eCog\_Ant\_mound

R Pienaar

2024-08-23

Libraries

library(ggplot2)  
library(tidyverse)  
library(stringr)  
library(lme4)  
library(lmerTest)  
library(glmmTMB)

Insert data: Input is a combined csv from arc that shows individual disks per treatment and block

d16 <- read.csv("Arc\_Outputs/disks\_2016.csv")  
d21 <- read.csv("Arc\_Outputs/disks\_2021.csv")  
d23 <- read.csv("Arc\_Outputs/disks\_2023.csv")  
d16$Year <- 2016  
d21$Year <- 2021  
  
  
d16 <- d16 %>% select(Block, Plot, Shape\_Area, Year)  
d21 <- d21 %>% select(Block, Plot, Shape\_Area, Year)  
d23 <- d23 %>% select(Block, Plot, Shape\_Area, Year)  
disks <- rbind(d16, d21, d23)

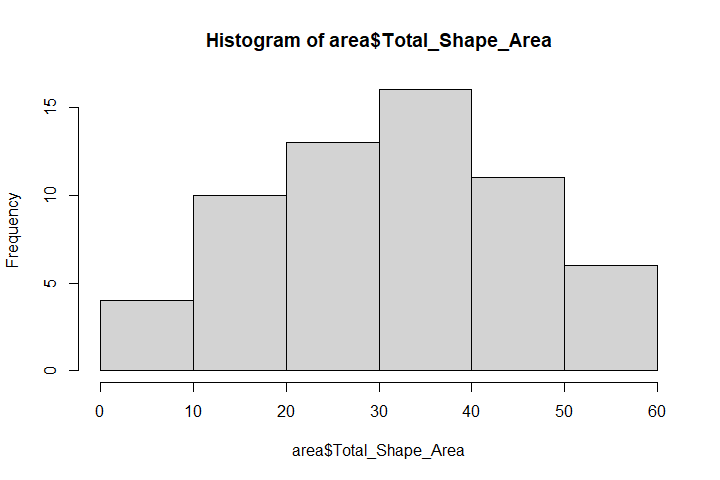
Create total area per treatment block

area <- disks %>%   
 group\_by(Block, Plot, Year) %>%  
 summarise(Total\_Shape\_Area = sum(Shape\_Area))  
  
complete\_combinations <- disks %>%  
 distinct(Block, Plot, Year) %>%  
 tidyr::expand(Block, Plot, Year)  
  
# Perform a full join with the original dataframe  
area <- complete\_combinations %>%  
 left\_join(area, by = c("Block", "Plot", "Year")) %>%  
 mutate(Total\_Shape\_Area = ifelse(is.na(Total\_Shape\_Area), 0, Total\_Shape\_Area))  
  
area <- area %>%  
 mutate(Burned = substr(Plot, 1, 1),  
 Rodents = substr(Plot, 2, 2))  
  
head(area)

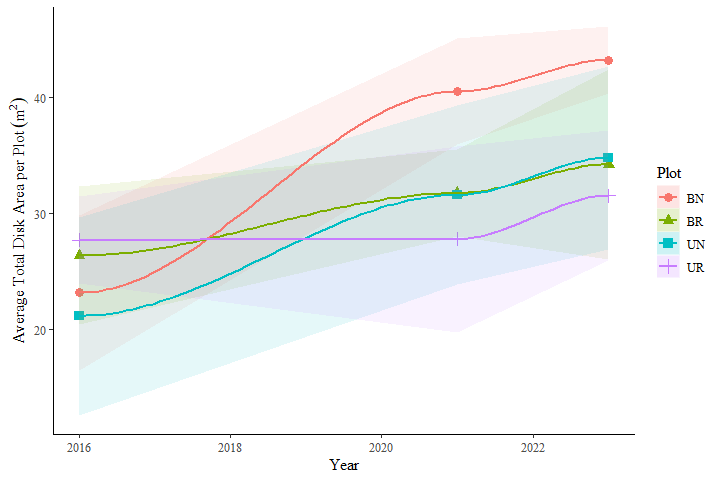
## # A tibble: 6 × 6  
## Block Plot Year Total\_Shape\_Area Burned Rodents  
## <int> <chr> <dbl> <dbl> <chr> <chr>   
## 1 1 BN 2016 49.3 B N   
## 2 1 BN 2021 54.9 B N   
## 3 1 BN 2023 47.8 B N   
## 4 1 BR 2016 32.5 B R   
## 5 1 BR 2021 30.4 B R   
## 6 1 BR 2023 21.3 B R

Compare total area of ant mounds and disks combined between treatments and years

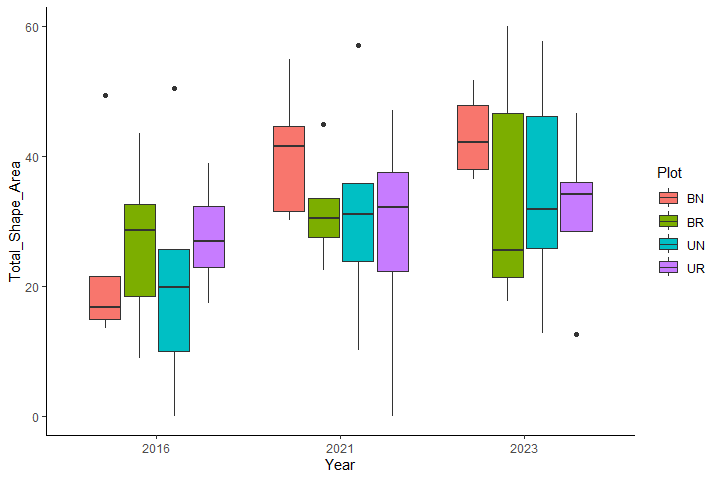
hist(area$Total\_Shape\_Area)



area %>% group\_by(Year, Plot) %>%  
 summarise(ave = mean(Total\_Shape\_Area),  
 se = sd(Total\_Shape\_Area)/ sqrt(n())) |>  
 ggplot(aes(Year, ave, color = Plot, fill = Plot, shape = Plot))+  
 geom\_point(size =3)+  
 geom\_ribbon(aes(ymin = ave - se, ymax = ave + se), alpha = 0.1, color = NA)+  
 geom\_smooth(aes(group = Plot), alpha = 0.1) +  
 theme\_classic() +  
 labs(x = "Year",  
 y = expression(paste("Average Total Disk Area per Plot ", (m^2))))+  
 theme(text = element\_text(size=12, family = "serif"))



area$Year <- as.factor(area$Year)  
   
ggplot(data = area, aes(Year, Total\_Shape\_Area, fill = Plot))+  
 geom\_boxplot()+  
 theme\_classic()



areadiff <- area |> pivot\_wider(names\_from = Year, values\_from = Total\_Shape\_Area)  
areadiff$tot.change <- areadiff$`2023` - areadiff$`2016`  
  
areaaov <- aov(data = areadiff, tot.change ~ Burned + Rodents + Burned \* Rodents)  
areaaov2 <- aov(data = areadiff, tot.change ~ Plot)  
summary(areaaov2)

