ESA\_Recovery\_overview\_doc

Annabelle

6/7/2021

##Introduction We are trying to answer the following Qs (which are the same as in the “Overview” doc) 1. What are the conservation initiatives for a species 2. What are the actions specified in the federal register important for the species? 3. Who are the organizations/players involved in doing these conservation initiatives? 4. What are the actions each organization is doing (optional?)

This document has the summary statistics for questions 1 & 2 (added 06/07/2021) and each section is question specific

Notes - Slight issue - missing at least 10 species data from tyler for initial draft of data (10 out of 38 species) … is this data accessible from the drives??

##Set up file structure

##load in different datasets

##Modify datasets

##Modify datasets Tyler Variables needed: # range size –> area c32 # area weighted footprint –> c33 ?have options # total threat count –> c 26 # type of threat –> c27 # employment (all relevant industries) -> 55:59 [avg\_employment ???] # percent public and percent federal c66, c68 # diversity of landowners - shannons H (land use complextity) # taxa (plants v animals // vert v invert) taxa c16

I think *missing* private lands?? (can get with TdataOLD because had variable information there..)

missing species (presumming Tyler dropped them..) # Euphilotes pallescens arenamontana # Nysius wekiuicola

Species Name changes/differences accross datasets Chorizanthe parryi var fernandina -> Chorizanthe parryi var. fernandina Moxostoma sp 2 -> Moxostoma sp.

tdata\_mod <- tdata %>% select(scientific\_name, common\_name, vert\_vs\_invert, species\_group, number\_of\_counties, range\_area, habitat\_modification, overutilization, pollution, species\_species\_interactions, env\_stochasticity, demographic\_stochasticity, total\_threats, public\_mang\_shannons\_diversity, avg\_employment)  
  
#put together with current list of species [doing Proactive dataset - may need to merge again]  
  
t\_pro\_df <- left\_join(proactive2, tdata\_mod, by = c("scientific\_name"))  
t\_pro\_df <- t\_pro\_df %>% select(-common\_name.y)  
t\_pro\_df$common\_name <- t\_pro\_df$common\_name.x  
t\_pro\_df <- t\_pro\_df %>% select(-common\_name.x)  
  
## Need to add values for % public/ private (located in tdata) bc not in new dataset   
## AND get info for species that have been recently removed from tyler's current dataset   
  
percent\_data <- tdataOLD %>% select(scientific\_name, common\_name, percentpublic:percentfederal)  
  
df <- left\_join(t\_pro\_df, percent\_data, by = c("common\_name"))  
df <- df %>% select(-scientific\_name.y)  
df$scientific\_name <- df$scientific\_name.x  
df <- df %>% select(-scientific\_name.x) #changing col name back  
  
## Need to rearrange column order  
  
df <- df[,c(27,23,1:22,24:26)]

## 1. What are the conservation initiatives for a species

#### 1 - Summary stats

* Number of iniatitves per species

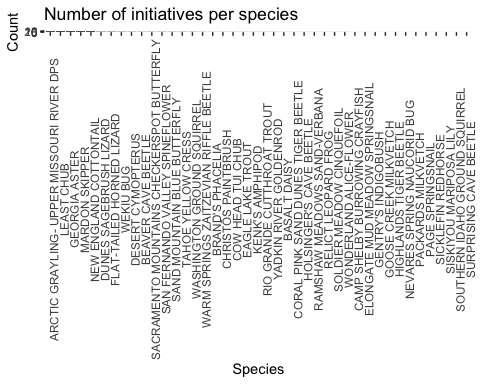
# iniatives per species (mean, median, quartiles)

* Count of efforts, agreements, groups per species

#### - Number of initiatives per species  
  
#number of partners in each of the agreements  
no\_init\_per\_sp <- sub\_initiatves\_partners %>% group\_by(common\_name, RowSpecInfo) %>% distinct(Agreement\_Name) %>% #count how many names are distinct   
summarise(n())

## `summarise()` has grouped output by 'common\_name'. You can override using the `.groups` argument.

## number of initiatives per species   
no\_init\_per\_sp$count <- no\_init\_per\_sp$`n()`   
no\_init\_per\_sp\_2 <- no\_init\_per\_sp %>% group\_by(common\_name) %>%  
summarize(sum = sum(count))  
  
ggplot(no\_init\_per\_sp\_2) + geom\_bar(mapping = aes(x = reorder(common\_name, -sum), y = sum), stat = "identity")+ theme(axis.text.x = element\_text(angle = 90)) + ggtitle("Number of initiatives per species") + scale\_x\_discrete(name ="Species") + scale\_y\_continuous(name ="Count")



