Ocurrencia de Bromelias

C.P.P.

29/2/2020

library(rmarkdown)  
#Maps Script\_ practice  
# https://ourcodingclub.github.io/2016/12/11/maps\_tutorial.html  
setwd("D:/USUARIO/Desktop/TRABAJOS R erre/Lab\_SISTE\_practica\_conR/PRoyecSISTE")  
  
 # Load packages  
 library(ggplot2) # ggplot() fortify()  
library(dplyr) # %>% select() filter() bind\_rows()

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(rgdal) # readOGR() spTransform()

## Loading required package: sp

## rgdal: version: 1.4-8, (SVN revision 845)  
## Geospatial Data Abstraction Library extensions to R successfully loaded  
## Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20  
## Path to GDAL shared files: D:/USUARIO/Documents/R/win-library/3.6/rgdal/gdal  
## GDAL binary built with GEOS: TRUE   
## Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ\_VERSION: 493]  
## Path to PROJ.4 shared files: D:/USUARIO/Documents/R/win-library/3.6/rgdal/proj  
## Linking to sp version: 1.3-2

library(raster) # intersect()

##   
## Attaching package: 'raster'

## The following object is masked from 'package:dplyr':  
##   
## select

library(ggsn) # north2() scalebar()

## Loading required package: grid

##   
## Attaching package: 'ggsn'

## The following object is masked from 'package:raster':  
##   
## scalebar

library(rworldmap) # getMap()

## ### Welcome to rworldmap ###

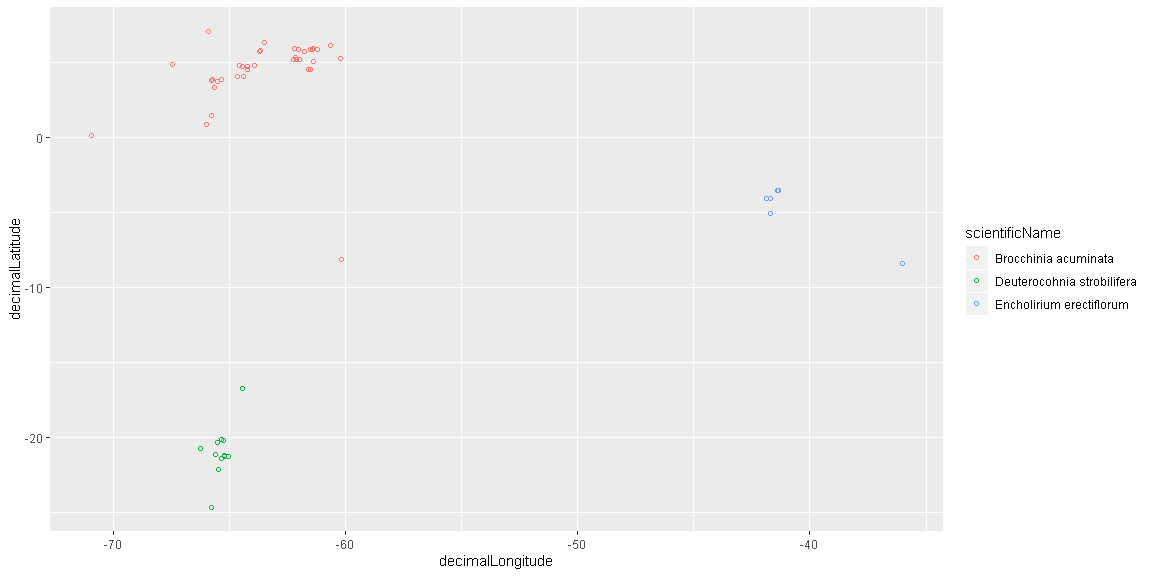
## For a short introduction type : vignette('rworldmap')

library(ggmap)

## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

## Please cite ggmap if you use it! See citation("ggmap") for details.

# Load species data  
B.ACUM<- read.csv("B.acum.csv", sep = ";")  
E.ERECTI<- read.csv("E.erect.csv", sep = ";")  
D.STRO<- read.csv("D.strob.csv", sep = ";")  
  
head(B.ACUM)  
vars1 <- c("key", "scientificName", "decimalLongitude", "decimalLatitude")  
bacum\_trim = B.ACUM %>% dplyr::select(one\_of(vars1))  
eerecti\_trim =E.ERECTI%>% dplyr::select(one\_of(vars1))  
dstro\_trim =D.STRO%>% dplyr::select(one\_of(vars1))  
  
write.table(bacum\_trim)  
write.table(eerecti\_trim)  
write.table(dstro\_trim)  
  
  
BROM\_trim <- bind\_rows(bacum\_trim,eerecti\_trim,dstro\_trim)  
str(BROM\_trim)  
  
# Check that species names are consistent  
unique(BROM\_trim$scientificName)  
# Needs cleaning up   
BROM\_trim$scientificName <- BROM\_trim$scientificName %>%  
 recode("Brocchinia acuminata L.B.Sm."= "Brocchinia acuminata", "Encholirium erectiflorum L.B.Sm." = "Encholirium erectiflorum", "Deuterocohnia strobilifera var. strobilifera" = "Deuterocohnia strobilifera","Deuterocohnia strobilifera var. inermis L.B.Sm." = "Deuterocohnia strobilifera","Deuterocohnia strobilifera Mez" = "Deuterocohnia strobilifera")  
  
# Checking names  
unique(BROM\_trim$scientificName)  
# Done  
  
#gráfico- preliminar  
ggplot(BROM\_trim, aes(x = decimalLongitude, y = decimalLatitude,   
 colour = scientificName)) +  
 geom\_point(shape = 1)

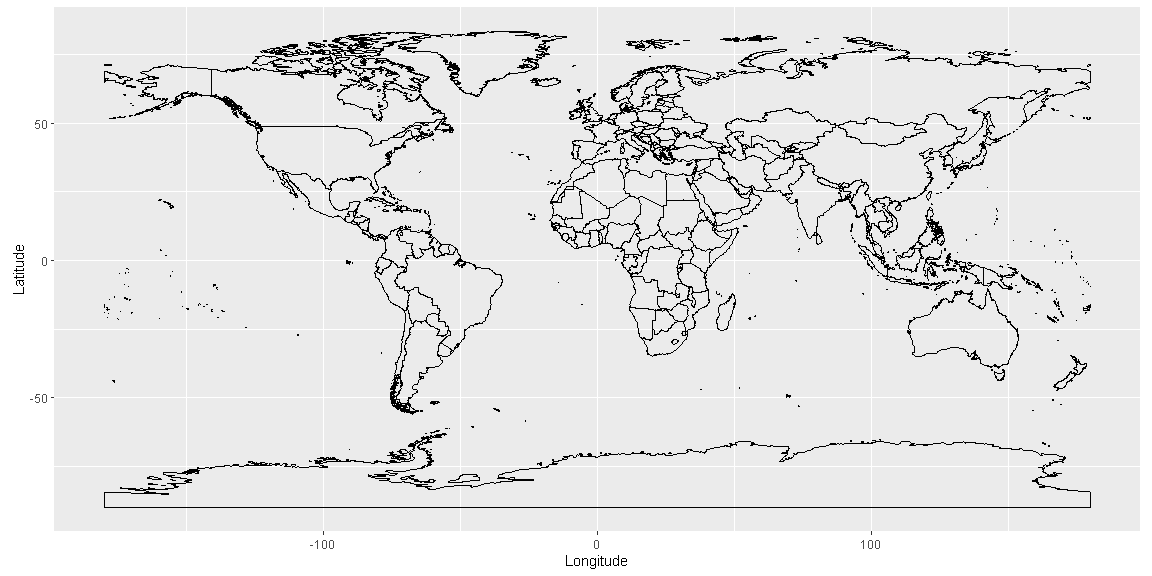


#?ggplot  
#?geom\_point  
BROM\_trim\_us = BROM\_trim %>% filter(decimalLongitude > 0)  
  
