

## #Table of Contents-----

```
# BH-01 starts on line 47
# BL-02 starts on line 192
# BH-02 start on line ...
```

```
#-----
```

```
#Set working directory to ensure R can find the files we wish to import.
setwd("C:/Users/aalda/Desktop/All plots 2018")
```

```
#Installing and Loading all packages-----
```

```
#install.packages("RStoolbox")
#install.packages("rasterVis")
#install.packages('raster')
#install.packages('gdalUtils')
#install.packages('tidyverse')
#install.packages('rgr')
#install.packages('uavRst')
#install.packages('rgdal') -Alex
```

```
library('gdalUtils')
library('RStoolbox')
library('rasterVis')
library('raster')
library('ggplot2')
library('rgr')
library('tidyverse')
library('rgdal') #alex wrote this.
#library('uavRst') #may not have been used.
```

```
# Load Data -----
#Raster
```

```
      # We will create a character vector list of raster files using the list.files() function in the directory
named "All plots 2018". This list will be used to generate a Rasterstack.
files <- list.files()
files      #See the list of all files in the directory named "All plots 2018".
dbf.files <- files[grep(".tif", files, fixed=T)]      #Creates a file that is list of names only having .tif extensions. Grep
function finds ".tiff" pattern in the created "files" and fixed=T means pattern is a text string.
for(i in dbf.files) { assign(unlist(strsplit(i, "[.]"))[1], raster(i)) }      #
```

```
BH1_RGB<-stack("BH-01 RGB_modified.tif") # Import multi-band raster data, using the stack() function.
BH1_IR<-raster('BH-01 IR modified.tif')   # Import and create a Rasterlayer file using the raster function.
BL2_RGB<-stack("BL-02 RGB.tif")
BL2_IR<-raster('BL-02 IR.tif')
```

```
# BH-01 -----
```

```
#align extent
```

```
BH1_IR_proj<-projectRaster(BH1_IR, BH1_RGB) #Alex: Project the data of a Raster object to a new Raster object with another
projection (crs). projectRaster(from, to).
```

```
#Shapefile
plot(BH1_IR_proj)      #press-clear broom stick in plots tab, if you get a error that margins are too large.
Alex+ani: line 58-Error-plot.new has not been called yet, occurred becuase plot name has not been called.
polygon<-shapefile("Polygons.shp")      #Alex: File format for storing geospatial data in polygon.shp.
plot(polygon, add=TRUE)      #Alex: This adds another raster ontop of another. This draws the boundary of the two
exclosures over the [BH1_IR_proj] image
BH1_shp_ug<-subset(polygon, PlotID=='BH-01 UG') #Ungrazed #Alex: Returns (selected variables) subsets of vectors, matirces or
dataframes which meet conditions. Subset(object to be subsetted, logical expression indicating elements or rows to keep).
BH1_shp_g<-subset(polygon, PlotID=='BH-01 G')#Grazed
```

```
# Mask and clip rasters to polygon -----
```

```
#Ungrazed
```

```
BH1_RGB_mask<-mask(BH1_RGB, BH1_shp_ug)
BH1_IR_mask<-mask(BH1_IR_proj, BH1_shp_ug)
```

```
BH1RGB_crop<- crop(BH1_RGB_mask, BH1_shp_ug)
plot(BH1RGB_crop)      #Alex: four images produced. (#1-image)
ex<-extent(BH1RGB_crop)
```

```
BH1IR_crop<- crop(BH1_IR_mask, ex)
plot(BH1IR_crop)      #Alex: one image produced. (#2-image)
```

```
# Stack and Brick IR and RGB -----
```

```
BH1_stack<-stack(BH1RGB_crop, BH1IR_crop)
nlayers(BH1_stack)
BH1_stack<-writeRaster(BH1_stack, filename="C:/Users/aalda/Desktop/All plots 2018/BH1_Stack.tif", format="GTiff", overwrite=TRUE)
```

```
# Calculate NDVI -----
```

```
BH1_ndvi_ungrazed<-((BH1_stack[[5]]-BH1_stack[[1]])/(BH1_stack[[5]]+BH1_stack[[1]]))
plot(BH1_ndvi_ungrazed)      #Alex: one image produced. (#3-image)
```

```

hist(BH1_ndvi_ungrazed)           #Alex: histogram produced. (#4-image)

# Do it all again for BH1 Grazed -----

# Mask and clip rasters to polygon -----

#Grazed

BH1_RGB_mask<-mask(BH1_RGB, BH1_shp_g)
BH1_IR_mask<-mask(BH1_IR_proj, BH1_shp_g)

BH1RGB_crop<- crop(BH1_RGB_mask, BH1_shp_g)
plot(BH1RGB_crop)                 #Alex: four images produced. (#5-image)
ex<-extent(BH1RGB_crop)

BH1IR_crop<- crop(BH1_IR_mask, ex)
plot(BH1IR_crop)                 #Alex: one image produced. (#6-image)

# Stack and Brick IR and RGB -----

BH1_stack<-stack(BH1RGB_crop, BH1IR_crop)
nlayers(BH1_stack)               #Alex: there are 5 layers.

