Ayata-et-al-2018

The objective of this section is to predict SOC across an area of interest based on terrain parameters and machine learning. We use 18 soil profile descriptions and a set of environmental information to predict the spatial variability of SOC across a water limited environment of Northeast Mexico, including its associated uncertainty.

data preparation: estimating SOC stocks 0-30cm depth

```
dat <- read.csv("horizon.csv")

sites <- read.csv("site.csv")

library(aqp)
```

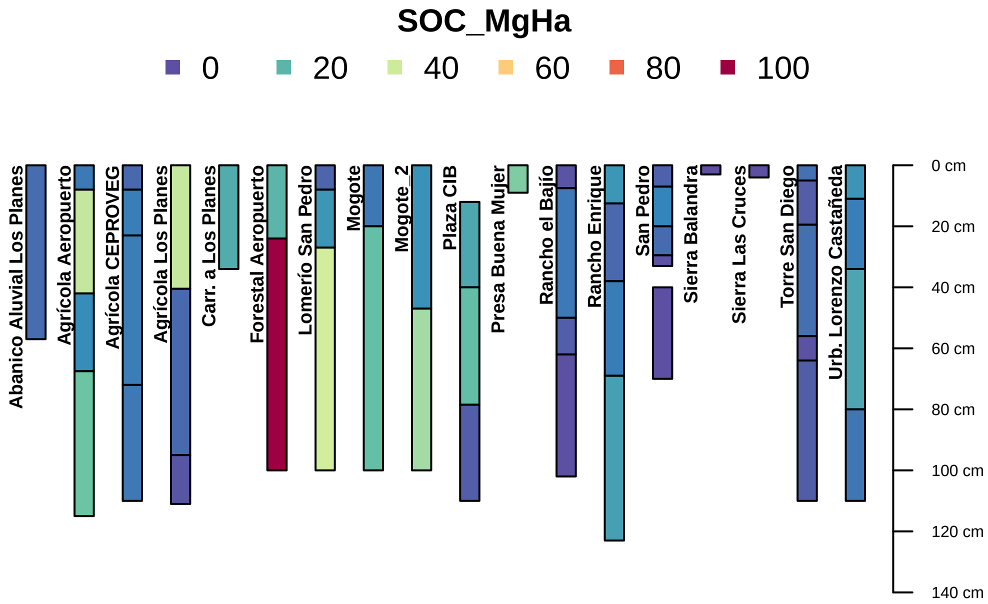
```
## This is aqp 1.16
```

```
depths(dat) <- ID ~ top + bottom
```

```
## Warning: converting IDs from factor to character
```

```
dataqp <- dat
#VISUALIZE SOC DATA
plot(dataqp, color='SOC_MgHa')
```

```
## unable to guess column containing horizon designations
```




```
site(dat) <- sites

coordinates(dat) <- ~ X + Y
library(GSIF)
```