# Plot again  
ggplot(BROM\_trim\_us, aes(x = decimalLongitude, y = decimalLatitude,   
 colour = scientificName)) +  
 geom\_point()



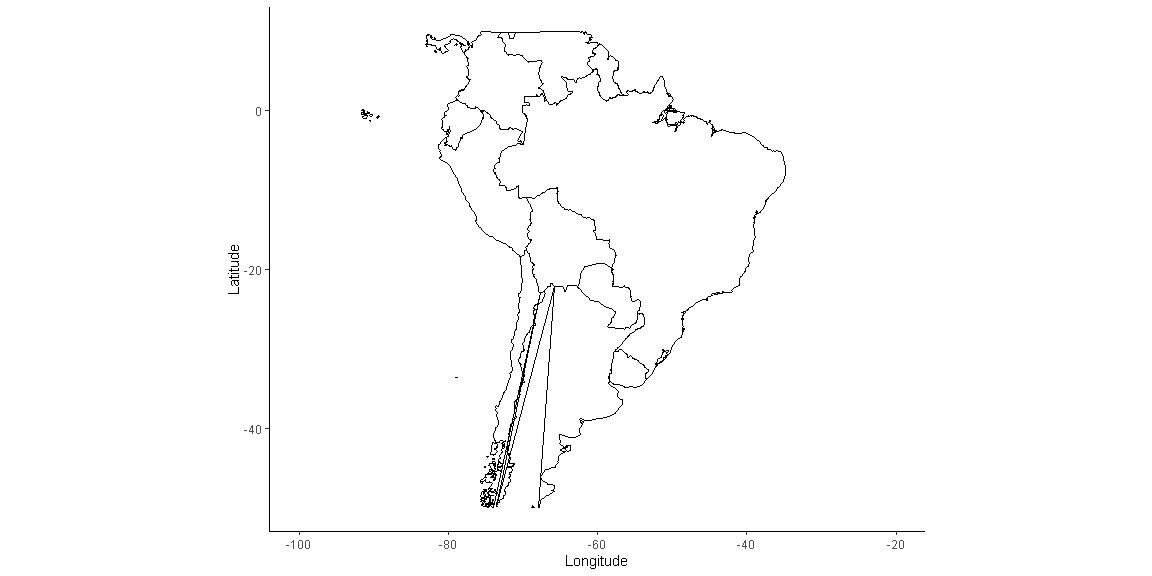
#get world map  
world <- getMap(resolution = "low")  
#BoLI <- getMap()  
  
# Plot data with map  
ggplot() +  
 geom\_polygon(data = world,   
 aes(x = long, y = lat, group = group),  
 fill = NA, colour = "black") +   
 geom\_point(data = BROM\_trim\_us, # Add and plot species data  
 aes(x = decimalLongitude, y = decimalLatitude,   
 colour = scientificName)) +  
 coord\_quickmap() + # Prevents stretching when resizing  
 theme\_gray() + # Remove ugly grey background  
 xlab("Longitude") +  
 ylab("Latitude") +   
 guides(colour=guide\_legend(title="Species"))