# Calculate NDVI -----

BH1_ndvi_grazed<-((BH1_stack[[5]]-BH1_stack[[1]])/(BH1_stack[[5]]+BH1_stack[[1]]))
plot(BH1_ndvi_grazed)            #Alex: one image produced. (#7-image)
BL2_IR_proj<-projectRaster(BL2_IR, BL2_RGB) #Alex: copied and pasted here.
BL2_shp_g<-subset(polygon, PlotID=='BL-02 G') #Grazed -Alex copied and pasted here.
BL2_IR_mask<-mask(BL2_IR_proj, BL2_shp_g)   #Alex: first run line 181 for BL2_shp_g and line 185 for BL2_IR_proj.

BL2_RGB_mask<-mask(BL2_RGB, BL2_shp_g)      #Alex: copied line 229 and pasted it here
BL2_shp_g<-subset(polygon, PlotID=='BL-02 G') #Grazed #Alex: copied and pasted it here
BL2RGB_crop<- crop(BL2_RGB_mask, BL2_shp_g) #Alex: first run line 229 (BL2_RGB_mask<-mask(BL2_RGB, BL2_shp_g). Problem: Error
in file(fn, "rb") : cannot open the connection. Solution: run line 72 and line 16 (library (raster)).
plot(BL2RGB_crop)                   #Alex: four images in the shape of parallelgrams. (#8 image)
ex<-extent(BL2RGB_crop)

BL2IR_crop<- crop(BL2_IR_mask, ex)
plot(BL2IR_crop)                   #Alex: one image produce in the shape of parallelgram. (#9-image)

# Stack and Brick IR and RGB -----

BL2_stack<-stack(BL2RGB_crop, BL2IR_crop)
nlayers(BL2_stack)

# Calculate NDVI -----

BL2_ndvi_grazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]]))
plot(BL2_ndvi_grazed)              #Alex: one image produced in the shape of a parallelgram. (#10 image)
hist(BL2_ndvi_grazed)              #Alex: histogram produced. (#11 image)

#compare to ungrazed
BL2_ndvi_ungrazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]])) #alex:copied/pasted here.
BL2_ndvi_ungrazed<-as.data.frame(BL2_ndvi_ungrazed) #Alex: First run line 217 for BL2_ndvi_ungrazed which is (
BL2_ndvi_ungrazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]])) .
BL2_ndvi_grazed<-as.data.frame(BL2_ndvi_grazed)

BH_01_g<-tibble(
  Value=BL2_ndvi_grazed$layer,
  Treatment="Grazed"
)

BH_01_ug<-tibble(
  Value=BL2_ndvi_ungrazed$layer,
  Treatment="Ungrazed"
)

BH_01<-rbind(BH_01_g, BH_01_ug)
ggplot(data=BH_01, aes(x=Treatment, y=Value))+
  geom_violin(scale='area')          #Alex: two images produced. (#12 image)
hist(BH1_ndvi_grazed)               #Alex: histogram produced. (#13 image)

#compare to ungrazed

BH1_ndvi_ungrazed<-as.data.frame(BH1_ndvi_ungrazed)
BH1_ndvi_grazed<-as.data.frame(BH1_ndvi_grazed)

BH_01_g<-tibble(
  Value=BH1_ndvi_grazed$layer,
  Treatment="Grazed"
)

BH_01_ug<-tibble(
  Value=BH1_ndvi_ungrazed$layer,
  Treatment="Ungrazed"
)

BH_01<-rbind(BH_01_g, BH_01_ug)
ggplot(data=BH_01, aes(x=Treatment, y=Value))+
  geom_violin(scale='area')          #Alex: two images produced. (#14 image)

```

```

# BL-02 -----

#align extent

BL2_IR_proj<-projectRaster(BL2_IR, BL2_RGB)      #Alex: two images produced. (#15 image)

#Shapefile

plot(BL2_IR_proj)                                #Alex created this line.
