

TREE ENSEMBLES_C2

QUETZALCOATL SINUE & V. Shalisko

29 de noviembre de 2017

```
data <- read.csv("tabla_fusionada_1.csv")
#data <- read.csv("tabLa_fusionada_1.csv")
```

```
dim(data)
```

```
## [1] 1272 27
```

```
data <- data[!is.na(data[,5]),]
dim(data)
```

```
## [1] 1167 27
```

```
library(dismo)
# good random seed 32
set.seed(32)
grupos <- kfold(data, k = 5)
```

```
data <- cbind(data,grupos)
head(data)
```

```
## X.1 X porc_veg rango fech_siem porc_veg_f b2n b3n b4n
## 1 1 1 0 3-12 2013 0 33.97 186.41 589.43
## 2 2 2 0 3-12 2011 0 7.68 14.78 197.04
## 3 3 3 20 3-12 2010 20 -24.12 -47.53 34.67
## 4 4 4 20 3-12 2010 20 33.93 10.98 166.91
## 5 5 5 0 3-12 2010 0 -181.33 -295.75 -559.35
## 6 6 6 0 3-12 2013 0 6.33 138.98 620.94
## b5n b6n b7n rango_f fech_siem_f rango_n b2n_c1
## 1 -813.65 816.1 836.24 3-12 2013 2 16.27941
## 2 -828.05 -1532.6 -444.77 3-12 2011 2 -10.01059
## 3 -544.45 39.7 50.24 3-12 2010 2 -41.81059
## 4 -961.85 -235.8 -4.00 3-12 2010 2 16.23941
## 5 1598.85 -813.2 -1120.21 3-12 2010 2 -199.02059
## 6 -1327.35 552.1 929.84 3-12 2013 2 -11.36059
## b2n_c2 b3n_c1 b3n_c2 b4n_c1 b4n_c2 b5n_c1
## 1 -6.745595 143.5601 121.66279 539.09304 493.91435 -975.7608
## 2 -33.035595 -28.0699 -49.96721 146.70304 101.52435 -990.1608
## 3 -60.340414 -90.3799 -113.99376 -15.66696 -40.36488 -706.5608
## 4 -2.290414 -31.8699 -55.48376 116.57304 91.87512 -1123.9608
## 5 -222.045595 -338.5999 -360.49721 -609.68696 -654.86565 1436.7392
## 6 -34.385595 96.1301 74.23279 570.60304 525.42435 -1489.4608
## b5n_c2 b6n_c1 b6n_c2 b7n_c1 b7n_c2 grupos
## 1 -1141.7801 620.5008 675.19090 888.70441 797.4770 3
## 2 -1156.1801 -1728.1992 -1673.50910 -392.30559 -483.5330 3
## 3 -882.1715 -155.8992 69.09977 102.70441 204.2085 4
## 4 -1299.5715 -431.3992 -206.40023 48.46441 149.9685 1
## 5 1270.7199 -1008.7992 -954.10910 -1067.74559 -1158.9730 2
## 6 -1655.4801 356.5008 411.19090 982.30441 891.0770 1
```

```
data_full <- data[data$grupos > 0,]
data_full_variables <- data_full[,c(18,20,22,24,26,28)]
data_full_clase <- as.factor(data_full[,5])
```

```
data_entrenamiento <- data[data$grupos > 1,]
dim(data_entrenamiento)
```

```
## [1] 934 28
```

```
data_entrenamiento_variables <- data_entrenamiento[,c(18,20,22,24,26,28)]
data_entrenamiento_clase <- as.factor(data_entrenamiento[,5])
```

```
summary(data_entrenamiento_variables)
```

```