## Df Sum Sq Mean Sq F value Pr(>F)  
## Plot 3 746 248.7 1.081 0.385  
## Residuals 16 3682 230.2

area %>% group\_by(Year, Plot) %>%  
 summarise(ave = mean(Total\_Shape\_Area),  
 se = sd(Total\_Shape\_Area)/ sqrt(n()))

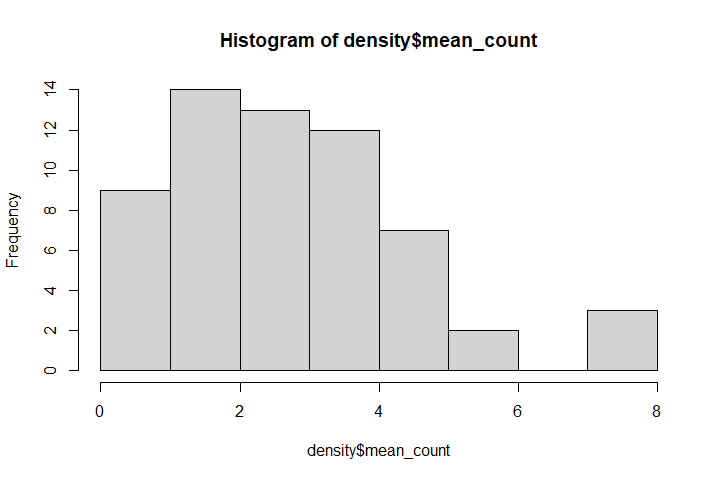
## # A tibble: 12 × 4  
## # Groups: Year [3]  
## Year Plot ave se  
## <fct> <chr> <dbl> <dbl>  
## 1 2016 BN 23.2 6.68  
## 2 2016 BR 26.4 5.94  
## 3 2016 UN 21.2 8.53  
## 4 2016 UR 27.7 3.73  
## 5 2021 BN 40.5 4.56  
## 6 2021 BR 31.7 3.75  
## 7 2021 UN 31.6 7.73  
## 8 2021 UR 27.8 8.02  
## 9 2023 BN 43.2 2.89  
## 10 2023 BR 34.2 8.16  
## 11 2023 UN 34.8 7.84  
## 12 2023 UR 31.5 5.60

Create density per treatment block

density <- disks  
  
density <- density %>% group\_by(Block, Plot, Year) %>%  
 summarise(mean\_count = mean(n()))  
  
density <- complete\_combinations %>% left\_join(density, by = c("Block", "Plot", "Year")) %>%  
 mutate(mean\_count = ifelse(is.na(mean\_count), 0, mean\_count))  
  
head(density)

## # A tibble: 6 × 4  
## Block Plot Year mean\_count  
## <int> <chr> <dbl> <dbl>  
## 1 1 BN 2016 6  
## 2 1 BN 2021 5  
## 3 1 BN 2023 5  
## 4 1 BR 2016 3  
## 5 1 BR 2021 3  
## 6 1 BR 2023 2

hist(density$mean\_count)



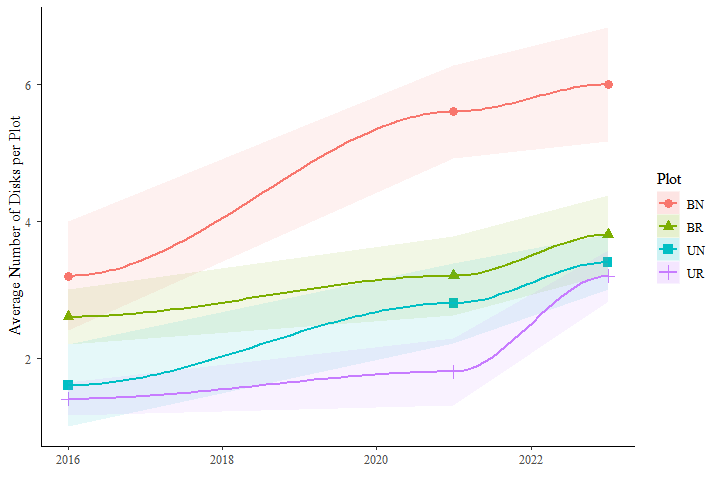
density[c("Burned", "Rodents")] <- str\_split\_fixed(density$Plot, "", 2)  
  
densityaov <- aov(data = density, mean\_count ~ Burned + Rodents)  
summary(densityaov)

## Df Sum Sq Mean Sq F value Pr(>F)   
## Burned 1 43.35 43.35 19.20 5.11e-05 \*\*\*  
## Rodents 1 18.15 18.15 8.04 0.00632 \*\*   
## Residuals 57 128.68 2.26   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

TukeyHSD(densityaov)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = mean\_count ~ Burned + Rodents, data = density)  
##   
## $Burned  
## diff lwr upr p adj  
## U-B -1.7 -2.476861 -0.9231393 5.11e-05  
##   
## $Rodents  
## diff lwr upr p adj  
## R-N -1.1 -1.876861 -0.3231393 0.0063231

density %>% group\_by(Year, Plot) %>%  
 summarise(ave = mean(mean\_count),  
 se = sd(mean\_count)/ sqrt(n())) |>  
 ggplot(aes(Year, ave, color = Plot, fill = Plot, shape = Plot))+  
 geom\_point(size = 3)+  
 scale\_y\_continuous(labels = function(x) paste0(x, " "))+  
 geom\_ribbon(aes(ymin = ave - se, ymax = ave + se), alpha = 0.1, color = NA)+  
 geom\_smooth(aes(group = Plot), alpha = 0.1) +  
 theme\_classic() +  
 labs(x = element\_blank(),  
 y = "Average Number of Disks per Plot")+  
 theme(text = element\_text(size=12, family = "serif"))

