Final project

RMarkdown version (FinalProject.rmd)

```
r   library(dismo)
```

```
## Loading required package: raster
```

```
## Loading required package: sp
```

```
r   library(maptools)
```

```
## Checking rgeos availability: TRUE
```

```
r   data(wrld_simpl)
```

Set filepaths and import Basemap

```
library(maptools)
library(rgdal)
```

```
## rgdal: version: 1.5-18, (SVN revision 1082)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 3.0.4, released 2020/01/28
## Path to GDAL shared files: C:/Users/Joe Receveur/Documents/R/win-library/4.0/rgdal/gdal
## GDAL binary built with GEOS: TRUE
## Loaded PROJ runtime: Rel. 6.3.1, February 10th, 2020, [PJ_VERSION: 631]
## Path to PROJ shared files: C:/Users/Joe Receveur/Documents/R/win-library/4.0/rgdal/proj
## Linking to sp version:1.4-4
## To mute warnings of possible GDAL/OSR exportToProj4() degradation,
## use options("rgdal_show_exportToProj4_warnings"="none") before loading rgdal.
```

```
library(dismo)
library(ggplot2)
library(GISTools)
```

```
## Loading required package: RColorBrewer
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following objects are masked from 'package:raster':
##
##     area, select
```

```
## Loading required package: rgeos
```

```
## rgeos version: 0.5-5, (SVN revision 640)
##  GEOS runtime version: 3.8.0-CAPI-1.13.1
##  Linking to sp version: 1.4-2
##  Polygon checking: TRUE
```

```
library(ggpubr) #For arranging ggplots
```

```
##
## Attaching package: 'ggpubr'
```

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

```
library(plyr)
```

```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:ggpubr':
##
##     mutate
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:rgeos':
##
##     intersect, setdiff, union
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:raster':
##
##     intersect, select, union
```

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

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

```
# Set the path for outputs
output_path<-("output")
# if this folder doesn't exist, create it
if(!dir.exists(output_path)){
  dir.create(output_path)
}


# Create the folders (directories) "data" and "Final project" - If they exist already, this
# command won't over-write them.
data_path<-(file.path("data","FinalProject"))
if(!dir.exists(data_path)){
  dir.create(data_path,recursive = TRUE)
}

#Download shapefile outline of Ghana
if(! file.exists(file.path(data_path,'countries.zip'))){
 download.file("http://www.naturalearthdata.com/http//www.naturalearthdata.com/download/10m/cult
ural/ne_10m_admin_0_countries.zip", dest=file.path(data_path,"countries.zip"), mode="wb")
}
#unzip (file.path(data_path,"countries.zip"), exdir = data_path)
# Read in the shapefile with rgdal package
world <- readOGR(file.path(data_path,"ne_10m_admin_0_countries.shp"))
```

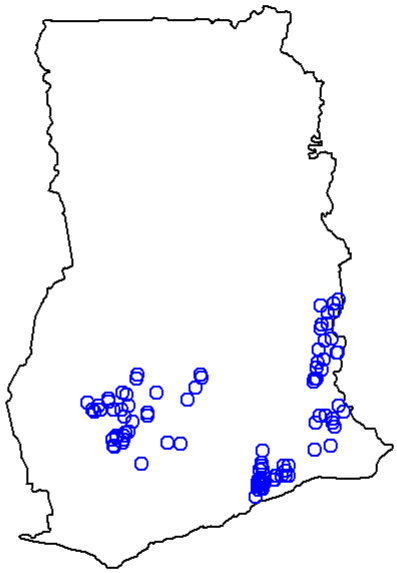
```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\Joe Receveur\Documents\MSU data\GhanaMUSpatialEco\Data\FinalProject\ne_10m_
admin_0_countries.shp", layer: "ne_10m_admin_0_countries"
## with 255 features
## It has 94 fields
## Integer64 fields read as strings:  POP_EST NE_ID
```

```
GhanaOutline<-(world[world$ADMIN=="Ghana",1]) #1=overall country outline




GhanaMeterGrid<-CRS("+proj=tmerc +lat_0=4.666666666666667 +lon_0=-1 +k=0.99975 +x_0=274319.51 +y
_0=0 +ellps=clrk80 +towgs84=-130,29,364,0,0,0,0 +units=m +no_defs") #GhanaMeter Grid EPSG:25000
GhanaOutline
```

```
## class       : SpatialPolygonsDataFrame
## features    : 1
## extent      : -3.262509, 1.187968, 4.737128, 11.16294  (xmin, xmax, ymin, ymax)
## crs         : +proj=longlat +datum=WGS84 +no_defs
## variables   : 1
## names       :      featurecla
## value       : Admin-0 country
```

Import Point Data

```
#Data available at https://github.com/JPReceveur/SpatialEcologyFinalProj
MUSurvey<-read.csv("GhanaMUSurvey.csv",header=T)
MUGeo <- subset(MUSurvey, !is.na(lon) & !is.na(lat)) #Remove NAs


#plot(GhanaOutline)
# restore the box around the map
#box()
# add the points
#points(MUGeo$lon, MUGeo$lat, col='orange', pch=20, cex=0.75)
# plot points again to add a border, for better visibility
#points(MUGeo$lon, MUGeo$lat, col='red', cex=0.75)
#WSGeo

wgs1984.proj <- CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs")
MUGeo.proj <- SpatialPoints(coords=MUGeo[,c("lon","lat")], proj4string=wgs1984.proj)

plot(GhanaOutline)
points(MUGeo.proj,col="blue")
```



```
MUGeo.Meter<-spTransform(MUGeo.proj, GhanaMeterGrid) #project to Ghana meter grid
Presence<-subset(MUGeo,Total.MU.Present..VNTR.==1)#Total MU Presence coded as 0 for absence and
 1 for presence
#Presence$Total.MU.Present..VNTR.
Absence<-subset(MUGeo,Total.MU.Present..VNTR.==0)
names(MUGeo)
```

```
##   [1] "ï..USITEID"                  "YSITEID"
##   [3] "SITECODE"                    "REGION"
##   [5] "lon"                         "lat"
##   [7] "MONTH"                       "YEAR"
##   [9] "SURVEY"                      "ANNUAL"
##  [11] "COUNTRY"                     "REGION.1"
##  [13] "DISTRICT"                    "COMMUNITY"
##  [15] "USWBTYPE"                    "WBFLOW"
##  [17] "HUMAN"                       "RAMSAR"
##  [19] "ENDEM_CASE"                  "ENDEM_DIST"
##  [21] "BU_2003"                     "BU_2004"
##  [23] "BU_2005"                     "BU_2006"
##  [25] "BU_2007"                     "BU_TOTAL"
##  [27] "Soil.Type"                   "FTemp"
##  [29] "FSPECCOND"                   "LELECTCOND"
##  [31] "FDO"                         "FPH"
##  [33] "LPH"                         "ORP"
##  [35] "FTURB"                       "LTURB"
##  [37] "CHLORO"                      "COLORAP"
##  [39] "SUSSOLIDS"                   "DISSOLIDS"
##  [41] "NA."                         "K"
##  [43] "CA"                          "MG"
##  [45] "FE"                          "CL"
##  [47] "SO4"                         "PO4P"
##  [49] "MN"                          "NO2N"
##  [51] "NO3N"                        "TOTHARD"
##  [53] "TOTALK"                      "CALHARD"
##  [55] "MAGHARD"                     "HCO3"
##  [57] "AS"                          "NH4"
##  [59] "F"                           "NF.ER..N."
##  [61] "NF.ER.NoPos.N."              "NF.ER.Positivity"
##  [63] "NF.NoforVNTR"                "NF.VNTR.MU.N."
##  [65] "NF.VNTR.MU.NoPos.N."         "NF.VNTR.Positivity.MU."
##  [67] "NF.Total.MU.Positivity"      "NF.VNTR.MPM.N."
##  [69] "NF.VNTR.NoPos.MPM.N."        "NF.VNTR.Positivity.MPM."
##  [71] "NF.Total.MPM.Positivity"     "FF.ER.N."
##  [73] "FF.ER.NoPos.N."              "FF.ER.Positivity"
##  [75] "FF.NoforVNTR"                "FF.VNTR.MU.N."
##  [77] "FF.VNTR.MU.NoPos.N."         "FF.VNTR.Positivity.MU."
##  [79] "FF.Total.MU.Positivity"      "FF.VNTR.MPM.N."
##  [81] "FFVNTR.NoPos.MPM.N."         "FF.VNTR.Positivity.MPM."
##  [83] "FF.Total.MPM.Positivity"     "Plant.A"
##  [85] "Plant.B"                     "Plant.C"
##  [87] "A.ER"                        "B.ER"
##  [89] "C.ER"                        "ER.N."
##  [91] "Plant.ER.NoPos.N."           "Plant.ER.Positivity"
##  [93] "MU.A.VNTR"                   "MU.B.VNTR"
##  [95] "MU.C.VNTR"                   "Plant.MU.VNTR.N."
##  [97] "Plant.MU.VNTR.NoPos.N."      "Plant.MU.VNTR.Positivity"
##  [99] "Plant.Total.MU.Positivity"   "MPM.A.VNTR"
## [101] "MPM.B.VNTR"                  "MPM.C.VNTR"
## [103] "MPM.VNTR..N."                "Plant.MPM.VNTR.NoPos.N."
## [105] "Plant.MPM.VNTR.Positivity"   "Plant.Total.MPM.Positivity"
```

```
## [107] "Avg.MU.Total.VNTR.Positivity"  "Total.MU.Present..VNTR."
## [109] "Avg.MPM.Total.VNTR.Positivity" "Total.MPM.Present..VNTR."
## [111] "Avg.Filter.VNTR.MU"            "Avg.Filter.VNTR.MPM"
## [113] "Filter.MU.Present..VNTR."      "Plant.MU.Present..VNTR."
## [115] "Filter.MPM.Present..VNTR."     "Plant.MPM.Present..VNTR."
## [117] "Total.ER.Present"              "Total.ER.NoPos.N."
## [119] "Avg.Total.ER.Positivity"
```

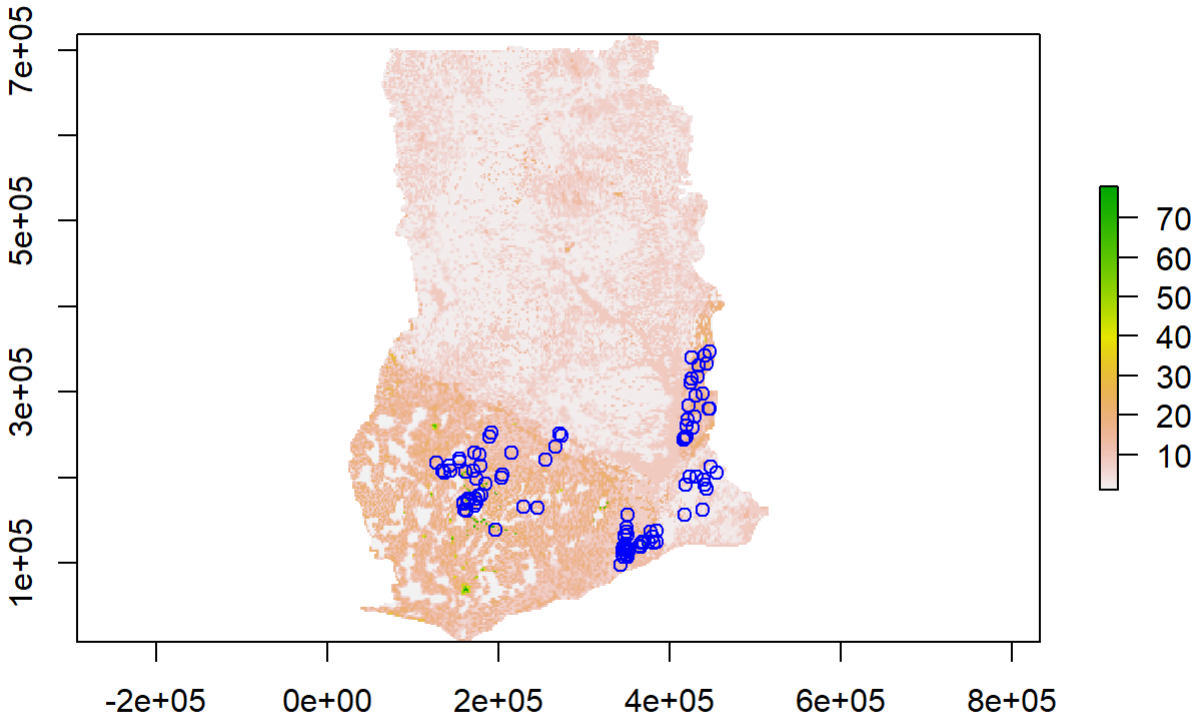
```
PresenceMPM<-subset(MUGeo,Total.ER.Present==1)#Total MPM Presence coded as 0 for absence and 1 f
or presence
#Presence$Total.MU.Present..VNTR.
AbsenceMPM<-subset(MUGeo,Total.ER.Present==0)


#Project the subsets into WGS84 and Ghana Meter Grid

wgs1984.proj <- CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs")
Presence.proj <- SpatialPoints(coords=Presence[,c("lon","lat")], proj4string=wgs1984.proj)
Absence.proj <- SpatialPoints(coords=Absence[,c("lon","lat")], proj4string=wgs1984.proj)
PresenceMPM.proj <- SpatialPoints(coords=PresenceMPM[,c("lon","lat")], proj4string=wgs1984.proj)
AbsenceMPM.proj <- SpatialPoints(coords=AbsenceMPM[,c("lon","lat")], proj4string=wgs1984.proj)


Presence.Meter<-spTransform(Presence.proj, GhanaMeterGrid)
Absence.Meter<-spTransform(Absence.proj, GhanaMeterGrid)

PresenceMPM.Meter<-spTransform(PresenceMPM.proj, GhanaMeterGrid)
AbsenceMPM.Meter<-spTransform(AbsenceMPM.proj, GhanaMeterGrid)


#MUGeo.Meter
GhanaOutline.Meter<-spTransform(GhanaOutline,GhanaMeterGrid)

# par(mfrow=c(1,2), mai=c(0.1,0.1,0.5,0.1))
# plot(GhanaOutline,main="WGS84")
# box()
# points(MUGeo.proj,col="blue")
#
#
#
# plot(GhanaOutline.Meter,main="GhanaMeterGrid")
# box()
# points(MUGeo.Meter,col="blue")
```

Import land cover data (Ghana 2013)

```
#Download land cover data from https://www.sciencebase.gov/catalog/item/5deffc05e4b02caea0f4f3fc

if(! file.exists(file.path(data_path,'LULC.zip'))){
 download.file("http://edcintl.cr.usgs.gov/downloads/sciweb1/shared/wafrica/downloads/data/west_africa_land-use_land-cover_2013_2km.zip", dest=file.path(data_path,"LULC.zip"), mode="wb")
}
#unzip (file.path(data_path,"LULC.zip"), exdir = data_path)

LandCover2k <- stack(file.path(data_path,"west_africa_land-use_land-cover_2013_2km/swa_2013lulc_2km.tif"))
#plot(LandCover2k)
LandCover2k.Meter<-projectRaster(LandCover2k,crs=GhanaMeterGrid)
#summary(GhanaOutline.Meter)
par(mfrow=c(1,1))
r<- raster(GhanaOutline.Meter)
res(r)<-2000   #Not really sure what number to put here (units of outline shapefile are m, so just picked the resoultion for the LandCover2k.Meter file)
r[] <- rnorm(ncell(r))
#plot(r)
#plot(GhanaOutline.Meter,add=T)
r[] <- 0

GhanaRaster<-rasterize(GhanaOutline.Meter,r,getCover=T)
GhanaRaster[GhanaRaster>0]<-1
GhanaRaster[GhanaRaster==0]<-NA
#plot(GhanaRaster)
#summary(LandCover2k)

#res(LandCover2k.Meter)
LULCRaster<-LandCover2k.Meter*GhanaRaster #Subset land use by the Ghana outline (all other raster points =0)
```

```
## Warning in LandCover2k.Meter * GhanaRaster: Raster objects have different
## extents. Result for their intersection is returned
```

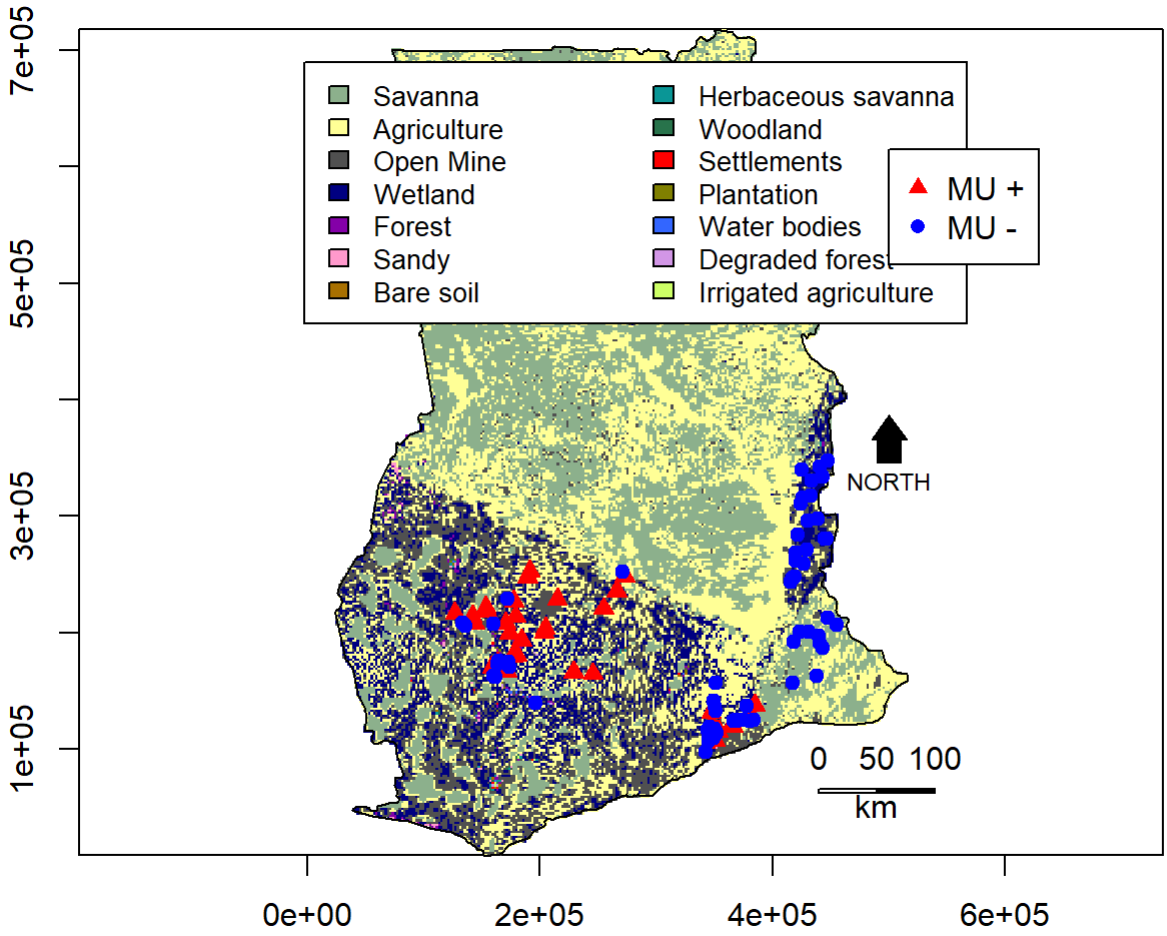
```
plot(LULCRaster)
points(MUGeo.Meter,col="blue")
```