## b3n_c1 b4n_c1 b5n_c1
## Min. : -626.47 Min. : -997.786 Min. : -3296.72
## 1st Qu.: -137.75 1st Qu.: -318.752 1st Qu.: -627.92
## Median : -10.01 Median : -55.306 Median : 156.71
## Mean : 11.34 Mean : 2.487 Mean : 91.17
## 3rd Qu.: 139.93 3rd Qu.: 292.526 3rd Qu.: 846.40
## Max. : 1562.72 Max. : 2808.274 Max. : 3464.23
## b6n_c1 b7n_c1 grupos
## Min. : -3654.26 Min. : -2407.8 Min. : 2.000
## 1st Qu.: -1086.67 1st Qu.: -761.4 1st Qu.: 2.250
## Median : -182.89 Median : -100.7 Median : 3.500
## Mean : -34.42 Mean : -24.2 Mean : 3.499
## 3rd Qu.: 863.05 3rd Qu.: 614.4 3rd Qu.: 4.000
## Max. : 6081.21 Max. : 4034.4 Max. : 5.000
```

```
summary(data_entrenamiento_clase)
```

```
## 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
## 35 16 13 79 100 204 171 147 137 32
```

```
levels(data_entrenamiento_clase)
```

```
## [1] "2006" "2007" "2008" "2009" "2010" "2011" "2012" "2013" "2014" "2015"
```

```
data_prueba <- data[data$grupos == "1",]
dim(data_prueba)
```

```
## [1] 233 28
```

```
data_prueba_variables <- data_prueba[,c(18,20,22,24,26,28)]
data_prueba_clase <- as.factor(data_prueba[,5])
```

```
str(data_prueba_variables)
```

```
## 'data.frame': 233 obs. of 6 variables:
## $ b3n_c1: num -31.9 96.1 157.4 405.3 163.1 ...
## $ b4n_c1: num 117 571 512 800 161 ...
## $ b5n_c1: num -1124 -1489.5 -1288.3 -40.9 980.9 ...
## $ b6n_c1: num -431 357 504 1121 775 ...
## $ b7n_c1: num 48.5 982.3 938.8 1488.5 336.1 ...
## $ grupos: int 1 1 1 1 1 1 1 1 1 1 ...
```

```
head(data_prueba_variables)
```

```
##      b3n_c1 b4n_c1 b5n_c1 b6n_c1 b7n_c1 grupos
## 4 -31.869902 116.5730 -1123.96078 -431.39922 48.46441 1
## 6 96.130098 570.6030 -1489.46078 356.50078 982.30441 1
## 7 157.440098 512.4130 -1288.26078 504.40078 938.80441 1
## 12 405.280098 800.0530 -40.86078 1120.50078 1488.50441 1
## 16 163.132619 161.1390 980.90476 775.05143 336.12714 1
## 24 5.707063 291.9239 -867.51971 88.80802 268.58178 1
```

```
head(data_entrenamiento_clase)
```

```
## [1] 2013 2011 2010 2010 2012 2012
## Levels: 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
```

```
#train <- rbind(iris3[1:25,,1], iris3[1:25,,2], iris3[1:25,,3])
#test <- rbind(iris3[26:50,,1], iris3[26:50,,2], iris3[26:50,,3])
#summary(train)
#summary(test)
#cl <- factor(c(rep("s",25), rep("c",25), rep("v",25)))
#ct <- factor(c(rep("s",25), rep("c",25), rep("v",25)))
```