## Regions defined for each Polygons

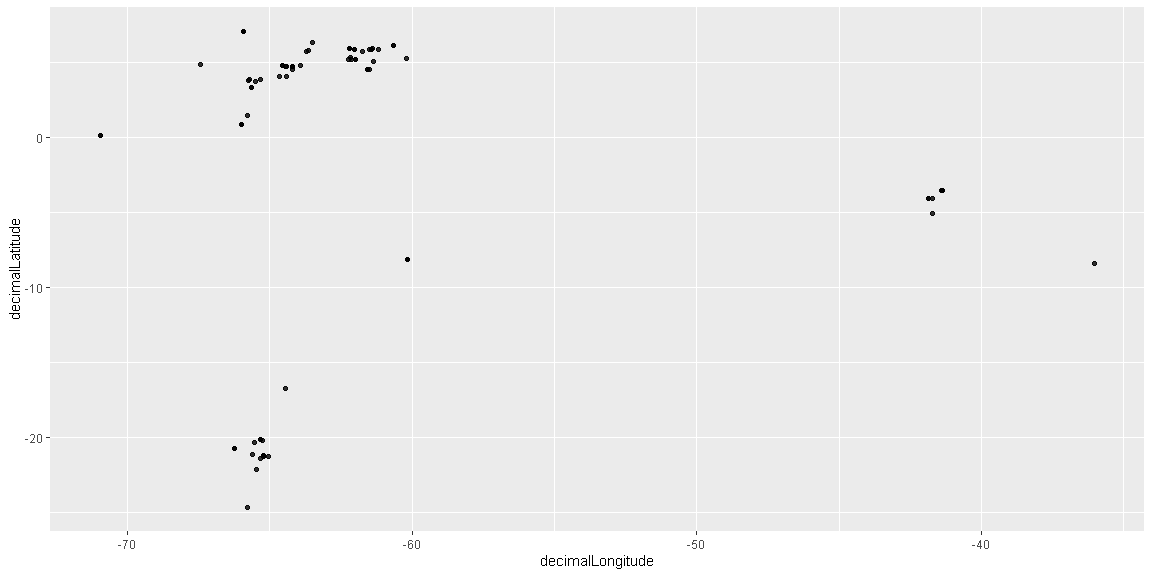


#hacer un vector de Nombres de paises  
saf\_countries1 <- c("Brazil", "Bolivia", "Argentina", "Peru", "Colombia", "Ecuador", "Venezuela", "Paraguay", "Uruguay", "Guyana", "Chile", "Panama")  
  
world\_saf1 <- world[world@data$ADMIN %in% saf\_countries1, ]  
  
ggplot() +  
 geom\_polygon(data = world\_saf1,   
 aes(x = long, y = lat, group = group),  
 fill = NA, colour = "black") +   
 geom\_point(data = BROM\_trim\_us, # Add and plot speices data  
aes(x = decimalLongitude, y = decimalLatitude,   
 colour = scientificName)) +  
 coord\_quickmap() +   
 xlim(-100,-20) + # Set x axis limits, xlim(min, max)  
 ylim(-50, 10) + # Set y axis limits  
 theme\_classic() + # Remove ugly grey background  
 xlab("Longitude") +  
 ylab("Latitude") +   
 guides(colour=guide\_legend(title="Species"))

## Regions defined for each Polygons



# Read in fosterellaaa data  
  
Foste\_rella = BROM\_trim  
# Preliminary plot  
  
ggplot(Foste\_rella, mapping = aes(x = decimalLongitude , y = decimalLatitude)) +  
 geom\_point(alpha = 0.8)