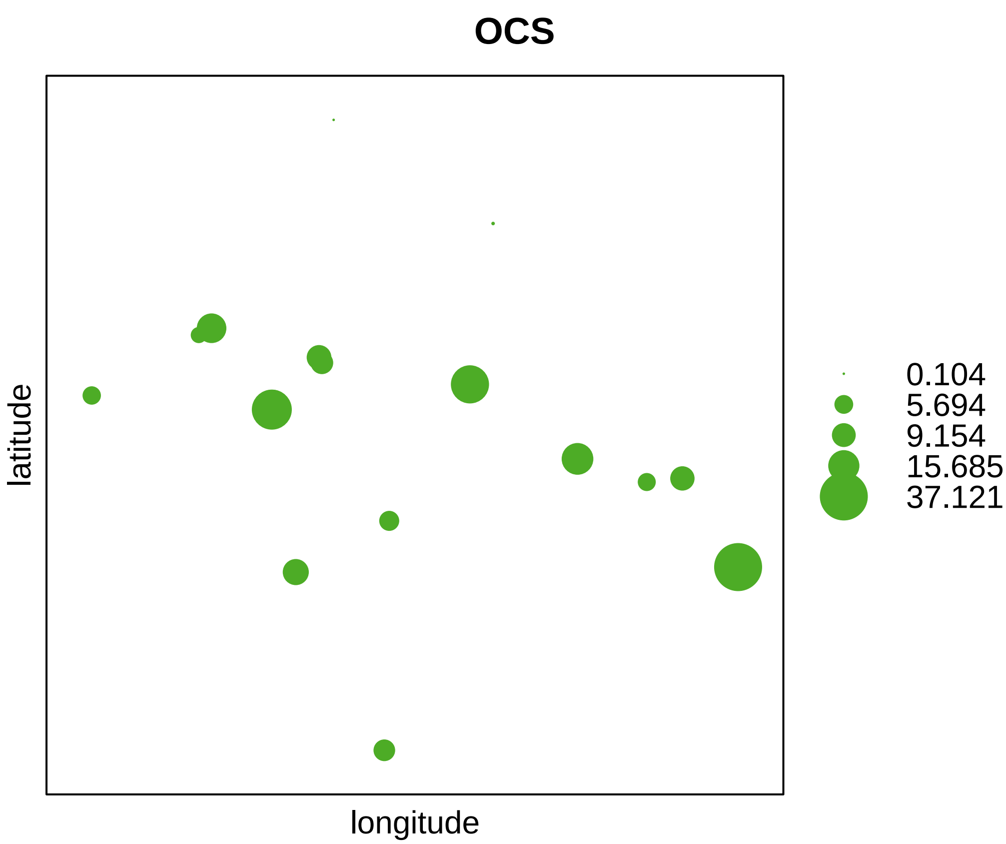
```
## GSIF version 0.5-4 (2017-04-25)
## URL: http://gsif.r-forge.r-project.org/
```

```r
try(OCS <- mpspline(dat, 'SOC_MgHa', d = t(c(0,30))))
```

```
## Fitting mass preserving splines per profile...
##
  |
  |                                                                 |   0%
## Spline not fitted to profile: Abanico Aluvial Los Planes
##
  |
  |====                                                             |   6%
  |
  |=======                                                          |  11%
  |
  |===========                                                      |  17%
  |
  |==============                                                   |  22%
## Spline not fitted to profile: Carr. a Los Planes
##
  |
  |==================                                               |  28%
  |
  |======================                                           |  33%
  |
  |=========================                                        |  39%
  |
  |=============================                                    |  44%
  |
  |================================                                 |  50%
  |
  |====================================                             |  56%
## Spline not fitted to profile: Presa Buena Mujer
##
  |
  |========================================                         |  61%
  |
  |===========================================                      |  67%
  |
  |===============================================                  |  72%
  |
  |===================================================              |  78%
## Spline not fitted to profile: Sierra Balandra
##
  |
  |======================================================           |  83%
## Spline not fitted to profile: Sierra Las Cruces
```

```
##
  |
  |==========================================================           |  89%
  |
  |=============================================================        |  94%
  |
  |=================================================================| 100%
```

```
dat <- data.frame(id = dat@site$ID,
                  Y = dat@sp@coords[,2],
                  X = dat@sp@coords[,1],
                  OCS = OCS$var.std[,1])

#write.csv(dat, "mexico_data.csv")


    datsp <- dat
    coordinates(datsp) <- ~ Y + X
  #SPATIAL LOCATIONS
    library(sp)
    bubble(datsp['OCS'], xlab='longitude', ylab='latitude')
```