Modelo 1 (con base en primera corrección)

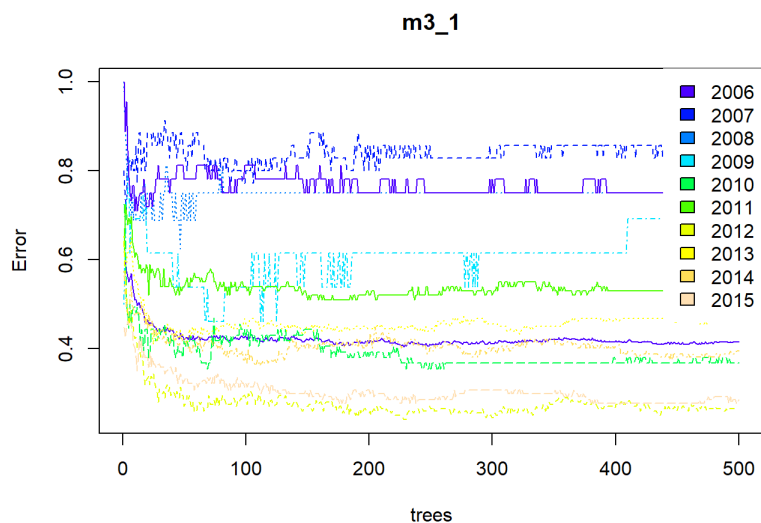
Para construir el modelo para posterior trabajo con raster se requiere utilizar únicamente resultados de primera corrección

```
library(randomForest)
data_entrenamiento$fech_siem_factor <- as.factor(data_entrenamiento$fech_siem)

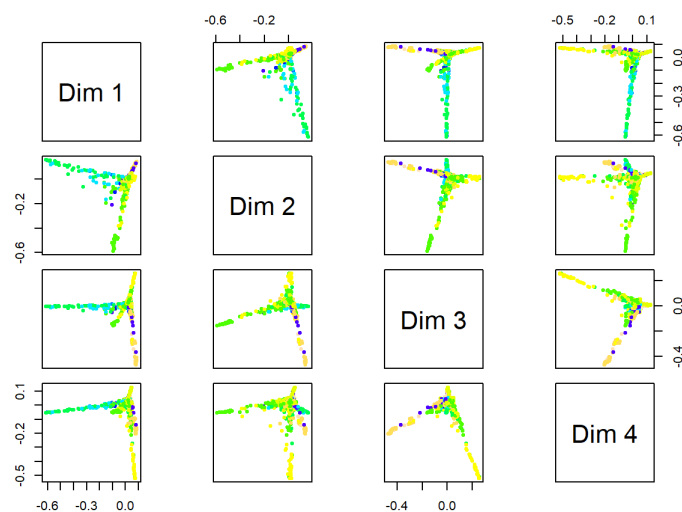
m3_1 <- randomForest(fech_siem_factor ~ b2n_c1 + b3n_c1 + b4n_c1 + b5n_c1 + b6n_c1 + b7n_c1, data=data_entrenamiento,proxim
ity=TRUE)
m3_1
```

```
##
## Call:
## randomForest(formula = fech_siem_factor ~ b2n_c1 + b3n_c1 + b4n_c1 + b5n_c1 + b6n_c1 + b7n_c1, data = data_entrenam
iento, proximity = TRUE)
##              Type of random forest: classification
##              Number of trees: 500
## No. of variables tried at each split: 2
##
## OOB estimate of error rate: 41.54%
## Confusion matrix:
##      2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 class.error
## 2006 6 1 0 1 1 4 13 1 7 1 0.8285714
## 2007 2 4 1 0 2 2 2 1 2 0 0.7500000
## 2008 0 0 4 1 6 0 1 0 1 0 0.6923077
## 2009 0 0 0 50 7 6 13 2 1 0 0.3670886
## 2010 1 0 1 8 46 15 6 6 17 0 0.5400000
## 2011 1 0 1 1 3 150 27 13 8 0 0.2647059
## 2012 1 1 0 13 3 32 91 10 6 14 0.4678363
## 2013 0 0 0 8 1 25 14 89 10 0 0.3945578
## 2014 3 0 0 2 17 3 6 8 98 0 0.2846715
## 2015 0 0 1 0 0 1 14 2 6 8 0.7500000
```

```
plot(m3_1, col = topo.colors(10))
legend('topright',levels(data_entrenamiento_clase),box.lty=0, fill=topo.colors(10))
```



```
MDSplot(m3_1, data_entrenamiento_clase, k=4, palette = topo.colors(10))
```



```
#text(m3_1)
```