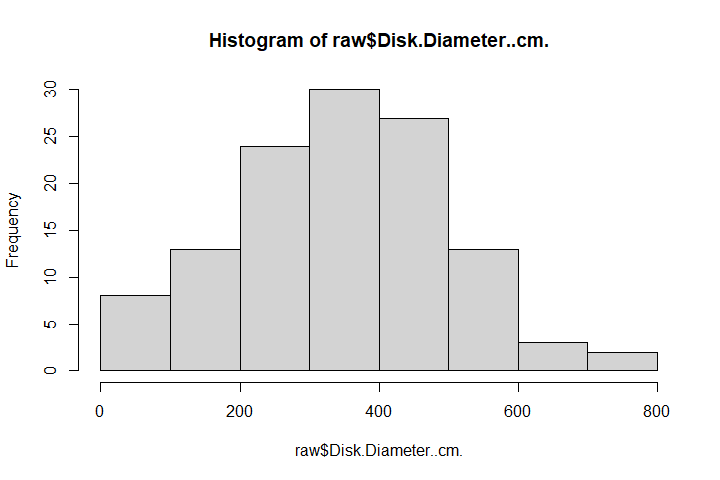


#### Repeated Measures ANOVA  
  
rma <- aov(data = density, mean\_count~ Burned\*Year+ Rodents\*Year+ Burned\*Rodents\*Year + Error(Block/Year))  
summary(rma)

##   
## Error: Block  
## Df Sum Sq Mean Sq F value Pr(>F)  
## Residuals 1 2.408 2.408   
##   
## Error: Block:Year  
## Df Sum Sq Mean Sq  
## Year 1 45.47 45.47  
##   
## Error: Within  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Burned 1 43.35 43.35 30.891 1.06e-06 \*\*\*  
## Year 1 0.05 0.05 0.033 0.856688   
## Rodents 1 18.15 18.15 12.934 0.000738 \*\*\*  
## Burned:Year 1 0.33 0.33 0.232 0.632442   
## Year:Rodents 1 2.63 2.63 1.876 0.176902   
## Burned:Rodents 1 6.02 6.02 4.288 0.043576 \*   
## Burned:Year:Rodents 1 1.62 1.62 1.153 0.288180   
## Residuals 50 70.17 1.40   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Load manual counts for 2021

manual\_dens\_2021 <- read.csv("Density\_manual\_2021.csv")  
raw <- read.csv("C:/Users/ryanp/Documents/Dissertation/Chapter1\_Ants/Ant\_analysis\_manua;/Ant Manuals.csv")  
hist(raw$Disk.Diameter..cm.)



manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "BS","BR"))  
  
manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "US","UR"))  
  
manual\_dens\_2021 <- manual\_dens\_2021 |> select(Block, Plot, Count, Area)  
  
head(manual\_dens\_2021)

## Block Plot Count Area  
## 1 1 BN 8 9503.317777  
## 2 2 BN 9 8607.963871  
## 3 3 BN 6 6565.928646  
## 4 4 BN 10 9000.662953  
## 5 5 BN 11 6418.273791  
## 6 1 BR 5 5089.380099

manual\_dens\_2021 <- raw  
  
manual\_dens\_2021 <- manual\_dens\_2021 |> group\_by(Block, Plot) %>%  
 summarise(mean\_count = mean(n()))  
  
manual\_dens\_2021 <- manual\_dens\_2021[-1,] #remove controls  
  
manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "BS","BR"))  
  
manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "US","UR"))  
  
  
#### Manual large is to do the same as before but to only compare disks above a certain size threshold  
# so same analysis repeated  
manual\_large <- raw[raw$Disk.Diameter..cm.>= 250,] ### I think there might be a size threshold for detecting mounds  
  
manual\_large <- manual\_large |> group\_by(Block, Plot) %>%  
 summarise(mean\_count = mean(n()))  
  
manual\_large <- manual\_large[-1,] # remove controls  
  
  
manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "BS","BR"))  
  
manual\_dens\_2021 <- manual\_dens\_2021 %>%  
 mutate\_all(~replace(., . == "US","UR"))  
  
  
head(manual\_dens\_2021)

## # A tibble: 6 × 3  
## # Groups: Block [2]  
## Block Plot mean\_count  
## <int> <chr> <chr>   
## 1 1 BN 8   
## 2 1 BR 5   
## 3 1 UN 6   
## 4 1 UR 3   
## 5 2 BN 9   
## 6 2 BR 8

Compare manual 2021 to drone 2021

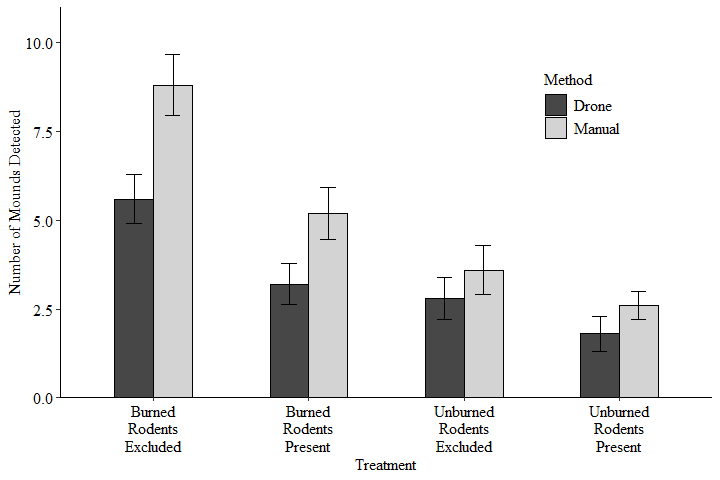
drone\_2021 <- disks[disks$Year== 2021,]  
drone\_2021 <- drone\_2021 %>% group\_by(Block, Plot) %>%  
 summarise(mean\_count = mean(n()),  
 Total\_Shape\_Area = sum(Shape\_Area))  
drone\_2021 <- complete\_combinations[complete\_combinations$Year==2021,] %>%   
 left\_join(drone\_2021, by = c("Block", "Plot")) %>%  
 mutate(mean\_count = ifelse(is.na(mean\_count), 0, mean\_count))  
  
drone\_2021 <- drone\_2021 %>% replace(is.na(.), 0)  
  
drone\_2021 <- drone\_2021 |> select(Block, Plot, mean\_count)  
#colnames(drone\_2021) <- colnames(manual\_dens\_2021)  
  
  
comp\_21 <- merge(drone\_2021,manual\_dens\_2021, by = c("Plot", "Block"), all.x = TRUE)  
colnames(comp\_21)