prepare covariates: harmonize all available prediction factors

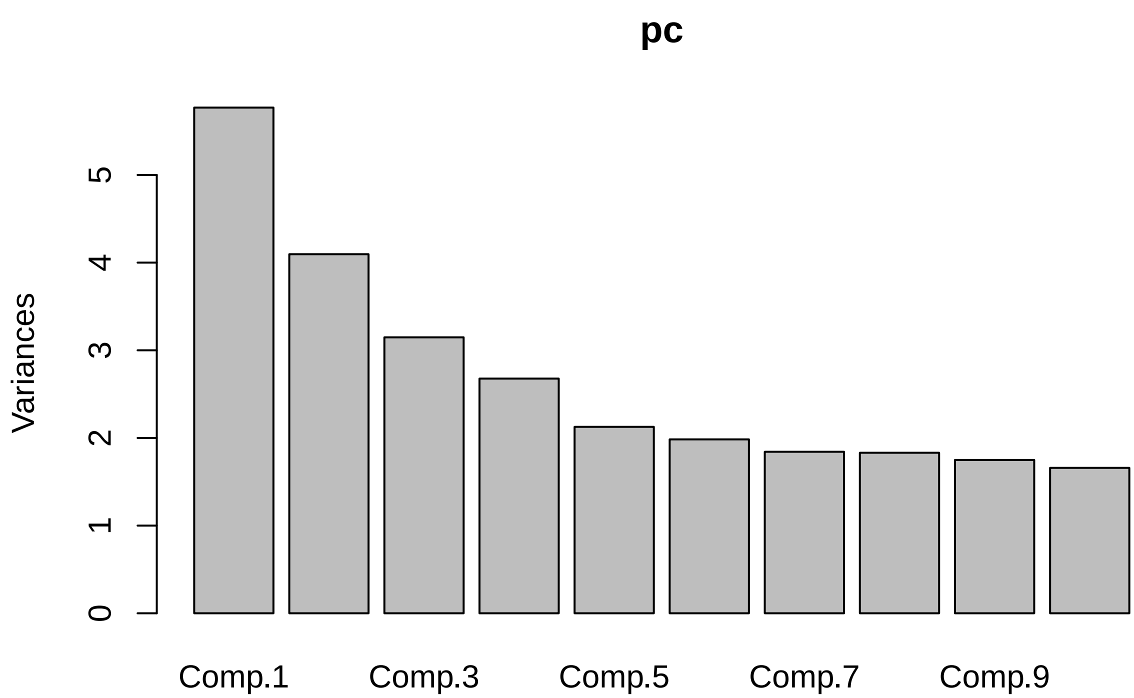
generate dummy variables for those categoriacal prediction factors

```
  dummyRaster <-function(rast){
      rast <-as.factor(rast)
      result <-list()
      for(i in 1:length(levels(rast)[[1]][[1]])){
      result[[i]] <-rast==levels(rast)[[1]][[1]][i]
      names(result[[i]]) <-paste0(names(rast),
      levels(rast)[[1]][[1]][i])}
      return(stack(result))
      }

  library (raster)

#SELECT THE COLUMN NUMBERS OF INTEREST
#lis all tif files
#lis1 the separated maps (i.e., landforms)
#lis2 continuos maps (i.e., prec)
#lis3 categorical maps (i.e., soil type)
  (lis <- list.files(pattern='tif$'))
  (lis1 <- lis[-c(5, 9, 10, 11, 12, 13, 35, 41)])
  (lis2 <- lis[c(35)])
  (lis3 <- lis[c(9, 10, 13, 41)])
  #AREA OF INTEREST
  aoi <- raster("Area de estudio.tif")
  #TOPOGRAPHIC TERRAIN PARAMETERS DERIVED ON SAGA GIS
dem <- stack('dem15/terrain/terrain.tif')
dem[is.na(dem)==TRUE]<- -9999
dem[is.infinite(dem)==TRUE]<- 9999
names(dem) <- c('dem','hillshade','curvature','convergenceIndex','flowAccumulation','wetnessIndex','l

dum <- stack()

  for (i in 1:length(lis1)){
      r <- raster (lis1[i])
      r <- projectRaster (r, aoi)
      r <- crop(r, aoi)
      r[is.na(r)==FALSE,] <- 1
      r[is.na(r)==TRUE,] <- 10
      #r <- mask (r, aoi)
      dum <- stack(dum, r)
   print(paste0(i, names(r), ' done!'))
      }

  cont <- stack()

  for (i in 1:length(lis2)){
      r <- raster (lis2[i])
      r <- projectRaster (r, aoi)
      r <-    crop(r, aoi)
```

