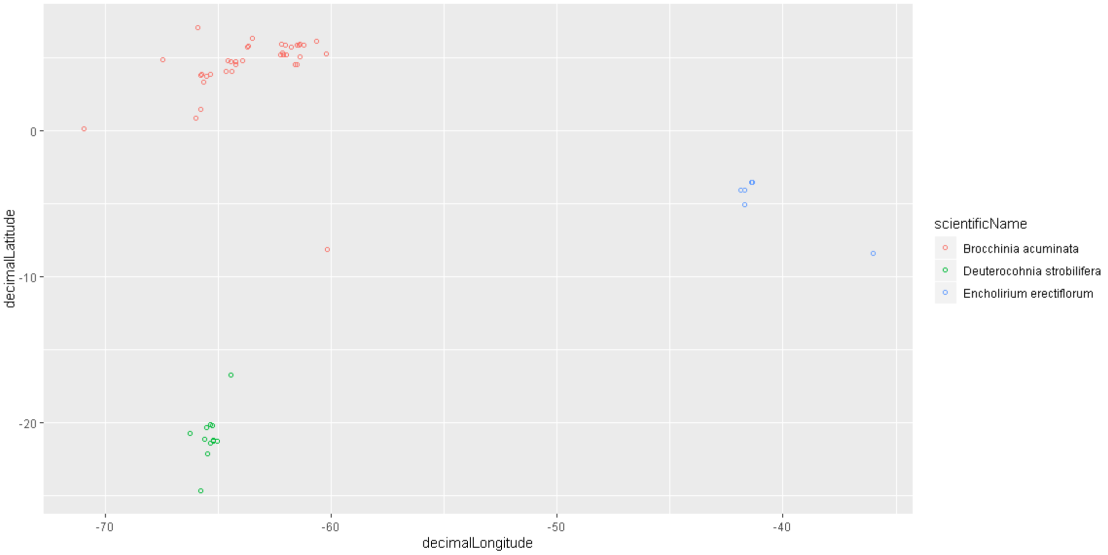
Ocurrencia de Bromelias

C.P.P.

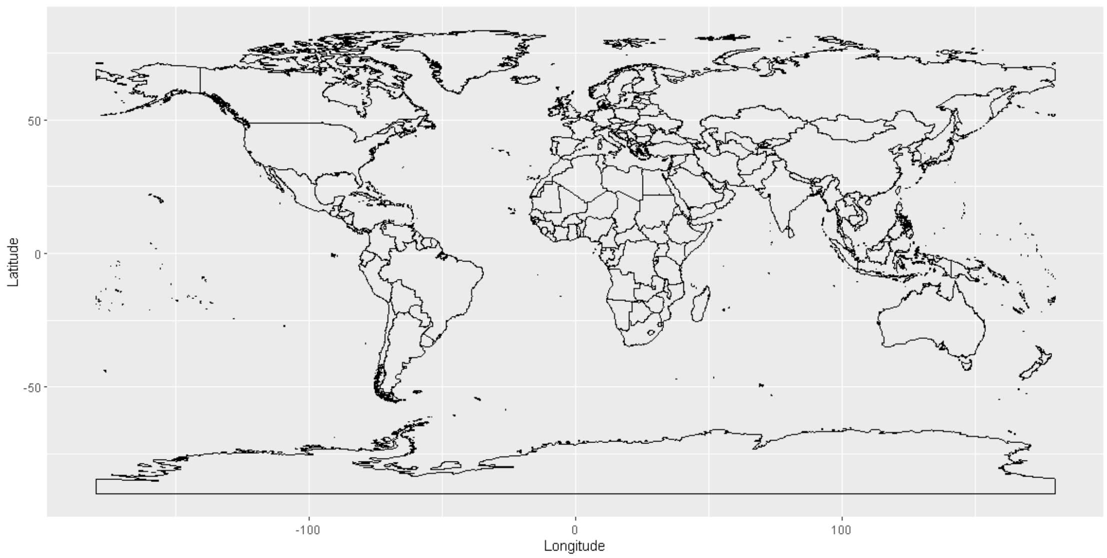
29/2/2020

```
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)
```