Land use graph MU (Figure 2a)

```
#pal_nlcd Function modified from https://space-lab-msu.github.io/MSUGradSpatialEcology/lab3_patch_design.html
#Table of values avialable at https://github.com/JPReceveur/SpatialEcologyFinalProj

pal_nlcd <- function() {
  data.frame(
    class = c("forest", "forest",
              "forest", "shrubland",
              "anthropogenic", "anthropogenic",
              "wetland","anthropogenic",
              "forest", "No Data",
              "Sand", "shrubland",
              "herbaceous", "herbaceous",
              "No Data", "anthropogenic",
              "shrubland", "forest",
              "anthropogenic", "anthropogenic",
              "No Data","shrubland",
              "shrubland","herbaceous",
              "Oasis","shrubland",
              "shrubland","water",
              "forest","anthropogenic",
              "No Data"),
    code = as.character(c(7, 15,
             28, 2, 8, 78,
             3,
             24, 1, 99,
             10,12, 31,
             4, 0, 27, 32,
             25, 13,
             6, 98,11,29,23,5,16,22,9,21,14,255)),
    description = c("Mangrove", "Gallery/ Riparian Forest",
                    "Swamp Forest", "Savanna",
                    "Agriculture", "Open Mine",
                    "Wetland","Agriculture/ flood recessional",
                    "Forest", "Cloud",
                    "Sandy","Bare soil",
                    "Herbaceous savanna", "Steppe",
                    "No data","Cropland/ oil palms",
                    "Shrubland",  "Woodland",
                    "Settlements","Plantation",
                    "Cloud shadow", "Rocky land",
                    "Sahelian short grass savanna", "Thicket",
                    "Oasis","Shrub and savanna",
                    "Bowe", "Water bodies",
                    "Degraded forest","Irrigated agriculture",
                    "No Data"),
    color = c("#33cccc", "#aa5ce8",
              "#beffa6", "#8cb08c",
              "#ffff96", "#505050",
              "#000081","#ebc961",
              "#8400a8", "#ffffff",
              "#ff99cc", "#a87000",
              "#0a9696", "#ffd09b",
```

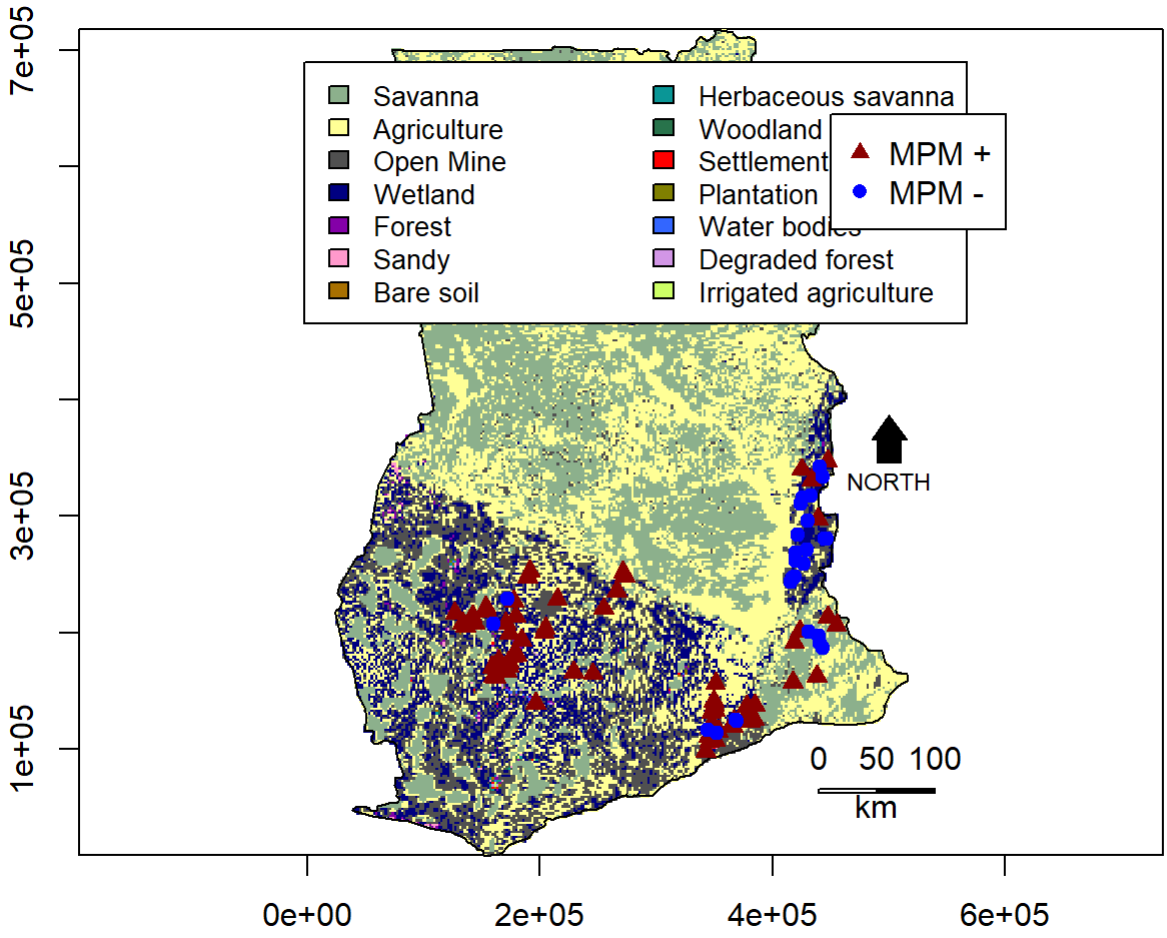
```
              "#ffffff", "#77AD93",
              "#749373", "#28734b",
              "#ff0000", "#808000",
              "#ffffff", "#969696",
              "#a7c38d", "#f8a37b",
              "#99C147", "#77AD93",
              "#DCD939", "#3366ff",
              "#d296e6", "#cdff66",
              "##ffffff"
              ),
    stringsAsFactors = FALSE)
}


NLCD <- as.matrix(table(raster::values(LULCRaster)))
cols <- dplyr::filter(pal_nlcd(), code %in% row.names(NLCD))
par(xpd = FALSE,mai = c(0.5, 0.5, 0.2, 0))

plot(LULCRaster, legend=FALSE,axes=TRUE,box=FALSE,col=cols$color,
     xlab="", ylab="")
par(xpd = TRUE) # reset to limit plotting to figure region
plot(GhanaOutline.Meter,add=T)
legend(x = -2000, y = 690000, legend = cols$description, fill = cols$color,
       ncol = 2,  cex = 0.85, inset = 0.9)
legend(x=500183,y=615115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
plot(Presence.Meter,col="red",bg="red",add=T,pch=24)
plot(Absence.Meter,col="blue",bg="blue",add=T,pch=21)
```




```
#
# dev.off()
# tiff("output/LUGhana.tiff", width = 174, height = 174, units = 'mm', res = 600)
# par(xpd = FALSE,mai = c(0.5, 0.5, 0.2, 0))
#
# plot(LULCRaster, legend=FALSE,axes=TRUE,box=FALSE,col=cols$color,
#      xlab="", ylab="")
# par(xpd = TRUE) # reset to limit plotting to figure region
# plot(GhanaOutline.Meter,add=T)
# legend(x = -2000, y = 690000, legend = cols$description, fill = cols$color,
#        ncol = 2,  cex = 0.85, inset = 0.9)
# legend(x=450183,y=645115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
# scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
# north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
# plot(Presence.Meter,col="red",bg="red",add=T,pch=24)
# plot(Absence.Meter,col="blue",bg="blue",add=T,pch=21)
#
# dev.off()
```

MPM Land Cover graph (Figure S1)

```
#Plotted as above but showing MPM pres/abs
par(xpd = FALSE,mai = c(0.5, 0.5, 0.2, 0))

plot(LULCRaster, legend=FALSE,axes=TRUE,box=FALSE,col=cols$color,
     xlab="", ylab="")
par(xpd = TRUE) # reset to limit plotting to figure region
plot(GhanaOutline.Meter,add=T)
legend(x = -2000, y = 690000, legend = cols$description, fill = cols$color,
       ncol = 2,  cex = 0.85, inset = 0.9)
legend(x=450183,y=645115,legend =c("MPM +","MPM -"),pch=c(17,16),col=c('red4','blue'))
scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
plot(PresenceMPM.Meter,col="red4",bg="red4",add=T,pch=24)
plot(AbsenceMPM.Meter,col="blue",bg="blue",add=T,pch=21)
```



```
# dev.off()
# tiff("output/LUGhana2.tiff", width = 174, height = 174, units = 'mm', res = 1200)
# par(xpd = FALSE,mai = c(0.5, 0.5, 0.2, 0))
#
# plot(LULCRaster, legend=FALSE,axes=TRUE,box=FALSE,col=cols$color,
#      xlab="", ylab="")
# par(xpd = TRUE) # reset to limit plotting to figure region
# plot(GhanaOutline.Meter,add=T)
# legend(x = -2000, y = 690000, legend = cols$description, fill = cols$color,
#        ncol = 2,  cex = 0.85, inset = 0.9)
# legend(x=450183,y=645115,legend =c("MPM +","MPM -"),pch=c(17,16),col=c('red4','blue'))
# scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
# north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
# plot(PresenceMPM.Meter,col="red4",bg="red4",add=T,pch=24)
# plot(AbsenceMPM.Meter,col="blue",bg="blue",add=T,pch=21)
#
# dev.off()
```

Landscape Metrics

Calculate 5k landscape metrics

```
#Load landscape pachages
library(landscapemetrics)
```

```
## Warning: package 'landscapemetrics' was built under R version 4.0.2
```

```
library(landscapetools)
```

```
## Warning: package 'landscapetools' was built under R version 4.0.2
```

```
#Data same as above
MUSurvey<-read.csv("GhanaMUSurvey.csv",header=T)

MUGeo <- subset(MUSurvey, !is.na(lon) & !is.na(lat))


wgs1984.proj <- CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs")
MUGeo.proj <- SpatialPoints(coords=MUGeo[,c("lon","lat")], proj4string=wgs1984.proj)

LandCover<-LandCover2k
MUGeo.proj<-spTransform(MUGeo.proj,crs(LandCover))
GhanaOutline2<-spTransform(GhanaOutline,crs(LandCover))



#Sample 5k buffer (5000 m) around each sample point using landscape metrics
Buffer5k<-sample_lsm(
  LandCover,
  MUGeo.proj,
  plot_id = NULL,
  shape = "square",
  size=5000,
  all_classes = FALSE,
  return_raster = FALSE,
  verbose = F,
  progress = FALSE,level="landscape",type="aggregation metric")

Buffer5k
```

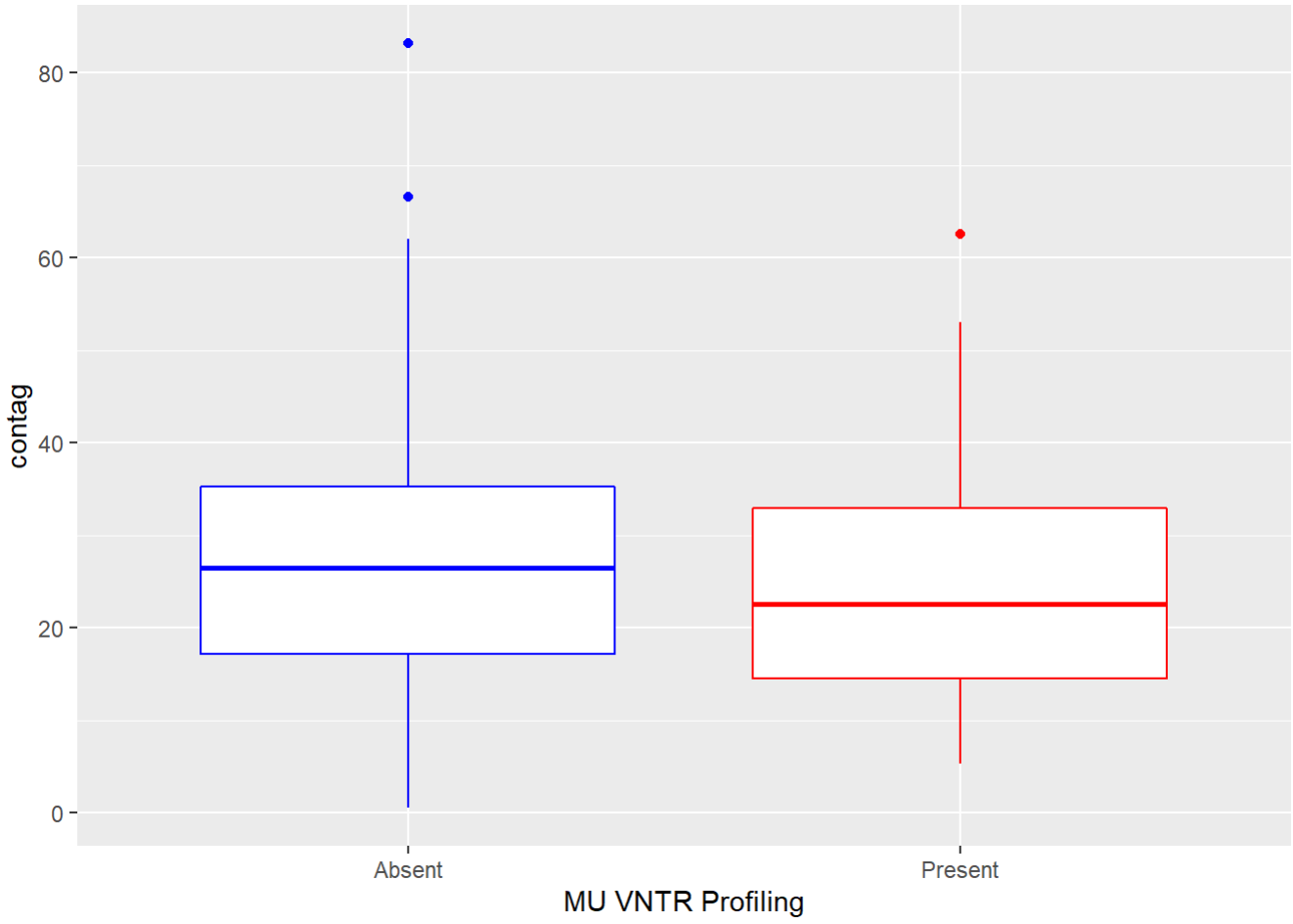
```
## # A tibble: 1,372 x 8
##    layer level     class    id metric      value plot_id percentage_inside
##    <int> <chr>     <int> <int> <chr>       <dbl>   <int>             <dbl>
##  1     1 landscape    NA    NA ai         62.9         1               100
##  2     1 landscape    NA    NA cohesion   64.6         1               100
##  3     1 landscape    NA    NA contag     17.9         1               100
##  4     1 landscape    NA    NA division    0.778       1               100
##  5     1 landscape    NA    NA enn_cv     NA           1               100
##  6     1 landscape    NA    NA enn_mn     NA           1               100
##  7     1 landscape    NA    NA enn_sd     NA           1               100
##  8     1 landscape    NA    NA iji        74.8         1               100
##  9     1 landscape    NA    NA lsi         2.05        1               100
## 10     1 landscape    NA    NA mesh     2224           1               100
## # ... with 1,362 more rows
```

```
#unique(Buffer$metric)
```