Precisión con datos completos (modelo con base en primera corrección)

```
resultado_full_1 <- predict(m3_1, data_full, type= "class")
```

```
#rpart.plot(m3_1)
#rpart.plot(m3_1, type=5, extra=234, fallen.leaves=T)
error_matrix_full_1 <- table(resultado_full_1, data_full_clase)
error_matrix_full_1
```

```
##          data_full_clase
## resultado_full_1 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
##          2006    38     0     0     0     0     0     0     0     0     0
##          2007     0    17     0     0     0     0     0     0     0     0
##          2008     0     0    20     0     0     0     1     0     1     0
##          2009     1     1     0    87     1     1     1     0     1     0
##          2010     1     0     1     0   107     1     2     0    11     0
##          2011     2     0     0     6     3   246    12     4     0     0
##          2012     0     3     0     2     0     6   188     6     4    12
##          2013     0     0     0     0     2     1     3   168     2     0
##          2014     0     0     0     1    10     0     1     4   155     2
##          2015     0     0     0     0     0     0     5     1     0    25
```

```
Kerror_matrix_full_1 <- Kappa(error_matrix_full_1)
Kerror_matrix_full_1
```

```
##          value    ASE    z Pr(>|z|)
## Unweighted 0.8828 0.01031 85.60      0
## Weighted   0.8897 0.01131 78.65      0
```

```
#summary(Kerror_matrix_full_1)
confint(Kerror_matrix_full_1, level = 0.95)
```

```
##
## Kappa          lwr          upr
## Unweighted 0.8626036 0.9030286
## Weighted    0.8674885 0.9118285
```

Precisión solo con datos de control (modelo con base en primera corrección)

```
resultado_prediccion_1 <- predict(m3_1, data_prueba , type= "class")
```

```
#rpart.plot(m3_1)
#rpart.plot(m3_1, type=5, extra=234, fallen.leaves=T)
error_matrix_1 <- table(resultado_prediccion_1, data_prueba_clase)
error_matrix_1
```

```
##
##          data_prueba_clase
## resultado_prediccion_1 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
##          2006      3    0    0    0    0    0    0    0    0    0
##          2007      0    1    0    0    0    0    0    0    0    0
##          2008      0    0    7    0    0    0    1    0    1    0
##          2009      1    1    0    8    1    1    1    0    1    0
##          2010      1    0    1    0    13   1    2    0    3    0
##          2011      2    0    0    6    3    42   12    4    0    0
##          2012      0    3    0    2    0    6    22    6    4    4
##          2013      0    0    0    0    2    1    3    21    2    0
##          2014      0    0    0    1    4    0    1    4    26    2
##          2015      0    0    0    0    0    0    0    1    0    1
```

```
Kerror_matrix_1 <- Kappa(error_matrix_1)
Kerror_matrix_1
```

```
##          value      ASE      z Pr(>|z|)
## Unweighted 0.5444 0.03764 14.46 2.071e-47
## Weighted    0.6108 0.04113 14.85 6.692e-50
```

```
confint(Kerror_matrix_1, level = 0.95)
```

```
##
## Kappa          lwr          upr
## Unweighted 0.4706018 0.6181421
## Weighted    0.5302283 0.6914410
```

Modelo 2 (con base en segunda corrección)