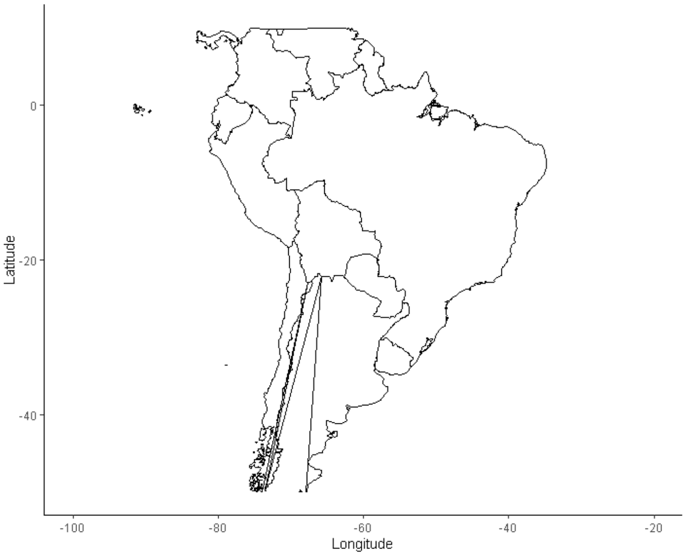


```
#?ggplot
#?geom_point
BROM_trim_us = BROM_trim %>% filter(decimalLongitude > 0)

# Plot again
ggplot(BROM_trim_us, aes(x = decimalLongitude, y = decimalLatitude,
                       colour = scientificName)) +
  geom_point()
```

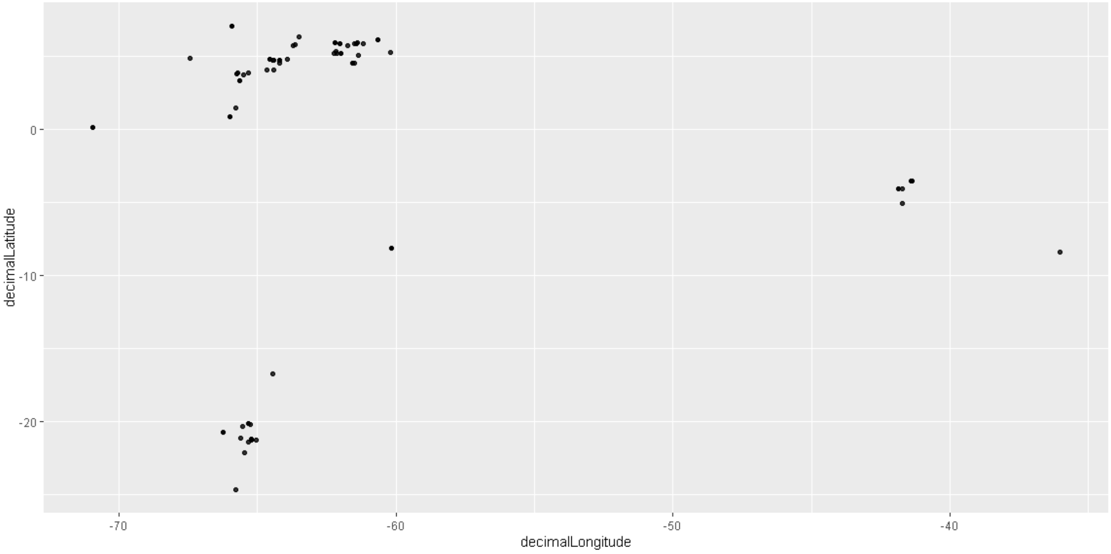
```
#get world map
world <- getMap(resolution = "low")
#BoLI <- getMap()

# 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
```