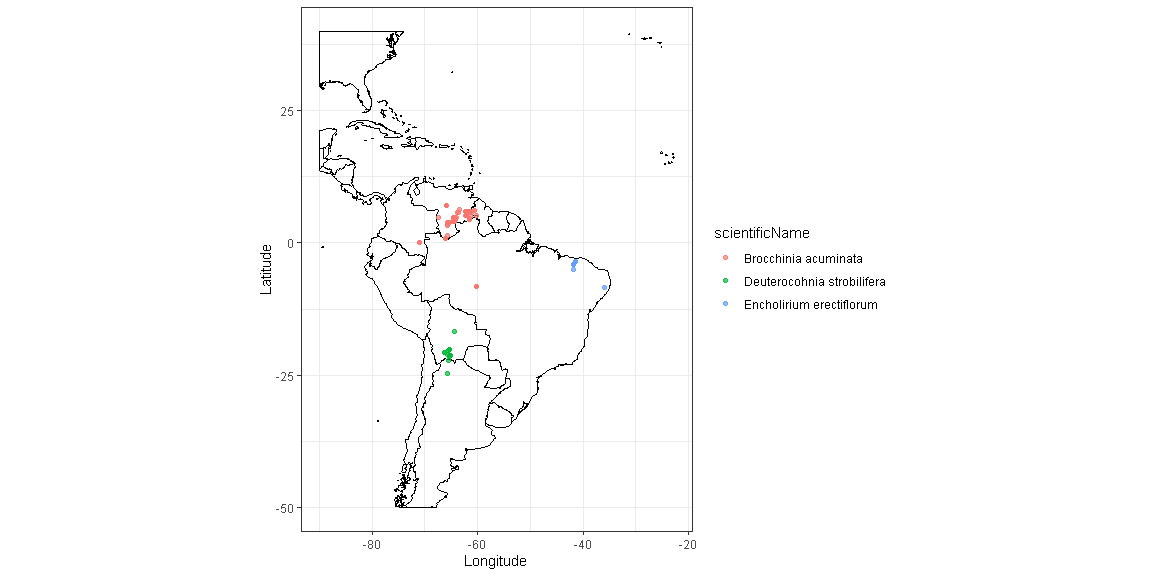
library(raster)  
  
# Get map data  
Clip\_BROM <- as(extent(-90, -20, -50, 40), "SpatialPolygons")  
proj4string(Clip\_BROM) <- CRS(proj4string(world))  
world\_clip <- raster::intersect(world, Clip\_BROM)

## Loading required namespace: rgeos

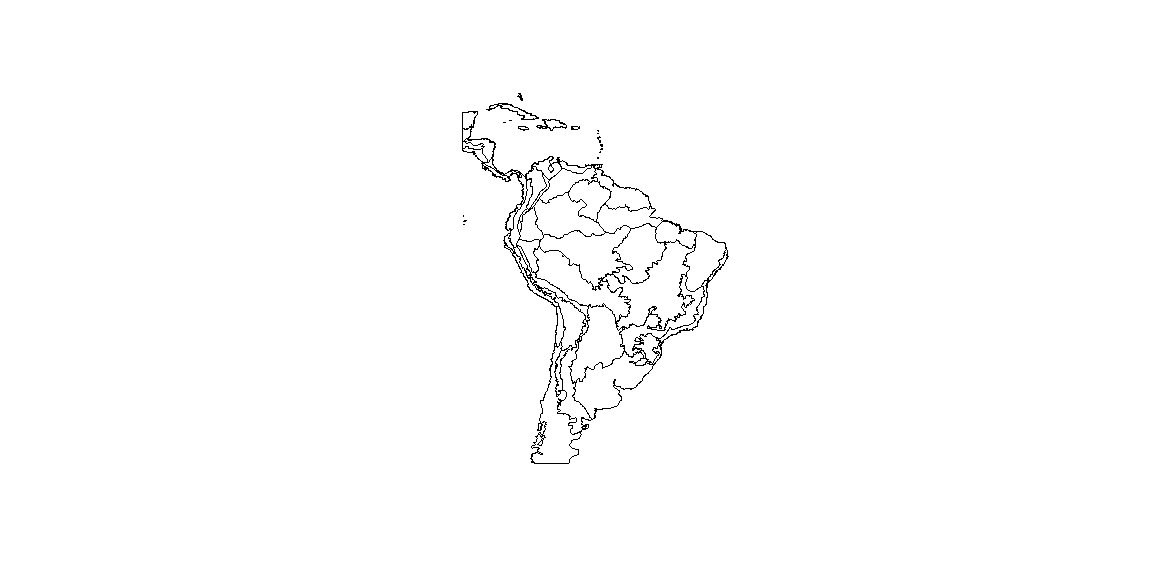
world\_clip\_f <- fortify(world\_clip)