## [1] "Plot" "Block" "mean\_count.x" "mean\_count.y"

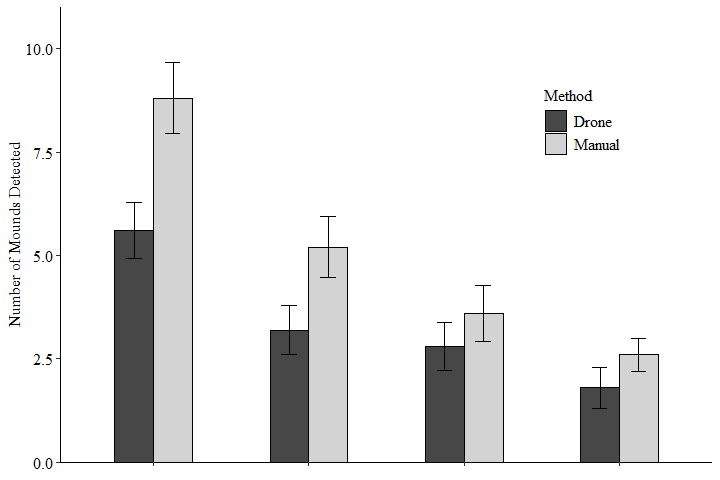
colnames(comp\_21) <- c("Plot", "Block", "Drone", "Manual")   
comp\_21$Manual <- as.numeric(comp\_21$Manual)  
  
summary(comp\_21)

## Plot Block Drone Manual   
## Length:20 Min. :1 Min. :0.00 Min. : 1.00   
## Class :character 1st Qu.:2 1st Qu.:2.00 1st Qu.: 3.00   
## Mode :character Median :3 Median :3.00 Median : 4.00   
## Mean :3 Mean :3.35 Mean : 5.05   
## 3rd Qu.:4 3rd Qu.:4.25 3rd Qu.: 6.50   
## Max. :5 Max. :8.00 Max. :11.00

comp\_21 <- comp\_21 |> pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Count")  
  
ggplot(data = comp\_21, aes(Plot, Count, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic() +   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 11)) + xlab( "Treatment") +  
 ylab("Number of Mounds Detected")+  
 theme(legend.position = c(0.8,0.75),  
 text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12))+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))+  
 scale\_x\_discrete(labels = c(paste("Burned", "Rodents", "Excluded", sep = "\n"),  
 paste("Burned", "Rodents", "Present", sep = "\n"),   
 paste("Unburned", "Rodents", "Excluded", sep = "\n"),   
 paste("Unburned", "Rodents", "Present", sep = "\n")))



#Exclude x axis labels for paper  
fig2a <- ggplot(data = comp\_21, aes(Plot, Count, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic() +   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 11)) + xlab( "Treatment") +  
 ylab("Number of Mounds Detected")+  
 xlab(element\_blank())+  
 theme(legend.position = c(0.8,0.75),  
 text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12),  
 axis.text.x = element\_blank())+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))  
  
fig2a



shapiro.test(sqrt(comp\_21$Count)) #### square root to normalize the data

##   
## Shapiro-Wilk normality test  
##   
## data: sqrt(comp\_21$Count)  
## W = 0.95441, p-value = 0.1076

aov21 <- aov(sqrt(comp\_21$Count)~ comp\_21$Method+ comp\_21$Plot + comp\_21$Method\*comp\_21$Plot)  
summary(aov21)

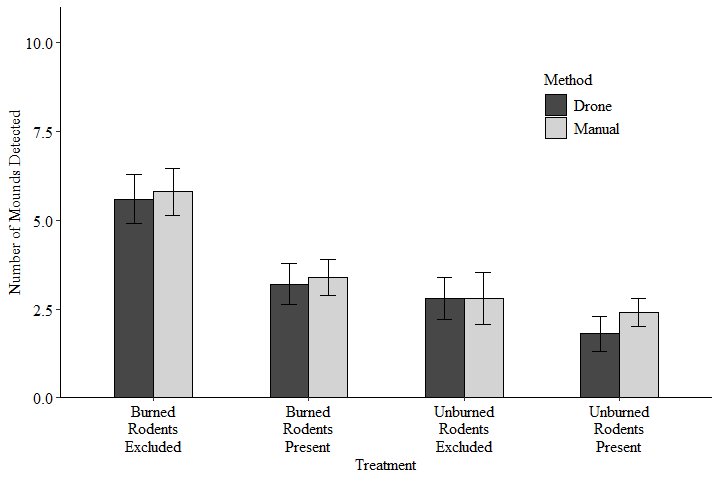
## Df Sum Sq Mean Sq F value Pr(>F)   
## comp\_21$Method 1 1.869 1.8688 11.01 0.00227 \*\*   
## comp\_21$Plot 3 8.488 2.8295 16.66 1.06e-06 \*\*\*  
## comp\_21$Method:comp\_21$Plot 3 0.184 0.0612 0.36 0.78205   
## Residuals 32 5.434 0.1698   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Do the same as above but only for mounds measuring greater than 2m in diameter

manual\_large[manual\_large=="BS"] <- "BR"  
manual\_large[manual\_large=="US"] <- "UR"  
comp\_21\_large <- merge(drone\_2021, manual\_large, by = c("Plot", "Block"), all.x = TRUE)  
colnames(comp\_21\_large)

## [1] "Plot" "Block" "mean\_count.x" "mean\_count.y"

colnames(comp\_21\_large) <- c("Plot", "Block", "Drone", "Manual")   
  
  
comp\_21\_large <- comp\_21\_large |> pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Count")  
  
fig2b <- ggplot(data = comp\_21\_large, aes(Plot, Count, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic() +   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 11)) +  
 theme(title = element\_text("serif"), text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12))+  
 xlab( "Treatment") +  
 ylab("Number of Mounds Detected")+   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 11)) +  
 theme(legend.position = c(0.8,0.75), text = element\_text(size=12, family = "serif"))+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))+  
 scale\_x\_discrete(labels = c(paste("Burned", "Rodents", "Excluded", sep = "\n"),  
 paste("Burned", "Rodents", "Present", sep = "\n"),   
 paste("Unburned", "Rodents", "Excluded", sep = "\n"),   
 paste("Unburned", "Rodents", "Present", sep = "\n")))  
  
fig2b



aov21\_large <-aov(sqrt(comp\_21\_large$Count)~ comp\_21\_large$Method+ comp\_21\_large$Plot +   
 comp\_21\_large$Method\*comp\_21\_large$Plot)  
summary(aov21\_large)

## Df Sum Sq Mean Sq F value Pr(>F)   
## comp\_21\_large$Method 1 0.109 0.1087 0.614 0.439   
## comp\_21\_large$Plot 3 5.526 1.8420 10.397 6.25e-05 \*\*\*  
## comp\_21\_large$Method:comp\_21\_large$Plot 3 0.174 0.0581 0.328 0.805   
## Residuals 32 5.670 0.1772   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Compare my manual area to my drone area with all disks