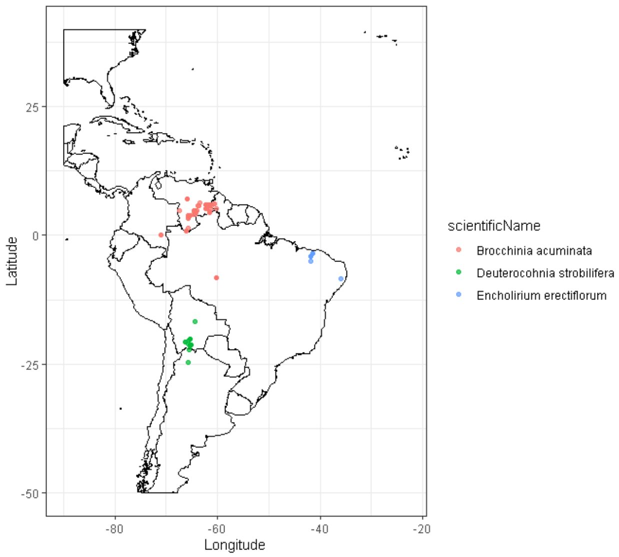


```
# Read in fosterellaaa data

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()
```

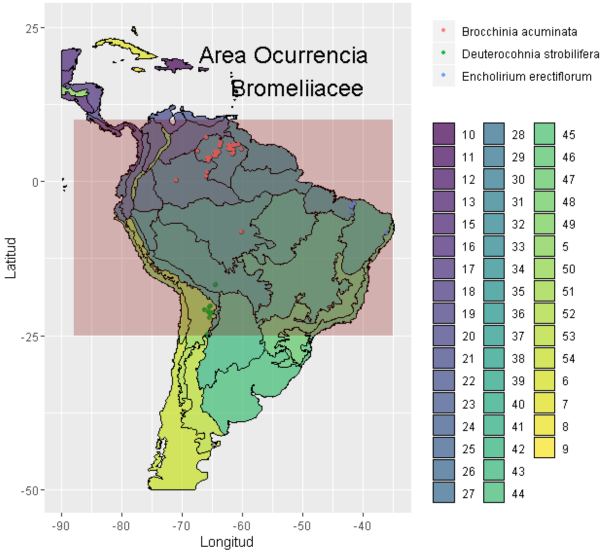


```
###########################################################
# Read shapefile
SHP_dataaBRO <- readOGR(dsn = "Lowenberg_Neto_2014_shapefile", layer =
"Lowenberg_Neto_2014")
# Check CRS
proj4string(SHP_dataaBRO)

# Transform CRS
SHP_dataaBRO <- spTransform(SHP_dataaBRO, CRS("+proj=longlat
+datum=WGS84"))

# Clip spatial polygons
s_clip <- raster::intersect(SHP_dataaBRO, Clip_BROM)
```

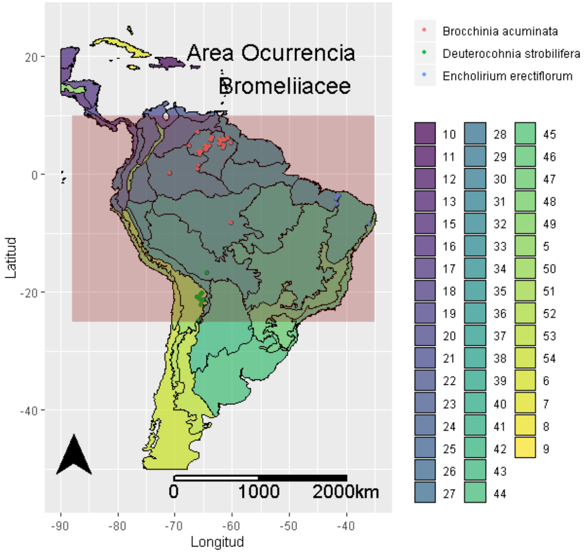
```
# Test plot
plot(s_clip)
```

```
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
```

```r
(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"))
```



```r
# 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.