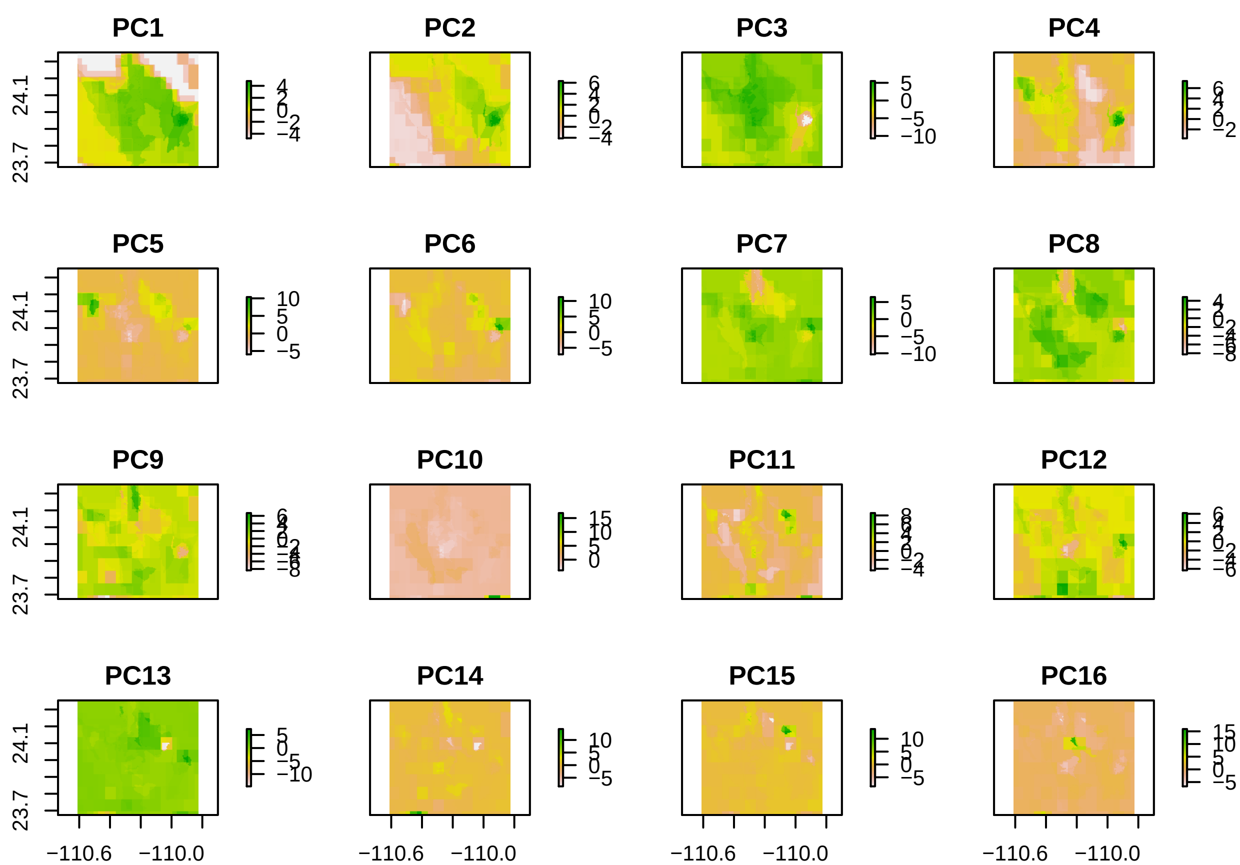
```r
      #r <- mask (r, aoi)

      cont <- stack(cont, r)

   print(paste0(i, names(r), ' done!'))
      }

   cat <- stack()

   for (i in 1:length(lis3)){
      r <- raster (lis3[i])
      r[is.na(r)==TRUE,] <- -9999
      r <- projectRaster (r, aoi, method='ngb')
      r <- crop(r, aoi)
      #r <- mask (r, aoi)
      r <-dummyRaster(r)
      cat <- stack(cat, r)
    print(paste0(i, names(r), ' done!'))
      }

   cat$Edafología_Serie_II4[is.na(cat$Edafología_Serie_II4)==TRUE] <- 2
   cat$Edafología_Serie_II6[is.na(cat$Edafología_Serie_II6)==TRUE] <- 2

   COVS <- stack(dum, cont, cat)
   COVS <- COVS[[-7]]
   COVS[is.infinite(COVS)==TRUE]<- -9999
   COVS[is.na(COVS)==TRUE]<- -9999
 library(RStoolbox)
 COVS <- scale(COVS)
```