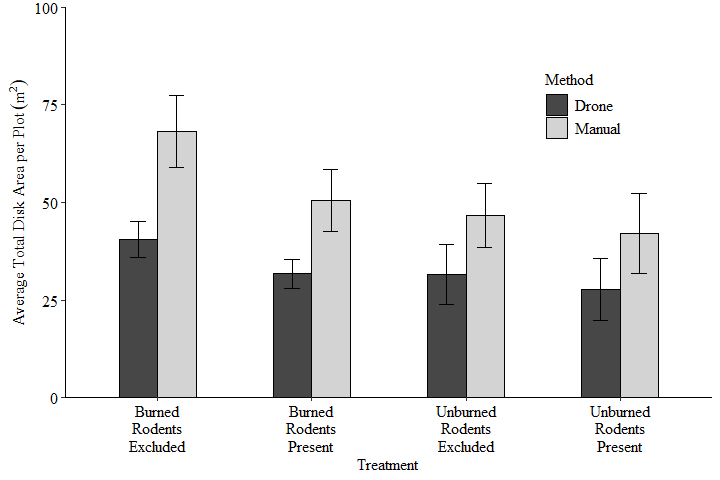
#### Organize manual areas #####  
man\_area\_21 <- read.csv("Manual\_area\_2021.csv")  
man\_area\_21 <- man\_area\_21[man\_area\_21$Plot != "C",] # Remove controls  
man\_area\_21 <- man\_area\_21 %>% group\_by(Block, Plot) %>%  
 summarise(total.area = sum(Disk.Area),  
 se = sd(Disk.Area)/ sqrt(n())) # Summarize to create total disk area per plot  
  
man\_area\_21 <- man\_area\_21 %>% replace(is.na(.), 0) #remove NAs  
man\_area\_21 <- man\_area\_21 %>%  
 mutate\_all(~replace(., . == "BS","BR")) # I used different naming conventions so I'm switching S to R  
man\_area\_21 <- man\_area\_21 %>%  
 mutate\_all(~replace(., . == "US","UR"))  
man\_area\_21

## # A tibble: 20 × 4  
## # Groups: Block [5]  
## Block Plot total.area se   
## <int> <chr> <chr> <chr>   
## 1 1 BN 91.665783145 1.17661691095255   
## 2 1 BR 45.297839072 2.884107217176   
## 3 1 UN 55.770338182 1.55922650316118   
## 4 1 UR 46.36118964 2.3456088964837   
## 5 2 BN 78.651028717 2.20948876863418   
## 6 2 BR 35.704436128 1.14267473972538   
## 7 2 UN 74.12658552 2.75518110150832   
## 8 2 UR 28.27433388 0   
## 9 3 BN 62.994430499 2.70522957303091   
## 10 3 BR 76.262554374 2.93096106536127   
## 11 3 UN 38.71400335 8.075463915   
## 12 3 UR 24.273515635 3.49565138736933   
## 13 4 BN 70.640674313 1.27776977200318   
## 14 4 BR 60.789817845 5.47325040694117   
## 15 4 UN 31.631911032 5.08758938205403   
## 16 4 UR 31.612276081 1.81575003969529   
## 17 5 BN 36.844363021 0.796527476116962  
## 18 5 BR 35.138164053 4.81996808558123   
## 19 5 UN 33.308736109 9.57229965191361   
## 20 5 UR 80.4404799 7.60346959702578

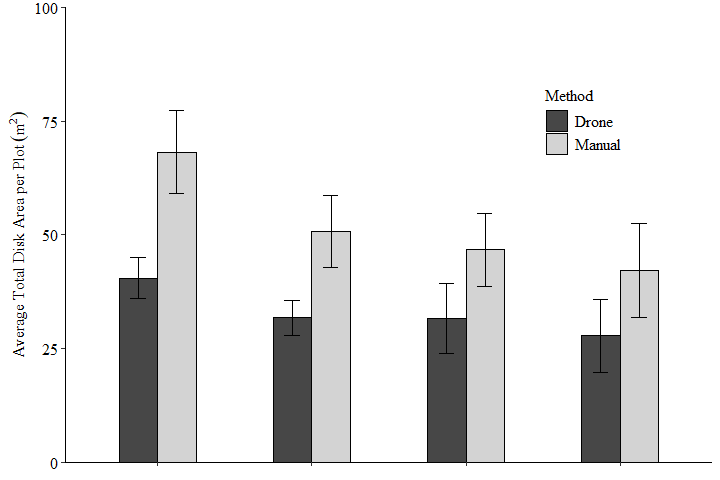
#### Organize 2021 drone areas ####  
drone\_2021 <- disks[disks$Year== 2021,]  
drone\_2021 <- drone\_2021 %>% group\_by(Block, Plot) %>%  
 summarise(total.area = sum(Shape\_Area),  
 se = sd(Shape\_Area)/ sqrt(n()))  
drone\_2021 <- complete\_combinations[complete\_combinations$Year==2021,] %>%   
 left\_join(drone\_2021, by = c("Block", "Plot"))   
  
drone\_2021 <- drone\_2021 %>% replace(is.na(.), 0)  
  
  
#### Merge together #####  
  
drone\_man21\_comp <- merge(drone\_2021, man\_area\_21, by = c("Block", "Plot"), all.x = TRUE)  
drone\_man21\_comp <- drone\_man21\_comp |> select(Block, Plot, total.area.x, total.area.y)  
  
colnames(drone\_man21\_comp) <- c("Block", "Plot","Drone", "Manual")  
drone\_man21\_comp$Manual <- as.numeric(drone\_man21\_comp$Manual)  
drone\_man21\_comp

## Block Plot Drone Manual  
## 1 1 BN 54.93658 91.66578  
## 2 1 BR 30.38121 45.29784  
## 3 1 UN 31.05475 55.77034  
## 4 1 UR 32.14975 46.36119  
## 5 2 BN 44.53588 78.65103  
## 6 2 BR 27.45346 35.70444  
## 7 2 UN 57.10453 74.12659  
## 8 2 UR 22.27423 28.27433  
## 9 3 BN 31.49449 62.99443  
## 10 3 BR 33.53372 76.26255  
## 11 3 UN 10.03832 38.71400  
## 12 3 UR 37.55020 24.27352  
## 13 4 BN 41.42948 70.64067  
## 14 4 BR 44.85330 60.78982  
## 15 4 UN 35.88195 31.63191  
## 16 4 UR 0.00000 31.61228  
## 17 5 BN 30.04699 36.84436  
## 18 5 BR 22.45231 35.13816  
## 19 5 UN 23.85633 33.30874  
## 20 5 UR 46.98408 80.44048