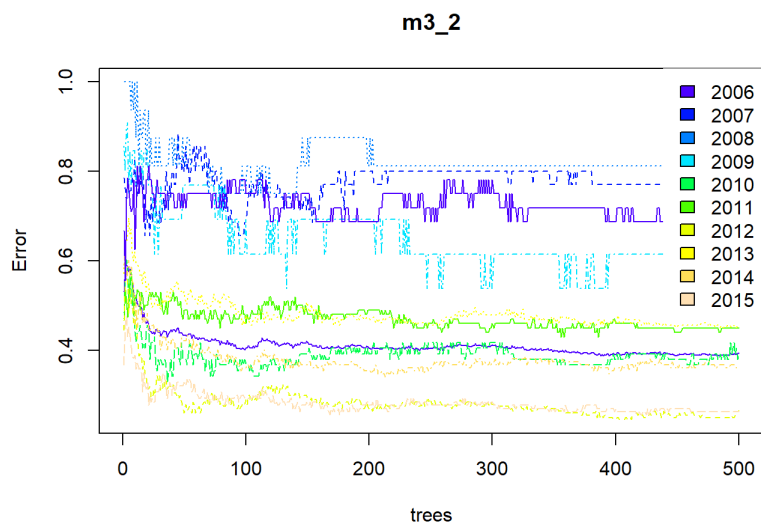
Para construir el modelo para posterior trabajo con raster se requiere utilizar únicamente resultados de primera corrección

```
data_entrenamiento$fech_siem_factor <- as.factor(data_entrenamiento$fech_siem)

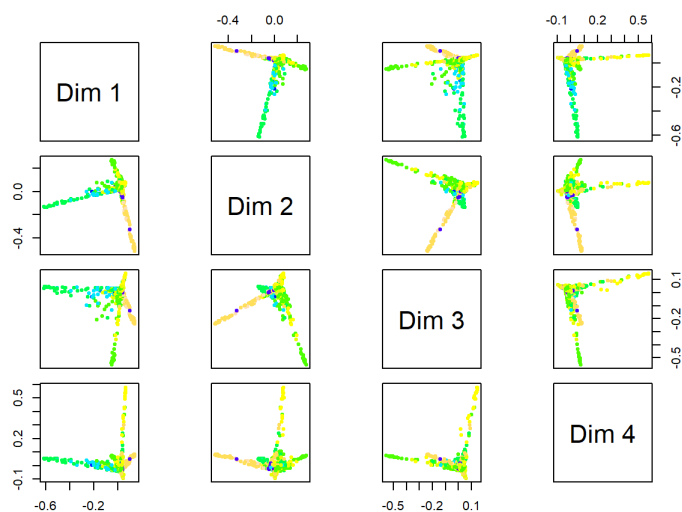
m3_2 <- randomForest(fech_siem_factor ~ b2n_c2 + b3n_c2 + b4n_c2 + b5n_c2 + b6n_c2 + b7n_c2, data=data_entrenamiento,proxim
ity=TRUE)
m3_2
```

```
##
## Call:
## randomForest(formula = fech_siem_factor ~ b2n_c2 + b3n_c2 + b4n_c2 +      b5n_c2 + b6n_c2 + b7n_c2, data = data_entrenam
## iento, proximity = TRUE)
##          Type of random forest: classification
##          Number of trees: 500
## No. of variables tried at each split: 2
##
##          OOB estimate of  error rate: 39.4%
## Confusion matrix:
##          2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 class.error
## 2006      8    2    0    2    1    4    9    3    6    0 0.7714286
## 2007      2    3    1    1    3    2    4    0    0    0 0.8125000
## 2008      0    0    5    0    0    7    1    0    0    0 0.6153846
## 2009      2    2    0    48    7    3    11    5    1    0 0.3924051
## 2010      0    0    0    8    55   19    5    8    5    0 0.4500000
## 2011      0    0    0    1    5   150   22   21    5    0 0.2647059
## 2012      6    0    0    8    3   32   93   13    9    7 0.4561404
## 2013      1    0    0    7    1   19   13   93   12    1 0.3673469
## 2014      1    0    0    3    4    7   11   10   101    0 0.2627737
## 2015      0    0    0    0    0    2   14    2    4   10 0.6875000
```

```
plot(m3_2, col = topo.colors(10))
legend('topright', levels(data_entrenamiento_clase),box.lty=0, fill=topo.colors(10))
```



```
MDSplot(m3_2,data_entrenamiento_clase, k=4, palette = topo.colors(10))
```



```
#text(m3_1)
```