##### - # iniatives per species (mean, median, quartiles)  
  
summary(no\_init\_per\_sp\_2)

## common\_name sum   
## Length:42 Min. : 1.000   
## Class :character 1st Qu.: 1.000   
## Mode :character Median : 3.000   
## Mean : 4.405   
## 3rd Qu.: 5.000   
## Max. :22.000

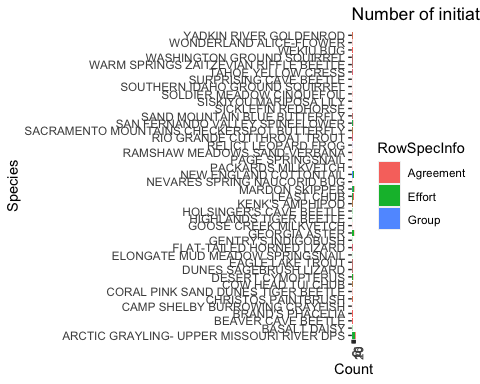
##### - Count of efforts, agreements, groups per species  
  
no\_init\_per\_sp <- sub\_initiatves\_partners %>% group\_by(common\_name, RowSpecInfo) %>% distinct(Agreement\_Name) %>% #count how many names are distinct   
summarise(n())

## `summarise()` has grouped output by 'common\_name'. You can override using the `.groups` argument.

## need to figure out how to add "Landowner" and "document" to other categories to count types of efforts   
no\_init\_per\_sp[which(no\_init\_per\_sp$RowSpecInfo == "Land Owner"),2] <- "Effort"  
no\_init\_per\_sp[which(no\_init\_per\_sp$RowSpecInfo == "Document"),2] <- "Agreement"  
  
no\_init\_per\_sp$count <- no\_init\_per\_sp$`n()`   
  
no\_init\_per\_sp\_3 <- no\_init\_per\_sp %>% group\_by(common\_name, RowSpecInfo) %>%  
summarize(sum = sum(count))

## `summarise()` has grouped output by 'common\_name'. You can override using the `.groups` argument.

ggplot(no\_init\_per\_sp\_3, aes(common\_name, sum)) +   
 geom\_bar(aes(fill = RowSpecInfo), position = "stack", stat="identity") + theme(axis.text.x = element\_text(angle = 90)) + ggtitle("Number of initiatives per species") + scale\_x\_discrete(name ="Species") + scale\_y\_continuous(name ="Count") + coord\_flip()



#### 1 - Regression Predictor Modification

* Working group mentioned – logistic regression
* Numbers of efforts or agreements – linear regression or negative binomial or poisson regression (since count data)

NoIniv <- no\_init\_per\_sp\_2   
names(NoIniv) <- c("common\_name", "no\_iniv")  
  
L\_df <- full\_join(df, NoIniv)

## Joining, by = "common\_name"

## taxa (coding for plants)  
plant <- c(0)  
L\_df <- add\_column(L\_df, plant)  
L\_df[which(L\_df$vert\_vs\_invert == "P"),29] <- 1  
  
reg\_prdctr <- L\_df %>% select(scientific\_name, common\_name, no\_iniv, range\_area, percentpublic, plant, public\_mang\_shannons\_diversity, total\_threats)  
  
  
## Add FR\_pg\_length variable and create dummy variables  
  
#type of FR doc - 12 monthns  
FR\_info <- FR\_pg\_length\_and\_doc\_type  
month\_12 <- c(0)  
FR\_info <- add\_column(FR\_info, month\_12)  
FR\_info[which(FR\_info$type\_FR\_doc == "12-Month Findings" ),5] <- 1  
  
#type of FR doc - withdrawal of proposed rule   
# method used above won't work because strings are all slightly different  
# grepl works to slect strings  
# case\_when is similar to ifelse   
  
FR\_info <- FR\_info %>% mutate(wd\_proposed = case\_when(grepl("withdrawal", type\_FR\_doc, ignore.case = TRUE) ~ 1)) %>%   
 replace(is.na(.), 0) # then replace all 0s   
  
#FR page count based on binary variable (1 = if greater than 1 page)  
FR\_info <- FR\_info %>% mutate(Big\_Pg\_count = case\_when(FR\_pg\_length > .5 ~ 1, FR\_pg\_length <= .5 ~ 0 ))   
  