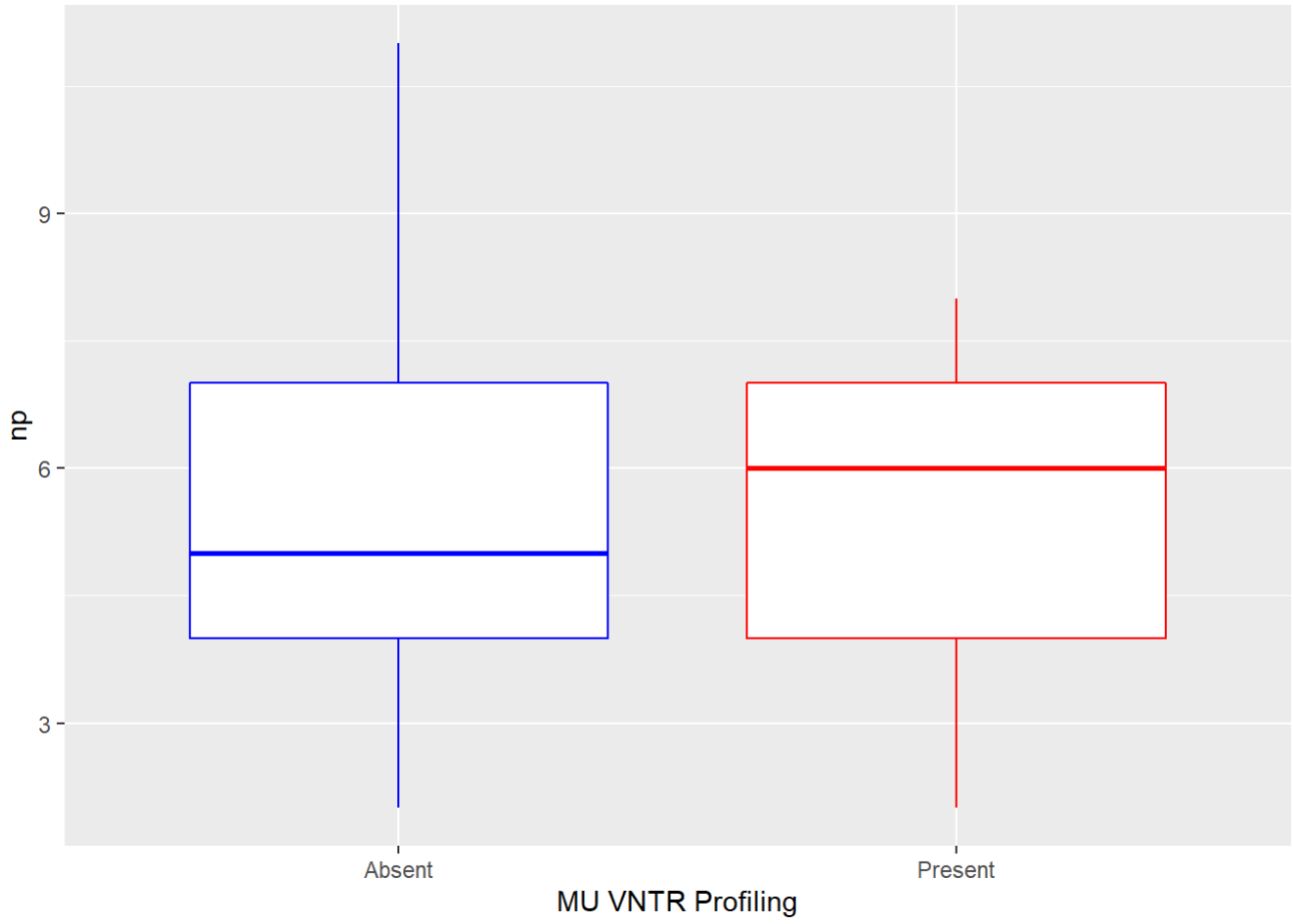
Landscape Metric MU 5k

```
#For loop testing differences between PresAbs values for the chosen metrics (MetricList)
MetricList<-c("contag","np","lsi","cohesion")

for(i in 1:length(MetricList)){
  Subset<-subset(Buffer5k,Buffer5k$metric==MetricList[i])
  print(unique(Subset$metric))



  Subset$plot_id<-MUGeo$USITEID
  Subset$PresAbs<-MUGeo$Total.MU.Present..VNTR.
  Subset$PresAbs[Subset$PresAbs==1]<-"Present"
  Subset$PresAbs[Subset$PresAbs==0]<-"Absent"
  Plot <- ggplot(Subset,aes( x=PresAbs,y=value,col=PresAbs))+ylab(MetricList[i])+xlab("MU VNTR P
rofiling")+geom_boxplot()+scale_color_manual(values=c("blue","red"))+theme(legend.position="non
e")
  print(Plot)
  print(kruskal.test(data=Subset,value~PresAbs))
  Trtdata <- ddply(Subset, c("metric","PresAbs"), summarise,
                N    = length(value),
                mean = mean(value),
                sd   = sd(value),
                se   = sd / sqrt(N)
  )
  print(Trtdata)

}
```

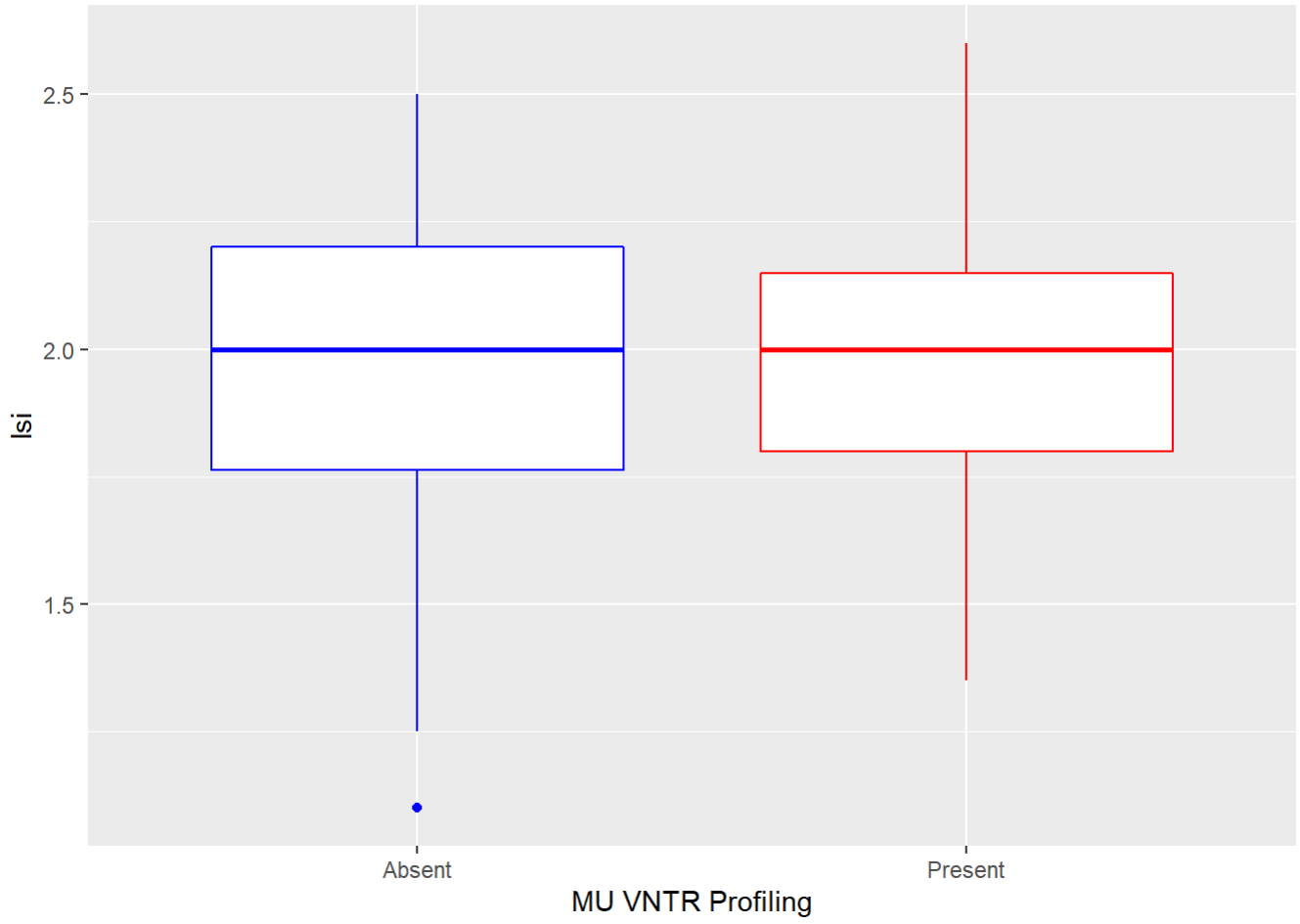
```
## [1] "contag"
```



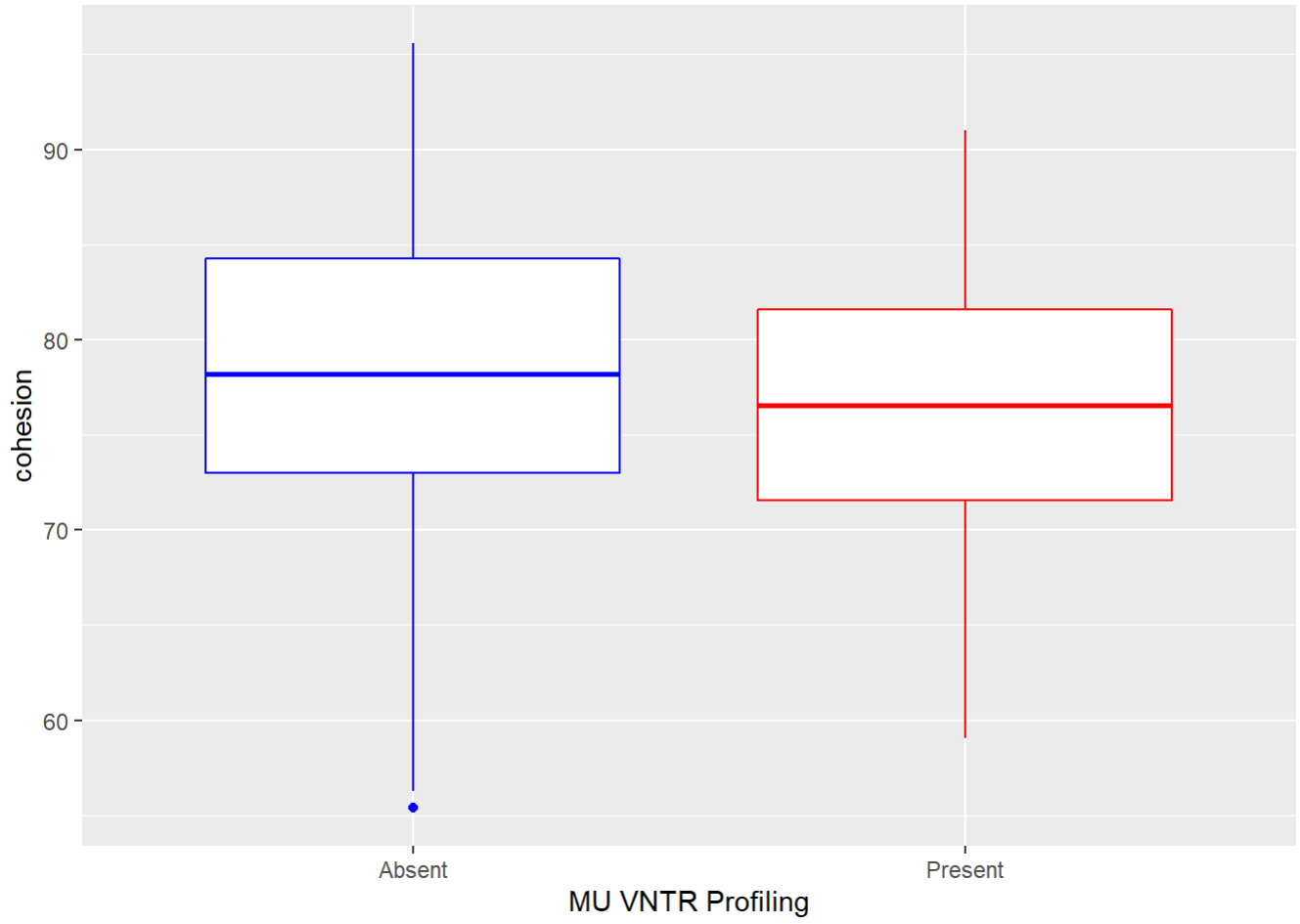

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.59306, df = 1, p-value = 0.4412
##
##   metric PresAbs  N     mean       sd       se
## 1 contag  Absent 62 28.62440 17.36018 2.204746
## 2 contag Present 36 25.19753 13.45430 2.242383
## [1] "np"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.24702, df = 1, p-value = 0.6192
##
##   metric PresAbs  N     mean       sd        se
## 1     np  Absent 62 5.516129 2.046524 0.2599088
## 2     np Present 36 5.583333 1.500000 0.2500000
## [1] "lsi"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.23018, df = 1, p-value = 0.6314
##
##   metric PresAbs  N     mean        sd         se
## 1    lsi  Absent 62 1.950806 0.3204111 0.04069225
## 2    lsi Present 36 2.002778 0.2937308 0.04895513
## [1] "cohesion"
```



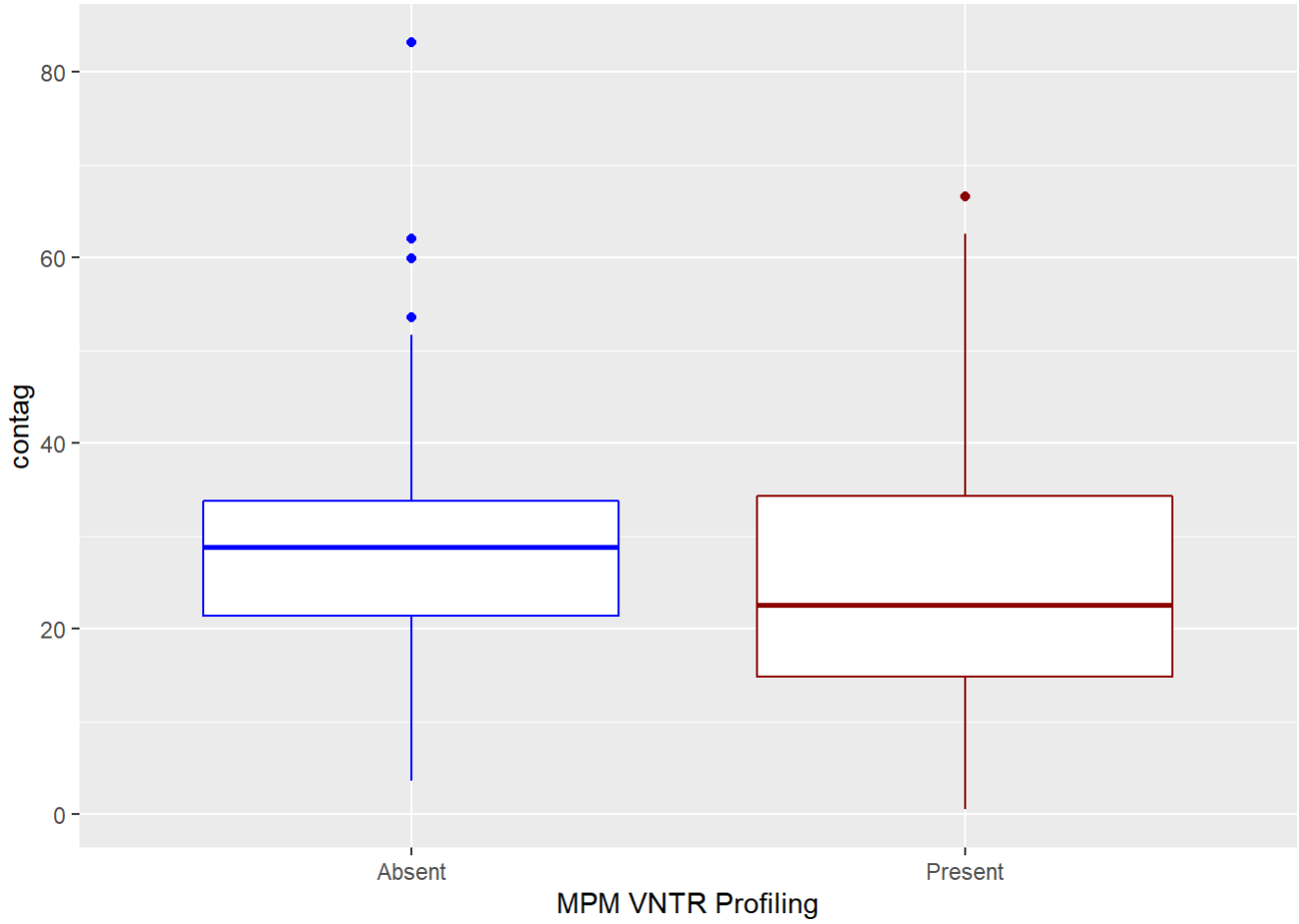

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.66312, df = 1, p-value = 0.4155
##
##     metric PresAbs  N     mean       sd       se
## 1 cohesion  Absent 62 77.65226 9.383922 1.191759
## 2 cohesion Present 36 76.31989 7.865909 1.310985
```

```
#LSI Landscape shape index #standardized measure of total edge LSI=1 =single patch increases fro
m there pg 158 Fragstats

#NP <- number of patches pg 149
#contagion -sum of proportional abundance of each patch, inverse to edge density pg154
```