Precisión con datos completos (modelo con base en segunda corrección)

```
resultado_full_2 <- predict(m3_2, data_full , type= "class")
```

```
#rpart.plot(m3_2)
#rpart.plot(m3_2, type=5, extra=234, fallen.leaves=T)
error_matrix_full_2 <- table(resultado_full_2, data_full_clase)
error_matrix_full_2
```

```
##          data_full_clase
## resultado_full_2 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
##          2006    39     0     0     0     0     0     0     1     0     0
##          2007     0    16     0     0     0     0     0     0     0     0
##          2008     0     0    20     0     0     0     0     0     0     0
##          2009     1     1     0    88     3     0     1     2     0     0
##          2010     0     0     0     1   114     0     3     1     2     1
##          2011     2     1     1     6     3   248     8     3     0     0
##          2012     0     3     0     0     2     7   195     7     1     3
##          2013     0     0     0     1     1     0     4   168     3     1
##          2014     0     0     0     0     0     0     1     1   168     1
##          2015     0     0     0     0     0     0     1     0     0    33
```

```
Kerror_matrix_full_2 <- Kappa(error_matrix_full_2)
Kerror_matrix_full_2
```

```
##          value      ASE      z Pr(>|z|)
## Unweighted 0.9212 0.008616 106.9      0
## Weighted   0.9377 0.008315 112.8      0
```

```
#summary(Kerror_matrix_full_2)
confint(Kerror_matrix_full_2, level = 0.95)
```

```
##
## Kappa          lwr          upr
## Unweighted 0.9043560 0.9381300
## Weighted   0.9213985 0.9539913
```

Precisión solo con datos de control (modelo con base en segunda corrección)

```
resultado_prediccion_2 <- predict(m3_2, data_prueba , type= "class")
```

```
#rpart.plot(m3_2)
#rpart.plot(m3_2, type=5, extra=234, fallen.leaves=T)
error_matrix_2 <- table(resultado_prediccion_2, data_prueba_clase)
error_matrix_2
```

```
##              data_prueba_clase
## resultado_prediccion_2 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
##              2006      4    0    0    0    0    0    0    1    0    0
##              2007      0    0    0    0    0    0    0    0    0    0
##              2008      0    0    7    0    0    0    0    0    0    0
##              2009      1    1    0    9    3    0    1    2    0    0
##              2010      0    0    0    1   14    0    3    1    2    1
##              2011      2    1    1    6    3   44    8    3    0    0
##              2012      0    3    0    0    2    7   24    7    1    3
##              2013      0    0    0    1    1    0    4   21    3    1
##              2014      0    0    0    0    0    0    1    1   31    1
##              2015      0    0    0    0    0    0    1    0    0    1
```

```
Kerror_matrix_2 <- Kappa(error_matrix_2)
Kerror_matrix_2
```

```
##          value      ASE      z Pr(>|z|)
## Unweighted 0.6012 0.03646 16.49 4.298e-61
## Weighted   0.6711 0.03859 17.39 1.000e-67
```

```
confint(Kerror_matrix_2, level = 0.95)
```

```
##
## Kappa          lwr          upr
## Unweighted 0.5297545 0.6726680
## Weighted   0.5954522 0.7467343
```