## Regions defined for each Polygons

#PL0T MAP  
ggplot(Foste\_rella, aes(x = decimalLongitude, y = decimalLatitude,   
 colour = scientificName)) +   
 geom\_polygon(data = world\_clip\_f,   
 aes(x = long, y = lat, group = group),  
 fill = NA, colour = "black") +   
geom\_point( alpha = 0.7,  
aes(x = decimalLongitude, y = decimalLatitude, colour = scientificName),  
 data = Foste\_rella) +  
 theme\_bw() +  
 xlab("Longitude") +  
 ylab("Latitude") +   
 coord\_quickmap()



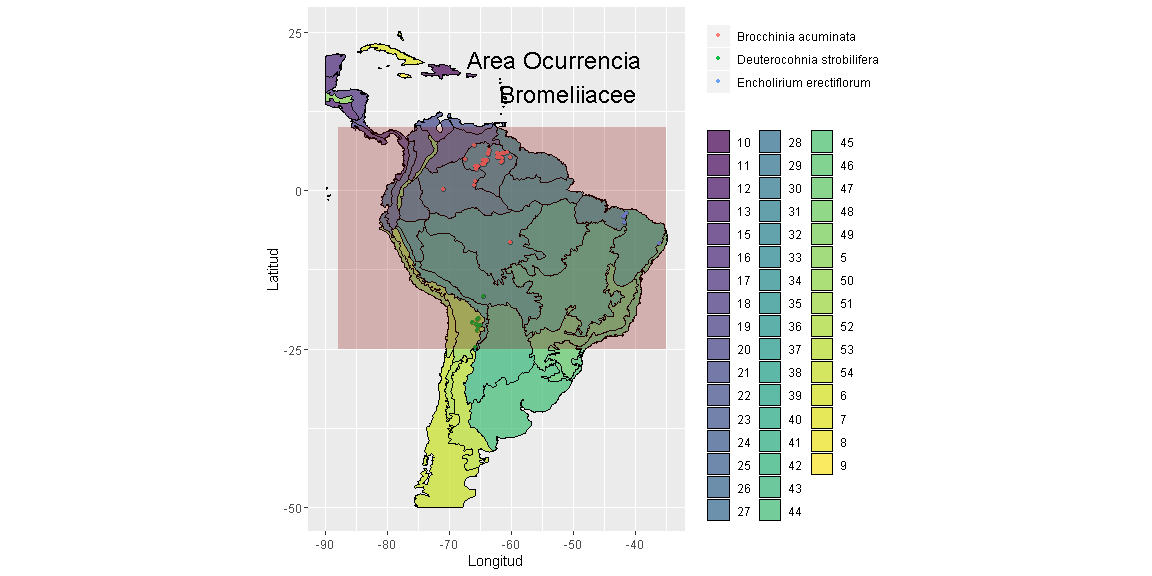
###########################################################  
# Read shapefile  
SHP\_dataaBRO <- readOGR(dsn = "Lowenberg\_Neto\_2014\_shapefile", layer = "Lowenberg\_Neto\_2014")  
# Check CRS  
proj4string(SHP\_dataaBRO)  
  