#select the predictors actually want   
FR\_info <- FR\_info %>% select(-FR\_pg\_length, -type\_FR\_doc)   
  
  
reg\_prdctr <- reg\_prdctr %>% left\_join(FR\_info)

## Joining, by = c("scientific\_name", "common\_name")

names(reg\_prdctr) <- c("scientific\_name" , "common\_name" ,   
 "no\_iniv" , "range\_area" ,   
 "percentpublic" , "plant" ,   
 "public\_mang\_shannons\_diversity", "total\_threats" ,   
 "month\_12" , "wd\_proposed" ,   
"Big\_Pg\_count" )  
  
skim(reg\_prdctr)

Data summary

|  |  |
| --- | --- |
| Name | reg\_prdctr |
| Number of rows | 42 |
| Number of columns | 11 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| character | 2 |
| numeric | 9 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: character**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | min | max | empty | n\_unique | whitespace |
| scientific\_name | 0 | 1 | 13 | 35 | 0 | 42 | 0 |
| common\_name | 0 | 1 | 9 | 42 | 0 | 42 | 0 |

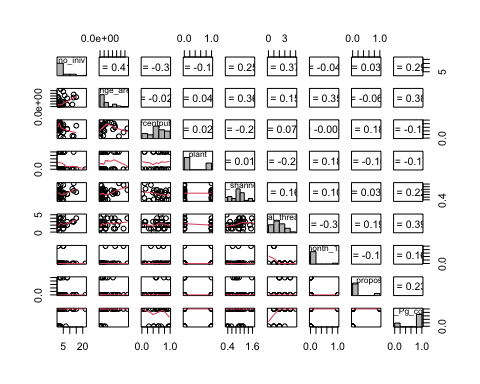
**Variable type: numeric**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| no\_iniv | 0 | 1.00 | 4.400000e+00 | 4.570000e+00 | 1.00 | 1.000000e+00 | 3.000000e+00 | 5.000000e+00 | 2.200000e+01 | ▇▁▁▁▁ |
| range\_area | 3 | 0.93 | 3.050419e+10 | 3.381441e+10 | 784468319.00 | 6.509527e+09 | 1.707273e+10 | 4.582297e+10 | 1.317573e+11 | ▇▁▂▁▁ |
| percentpublic | 12 | 0.71 | 5.600000e-01 | 2.600000e-01 | 0.01 | 4.200000e-01 | 5.900000e-01 | 7.500000e-01 | 9.800000e-01 | ▃▂▇▇▅ |
| plant | 0 | 1.00 | 3.600000e-01 | 4.800000e-01 | 0.00 | 0.000000e+00 | 0.000000e+00 | 1.000000e+00 | 1.000000e+00 | ▇▁▁▁▅ |
| public\_mang\_shannons\_diversity | 3 | 0.93 | 9.100000e-01 | 3.400000e-01 | 0.33 | 6.500000e-01 | 9.500000e-01 | 1.130000e+00 | 1.650000e+00 | ▅▃▇▃▂ |
| total\_threats | 3 | 0.93 | 2.330000e+00 | 1.110000e+00 | 0.00 | 2.000000e+00 | 2.000000e+00 | 3.000000e+00 | 5.000000e+00 | ▅▇▆▃▁ |
| month\_12 | 0 | 1.00 | 7.000000e-02 | 2.600000e-01 | 0.00 | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 | 1.000000e+00 | ▇▁▁▁▁ |
| wd\_proposed | 0 | 1.00 | 1.400000e-01 | 3.500000e-01 | 0.00 | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 | 1.000000e+00 | ▇▁▁▁▁ |
| Big\_Pg\_count | 0 | 1.00 | 7.600000e-01 | 4.300000e-01 | 0.00 | 1.000000e+00 | 1.000000e+00 | 1.000000e+00 | 1.000000e+00 | ▂▁▁▁▇ |

reg\_prdctr\_init <- reg\_prdctr

### Base model

# Correlation Matrix of Predictors vs Responses ##  
## modified some code from http://handlesman.blogspot.com/2011/03/matrix-plot-with-confidence-intervals.html  
#Not working because of NAs in data - getting error that x is not numeric   
  
  
#set up  
PredictorsOnlyPixel <- reg\_prdctr\_init[,c(3)]  
PredictAndResponsePixel <- reg\_prdctr\_init[,-c(1:2)]  
PredictAndResponseGrid <- reg\_prdctr\_init[,-c(1:2)]  
   