Cargar capas raster (con primera corrección aplicada)

```
raster_2013_corregido <- stack("IMAGENES_2013/b2n_c1_r.tif",
                              "IMAGENES_2013/b3n_c1_r.tif",
                              "IMAGENES_2013/b4n_c1_r.tif",
                              "IMAGENES_2013/b5n_c1_r.tif",
                              "IMAGENES_2013/b6n_c1_r.tif",
                              "IMAGENES_2013/b7n_c1_r.tif")
names(raster_2013_corregido) <- c("b2n_c1", "b3n_c1", "b4n_c1", "b5n_c1", "b6n_c1", "b7n_c1")

raster_2013_corregido
```

```
## class      : RasterStack
## dimensions  : 684, 1389, 950076, 6 (nrow, ncol, ncell, nlayers)
## resolution  : 30, 30 (x, y)
## extent      : 612160.6, 653830.6, 2295131, 2315651 (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=utm +zone=13 +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0
## names       : b2n_c1, b3n_c1, b4n_c1, b5n_c1, b6n_c1, b7n_c1
## min values  : -1473.917, -2157.007, -2733.408, -8057.168, -6856.770, -5282.532
## max values  : 25902.49, 26999.84, 29612.49, 31957.48, 27250.04, 23644.11
```

```
raster_2014_corregido <- stack("IMAGENES_2014/b2n_c1_r.tif",
                              "IMAGENES_2014/b3n_c1_r.tif",
                              "IMAGENES_2014/b4n_c1_r.tif",
                              "IMAGENES_2014/b5n_c1_r.tif",
                              "IMAGENES_2014/b6n_c1_r.tif",
                              "IMAGENES_2014/b7n_c1_r.tif")
names(raster_2014_corregido) <- c("b2n_c1", "b3n_c1", "b4n_c1", "b5n_c1", "b6n_c1", "b7n_c1")

raster_2014_corregido
```

```
## class      : RasterStack
## dimensions  : 684, 1389, 950076, 6 (nrow, ncol, ncell, nlayers)
## resolution  : 30, 30 (x, y)
## extent      : 612160.6, 653830.6, 2295131, 2315651 (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=utm +zone=13 +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0
## names       : b2n_c1, b3n_c1, b4n_c1, b5n_c1, b6n_c1, b7n_c1
## min values  : -1259.752, -2273.587, -2778.002, -8482.090, -6985.296, -5360.985
## max values  : 10021.97, 10694.88, 11966.96, 18991.83, 19125.94, 53696.88
```

```
# se detecto un problema con el raster "IMAGENES_2015/b5n_c1_r.tif" - su extent no coincide con el resto - a corregir

raster_2015_corregido <- stack("IMAGENES_2015/b2n_c1_r.tif",
                              "IMAGENES_2015/b3n_c1_r.tif",
                              "IMAGENES_2015/b4n_c1_r.tif",
                              "IMAGENES_2014/b5n_c1_r.tif",
                              "IMAGENES_2015/b6n_c1_r.tif",
                              "IMAGENES_2015/b7n_c1_r.tif")

names(raster_2015_corregido) <- c("b2n_c1", "b3n_c1", "b4n_c1", "b5n_c1", "b6n_c1", "b7n_c1")

raster_2015_corregido
```

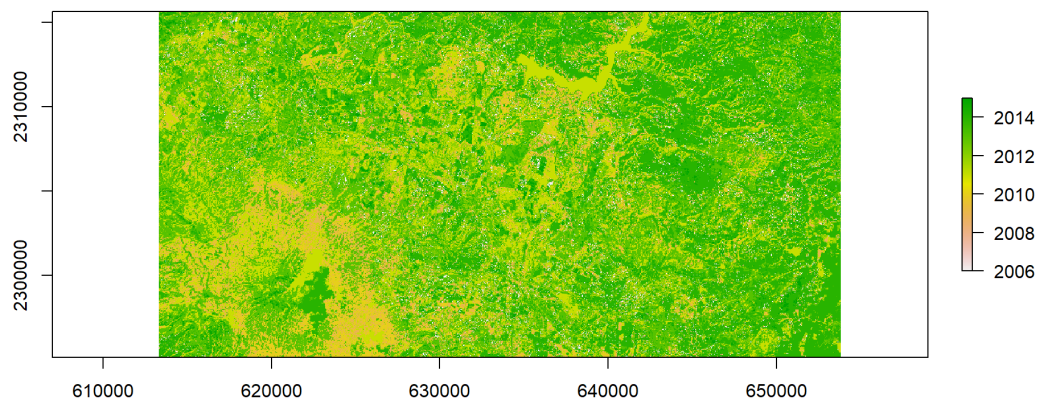
```
## class      : RasterStack
## dimensions  : 684, 1389, 950076, 6  (nrow, ncol, ncell, nlayers)
## resolution  : 30, 30  (x, y)
## extent     : 612160.6, 653830.6, 2295131, 2315651  (xmin, xmax, ymin, ymax)
## coord. ref. : +proj=utm +zone=13 +datum=WGS84 +units=m +no_defs +ellps=WGS84 +towgs84=0,0,0
## names      :  b2n_c1,   b3n_c1,   b4n_c1,   b5n_c1,   b6n_c1,   b7n_c1
## min values  : -1265.148, -2201.421, -2598.537, -8482.090, -7314.241, -5213.068
## max values  : 12027.87, 13469.16, 14746.98, 18991.83, 20667.99, 15481.62
```

Realizar predicción con el modelo 1 (en raster)

En este análisis significado solo tienen los píxeles que se encuentran en la máscara de campos de Agave, todo lo que se encuentra fuera de esta máscara es el río. Aplicación de máscara se puede realizar después de realizar la clasificación, o al momento.