#### Plot comparison between manual and drone ####  
drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area") |>   
 ggplot(aes(Plot, Area, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic()+  
 theme(title = element\_text("serif"), text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12))+  
 xlab( "Treatment") +  
 ylab(expression(paste("Average Total Disk Area per Plot ", (m^2))))+   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 100))+  
 theme(legend.position = c(0.8,0.75), text = element\_text(size=12, family = "serif"))+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))+  
 scale\_x\_discrete(labels = c(paste("Burned", "Rodents", "Excluded", sep = "\n"),  
 paste("Burned", "Rodents", "Present", sep = "\n"),   
 paste("Unburned", "Rodents", "Excluded", sep = "\n"),   
 paste("Unburned", "Rodents", "Present", sep = "\n")))



#### Remove x axis for paper  
drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area") |>   
 ggplot(aes(Plot, Area, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic()+  
 theme(title = element\_text("serif"), text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12),  
 axis.text.x = element\_blank())+  
 xlab(element\_blank()) +  
 ylab(expression(paste("Average Total Disk Area per Plot ", (m^2))))+   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 100))+  
 theme(legend.position = c(0.8,0.75), text = element\_text(size=12, family = "serif"))+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))



ano.df2 <- drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area")  
  
aov21\_b <- aov(data = ano.df2, Area ~ Plot + Method + Plot\*Method)  
summary(aov21\_b)

## Df Sum Sq Mean Sq F value Pr(>F)   
## Plot 3 2091 697 2.344 0.09155 .   
## Method 1 3619 3619 12.171 0.00144 \*\*  
## Plot:Method 3 278 93 0.312 0.81649   
## Residuals 32 9516 297   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area") |>  
 group\_by(Plot, Method) |>  
 summarise(mean\_area = mean(Area))

## # A tibble: 8 × 3  
## # Groups: Plot [4]  
## Plot Method mean\_area  
## <chr> <chr> <dbl>  
## 1 BN Drone 40.5  
## 2 BN Manual 68.2  
## 3 BR Drone 31.7  
## 4 BR Manual 50.6  
## 5 UN Drone 31.6  
## 6 UN Manual 46.7  
## 7 UR Drone 27.8  
## 8 UR Manual 42.2

Compare my manual area to my drone area with the 250cm diameter cutoff

#### Organize manual areas #####  
man\_area\_21 <- read.csv("Manual\_area\_2021.csv")  
man\_area\_21 <- man\_area\_21[man\_area\_21$Plot != "C",] # Remove controls  
man\_area\_21 <- man\_area\_21[man\_area\_21$Disk.Diameter..cm.>=250,]  
man\_area\_21 <- man\_area\_21 %>% group\_by(Block, Plot) %>%  
 summarise(total.area = sum(Disk.Area),  
 se = sd(Disk.Area)/ sqrt(n())) # Summarize to create total disk area per plot  
  
man\_area\_21 <- man\_area\_21 %>% replace(is.na(.), 0) #remove NAs  
man\_area\_21 <- man\_area\_21 %>%  
 mutate\_all(~replace(., . == "BS","BR")) # I used different naming conventions so I'm switching S to R  
man\_area\_21 <- man\_area\_21 %>%  
 mutate\_all(~replace(., . == "US","UR"))  
man\_area\_21

## # A tibble: 20 × 4  
## # Groups: Block [5]  
## Block Plot total.area se   
## <int> <chr> <chr> <chr>   
## 1 1 BN 91.665783145 1.17661691095255   
## 2 1 BR 37.119880695 3.72432248155007   
## 3 1 UN 51.395984571 1.48120963993429   
## 4 1 UR 46.36118964 2.3456088964837   
## 5 2 BN 73.998958316 1.97515748059681   
## 6 2 BR 22.114063388 2.02601289078946   
## 7 2 UN 74.12658552 2.75518110150832   
## 8 2 UR 28.27433388 0   
## 9 3 BN 60.983811201 2.57956799921602   
## 10 3 BR 76.262554374 2.93096106536127   
## 11 3 UN 38.71400335 8.075463915   
## 12 3 UR 22.262896337 2.9881258515   
## 13 4 BN 59.038379944 0.956120074043167  
## 14 4 BR 60.789817845 5.47325040694117   
## 15 4 UN 30.92308919 2.259001465   
## 16 4 UR 31.612276081 1.81575003969529   
## 17 5 BN 24.339489083 0.552077509250291  
## 18 5 BR 31.682961912 6.220353454   
## 19 5 UN 30.1907054 0   
## 20 5 UR 80.4404799 7.60346959702578

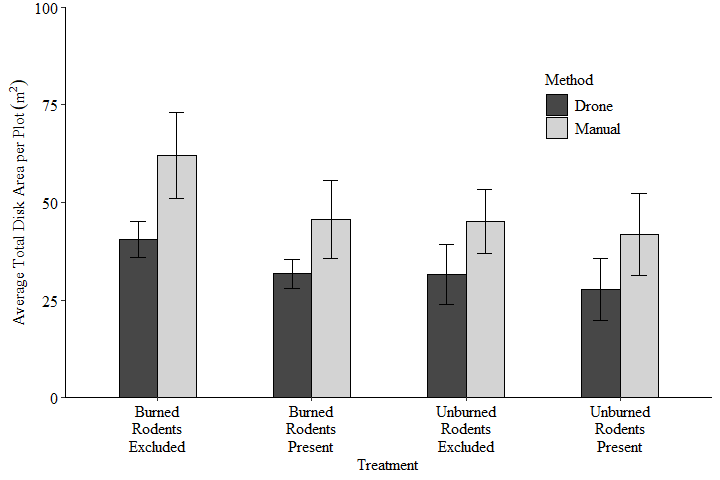
#### Organize 2021 drone areas ####  
drone\_2021

## # A tibble: 20 × 5  
## Block Plot Year total.area se  
## <int> <chr> <dbl> <dbl> <dbl>  
## 1 1 BN 2021 54.9 1.21   
## 2 1 BR 2021 30.4 2.42   
## 3 1 UN 2021 31.1 1.16   
## 4 1 UR 2021 32.1 2.48   
## 5 2 BN 2021 44.5 1.31   
## 6 2 BR 2021 27.5 1.05   
## 7 2 UN 2021 57.1 2.63   
## 8 2 UR 2021 22.3 8.73   
## 9 3 BN 2021 31.5 1.62   
## 10 3 BR 2021 33.5 0.826  
## 11 3 UN 2021 10.0 0   
## 12 3 UR 2021 37.6 8.47   
## 13 4 BN 2021 41.4 1.26   
## 14 4 BR 2021 44.9 2.46   
## 15 4 UN 2021 35.9 1.47   
## 16 4 UR 2021 0 0   
## 17 5 BN 2021 30.0 1.53   
## 18 5 BR 2021 22.5 2.66   
## 19 5 UN 2021 23.9 7.69   
## 20 5 UR 2021 47.0 15.1