# put histograms on the diagonal panel   
panel.hist <- function (x,...) # define a function that says what we want to plot in the diagonal  
{  
 usr <- par("usr"); on.exit(par(usr)) # not sure what usr is for?  
 par(usr = c(usr[1:2],0,1.5))  
 h <- hist(x, plot = FALSE)  
 breaks <- h$breaks; nB <- length(breaks) # make the hist   
 y <- h$counts; y <- y/max(y)  
 rect(breaks[-nB], 0, breaks[-1], y, col="grey", ...) # defines what the histogram is going to look like  
}  
  
# put correlations on the upper panels,  
panel.cor <- function(x, y, digits=2, prefix="", cex.cor, ...)  
{  
 usr <- par("usr"); on.exit(par(usr))  
 par(usr = c(0, 1, 0, 1))  
 r <- cor(x, y,use="complete.obs")   
 txt <- format(c(r, 0.123456789), digits=digits)[1]  
 prefix <- "r = "  
 rc <- cor.test(x,y,method = c("pearson")) ## calculate pearsons rho for upper grid  
 txt <- paste(prefix,txt,sep="")  
 text(0.5, 0.5, txt, cex = 1)  
}  
  
## plot a correlation matrix plot that uses the functions specified above to say what to plot where  
 ## this was taken directly from website and still not plotting r values for all   
#pairs(PredictAndResponsePixel[1:6], lower.panel=panel.smooth, cex = .8, diag.panel=panel.hist, cex.labels = 1.2, font.labels=2, upper.panel=panel.cor)  
  
##Pixel level  
  
pairs(PredictAndResponsePixel,lower.panel = panel.smooth, diag.panel=panel.hist,upper.panel=panel.cor)



### Predictors only --> only have one response so not going to run this one  
##pairs(PredictorsOnlyPixel,lower.panel = panel.smooth, diag.panel=panel.hist,upper.panel=panel.cor)  
  
  
#test correlations to look for covariation between predictors  
cor(reg\_prdctr[,-c(1:2)], use = "complete.obs") # matrix seems fine

## no\_iniv range\_area percentpublic  
## no\_iniv 1.00000000 0.32943096 -0.37118860  
## range\_area 0.32943096 1.00000000 -0.02396985  
## percentpublic -0.37118860 -0.02396985 1.00000000  
## plant -0.18065813 -0.03791189 0.02802639  
## public\_mang\_shannons\_diversity 0.36935612 0.52359818 -0.24839380  
## total\_threats 0.40217206 0.20681469 0.07563005  
## month\_12 -0.02107239 0.38029754 -0.00708419  
## wd\_proposed 0.14346151 -0.02961601 0.17741864  
## Big\_Pg\_count 0.39210574 0.39920159 -0.10515034  
## plant public\_mang\_shannons\_diversity  
## no\_iniv -0.18065813 0.36935612  
## range\_area -0.03791189 0.52359818  
## percentpublic 0.02802639 -0.24839380  
## plant 1.00000000 0.15024899  
## public\_mang\_shannons\_diversity 0.15024899 1.00000000  
## total\_threats -0.35360595 0.12078957  
## month\_12 0.18144368 0.17417738  
## wd\_proposed -0.32025631 -0.08749337  
## Big\_Pg\_count -0.27695585 0.17006441  
## total\_threats month\_12 wd\_proposed  
## no\_iniv 0.402172056 -0.02107239 0.143461510  
## range\_area 0.206814688 0.38029754 -0.029616006  
## percentpublic 0.075630054 -0.00708419 0.177418639  
## plant -0.353605952 0.18144368 -0.320256308  
## public\_mang\_shannons\_diversity 0.120789570 0.17417738 -0.087493365  
## total\_threats 1.000000000 -0.36836441 0.005857476  
## month\_12 -0.368364410 1.00000000 -0.130744090  
## wd\_proposed 0.005857476 -0.13074409 1.000000000  
## Big\_Pg\_count 0.396235632 0.20100756 0.236524958  
## Big\_Pg\_count  
## no\_iniv 0.3921057  
## range\_area 0.3992016  
## percentpublic -0.1051503  
## plant -0.2769559  
## public\_mang\_shannons\_diversity 0.1700644  
## total\_threats 0.3962356  
## month\_12 0.2010076  
## wd\_proposed 0.2365250  
## Big\_Pg\_count 1.0000000

#no na.rm in this one instead it's "use"

* Note: percent public variable is still in matrix but was removed from model

VIFS

#### Previous  
###vif(lm(no\_iniv ~ range\_area +percentpublic + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr))  
  
vif(lm(log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr\_init))