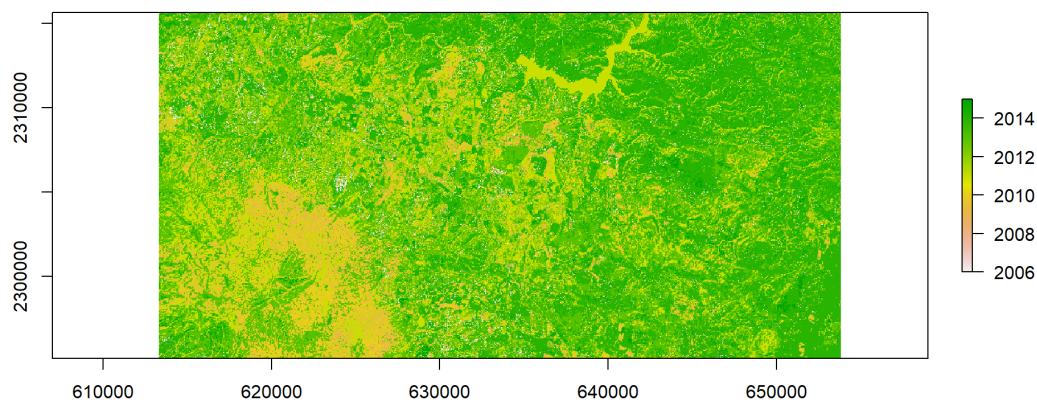
```
clasificado_raster_2013 <- predict(raster_2013_corregido, m3_1, type= "class")

plot(clasificado_raster_2013)
```



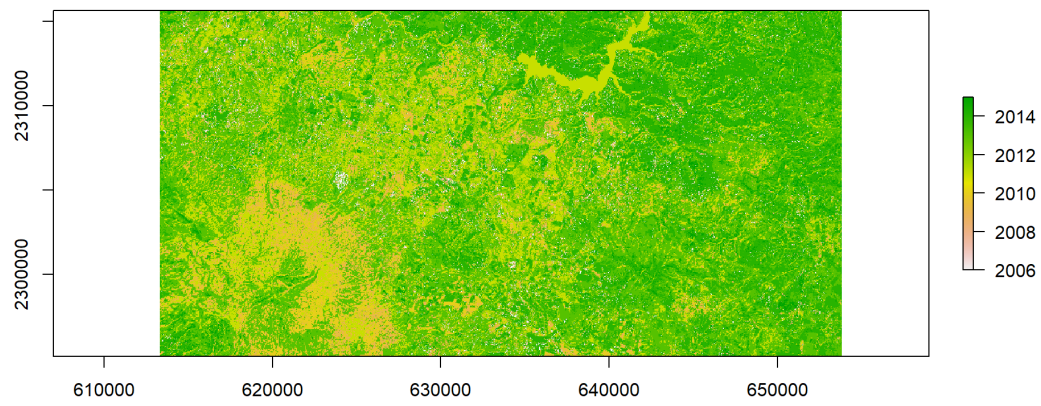
```
clasificado_raster_2014 <- predict(raster_2014_corregido, m3_1, type= "class")

plot(clasificado_raster_2014)
```



```
clasificado_raster_2015 <- predict(raster_2015_corregido, m3_1, type= "class")

plot(clasificado_raster_2015)
```



Guardar los resultados en TIFF

```
writeRaster(clasificado_raster_2013, filename="clasificado_preliminar_raster_2013.tif", format="GTiff", overwrite=TRUE)
writeRaster(clasificado_raster_2014, filename="clasificado_preliminar_raster_2014.tif", format="GTiff", overwrite=TRUE)
writeRaster(clasificado_raster_2015, filename="clasificado_preliminar_raster_2015.tif", format="GTiff", overwrite=TRUE)
```