#### Merge together #####  
  
drone\_man21\_comp <- merge( drone\_2021, man\_area\_21, by = c("Block", "Plot"), all.x = TRUE)  
drone\_man21\_comp <- drone\_man21\_comp |> select(Block, Plot, total.area.x, total.area.y)  
colnames(drone\_man21\_comp) <- c("Block", "Plot", "Drone", "Manual")  
drone\_man21\_comp$Manual <- as.numeric(drone\_man21\_comp$Manual)  
drone\_man21\_comp

## Block Plot Drone Manual  
## 1 1 BN 54.93658 91.66578  
## 2 1 BR 30.38121 37.11988  
## 3 1 UN 31.05475 51.39598  
## 4 1 UR 32.14975 46.36119  
## 5 2 BN 44.53588 73.99896  
## 6 2 BR 27.45346 22.11406  
## 7 2 UN 57.10453 74.12659  
## 8 2 UR 22.27423 28.27433  
## 9 3 BN 31.49449 60.98381  
## 10 3 BR 33.53372 76.26255  
## 11 3 UN 10.03832 38.71400  
## 12 3 UR 37.55020 22.26290  
## 13 4 BN 41.42948 59.03838  
## 14 4 BR 44.85330 60.78982  
## 15 4 UN 35.88195 30.92309  
## 16 4 UR 0.00000 31.61228  
## 17 5 BN 30.04699 24.33949  
## 18 5 BR 22.45231 31.68296  
## 19 5 UN 23.85633 30.19071  
## 20 5 UR 46.98408 80.44048

#### Plot comparison between manual and drone with 250cm cut off####  
drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area") |>  
 ggplot(aes(Plot, Area, fill = Method))+  
 stat\_summary(geom = "bar", fun = mean, position = "dodge", color = "black", width = 0.5) +  
 stat\_summary(geom = "errorbar", fun.data = mean\_se, width = 0.2, position = position\_dodge(0.5))+  
 theme\_classic()+  
 theme(title = element\_text("serif"),  
 text=element\_text("serif", size=12), #change font size of all text  
 axis.text=element\_text("serif", size=12, colour = "black"), #change font size of axis text  
 axis.title=element\_text("serif", size=12), #change font size of axis titles  
 plot.title=element\_text("serif", size=12), #change font size of plot title  
 legend.text=element\_text("serif", size=12), #change font size of legend text  
 legend.title=element\_text("serif", size=12))+  
 xlab( "Treatment") +  
 ylab(expression(paste("Average Total Disk Area per Plot ", (m^2))))+   
 scale\_y\_continuous(expand = c(0, 0), limits = c(0, 100))+  
 theme(legend.position = c(0.8,0.75), text = element\_text(size=12, family = "serif"))+  
 scale\_fill\_manual(values = c("grey28", "lightgrey"))+  
 scale\_x\_discrete(labels = c(paste("Burned", "Rodents", "Excluded", sep = "\n"),  
 paste("Burned", "Rodents", "Present", sep = "\n"),   
 paste("Unburned", "Rodents", "Excluded", sep = "\n"),   
 paste("Unburned", "Rodents", "Present", sep = "\n")))



drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area") |>  
 group\_by(Plot, Method) |>  
 summarise(mean\_area = mean(Area))

## # A tibble: 8 × 3  
## # Groups: Plot [4]  
## Plot Method mean\_area  
## <chr> <chr> <dbl>  
## 1 BN Drone 40.5  
## 2 BN Manual 62.0  
## 3 BR Drone 31.7  
## 4 BR Manual 45.6  
## 5 UN Drone 31.6  
## 6 UN Manual 45.1  
## 7 UR Drone 27.8  
## 8 UR Manual 41.8

ano.df1 <- drone\_man21\_comp |>   
 pivot\_longer(cols = c("Drone", "Manual"), names\_to = "Method", values\_to = "Area")   
   
aov21\_a <- aov(data = ano.df1, Area ~ Method + Plot + Method\*Plot)  
summary(aov21\_a)

## Df Sum Sq Mean Sq F value Pr(>F)   
## Method 1 2469 2469.4 7.090 0.012 \*  
## Plot 3 1559 519.7 1.492 0.235   
## Method:Plot 3 113 37.5 0.108 0.955   
## Residuals 32 11146 348.3   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

GLMM for density

typeof(density$Year)

## [1] "double"

dens.mod <- lmer(data = density, mean\_count ~ Burned \* Rodents + Year + (1|Block))  
int.mod <- lmer(data = density, mean\_count ~ Burned \* Rodents \* Year +(1|Block))  
summary(int.mod)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: mean\_count ~ Burned \* Rodents \* Year + (1 | Block)  
## Data: density  
##   
## REML criterion at convergence: 201.1  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -2.0296 -0.6431 -0.0361 0.5688 2.0853   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## Block (Intercept) 0.07431 0.2726   
## Residual 1.50828 1.2281   
## Number of obs: 60, groups: Block, 5  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) -834.1436 217.5812 48.0123 -3.834 0.000367 \*\*\*  
## BurnedU 323.9744 307.7062 48.0114 1.053 0.297671   
## RodentsR 511.0359 307.7062 48.0112 1.661 0.103273   
## Year 0.4154 0.1077 48.0123 3.856 0.000342 \*\*\*  
## BurnedU:RodentsR -449.3487 435.1623 48.0117 -1.033 0.306965   
## BurnedU:Year -0.1615 0.1523 48.0114 -1.060 0.294244   
## RodentsR:Year -0.2538 0.1523 48.0112 -1.666 0.102140   
## BurnedU:RodentsR:Year 0.2231 0.2154 48.0117 1.036 0.305617   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) BurndU RdntsR Year BrU:RR BrnU:Y RdnR:Y  
## BurnedU -0.707   
## RodentsR -0.707 0.500   
## Year -1.000 0.707 0.707   
## BrndU:RdntR 0.500 -0.707 -0.707 -0.500   
## BurnedU:Yer 0.707 -1.000 -0.500 -0.707 0.707   
## RodentsR:Yr 0.707 -0.500 -1.000 -0.707 0.707 0.500   
## BrndU:RdR:Y -0.500 0.707 0.707 0.500 -1.000 -0.707 -0.707  
## fit warnings:  
## Some predictor variables are on very different scales: consider rescaling