prepare predictors for PCA, include the terrain parameters to the covariate space and generate a regression matrix (couple with poins of soil profiles)

```r
COVSpca <- rasterPCA(COVS, maskCheck=FALSE)
pc <- COVSpca$model
#PLOT PCA INERTIA
plot(pc)
```




```r
#PLOT PCA maps
plot(stack(COVSpca$map))
```




```r
COVSpca <- stack(COVSpca$map)[[1:3]]
```

```r
x <- stack(resample( dem, COVSpca), COVSpca)

training <- cbind(data.frame(extract(x, datsp)), OCS = dat$OCS, Y = dat$X, X = dat$Y)

#summary(training)
#CHECK CORRELATED PREDICTORS
round(cor(training), 2)
```

```
##                     dem hillshade curvature convergenceIndex
## dem                1.00      0.32      0.16             0.15
## hillshade          0.32      1.00      0.00            -0.08
## curvature          0.16      0.00      1.00             0.75
## convergenceIndex   0.15     -0.08      0.75             1.00
## flowAccumulation -0.29     -0.24      0.01             0.14
## wetnessIndex     -0.68     -0.25     -0.04             0.10
## lsFactor         -0.48     -0.06     -0.05             0.07
## slope             0.68      0.50     -0.21            -0.33
## aspect            0.12     -0.73     -0.01             0.01
## PC1               0.00      0.06     -0.09             0.25
## PC2               0.06      0.23      0.12             0.20
## PC3               0.14     -0.24     -0.14            -0.35
## OCS               0.07      0.02     -0.25             0.06
## Y                -0.23      0.21      0.10            -0.36
## X                 0.27      0.29      0.16             0.31
##                  flowAccumulation wetnessIndex lsFactor slope aspect   PC1
## dem                         -0.29        -0.68    -0.48  0.68   0.12  0.00
## hillshade                   -0.24        -0.25    -0.06  0.50  -0.73  0.06
## curvature                    0.01        -0.04    -0.05 -0.21  -0.01 -0.09
## convergenceIndex             0.14         0.10     0.07 -0.33   0.01  0.25
## flowAccumulation             1.00         0.65     0.79 -0.26   0.36  0.19
## wetnessIndex                 0.65         1.00     0.91 -0.75   0.14  0.25
## lsFactor                     0.79         0.91     1.00 -0.45   0.20  0.25
## slope                       -0.26        -0.75    -0.45  1.00  -0.05 -0.18
## aspect                       0.36         0.14     0.20 -0.05   1.00 -0.10
## PC1                          0.19         0.25     0.25 -0.18  -0.10  1.00
## PC2                          0.18         0.04     0.20  0.12  -0.09  0.80
## PC3                         -0.05        -0.24    -0.22  0.21   0.40 -0.74
## OCS                         -0.24        -0.32    -0.46  0.06  -0.38  0.48
## Y                            0.10        -0.13     0.03  0.32  -0.09 -0.46
## X                            0.08        -0.04     0.12  0.20  -0.11  0.75
##                     PC2   PC3   OCS     Y     X
## dem                0.06  0.14  0.07 -0.23  0.27
## hillshade          0.23 -0.24  0.02  0.21  0.29
## curvature          0.12 -0.14 -0.25  0.10  0.16
## convergenceIndex   0.20 -0.35  0.06 -0.36  0.31
## flowAccumulation   0.18 -0.05 -0.24  0.10  0.08
## wetnessIndex       0.04 -0.24 -0.32 -0.13 -0.04
## lsFactor           0.20 -0.22 -0.46  0.03  0.12
## slope              0.12  0.21  0.06  0.32  0.20
## aspect            -0.09  0.40 -0.38 -0.09 -0.11
## PC1                0.80 -0.74  0.48 -0.46  0.75
## PC2                1.00 -0.73  0.23 -0.04  0.89
## PC3               -0.73  1.00 -0.38  0.37 -0.76
## OCS                0.23 -0.38  1.00 -0.31  0.23
```

```
## Y                -0.04  0.37 -0.31  1.00 -0.30
## X                 0.89 -0.76  0.23 -0.30  1.00
```

define color pallete for maps, remove non assigned values and mask the prediction space to the area of interest