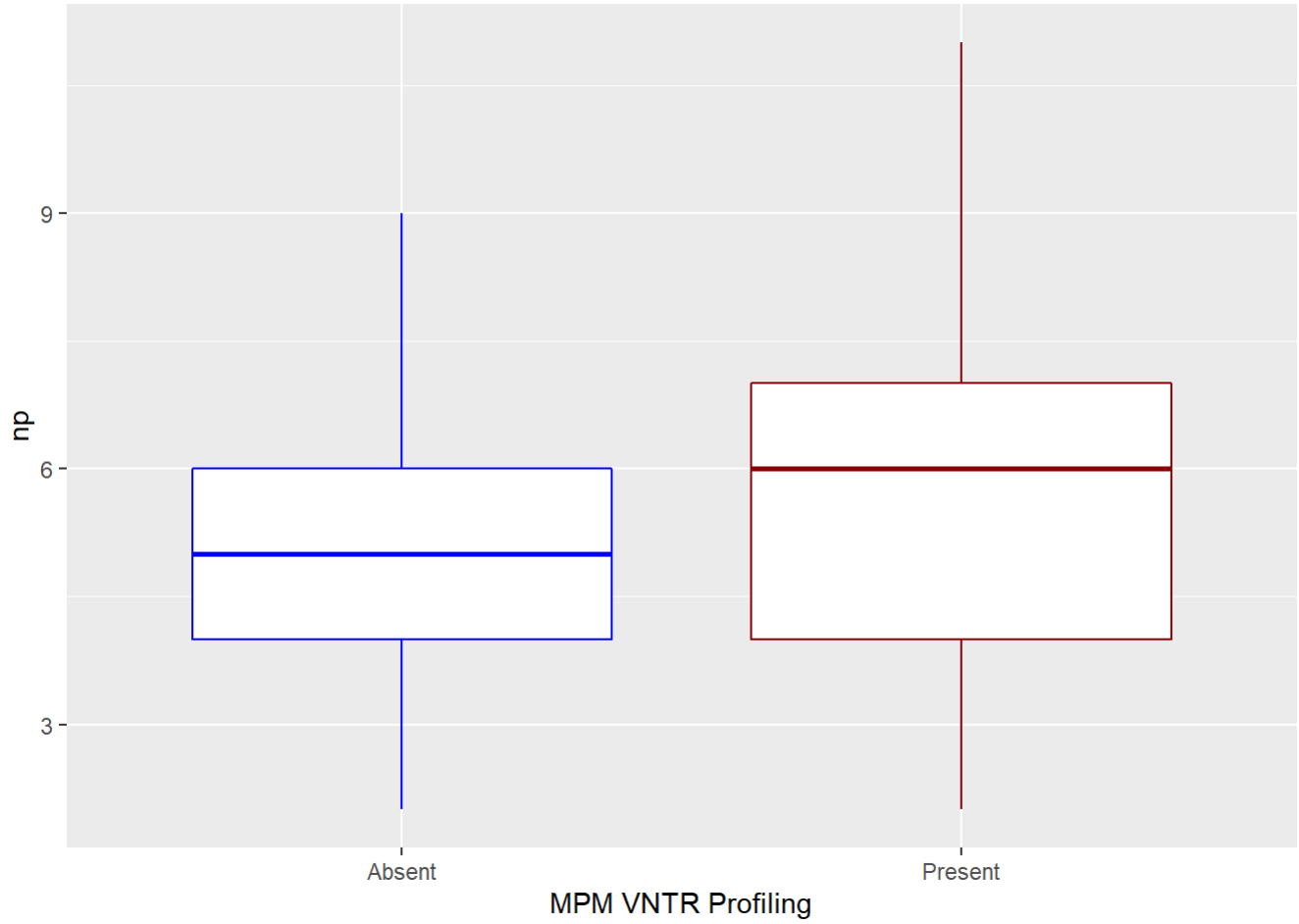
Landscape Metric MPM 5k

```
MetricList<-c("contag","np","lsi","cohesion")
#Same as above but using Total.ER.Present to calculate for MPM abundance
for(i in 1:length(MetricList)){
  Subset<-subset(Buffer5k,Buffer5k$metric==MetricList[i])
  print(unique(Subset$metric))



  Subset$plot_id<-MUGeo$USITEID
  Subset$PresAbs<-MUGeo$Total.ER.Present #Total ER presence coded as 1= presence, 0 =absence
  Subset$PresAbs[Subset$PresAbs==1]<-"Present"
  Subset$PresAbs[Subset$PresAbs==0]<-"Absent"
  Plot <- ggplot(Subset,aes( x=PresAbs,y=value,col=PresAbs))+ylab(MetricList[i])+xlab("MPM VNTR
 Profiling")+geom_boxplot()+scale_color_manual(values=c("blue","red4"))+theme(legend.position="n
one")
  print(Plot)
  print(kruskal.test(data=Subset,PresAbs~value))

}
```

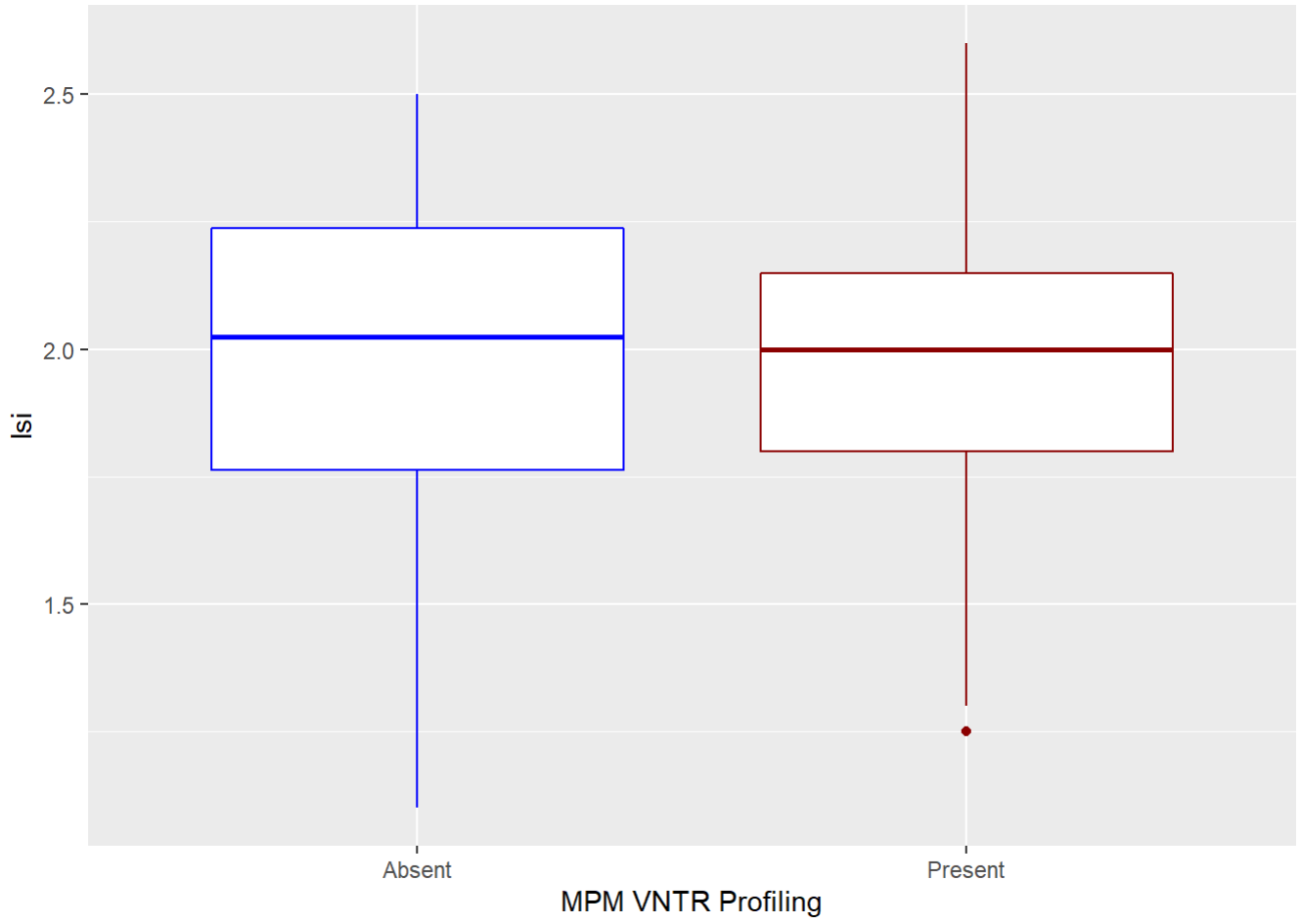
```
## [1] "contag"
```



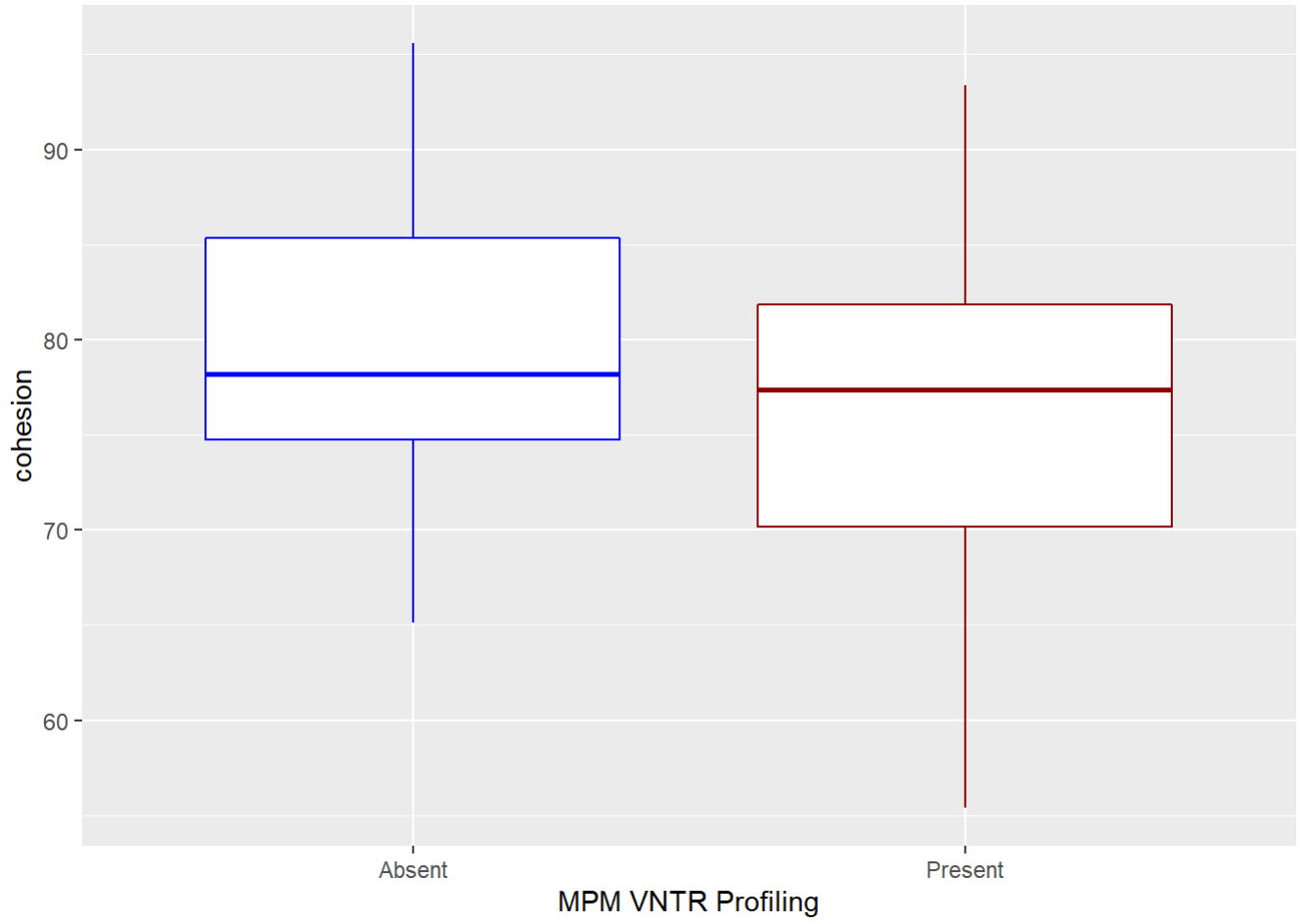
```
##
##  Kruskal-Wallis rank sum test
##
## data:  PresAbs by value
## Kruskal-Wallis chi-squared = 94.461, df = 92, p-value = 0.4095
##
## [1] "np"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  PresAbs by value
## Kruskal-Wallis chi-squared = 7.2582, df = 9, p-value = 0.6103
##
## [1] "lsi"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  PresAbs by value
## Kruskal-Wallis chi-squared = 25.4, df = 25, p-value = 0.4401
##
## [1] "cohesion"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  PresAbs by value
## Kruskal-Wallis chi-squared = 94.461, df = 92, p-value = 0.4095
```

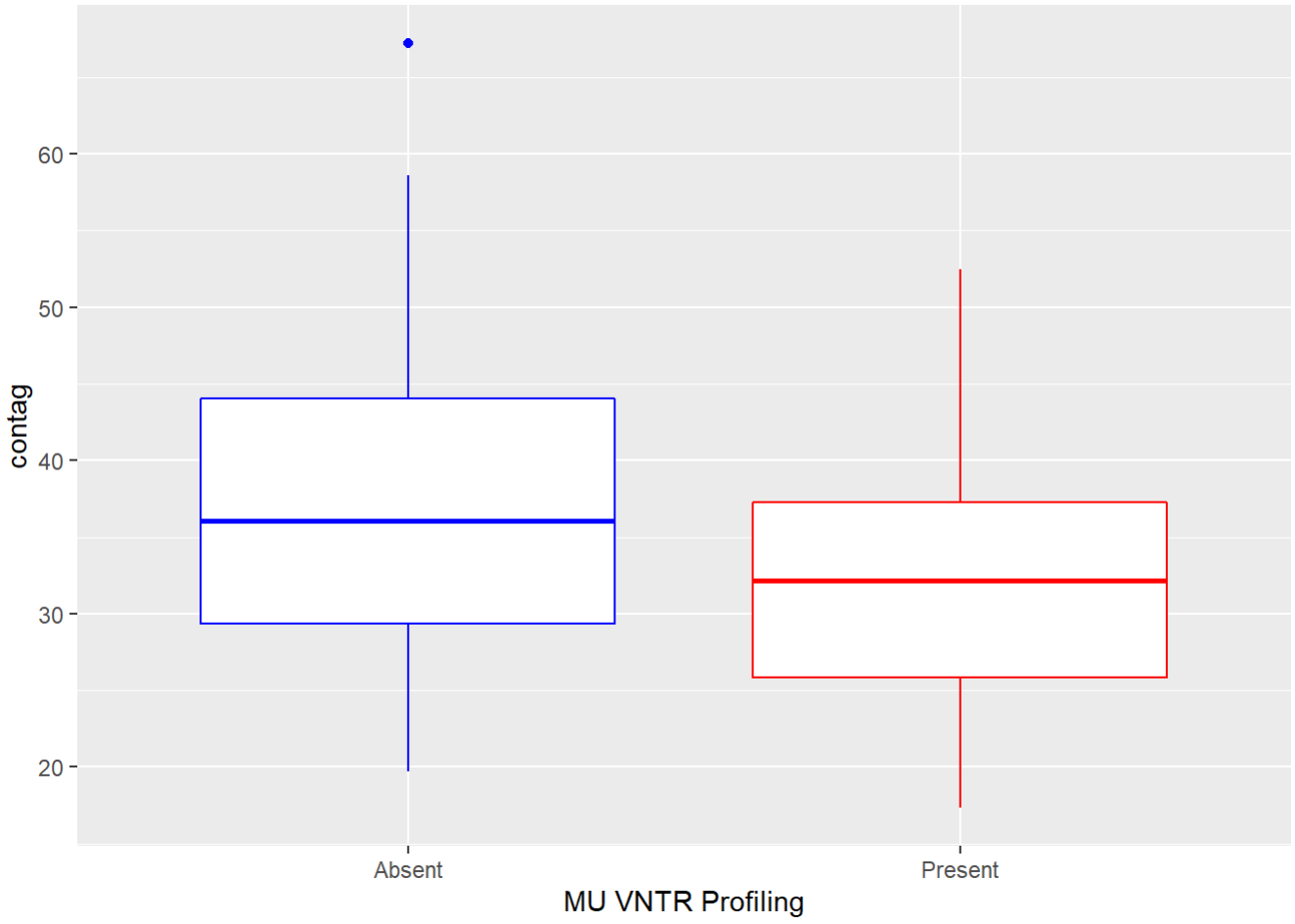
10k landscape metrics MU

```
#Data same as above
MUSurvey<-read.csv("GhanaMUSurvey.csv",header=T)

MUGeo <- subset(MUSurvey, !is.na(lon) & !is.na(lat))


#Project to WGS84
wgs1984.proj <- CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs")
MUGeo.proj <- SpatialPoints(coords=MUGeo[,c("lon","lat")], proj4string=wgs1984.proj)

LandCover<-LandCover2k
MUGeo.proj<-spTransform(MUGeo.proj,crs(LandCover))
GhanaOutline2<-spTransform(GhanaOutline,crs(LandCover))



#Calculate 10 k buffer (10000 m) around sample points for landscape metrics
Buffer10k<-sample_lsm(
  LandCover,
  MUGeo.proj,
  plot_id = NULL,
  shape = "square",
  size=10000,
  all_classes = FALSE,
  return_raster = FALSE,
  verbose = F,
  progress = FALSE,level="landscape",type="aggregation metric")

#Buffer10k
#unique(Buffer$metric)
```

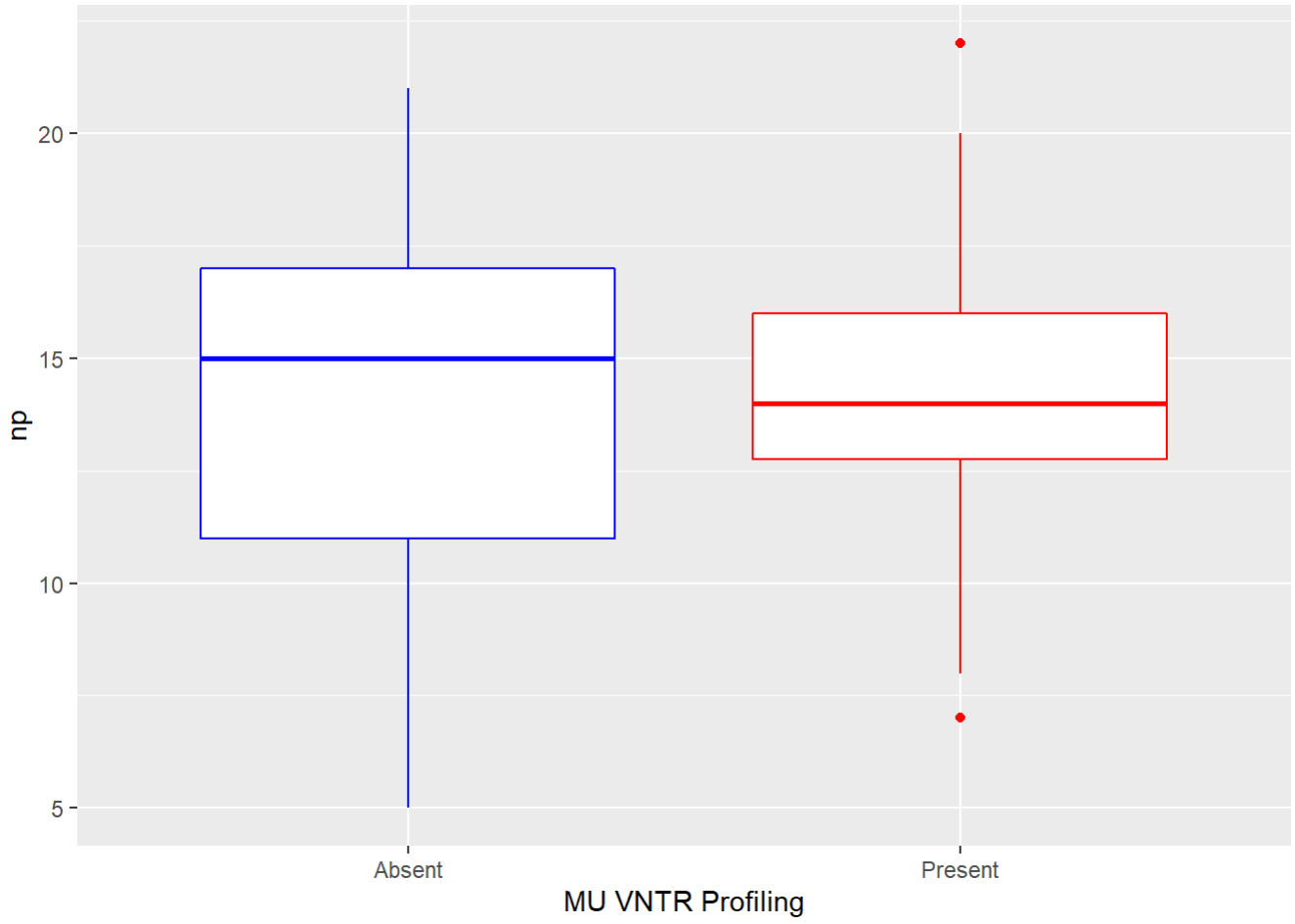
Landscape Metric MU 10 k

```
MetricList<-c("contag","np","lsi","cohesion")

for(i in 1:length(MetricList)){
  Subset<-subset(Buffer10k,Buffer10k$metric==MetricList[i])
  print(unique(Subset$metric))



  Subset$plot_id<-MUGeo$USITEID
  Subset$PresAbs<-MUGeo$Total.MU.Present..VNTR.
  Subset$PresAbs[Subset$PresAbs==1]<-"Present"
  Subset$PresAbs[Subset$PresAbs==0]<-"Absent"
  Plot <- ggplot(Subset,aes( x=PresAbs,y=value,col=PresAbs))+ylab(MetricList[i])+xlab("MU VNTR P
rofiling")+geom_boxplot()+scale_color_manual(values=c("blue","red"))+theme(legend.position="non
e")
  print(Plot)
  print(kruskal.test(data=Subset,value~PresAbs))
  Trtdata <- ddply(Subset, c("metric","PresAbs"), summarise,
                N    = length(value),
                mean = mean(value),
                sd   = sd(value),
                se   = sd / sqrt(N)
  )
  print(Trtdata)

}
```