## range\_area plant   
## 1.166965 1.069748   
## public\_mang\_shannons\_diversity total\_threats   
## 1.166627 1.108348

model summary - variations

lm1 <- lm(log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr\_init)  
  
lm2 <- lm(log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr\_init)  
  
lm3 <- lm(log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats + Big\_Pg\_count,data = reg\_prdctr\_init)  
  
lm4 <- lm(log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity + total\_threats + Big\_Pg\_count,data = reg\_prdctr\_init)  
  
lm5 <- lm(log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats + month\_12,data = reg\_prdctr\_init)  
  
lm6 <- lm(log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity + total\_threats + month\_12,data = reg\_prdctr\_init)  
  
lm7 <- lm(log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats + wd\_proposed,data = reg\_prdctr\_init)  
  
lm8 <- lm(log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity + total\_threats + wd\_proposed,data = reg\_prdctr\_init)  
  
  
summary(lm1)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.17757 -0.64524 0.00773 0.45359 1.44443   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.575e-02 4.410e-01 0.172 0.8646   
## range\_area 8.308e-12 3.997e-12 2.078 0.0453 \*  
## plant 6.471e-03 2.626e-01 0.025 0.9805   
## public\_mang\_shannons\_diversity 1.701e-02 3.942e-01 0.043 0.9658   
## total\_threats 2.716e-01 1.189e-01 2.284 0.0287 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7713 on 34 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2696, Adjusted R-squared: 0.1837   
## F-statistic: 3.138 on 4 and 34 DF, p-value: 0.02674

summary(lm2)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.24287 -0.77443 -0.05437 0.60042 1.27104   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -3.99252 2.11341 -1.889 0.0674 .  
## log(range\_area) 0.18398 0.09537 1.929 0.0621 .  
## plant -0.03972 0.26760 -0.148 0.8829   
## public\_mang\_shannons\_diversity 0.10583 0.38636 0.274 0.7858   
## total\_threats 0.25265 0.12151 2.079 0.0452 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7774 on 34 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2581, Adjusted R-squared: 0.1708   
## F-statistic: 2.956 on 4 and 34 DF, p-value: 0.0337

summary(lm3)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats + Big\_Pg\_count, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.13881 -0.61015 0.01713 0.58371 1.38541   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.759e-02 4.558e-01 -0.148 0.8830   
## range\_area 6.676e-12 4.218e-12 1.583 0.1230   
## plant 5.313e-02 2.643e-01 0.201 0.8419   
## public\_mang\_shannons\_diversity -9.826e-03 3.929e-01 -0.025 0.9802   
## total\_threats 2.257e-01 1.247e-01 1.810 0.0795 .  
## Big\_Pg\_count 3.986e-01 3.434e-01 1.161 0.2541   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7674 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2983, Adjusted R-squared: 0.192   
## F-statistic: 2.806 on 5 and 33 DF, p-value: 0.03219

summary(lm4)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity +   
## total\_threats + Big\_Pg\_count, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.25999 -0.64742 -0.00799 0.55009 1.24403   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -3.01792 2.35085 -1.284 0.2082   
## log(range\_area) 0.13374 0.10911 1.226 0.2290   
## plant 0.01843 0.27484 0.067 0.9470   
## public\_mang\_shannons\_diversity 0.08469 0.38753 0.219 0.8284   
## total\_threats 0.21944 0.12658 1.734 0.0923 .  
## Big\_Pg\_count 0.35727 0.37522 0.952 0.3479   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7785 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2779, Adjusted R-squared: 0.1685   
## F-statistic: 2.54 on 5 and 33 DF, p-value: 0.04735

summary(lm5)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats + month\_12, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.13714 -0.62556 0.00507 0.50482 1.43715   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.383e-02 4.582e-01 0.096 0.9244   
## range\_area 7.746e-12 4.426e-12 1.750 0.0894 .  
## plant 1.411e-03 2.666e-01 0.005 0.9958   
## public\_mang\_shannons\_diversity 1.374e-02 3.997e-01 0.034 0.9728   
## total\_threats 2.889e-01 1.324e-01 2.182 0.0364 \*  
## month\_12 1.762e-01 5.586e-01 0.315 0.7544   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7818 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2718, Adjusted R-squared: 0.1615   
## F-statistic: 2.464 on 5 and 33 DF, p-value: 0.05291