summary(dens.mod)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: mean\_count ~ Burned \* Rodents + Year + (1 | Block)  
## Data: density  
##   
## REML criterion at convergence: 197.7  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -2.06173 -0.62154 -0.00615 0.67259 2.17100   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## Block (Intercept) 0.07423 0.2724   
## Residual 1.50926 1.2285   
## Number of obs: 60, groups: Block, 5  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) -527.25897 108.82642 50.99989 -4.845 1.22e-05 \*\*\*  
## BurnedU -2.33333 0.44859 51.00000 -5.201 3.54e-06 \*\*\*  
## RodentsR -1.73333 0.44859 51.00000 -3.864 0.000316 \*\*\*  
## Year 0.26346 0.05387 50.99954 4.890 1.05e-05 \*\*\*  
## BurnedU:RodentsR 1.26667 0.63441 51.00000 1.997 0.051217 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) BurndU RdntsR Year   
## BurnedU -0.002   
## RodentsR -0.002 0.500   
## Year -1.000 0.000 0.000   
## BrndU:RdntR 0.001 -0.707 -0.707 0.000

dens.diff <- density |> pivot\_wider(names\_from = "Year", values\_from = "mean\_count")  
dens.diff$change <- dens.diff$'2023' - dens.diff$'2016'

GLMM for area

shapiro.test(area$Total\_Shape\_Area) ## Normal distribution

##   
## Shapiro-Wilk normality test  
##   
## data: area$Total\_Shape\_Area  
## W = 0.98697, p-value = 0.7714

area$Year <- as.numeric(area$Year)  
  
area.mod <- lmer(data = area, Total\_Shape\_Area ~ Burned \* Rodents + Year + (1|Block))  
  
summary(area.mod)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: Total\_Shape\_Area ~ Burned \* Rodents + Year + (1 | Block)  
## Data: area  
##   
## REML criterion at convergence: 459.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -2.10142 -0.60520 -0.02896 0.64747 2.02340   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## Block (Intercept) 0.0 0.0   
## Residual 190.5 13.8   
## Number of obs: 60, groups: Block, 5  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 24.293 5.635 55.000 4.311 6.78e-05 \*\*\*  
## BurnedU -6.440 5.040 55.000 -1.278 0.2067   
## RodentsR -4.845 5.040 55.000 -0.961 0.3406   
## Year 5.662 2.182 55.000 2.595 0.0121 \*   
## BurnedU:RodentsR 4.671 7.127 55.000 0.655 0.5150   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) BurndU RdntsR Year   
## BurnedU -0.447   
## RodentsR -0.447 0.500   
## Year -0.775 0.000 0.000   
## BrndU:RdntR 0.316 -0.707 -0.707 0.000  
## optimizer (nloptwrap) convergence code: 0 (OK)  
## boundary (singular) fit: see help('isSingular')

Comparison of change

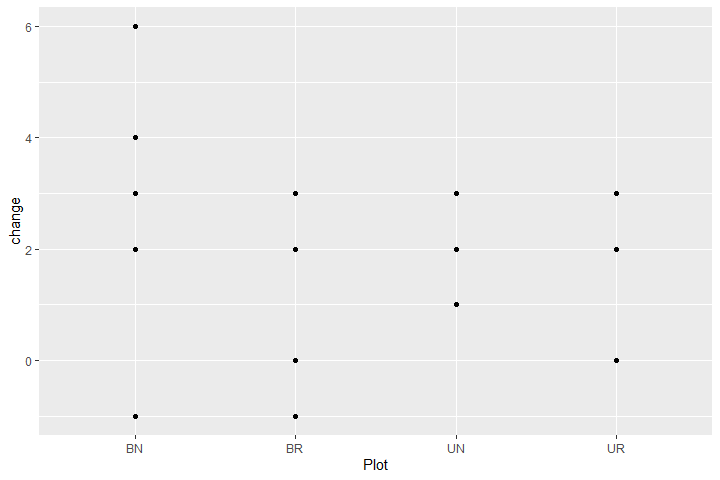
head(dens.diff)

## # A tibble: 6 × 8  
## Block Plot Burned Rodents `2016` `2021` `2023` change  
## <int> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 1 BN B N 6 5 5 -1  
## 2 1 BR B R 3 3 2 -1  
## 3 1 UN U N 3 4 4 1  
## 4 1 UR U R 2 3 4 2  
## 5 2 BN B N 2 6 4 2  
## 6 2 BR B R 3 5 3 0

summary(dens.diff)

## Block Plot Burned Rodents   
## Min. :1 Length:20 Length:20 Length:20   
## 1st Qu.:2 Class :character Class :character Class :character   
## Median :3 Mode :character Mode :character Mode :character   
## Mean :3   
## 3rd Qu.:4   
## Max. :5   
## 2016 2021 2023 change   
## Min. :0.0 Min. :0.00 Min. :2.0 Min. :-1.0   
## 1st Qu.:1.0 1st Qu.:2.00 1st Qu.:3.0 1st Qu.: 1.0   
## Median :2.0 Median :3.00 Median :4.0 Median : 2.0   
## Mean :2.2 Mean :3.35 Mean :4.1 Mean : 1.9   
## 3rd Qu.:3.0 3rd Qu.:4.25 3rd Qu.:5.0 3rd Qu.: 3.0   
## Max. :6.0 Max. :8.00 Max. :8.0 Max. : 6.0

ggplot(data= dens.diff, aes(Plot,change))+  
 geom\_point()



dens.c.ano <- aov(data = dens.diff, change ~ Plot + (1|Block))  
summary(dens.c.ano)

## Df Sum Sq Mean Sq F value Pr(>F)  
## Plot 3 6.6 2.200 0.779 0.523  
## Residuals 16 45.2 2.825

area.diff <- area |> pivot\_wider(names\_from = "Year", values\_from = "Total\_Shape\_Area")  
area.diff$change <- area.diff$'3' - area.diff$'1'  
  
area.c.ano <- aov(data = area.diff, change~ Plot)  
summary(area.c.ano)

## Df Sum Sq Mean Sq F value Pr(>F)  
## Plot 3 746 248.7 1.081 0.385  
## Residuals 16 3682 230.2

combined figures for paper

library(ggpubr)  
  
ggarrange(fig2a, fig2b, ncol = 1, common.legend = TRUE, legend = "right",  
 widths = 5, heights = 7.5)