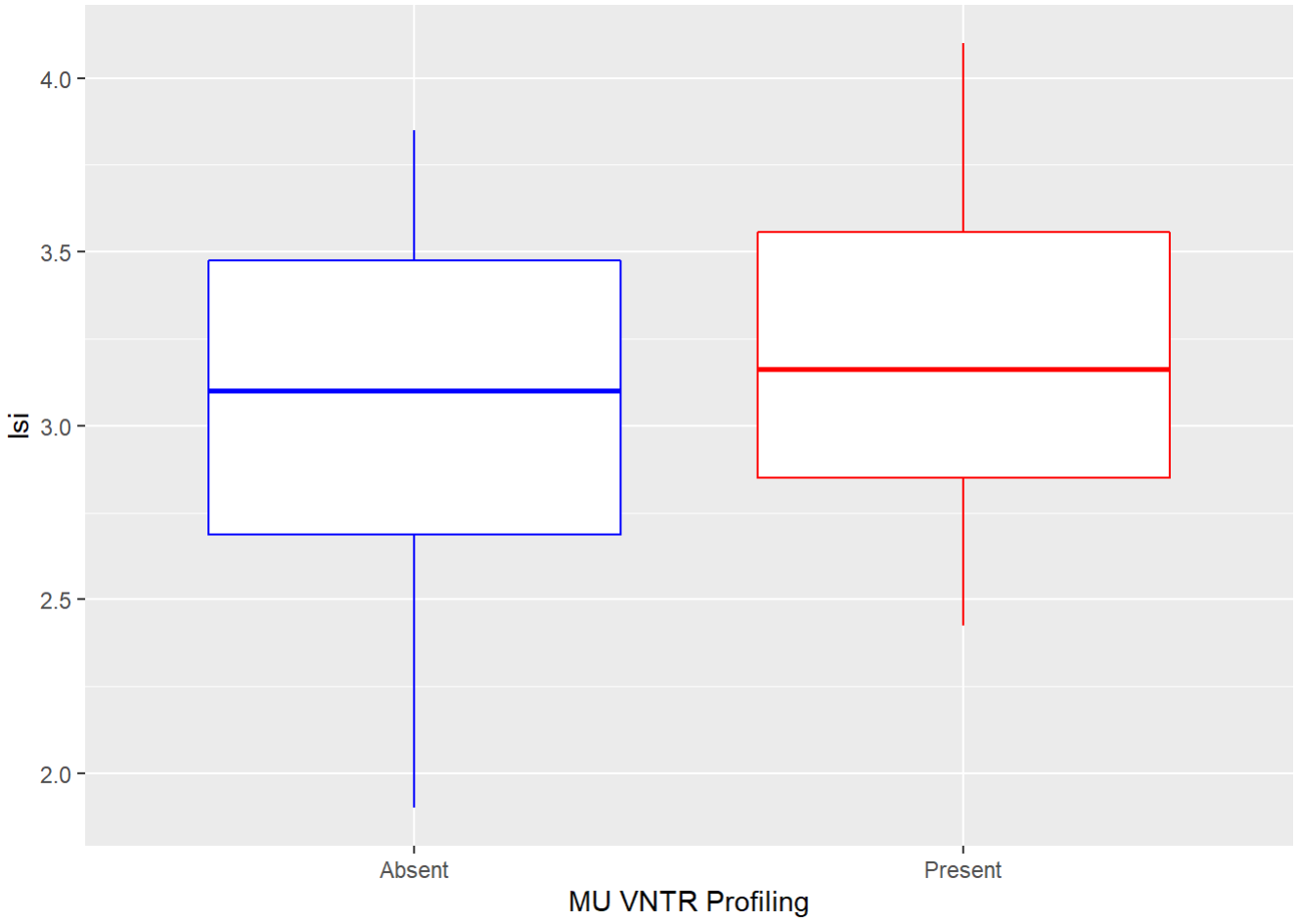
```
## [1] "contag"
```



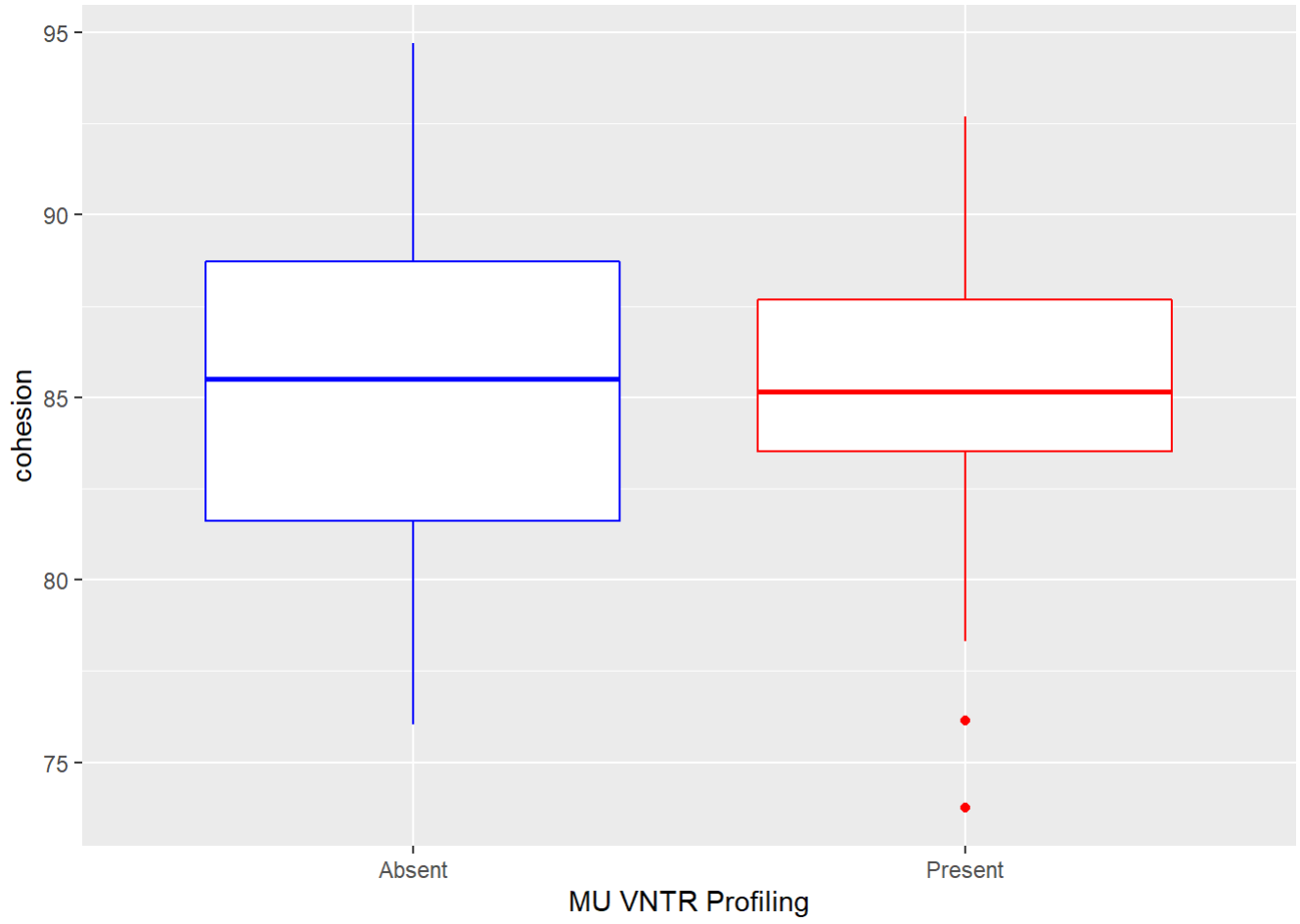

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 3.573, df = 1, p-value = 0.05873
##
##   metric PresAbs  N     mean        sd       se
## 1 contag  Absent 62 36.90062 10.520452 1.336099
## 2 contag Present 36 32.42555  8.259002 1.376500
## [1] "np"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.089664, df = 1, p-value = 0.7646
##
##   metric PresAbs  N     mean       sd        se
## 1     np  Absent 62 14.01613 4.162644 0.5286563
## 2     np Present 36 14.05556 3.380147 0.5633579
## [1] "lsi"
```

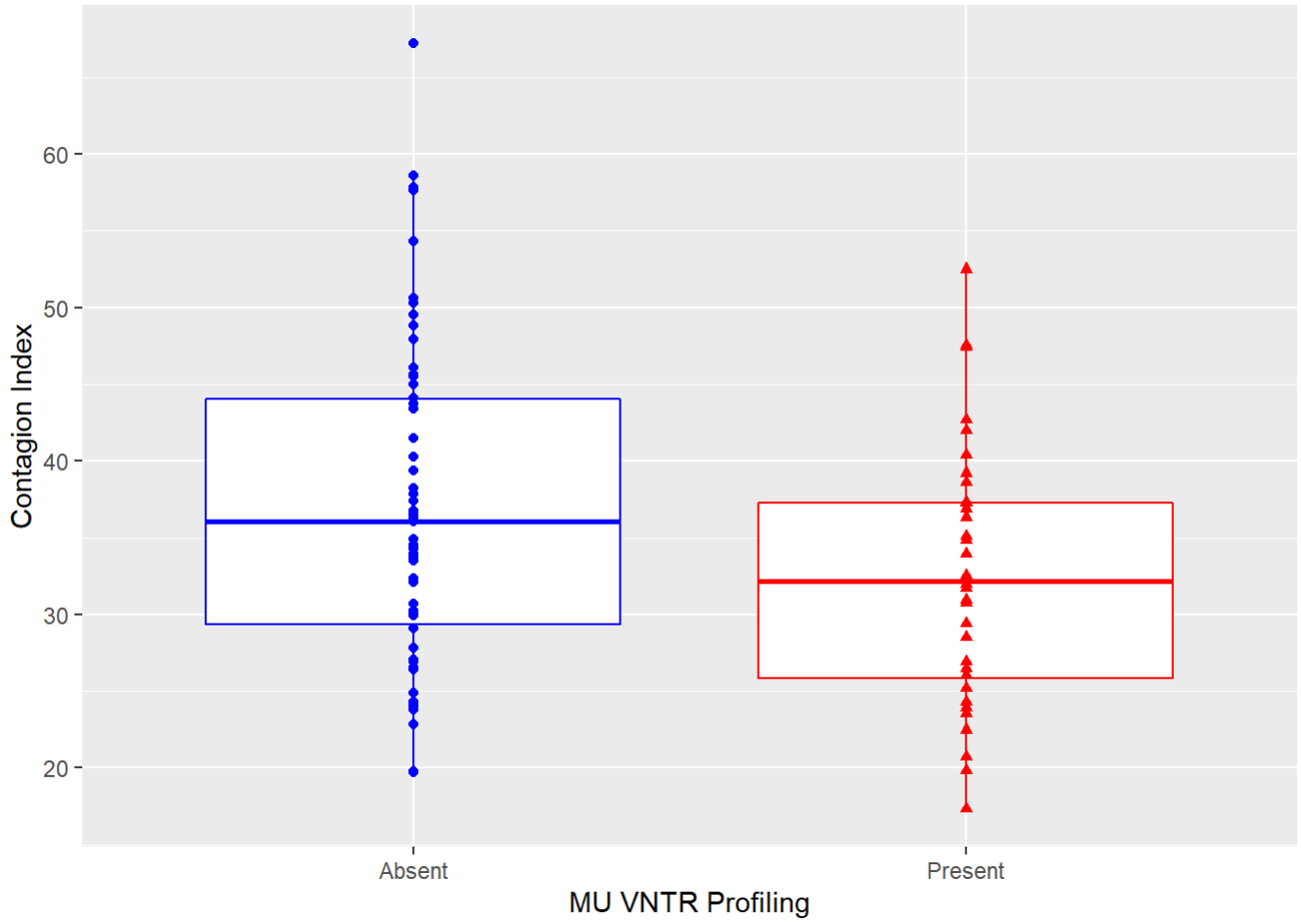



```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 1.4704, df = 1, p-value = 0.2253
##
##   metric PresAbs  N     mean        sd         se
## 1    lsi  Absent 62 3.056474 0.4991734 0.06339509
## 2    lsi Present 36 3.202778 0.3993199 0.06655331
## [1] "cohesion"
```




```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.0084855, df = 1, p-value = 0.9266
##
##     metric PresAbs  N     mean       sd        se
## 1 cohesion  Absent 62 85.24231 5.017880 0.6372714
## 2 cohesion Present 36 85.21881 3.980391 0.6633986
```

```
#LSI Landscape shape index #standardized measure of total edge LSI=1 =single patch increases fro
m there pg 158 Fragstats

#NP <- number of patches pg 149
#contagion -sum of proportional abundance of each patch, inverse to edge density pg154

#Subset for Contag only plot
Contag<-subset(Buffer10k,Buffer10k$metric=="contag")
Contag$plot_id<-MUGeo$USITEID
Contag$PresAbs<-MUGeo$Total.MU.Present..VNTR.
Contag$PresAbs[Contag$PresAbs==1]<-"Present"
Contag$PresAbs[Contag$PresAbs==0]<-"Absent"

Plot <- ggplot(Contag,aes( x=PresAbs,y=value,col=PresAbs,shape=PresAbs))+ylab(MetricList[i])+xla
b("MU VNTR Profiling")+geom_boxplot()+scale_color_manual(values=c("blue","red"))+theme(legend.po
sition="none")+ylab("Contagion Index")+geom_point()
Plot
```




```
#
# dev.off()
# tiff("output/ContagPlot.tiff", width = 84, height = 84, units = 'mm', res = 600)
# Plot
# dev.off()
```

Landscape Metric MPM Presence Absence

5 kilometer

```
#List of metrics to run kruskal.wallis tests on
MetricList<-c("contag","np","lsi","cohesion")

for(i in 1:length(MetricList)){
  Subset<-subset(Buffer5k,Buffer5k$metric==MetricList[i])
  print(unique(Subset$metric))



  Subset$plot_id<-MUGeo$USITEID #unique site ID from file
  Subset$PresAbs<-MUGeo$Total.ER.Present
  Subset$PresAbs[Subset$PresAbs==1]<-"Present"
  Subset$PresAbs[Subset$PresAbs==0]<-"Absent"
  Plot <- ggplot(Subset,aes( x=PresAbs,y=value,col=PresAbs))+ylab(MetricList[i])+xlab("MPM  Prof
iling")+geom_boxplot()+scale_color_manual(values=c("blue","red4"))+theme(legend.position="none")
  #print(Plot)
  print(kruskal.test(data=Subset,value~PresAbs))
  Trtdata <- ddply(Subset, c("metric","PresAbs"), summarise,
                N    = length(value),
                mean = mean(value),
                sd   = sd(value),
                se   = sd / sqrt(N)
  )
  print(Trtdata)

}
```

```
## [1] "contag"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 1.3708, df = 1, p-value = 0.2417
##
##   metric PresAbs  N     mean       sd       se
## 1 contag  Absent 26 30.81137 18.37608 3.603845
## 2 contag Present 72 26.12122 15.07018 1.776038
## [1] "np"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 1.9361, df = 1, p-value = 0.1641
##
##   metric PresAbs  N     mean       sd        se
## 1     np  Absent 26 5.076923 1.916728 0.3759013
## 2     np Present 72 5.708333 1.818818 0.2143497
## [1] "lsi"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.021046, df = 1, p-value = 0.8847
##
##   metric PresAbs  N     mean        sd         se
## 1    lsi  Absent 26 1.959615 0.3552518 0.06967061
## 2    lsi Present 72 1.973611 0.2951453 0.03478320
## [1] "cohesion"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 2.3746, df = 1, p-value = 0.1233
##
##     metric PresAbs  N     mean       sd       se
## 1 cohesion  Absent 26 79.98346 8.110159 1.590533
## 2 cohesion Present 72 76.14425 8.921870 1.051452
```

10k MPM landscape metrics

```
MetricList<-c("contag","np","lsi","cohesion")

for(i in 1:length(MetricList)){
  Subset<-subset(Buffer10k,Buffer5k$metric==MetricList[i])
  print(unique(Subset$metric))