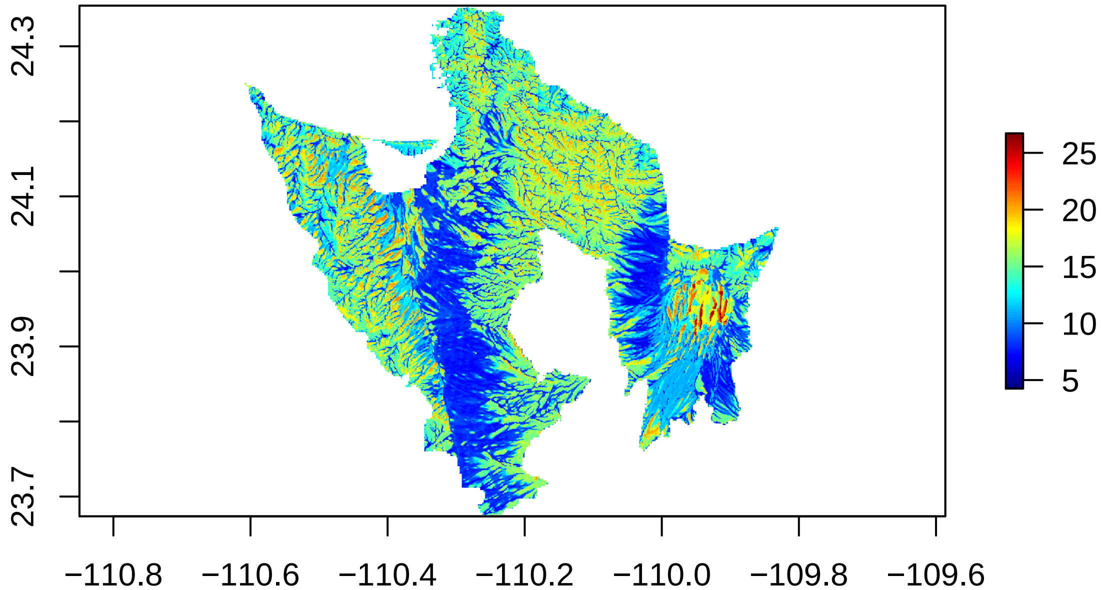
```
jet.colors <-   colorRampPalette(c("#00007F", "blue", "#007FFF", "cyan",
                      "#7FFF7F", "yellow", "#FF7F00", "red", "#7F0000"))

x <- stack(resample(COVSpca, dem), dem)
x[is.na(x)==TRUE]<- -9999
x[is.infinite(x)==TRUE]<- 9999

aoi <- resample(aoi, x, method='ngb')
x <- mask(x, aoi)
```

