(prediccion, especie_reclass_table)

plot(prediccion_presencia, main = "Probable presencia", legend = FALSE)
points(ocurrencias_especie[,2], ocurrencias_especie[,3],
       col = "black", cex = 0.8)
```

Probable presencia



Guardar los resultados en formato raster

```
writeRaster(prediccion, filename = "datos/probabilidad_svm_32bits.tif",  
            format="GTiff", datatype = "FLT4S", overwrite = TRUE)  
  
writeRaster(prediccion_presencia, filename = "datos/presencia_svm_16bits.tif",  
            format="GTiff", datatype = "INT2S", overwrite = TRUE)
```