  Subset$plot_id<-MUGeo$USITEID
  Subset$PresAbs<-MUGeo$Total.ER.Present
  Subset$PresAbs[Subset$PresAbs==1]<-"Present"
  Subset$PresAbs[Subset$PresAbs==0]<-"Absent"
  Plot <- ggplot(Subset,aes( x=PresAbs,y=value,col=PresAbs))+ylab(MetricList[i])+xlab("MPM  Prof
iling")+geom_boxplot()+scale_color_manual(values=c("blue","red4"))+theme(legend.position="none")
  #print(Plot)
  print(kruskal.test(data=Subset,value~PresAbs))
  Trtdata <- ddply(Subset, c("metric","PresAbs"), summarise,
                N    = length(value),
                mean = mean(value),
                sd   = sd(value),
                se   = sd / sqrt(N)
  )
  print(Trtdata)

}
```

```
## [1] "contag"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 2.8014, df = 1, p-value = 0.09418
##
##   metric PresAbs  N     mean       sd       se
## 1 contag  Absent 26 38.27226 10.35111 2.030019
## 2 contag Present 72 34.16776  9.63746 1.135786
## [1] "np"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.0041713, df = 1, p-value = 0.9485
##
##   metric PresAbs  N     mean       sd        se
## 1     np  Absent 26 13.92308 4.638302 0.9096459
## 2     np Present 72 14.06944 3.597050 0.4239165
## [1] "lsi"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 0.0010366, df = 1, p-value = 0.9743
##
##   metric PresAbs  N     mean        sd         se
## 1    lsi  Absent 26 3.070192 0.5688594 0.11156250
## 2    lsi Present 72 3.124672 0.4298932 0.05066339
## [1] "cohesion"
##
##  Kruskal-Wallis rank sum test
##
## data:  value by PresAbs
## Kruskal-Wallis chi-squared = 1.028, df = 1, p-value = 0.3106
##
##     metric PresAbs  N     mean       sd        se
## 1 cohesion  Absent 26 86.07564 4.424882 0.8677908
## 2 cohesion Present 72 84.92963 4.711217 0.5552222
```

Overview Plots (Figure 1a-d)

Figures 1a-b

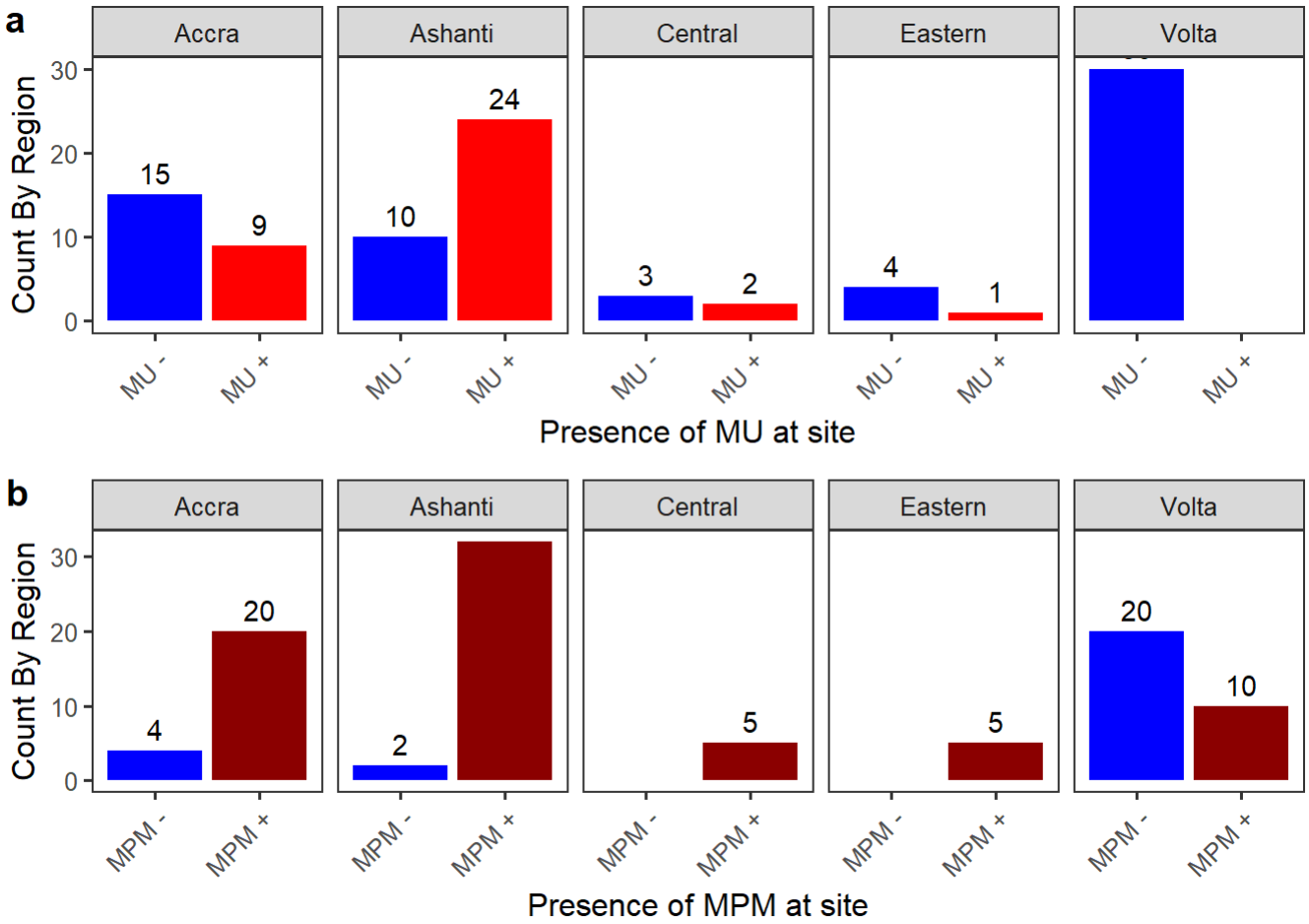
```
Overview<-MUGeo

#Total.MU.Present..VNTR. coded as 1 =pres,0=abs
Overview$PresAbsMU[Overview$Total.MU.Present..VNTR.==1]<-"MU +"
Overview$PresAbsMU[Overview$Total.MU.Present..VNTR.==0]<-"MU -"

Overview$PresAbsMPM[Overview$Total.ER.Present==1]<-"MPM +"
Overview$PresAbsMPM[Overview$Total.ER.Present==0]<-"MPM -"
theme_set(theme_bw(base_size = 12)+theme(panel.grid.major = element_blank(), panel.grid.minor =
 element_blank()))

#Count of MU by region
MUCount<-ggplot(Overview,aes( x=PresAbsMU,fill=PresAbsMU))+geom_histogram(stat="count")+facet_gr
id(~REGION)+scale_fill_manual(values=c('blue','red'))+ylab("Count By Region")+xlab("Presence of
 MU at site")+theme(legend.position = "none")+ theme(axis.text.x = element_text(angle = 45, hjus
t = 1))+stat_count(aes(y=..count.., label=..count..), geom="text", vjust=-.5)
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
#Count of MPM by region
MPMCount<-ggplot(Overview,aes( x=PresAbsMPM,fill=PresAbsMPM))+geom_histogram(stat="count")+facet
_grid(~REGION)+scale_fill_manual(values=c('blue','red4'))+ylab("Count By Region")+xlab("Presence
of MPM at site")+theme(legend.position = "none")+ theme(axis.text.x = element_text(angle = 45, h
just = 1))+stat_count(aes(y=..count.., label=..count..), geom="text", vjust=-.5)
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
ggarrange(MUCount,MPMCount,
          labels = c("a", "b"), ncol = 1,nrow=2)
```




```
# dev.off()
# tiff("output/MU.MPMCountByRegion.tiff", width = 174, height = 174, units = 'mm', res = 1200)
# ggarrange(MUCount,MPMCount,
#           labels = c("a", "b"), ncol = 1,nrow=2)
# dev.off()
```

Figures 1c-d

```
#Extract Land cover from locations where there is a sample point
Patch<-extract_lsm(LandCover, y = MUGeo.proj, what = "lsm_p_area")
```

```
## Warning: Only using 'what' argument.
```

```
unique(Patch$class)
```

```
## [1]  8  2  3 13 78 21 15  1
```

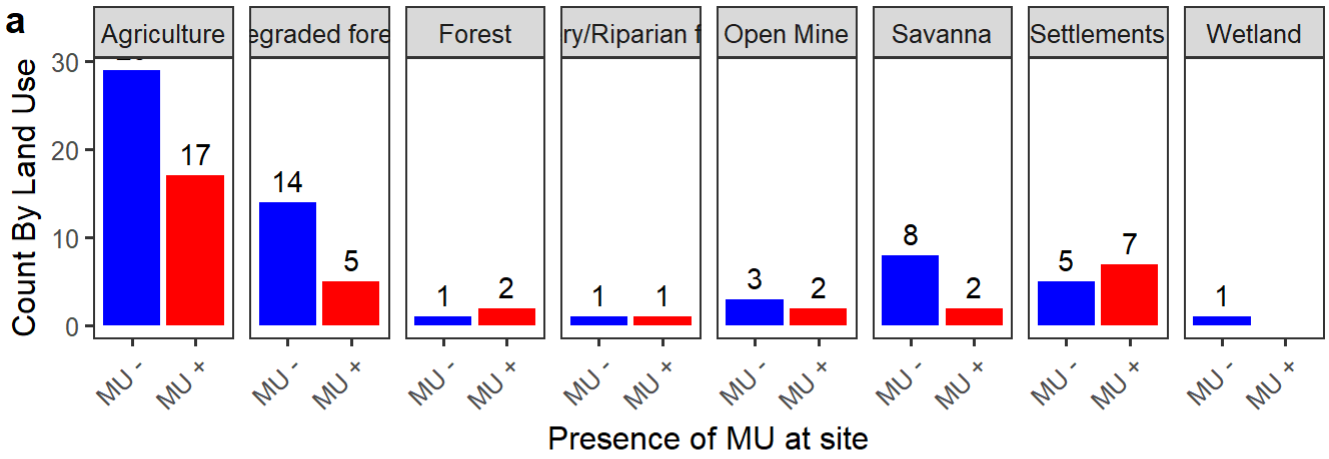
```
#Transform values to LU classes, list available at https://github.com/JPReceveur/SpatialEcologyFinalProj
Patch$class[Patch$class ==8]<-"Agriculture"
Patch$class[Patch$class ==2]<-"Savanna"
Patch$class[Patch$class ==3]<-"Wetland"
Patch$class[Patch$class ==13]<-"Settlements"
Patch$class[Patch$class ==21]<-"Degraded forest"
Patch$class[Patch$class ==1]<-"Forest"
Patch$class[Patch$class ==78]<-"Open Mine"
Patch$class[Patch$class ==15]<-"Gallery/Riparian forest"

Patch$extract_id<-MUGeo$ï..USITEID
Patch$MUPresAbs<-as.character(MUGeo$Total.MU.Present..VNTR.)
Patch$MUPresAbs[Patch$MUPresAbs==0]<-"MU -"
Patch$MUPresAbs[Patch$MUPresAbs==1]<-"MU +"

Patch$MPMPresAbs<-as.character(MUGeo$Total.ER.Present)
Patch$MPMPresAbs[Patch$MPMPresAbs==0]<-"MPM -"
Patch$MPMPresAbs[Patch$MPMPresAbs==1]<-"MPM +"

#MU presence by Land use
MUByLocation<-ggplot(Patch,aes( x= MUPresAbs,fill=MUPresAbs))+geom_histogram(stat="count")+facet_grid(~class)+scale_fill_manual(values=c('blue','red'))+ylab("Count By Land Use")+xlab("Presence of MU at site")+theme(legend.position = "none")+ theme(axis.text.x = element_text(angle = 45, hjust = 1))+stat_count(aes(y=..count.., label=..count..), geom="text", vjust=-.5)
```

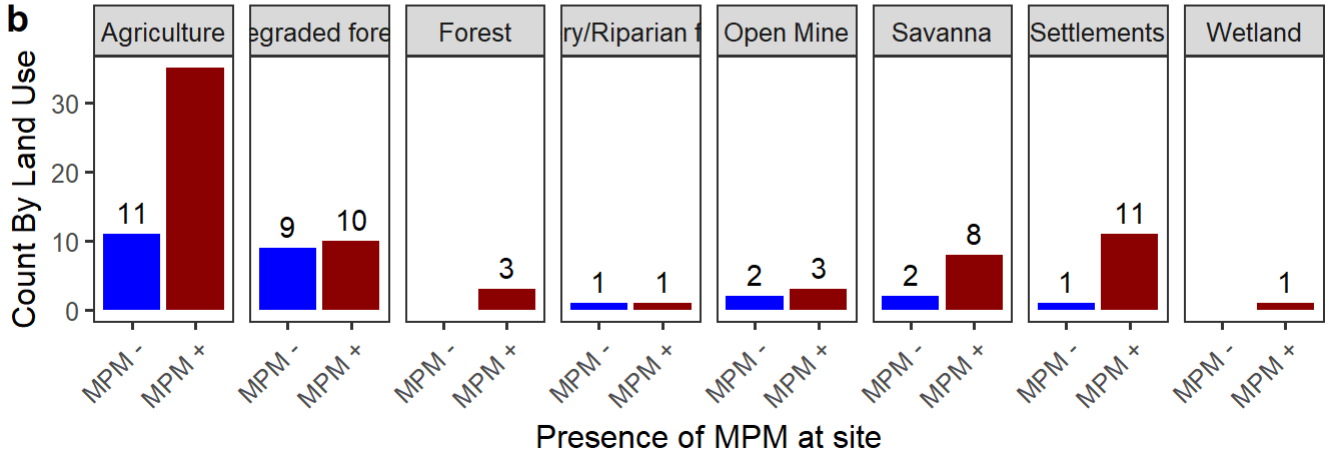
```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
#MPM pres/abs by Land use
MPMByLocation<-ggplot(Patch,aes( x= MPMPresAbs,fill=MPMPresAbs))+geom_histogram(stat="count")+facet_grid(~class)+scale_fill_manual(values=c('blue','red4'))+ylab("Count By Land Use")+xlab("Presence of MPM at site")+theme(legend.position = "none")+ theme(axis.text.x = element_text(angle = 45, hjust = 1))+stat_count(aes(y=..count.., label=..count..), geom="text", vjust=-.5)
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
ggarrange(MUByLocation,MPMByLocation,
          labels = c("a", "b"), ncol = 1,nrow=2)
```






```
#
# dev.off()
# tiff("output/PresenceByLU.tiff", width = 174, height = 174, units = 'mm', res = 600)
# ggarrange(MUByLocation,MPMByLocation,
#           labels = c("c", "d"), ncol = 1,nrow=2)
#
#
#
# dev.off()
```

Import Climate Data

```
#Climate Data downloaded
#if(! file.exists(file.path(data_path,'ClimateData.zip'))){
# download.file("https://biogeo.ucdavis.edu/data/worldclim/v2.1/base/wc2.1_30s_bio.zip", dest=fi
le.path(data_path,"ClimateData.zip"), mode="wb")
#}
#unzip (file.path(data_path,"ClimateData.zip"), exdir = data_path)
#Folder Climate Data subset contains
#wc2.1_30s_bio_1.tif Annual Mean Temperature
#wc2.1_30s_bio_12.tif Annual Precipitation
#wc2.1_30s_bio_15.tif Precipitation seasonality
#wc2.1_30s_bio_7.tif Temperature Annual range
#wc2.1_30s_bio_14.tif Precipitation of driest month

ClimateTif<-list.files(file.path(data_path,"ClimateDataSubset"), pattern='tif$', full.names=TRUE
)
ClimateSubset <- stack(ClimateTif)


ma.area <- extent(GhanaOutline)

ClimateSubsetGhana <- crop(ClimateSubset, ma.area)
GhanaClimateSubset.Meter<-projectRaster(ClimateSubsetGhana,crs=GhanaMeterGrid)
```

SDM Models

```
#Import survey data and seperate into pres absence data
MUSurvey<-read.csv("GhanaMUSurvey.csv",header=T)
MUGeo <- subset(MUSurvey, !is.na(lon) & !is.na(lat))
Presence<-subset(MUGeo,Total.MU.Present..VNTR.==1)
Presence$Total.MU.Present..VNTR.
```

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

```
Absence<-subset(MUGeo,Total.MU.Present..VNTR.==0)

wgs1984.proj <- CRS("+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs")
Presence.proj <- SpatialPoints(coords=Presence[,c("lon","lat")], proj4string=wgs1984.proj)
Absence.proj <- SpatialPoints(coords=Absence[,c("lon","lat")], proj4string=wgs1984.proj)


Presence.Meter<-spTransform(Presence.proj, GhanaMeterGrid)
Absence.Meter<-spTransform(Absence.proj, GhanaMeterGrid)


#MUGeo.Meter


presvals <- extract(GhanaClimateSubset.Meter, Presence.Meter)
absvals <- extract(GhanaClimateSubset.Meter, Absence.Meter)
pb <- c(rep(1, nrow(presvals)), rep(0, nrow(absvals)))
sdmdata <- data.frame(cbind(pb, rbind(presvals, absvals)))
#head(sdmdata)

#pairs(sdmdata[,2:5], cex=0.1)



PresLat<-Presence.Meter$lat
PresLong<-Presence.Meter$lon

PresLongLat<-data.frame(PresLong,PresLat)


pred_nf<-GhanaClimateSubset.Meter

#See dismo vignette
set.seed(0)
group <- kfold(PresLongLat, 5)
pres_train <- PresLongLat[group != 1, ]
pres_test <- PresLongLat[group == 1, ]

ext<-extent(GhanaOutline.Meter)

AbsLat<-Absence.Meter$lat
AbsLong<-Absence.Meter$lon

AbsLongLat<-data.frame(AbsLong,AbsLat)


pred_nf<-GhanaClimateSubset.Meter


set.seed(0)
group <- kfold(AbsLongLat, 5)
#group
abs_train <- AbsLongLat[group != 1, ]
```

```
abs_test <- AbsLongLat[group == 1, ]


r <- raster(pred_nf, 1)
# plot(!is.na(r), col=c('white', 'light grey'), legend=FALSE)
# plot(ext, add=TRUE, col='red', lwd=2)
# points(abs_train, pch='-', cex=0.5, col='black')
# points(abs_test, pch='-',  cex=0.5, col='black')
# points(pres_train, pch= '+', col='green')
# points(pres_test, pch='+', col='blue')
```

Bioclim model MU pres/abs