run 1 predictive model with all the 18 points

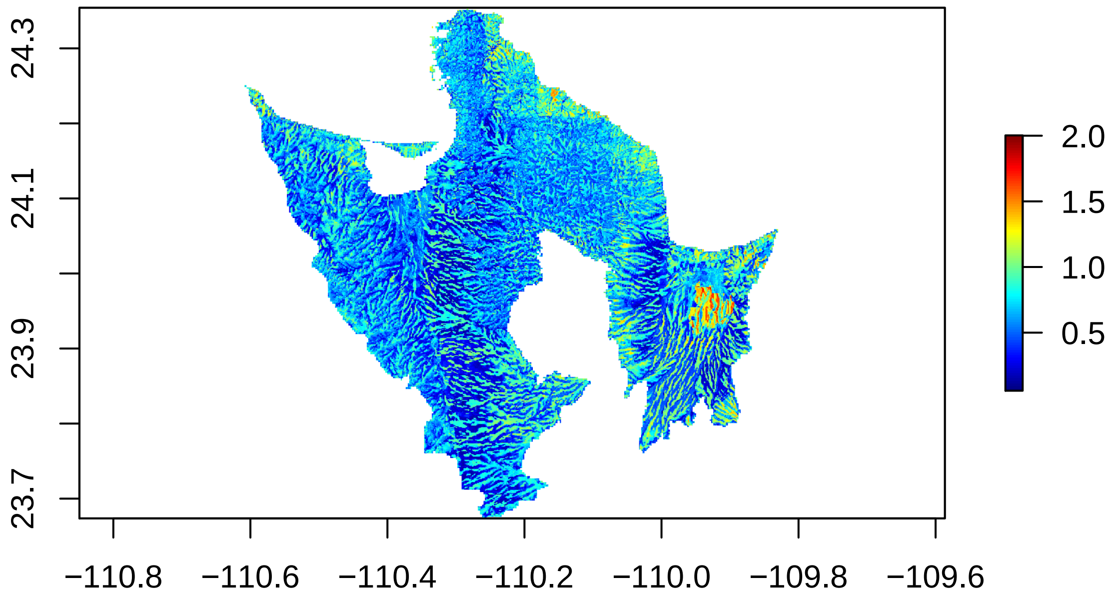
```
library(caret)

s <- stack()
m <- list()
r2 <- numeric()
rmse <- numeric()
#REPEATED CROSS-VALIDATION
control <- rfeControl(functions=rfFuncs, method="repeatedcv",       number=2, repeats=5)

#10 MODELS FOR TESTING
for (i in 1:10){
#RFE recursive feature elimination based on RANDOM FORESTS
rfProfile <- rfe(training[,1:12], training[,13], sizes=c(1:12),         rfeControl=control)

        #BEST FIT
        m[[i]] <- rfProfile
        rmse[i] <- max(m[[i]]$results[2])
        r2[i] <- max(m[[i]]$results[3])

    print(rfProfile)
    predictors(rfProfile)
    predRFE <- predict(x, rfProfile)
    #plot(predRFE, col=jet.colors(100))
    s <- stack(s, predRFE)
  names(s)[[i]] <- paste0('model-', i)
    }
```

plot the predicted maps and the variance map

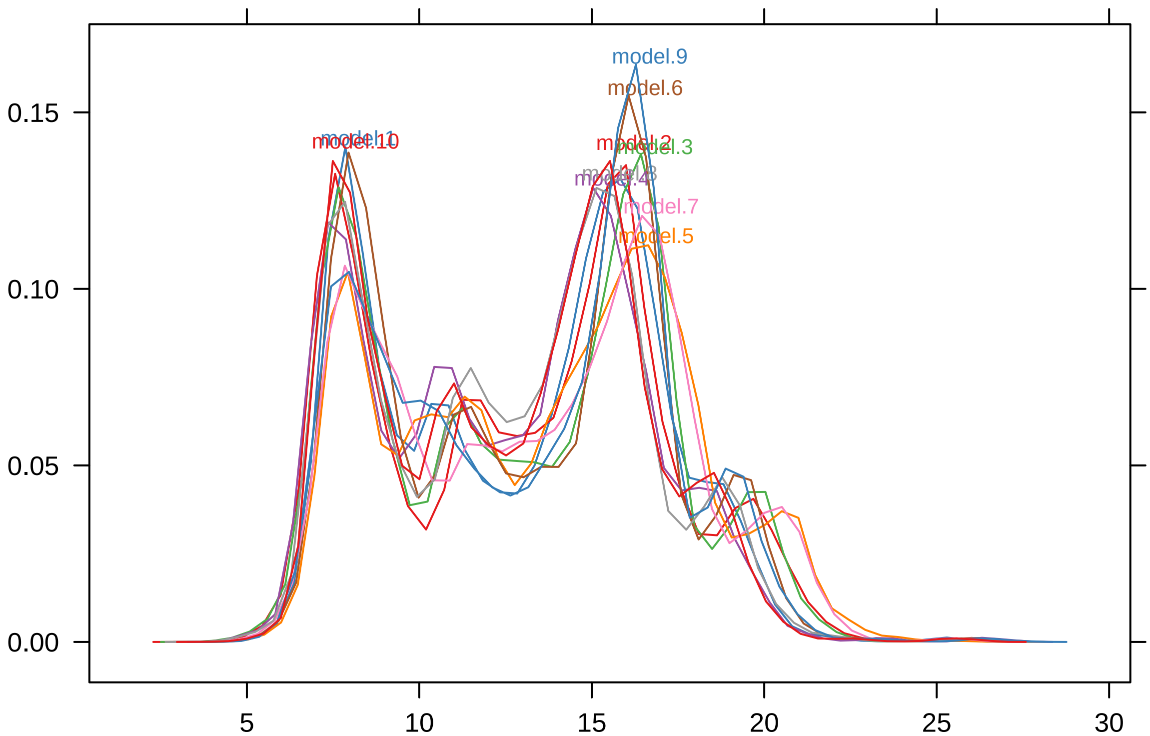
```
#MEAN PREDICTION
plot(calc(s, mean),  col=jet.colors(100))
```