polygon<-shapefile("Polygons.shp")
plot(polygon, add=TRUE)
BL2_shp_ug<-subset(polygon, PlotID=='BL-02 UG') #Ungrazed
BL2_shp_g<-subset(polygon, PlotID=='BL-02 G') #Grazed

# Wrong: BL2_shp_ug<-subset(polygon, PlotID=='BH-01 UG') #Ungrazed      #Alex: warming messages.
#wrong: BL2_shp_g<-subset(polygon, PlotID=='BH-01 G') #Grazed

# Mask and clip rasters to polygon -----

#Ungrazed

BL2_RGB_mask<-mask(BL2_RGB, BL2_shp_ug)
BL2_IR_mask<-mask(BL2_IR_proj, BL2_shp_ug)

BL2RGB_crop<- crop(BL2_RGB_mask, BL2_shp_ug)      #Alex: first run line 191 for BL2_shp_ug. Problem: Error in .local(x, y, ...) :
extents do not overlap
plot(BL2RGB_crop)                                #Alex: four images produced in shape of parallelograms. (#16 image)
ex<-extent(BL2RGB_crop)

BL2IR_crop<- crop(BL2_IR_mask, ex)
plot(BL2IR_crop)                                #Alex: (check) one image not produced. (#17 image)

# Stack and Brick IR and RGB -----

BL2_stack<-stack(BL2RGB_crop, BL2IR_crop)
nlayers(BL2_stack)                                #Alex: 5 layers.

# Calculate NDVI -----

BL2_ndvi_ungrazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]]))
plot(BL2_ndvi_ungrazed)                          #Alex: (check) one image produced. (#18 image)
hist(BL2_ndvi_ungrazed)                          #Alex: (check) histogram produced. (#19 image)

# Do it all again for BL2 Grazed -----

# Mask and clip rasters to polygon -----

#Grazed

BL2_RGB_mask<-mask(BL2_RGB, BL2_shp_g)
BL2_IR_mask<-mask(BL2_IR_proj, BL2_shp_g)        #Ales wrote this line.

#pasted here.

BL2RGB_crop<- crop(BL2_RGB_mask, BL2_shp_g)      #Alex: first run line 229 (BL2_RGB_mask<-mask(BL2_RGB, BL2_shp_g). Problem: Error
in file(fn, "rb") : cannot open the connection. Solution: run line 72 and line 16 (library (raster))).
plot(BL2RGB_crop)                                #Alex: four images in the shape of parallelograms. (#8 image)
ex<-extent(BL2RGB_crop)

BL2IR_crop<- crop(BL2_IR_mask, ex)
plot(BL2IR_crop)                                #Alex: one image produce in the shape of parallelogram. (#9-image)

# Stack and Brick IR and RGB -----

BL2_stack<-stack(BL2RGB_crop, BL2IR_crop)
nlayers(BL2_stack)

# Calculate NDVI -----

BL2_ndvi_grazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]]))
plot(BL2_ndvi_grazed)                            #Alex: one image produced in the shape of a parallelogram. (#10 image)
hist(BL2_ndvi_grazed)                            #Alex: histogram produced. (#11 image)

#compare to ungrazed
BL2_ndvi_ungrazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]])) #alex:copied/pasted here.
BL2_ndvi_ungrazed<-as.data.frame(BL2_ndvi_ungrazed) #Alex: First run line 217 for BL2_ndvi_ungrazed which is (
BL2_ndvi_ungrazed<-((BL2_stack[[5]]-BL2_stack[[1]])/(BL2_stack[[5]]+BL2_stack[[1]])) .
BL2_ndvi_grazed<-as.data.frame(BL2_ndvi_grazed)

#alex wrote this
BL_02_g<-tibble(
  Value= BL2_ndvi_ungrazed$layer,
  Treatment="Grazed"
)

```

```

BL_02_ug<-tibble(
  Value=BL2_ndvi_ungrazed$layer,
  Treatment= "Ungrazed"
)

BL_02<-rbind(BL_02_g, BL_02_ug)
ggplot(data= BL_02, aes (x= Treatment, y=Value))+
  geom_violin(scale='area')
hist(BL2_ndvi_grazed)      # Error in hist.default(BL2_ndvi_grazed) : 'x' must be numeric. Image#

```