```
bc<-bioclim(pred_nf,pres_train) #pres_train

#plot(bc, a=1,b=2,p=0.85)
```

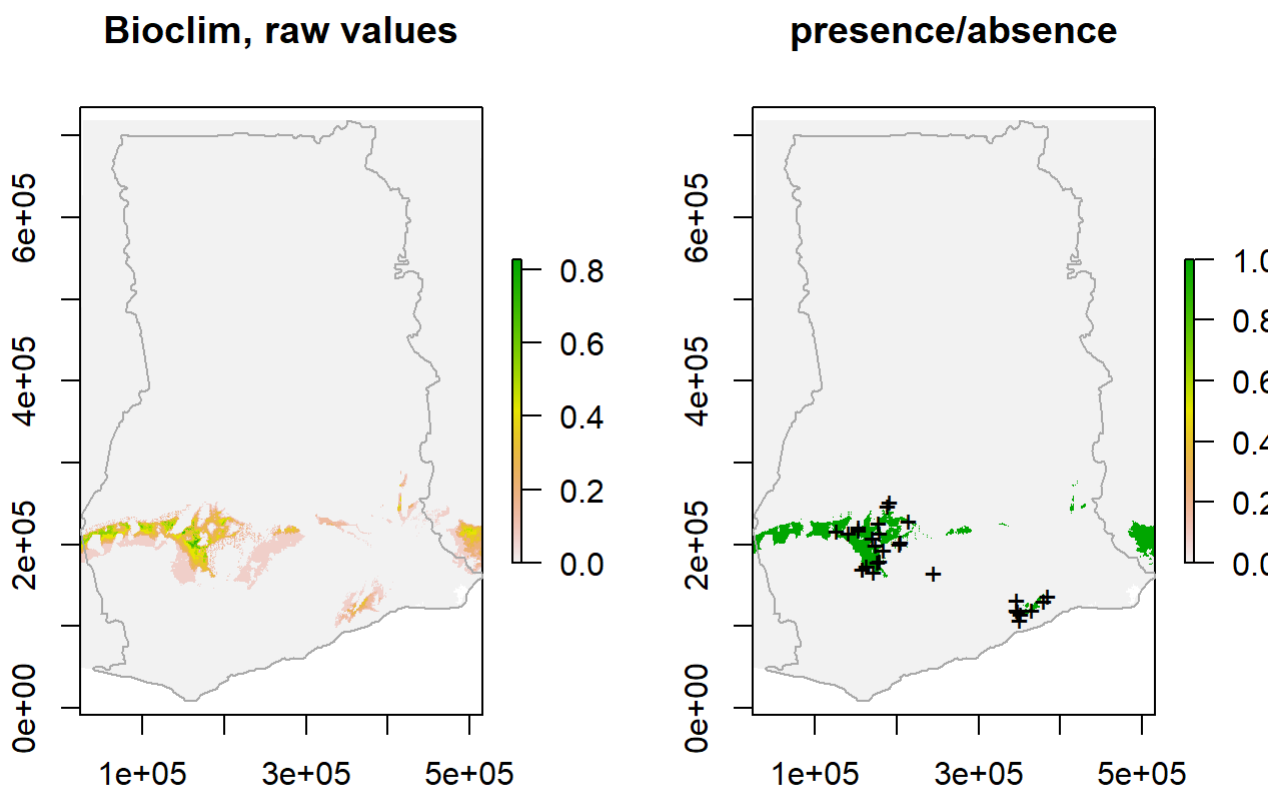
```
e<-evaluate(pres_test,abs_test,bc,pred_nf)
e
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.4821429
## cor            : 0.03370387
## max TPR+TNR at : 0.2067966
```

```
tr<-threshold(e,'spec_sens')
#tr

pb<-predict(pred_nf,bc,ext=ext,progress='')
pb
```

```
## class      : RasterLayer
## dimensions : 772, 536, 413792  (nrow, ncol, ncell)
## resolution : 919, 921  (x, y)
## extent     : 23762.34, 516346.3, 7649.294, 718661.3  (xmin, xmax, ymin, ymax)
## crs        : +proj=tmerc +lat_0=4.66666666666667 +lon_0=-1 +k=0.99975 +x_0=274319.51 +y_0=0 +
ellps=clrk80 +units=m +no_defs
## source     : memory
## names      : layer
## values     : 0, 0.8275862  (min, max)
```

```
par(mfrow=c(1,2))
plot(pb, main='Bioclim, raw values')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
plot(pb > tr, main='presence/absence')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
points(pres_train, pch='+')
```

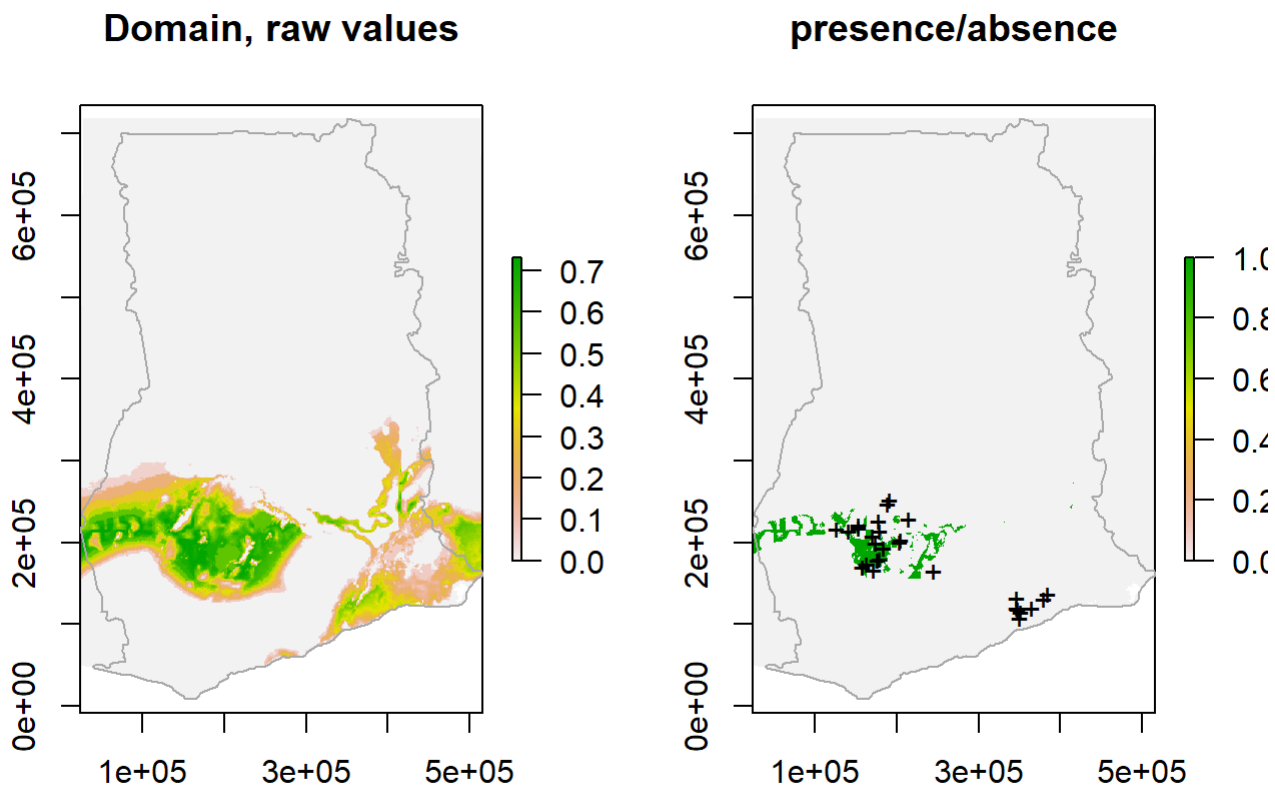


Domain Model

```
dm <- domain(pred_nf, pres_train)
e <- evaluate(pres_test, abs_test, dm, pred_nf)
e
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.6309524
## cor            : 0.2053091
## max TPR+TNR at : 0.6593678
```

```
pd = predict(pred_nf, dm, ext=ext, progress='')
par(mfrow=c(1,2))
plot(pd, main='Domain, raw values')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
tr <- threshold(e, 'spec_sens')
plot(pd > tr, main='presence/absence')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
points(pres_train, pch='+')
```




```
predictors<-GhanaClimateSubset.Meter
colnames(pres_train)<-colnames(abs_train)
train <- rbind(pres_train, abs_train)


pb_train <- c(rep(1, nrow(pres_train)), rep(0, nrow(abs_train)))
envtrain <- extract(predictors, train)
envtrain <- data.frame( cbind(pa=pb_train, envtrain) )
#head(envtrain)

testpres <- data.frame( extract(predictors, pres_test) )
testabs <- data.frame( extract(predictors, abs_test) )
```

GLM Model MU pres/abs

```
# logistic regression:
#names(envtrain)
gm1 <- glm(pa ~ wc2.1_30s_bio_1+wc2.1_30s_bio_12+wc2.1_30s_bio_15+wc2.1_30s_bio_7+wc2.1_30s_bio_
14, family = binomial(link = "logit"), data=envtrain)

summary(gm1)
```

```
##
## Call:
## glm(formula = pa ~ wc2.1_30s_bio_1 + wc2.1_30s_bio_12 + wc2.1_30s_bio_15 +
##     wc2.1_30s_bio_7 + wc2.1_30s_bio_14, family = binomial(link = "logit"),
##     data = envtrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5562  -0.7862  -0.2291   0.5973   2.1629
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      95.998640  33.289258   2.884  0.00393 **
## wc2.1_30s_bio_1  -2.566436   1.163980  -2.205  0.02746 *
## wc2.1_30s_bio_12 -0.006619   0.005202  -1.272  0.20324
## wc2.1_30s_bio_15 -0.111485   0.161049  -0.692  0.48878
## wc2.1_30s_bio_7  -0.782073   0.673061  -1.162  0.24525
## wc2.1_30s_bio_14 -0.301353   0.158132  -1.906  0.05669 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 103.867  on 78  degrees of freedom
## Residual deviance:  70.846  on 73  degrees of freedom
## AIC: 82.846
##
## Number of Fisher Scoring iterations: 5
```

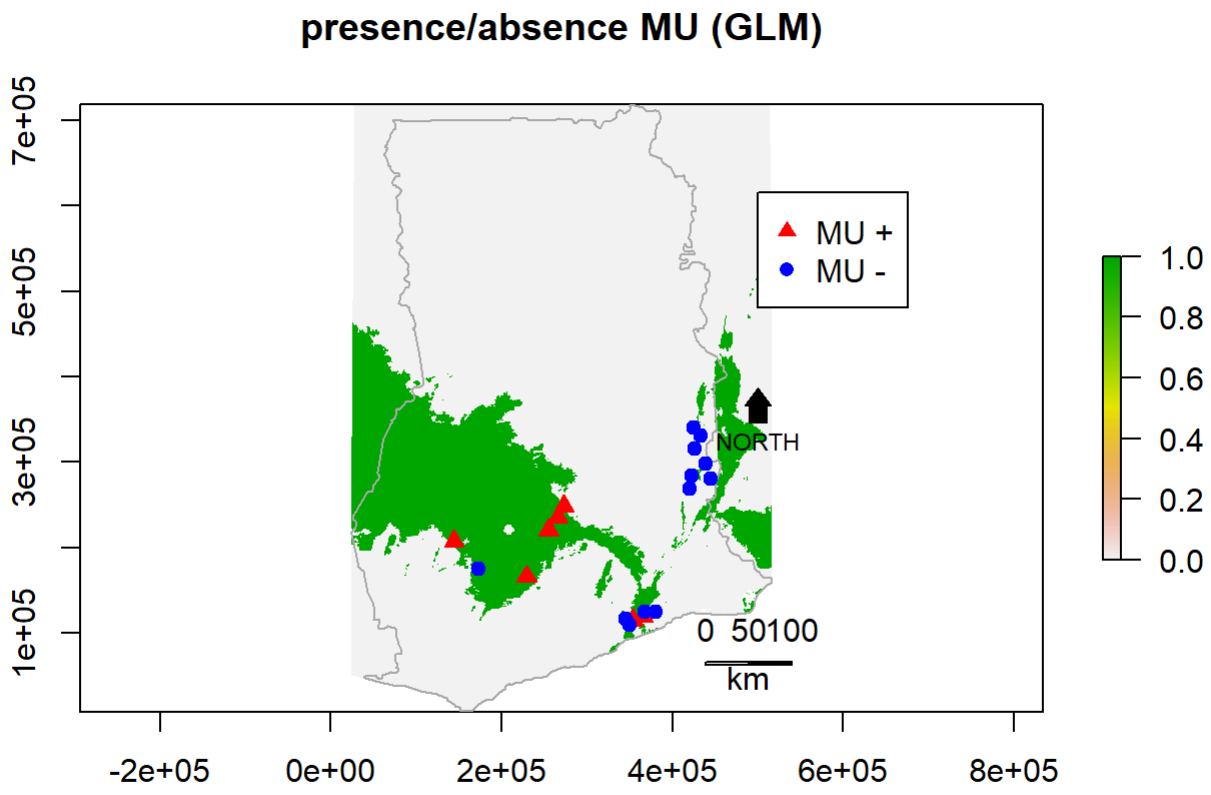
```
#coef(gm1)
gm2 <- glm(pa ~ wc2.1_30s_bio_1+wc2.1_30s_bio_12+wc2.1_30s_bio_15+wc2.1_30s_bio_7+wc2.1_30s_bio_
14,
           family = gaussian(link = "identity"), data=envtrain)
evaluate(testpres,testabs,gm1)
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.9047619
## cor            : 0.6714574
## max TPR+TNR at : -0.177097
```

```
ge2 <- evaluate(testpres, testabs, gm1)
ge2
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.9047619
## cor            : 0.6714574
## max TPR+TNR at : -0.177097
```

```
pg <- predict(predictors, gm1, ext=ext)

#plot(pg, main='GLM/binomial, raw values')
#plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
tr <- threshold(ge2, 'spec_sens')

plot(pg > tr, main='presence/absence MU (GLM)')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')

legend(x=500183,y=615115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
points(pres_test,col="red",bg="red",pch=24)
points(abs_test,col="blue",bg="blue",pch=21)
```



```
#
#
# dev.off()
# tiff("output/GLMSDMWTestSet.tiff", width = 174, height = 174, units = 'mm', res = 1200)
# plot(pg > tr, main='presence/absence MU (GLM)')
# plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
# legend(x=500183,y=615115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
# scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
# north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
# points(pres_test,col="red",bg="red",pch=24)
# points(abs_test,col="blue",bg="blue",pch=21)
# dev.off()
```

Random Forest SDM Model MU

```
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

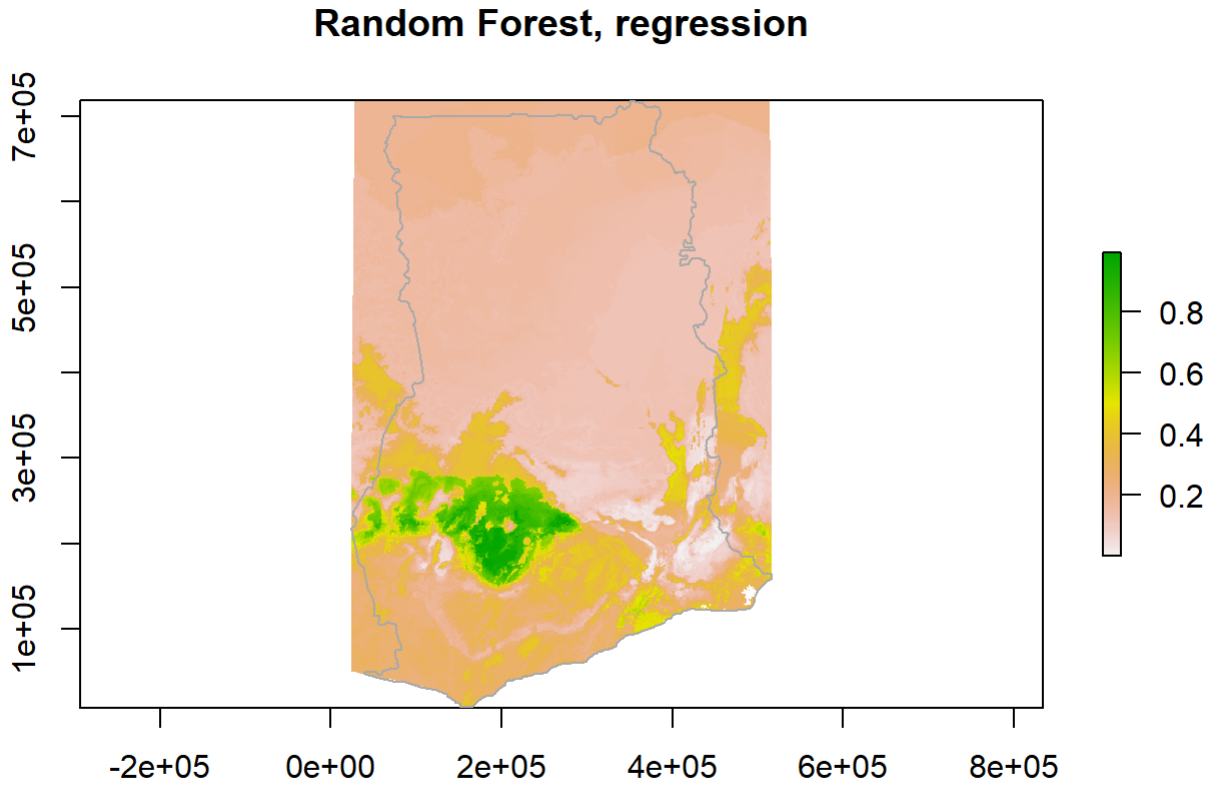
```
model <- pa ~ wc2.1_30s_bio_1+wc2.1_30s_bio_12+wc2.1_30s_bio_15+wc2.1_30s_bio_7+wc2.1_30s_bio_14
rf1 <- randomForest(model, data=envtrain)
```

```
## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values. Are you sure you want to do regression?
```