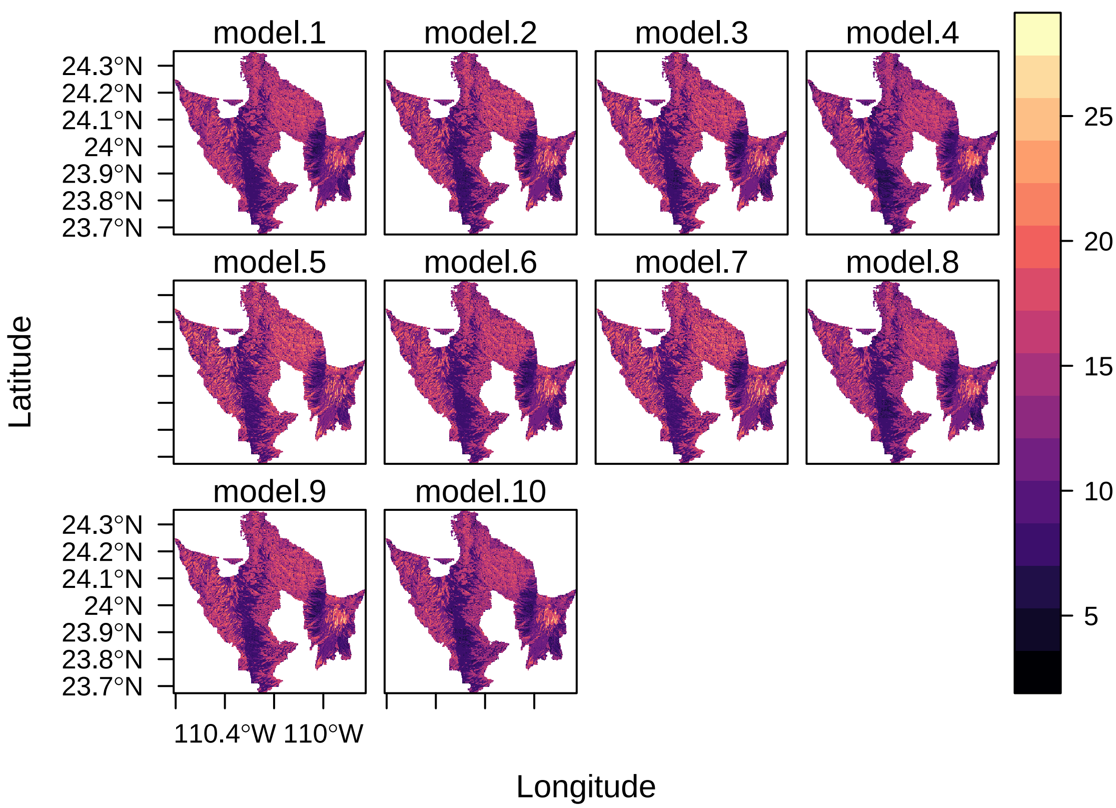

```
#PREDICTION VARIANCE (UNCERTAINTY)
plot(calc(s, sd), col=jet.colors(100))
```




```
#UNCERTANTY
rasterVis::densityplot(s)
```




```r
#ALL PREDICTIONS
rasterVis::levelplot(s)
```



```r
#writeRaster(s, file='SOCpredictions.tif')
```

accuracy numbers

```r
#EXPLAINED VARIANCE
summary(r2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2235  0.2804  0.3300  0.3121  0.3475  0.3535
```

```r
#RMSE
summary(rmse)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   9.756   9.977  10.300  10.350  10.520  11.510
```

```r
#sum pixes and calculate the total SOC stocks for all the area
#cellStats(calc(s, mean), sum)
#and the uncertainty
#cellStats(calc(s, sd), sum)
```

around 30% of explained variance with a mean error of 9.9 Mg.Ha.

end of exercise