# Transform CRS  
SHP\_dataaBRO <- spTransform(SHP\_dataaBRO, CRS("+proj=longlat +datum=WGS84"))  
  
# Clip spatial polygons  
s\_clip <- raster::intersect(SHP\_dataaBRO, Clip\_BROM)  
  
# Test plot   
plot(s\_clip)



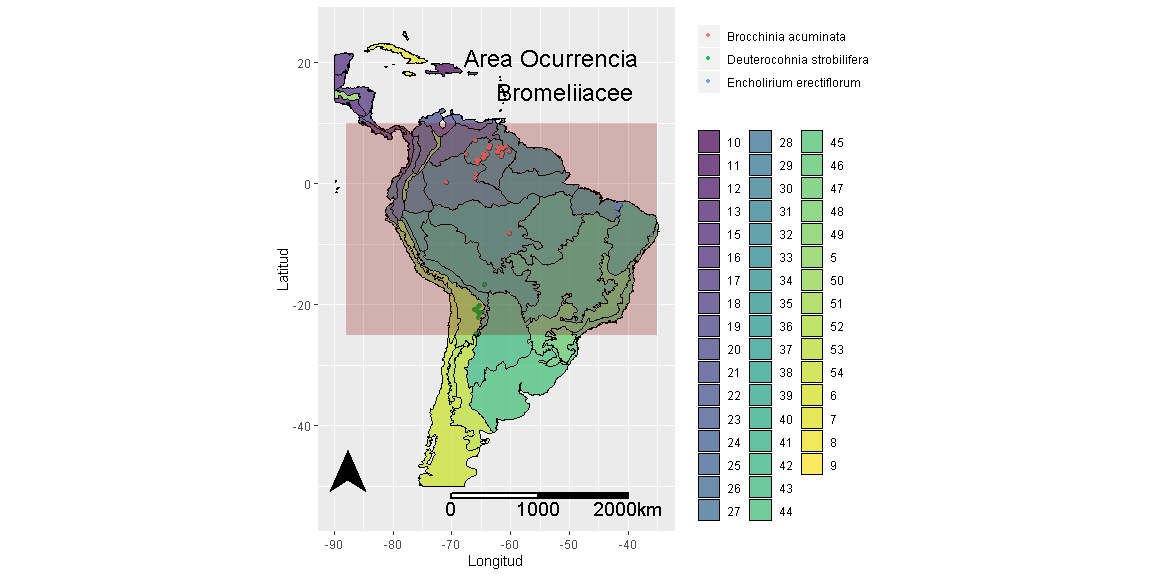
str(s\_clip)  
names(s\_clip)  
# Fortify for ggplot2  
  
s\_clip\_fo <- fortify(s\_clip, region = "NUMB")  
  
library(RColorBrewer)  
library("viridis")

## Loading required package: viridisLite

# ggplot of ecoregions  
map\_BROM <- ggplot(Foste\_rella, aes(x = decimalLongitude, y = decimalLatitude)) +  
 geom\_point(shape =1 )+  
 geom\_polygon(data = s\_clip\_fo,  
 aes(x = long, y = lat, group = group, fill = id),  
 color = "black", size = 0.5) + scale\_fill\_viridis(discrete = TRUE, alpha = 0.7)+  
 geom\_point(alpha = 0.9, size = 1,  
 aes(x = decimalLongitude, y = decimalLatitude, colour = scientificName),  
 data = Foste\_rella) +  
 theme\_grey() +  
 theme(legend.position="right") +  
 theme(legend.title=element\_blank()) +   
 xlab("Longitud") +  
 ylab("Latitud") +   
 coord\_quickmap()  
  
# Add annotations  
(map\_BROM\_annot <- map\_BROM +  
 annotate("rect", xmin = -88 , xmax = -35, ymin = -25, ymax = 10, fill="red4", alpha=0.25) +  
 annotate("text", x = -53, y = 18, size = 6.3, label = "Area Ocurrencia  
 Bromeliiacee"))



# Add scalebar  
map\_BROM\_scale <- map\_BROM\_annot +  
 scalebar(data = s\_clip\_fo,  
 transform = TRUE, dist = 1000, dist\_unit = "km", model='WGS84',  
 height = 0.01,   
 location = "bottomleft", anchor = c(x = -70, y = -52))  
  
# Add north arrow  
 north2(map\_BROM\_scale, x = 0.3, y = 0.18, scale = 0.1, symbol = 12)



## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.