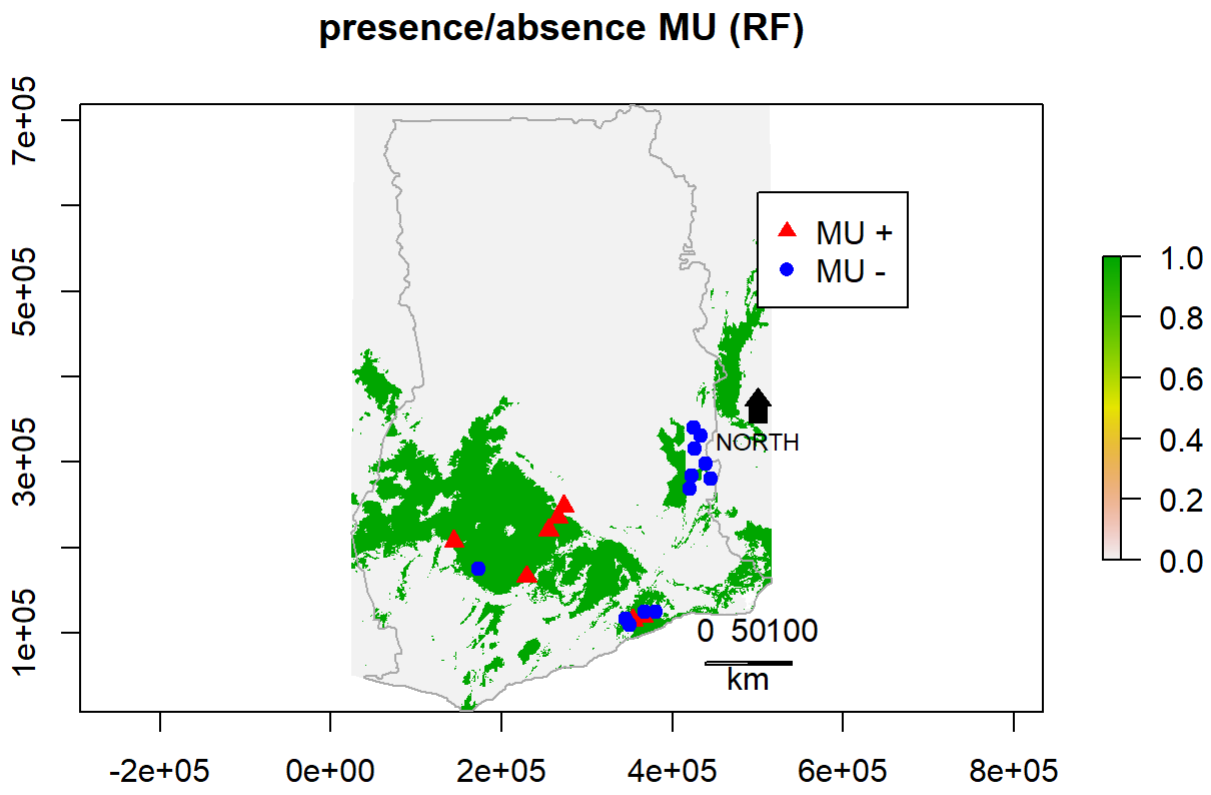
```
model <- factor(pa) ~ wc2.1_30s_bio_1+wc2.1_30s_bio_12+wc2.1_30s_bio_15+wc2.1_30s_bio_7+wc2.1_30
s_bio_14
rf2 <- randomForest(model, data=envtrain)
erf <- evaluate(testpres, testabs, rf1)
erf
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.797619
## cor            : 0.539484
## max TPR+TNR at : 0.3489333
```

```
pr <- predict(predictors, rf1, ext=ext)
plot(pr, main='Random Forest, regression')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
```




```
tr <- threshold(erf, 'spec_sens')
plot(pr > tr, main='presence/absence MU (RF)')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
legend(x=500183,y=615115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
points(pres_test,col="red",bg="red",pch=24)
points(abs_test,col="blue",bg="blue",pch=21)
```




```
importance(rf2)
```

```
##                  MeanDecreaseGini
## wc2.1_30s_bio_1          9.945281
## wc2.1_30s_bio_12         4.672312
## wc2.1_30s_bio_15         4.792521
## wc2.1_30s_bio_7          8.617284
## wc2.1_30s_bio_14         7.550353
```

```
# dev.off()
# tiff("output/rfSDMWTestSet.tiff", width = 174, height = 174, units = 'mm', res = 1200)
# plot(pr > tr, main='presence/absence MU (RF)')
# plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
# legend(x=500183,y=615115,legend =c("MU +","MU -"),pch=c(17,16),col=c('red','blue'))
# scalebar(100000,xy = c(439429.8,61762), type="bar", divs = 2, label = c(0,50,100),below="km")
# north.arrow(x=500183, y=345827, len = 10000, cex.lab = 0.75, col = "black", fg = "red")
# points(pres_test,col="red",bg="red",pch=24)
# points(abs_test,col="blue",bg="blue",pch=21)
#
# dev.off()
```

Support Vector machines

```
library(kernlab)
```

```
## Warning: package 'kernlab' was built under R version 4.0.3
```

```
##
## Attaching package: 'kernlab'
```

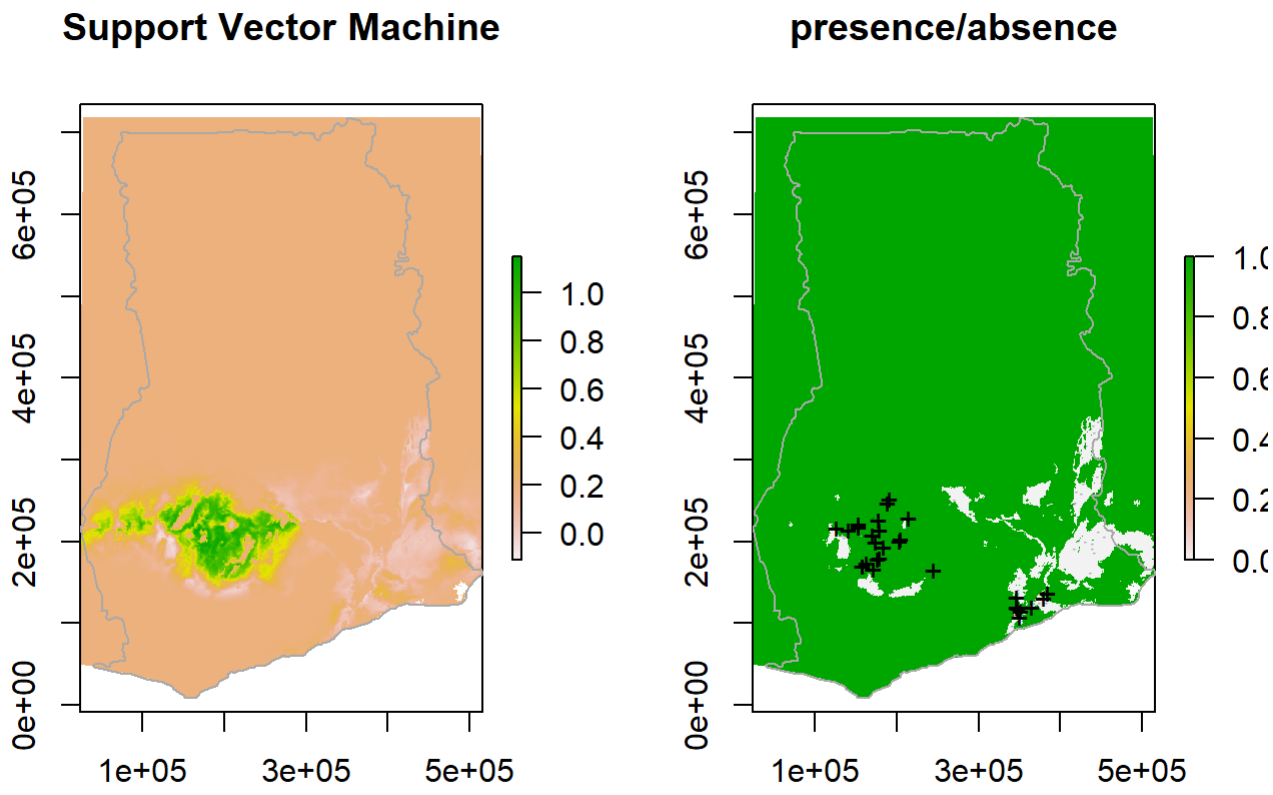
```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
## The following objects are masked from 'package:raster':
##
##     buffer, rotated
```

```
svm <- ksvm(pa ~ wc2.1_30s_bio_1+wc2.1_30s_bio_12+wc2.1_30s_bio_15+wc2.1_30s_bio_7+wc2.1_30s_bio
_14, data=envtrain)

esv <- evaluate(testpres, testabs, svm)
esv
```

```
## class          : ModelEvaluation
## n presences    : 7
## n absences     : 12
## AUC            : 0.8333333
## cor            : 0.4623423
## max TPR+TNR at : 0.1054004
```

```
ps <- predict(predictors, svm, ext=ext)
par(mfrow=c(1,2))
plot(ps, main='Support Vector Machine')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
tr <- threshold(esv, 'spec_sens')
plot(ps > tr, main='presence/absence')
plot(GhanaOutline.Meter, add=TRUE, border='dark grey')
points(pres_train, pch='+')
points(abs_train, pch='-', cex=0.25)
```



Combining model predictions (Final two models (rf,glm) shown seperately and not combined)

```
# Final
# models <- stack(pg, pr )
# names(models) <- c("glm", "rf")
# plot(models)
#
# m <- mean(models)
# plot(m, main='average score')
#
# auc <- sapply(list(ge2, erf), function(x) x@auc)
# w <- (auc-0.5)^2
# m2 <- weighted.mean( models[[c("glm", "rf")]], w)
# plot(m2)
# plot(GhanaOutline.Meter,add=T)
# plot(Presence.Meter,col="red",bg="red",add=T,pch=24)
# plot(Absence.Meter,col="blue",bg="blue",add=T,pch=21)
```

Correlograms MU/MPM presence absence by distance (Figure 2b)

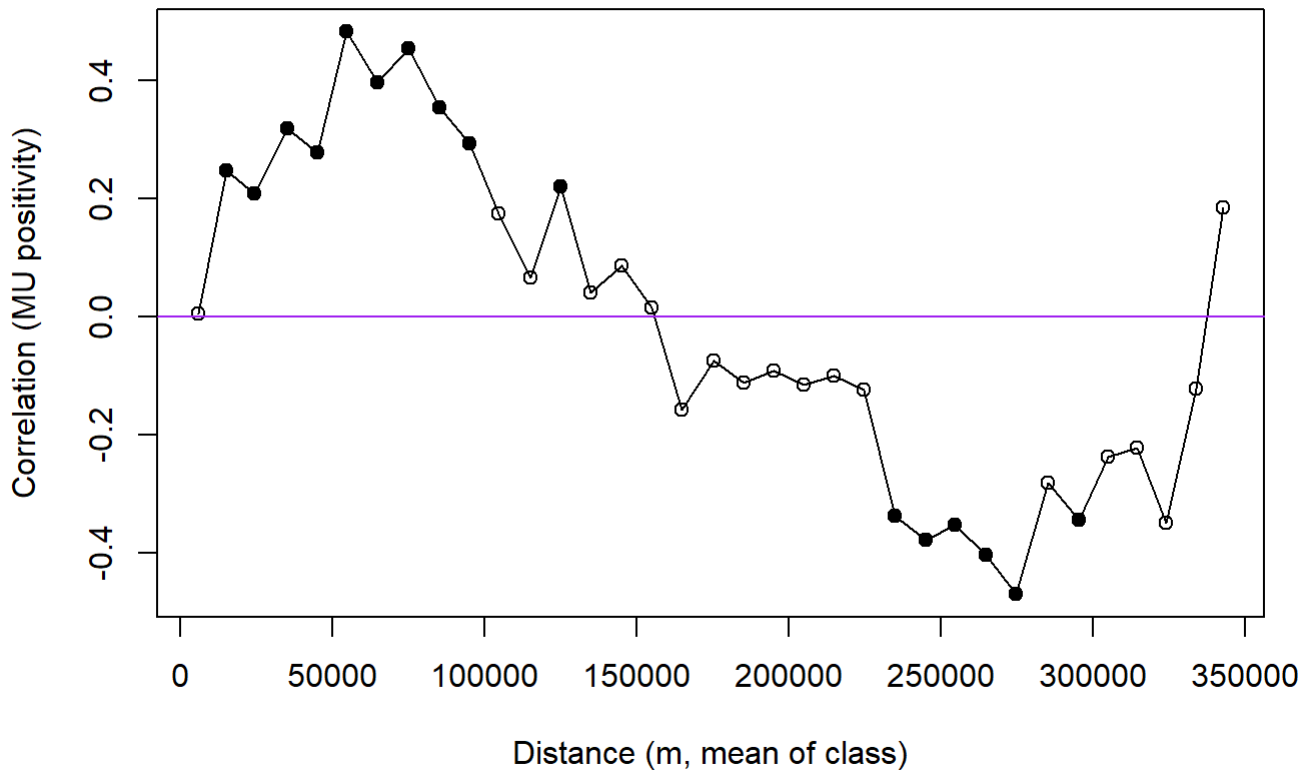
```
library(ncf)
```

```
## Warning: package 'ncf' was built under R version 4.0.3
```

```
#Not used Moran I assumption data is normal
#
# qqnorm(MUGeo$Total.ER.Present)
# qqline(MUGeo$Total.ER.Present,col="blue")
#
#

cor.MUPos <- correlog(MUGeo.proj$lat, MUGeo.proj$lon, MUGeo$Total.MU.Present..VNTR., increment =
10000, resamp = 100)
```

```
## 10  of  100
20  of  100
30  of  100
40  of  100
50  of  100
60  of  100
70  of  100
80  of  100
90  of  100
100  of  100
```

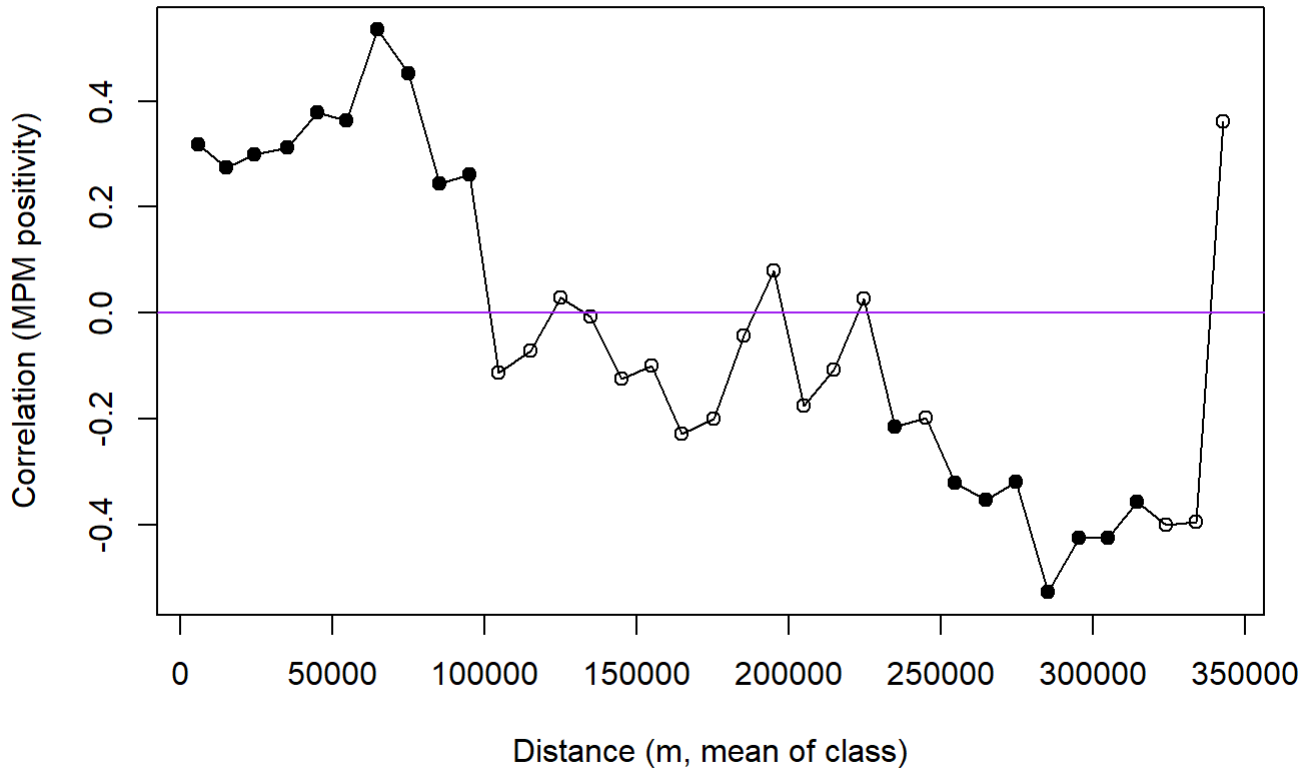
```
plot(cor.MUPos,xlab= "Distance (m, mean of class)",ylab="Correlation (MU positivity)",main="")
abline(h = 0, col = "purple")
```




```
#
# dev.off()
# tiff("output/CorrelogramMUPresence.tiff", width = 84, height = 84, units = 'mm', res = 1200)
# plot(cor.MUPos,xlab= "Distance (m, mean of class)",ylab="Correlation (MU positivity)",main="")
# abline(h = 0, col = "purple")
#
# dev.off()


cor.MPMPos <- correlog(MUGeo.proj$lat, MUGeo.proj$lon, MUGeo$Total.ER.Present, increment = 10000
, resamp = 100)
```

```
## 10  of  100
20  of  100
30  of  100
40  of  100
50  of  100
60  of  100
70  of  100
80  of  100
90  of  100
100  of  100
```

```
plot(cor.MPMPos,xlab= "Distance (m, mean of class)",ylab="Correlation (MPM positivity)",main="")
abline(h = 0, col = "purple")
```




```
#
# dev.off()
# tiff("output/CorelogramMPMPresence.tiff", width = 84, height = 84, units = 'mm', res = 1200)
# plot(cor.MPMPos,xlab= "Distance (m, mean of class)",ylab="Correlation (MPM positivity)",main
="")
# abline(h = 0, col = "purple")
#
# dev.off()
```