summary(lm6)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity +   
## total\_threats + month\_12, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.23493 -0.73435 -0.06814 0.62614 1.27728   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -3.66729 2.26403 -1.620 0.1148   
## log(range\_area) 0.16730 0.10376 1.612 0.1164   
## plant -0.04258 0.27092 -0.157 0.8761   
## public\_mang\_shannons\_diversity 0.09260 0.39220 0.236 0.8148   
## total\_threats 0.27814 0.13604 2.045 0.0489 \*  
## month\_12 0.24247 0.55323 0.438 0.6640   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7868 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2624, Adjusted R-squared: 0.1506   
## F-statistic: 2.347 on 5 and 33 DF, p-value: 0.06275

summary(lm7)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats + wd\_proposed, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.0882 -0.5597 -0.1403 0.4547 1.5679   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.838e-02 4.364e-01 0.111 0.9124   
## range\_area 8.809e-12 3.970e-12 2.219 0.0335 \*  
## plant 5.751e-02 2.624e-01 0.219 0.8279   
## public\_mang\_shannons\_diversity -4.109e-03 3.901e-01 -0.011 0.9917   
## total\_threats 2.456e-01 1.191e-01 2.062 0.0471 \*  
## wd\_proposed 4.688e-01 3.506e-01 1.337 0.1903   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7626 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.3072, Adjusted R-squared: 0.2022   
## F-statistic: 2.926 on 5 and 33 DF, p-value: 0.02706

summary(lm8)

##   
## Call:  
## lm(formula = log(no\_iniv) ~ log(range\_area) + plant + public\_mang\_shannons\_diversity +   
## total\_threats + wd\_proposed, data = reg\_prdctr\_init)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.15588 -0.69251 -0.06053 0.59142 1.35663   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -3.932291 2.111950 -1.862 0.0715 .  
## log(range\_area) 0.180033 0.095346 1.888 0.0678 .  
## plant 0.004023 0.270632 0.015 0.9882   
## public\_mang\_shannons\_diversity 0.106592 0.385952 0.276 0.7841   
## total\_threats 0.234632 0.122622 1.913 0.0644 .  
## wd\_proposed 0.368549 0.355794 1.036 0.3078   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7766 on 33 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.2814, Adjusted R-squared: 0.1726   
## F-statistic: 2.585 on 5 and 33 DF, p-value: 0.04435

AIC(lm1)

## [1] 97.07528

AIC(lm2)

## [1] 97.68863

AIC(lm3)

## [1] 97.51452

AIC(lm4)

## [1] 98.63164

AIC(lm5)

## [1] 98.95787

AIC(lm6)

## [1] 99.46227

AIC(lm7)

## [1] 97.01775

AIC(lm8)

## [1] 98.44074

#kable(table(lm2$coefficients))  
  
#par(mfrow = c(1, 1))  
#plot(lm2)  
  
#qqPlot(residuals(lm2))  
#par(mfrow = c(1, 1))  
  
#plot(predicted(lm2), residuals(lm2))  
#hist(residuals(lm2))  
# how add CI equivalent around q-q plot?   
AIC(lm2)

## [1] 97.68863

#check\_overdispersion(lm2)

## 2. What are the actions specified in the federal register important for the species?

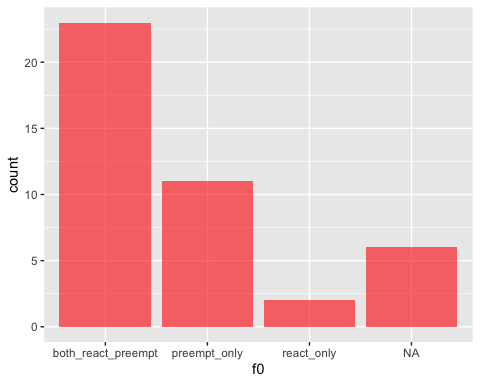
#### 1 - Summary stats

outcomes <- df  
  
outcomes[,c(3:11)] <- sapply(outcomes[ ,c(3:11)], as.numeric)  
  
### ### ### ### ### setting up data mannipulations (unsure if need) by making new columns   
  
### ### ### ### Actions  
outcomes <- outcomes %>% rowwise() %>% mutate("preempt" = sum(preempt\_private, preempt\_public, preempt\_unclear)) %>% #rowSums with mutate (need rowwise)   
 mutate\_at(vars(preempt), ~1 \* (. > 0)) #changed all values back to ones and zeros ## this doesn't work because changes whole df   
  
outcomes <- outcomes %>% rowwise() %>% mutate("react" = sum(respon\_private, respon\_public, respon\_unclear)) %>% #rowSums with mutate (need rowwise)  
 mutate\_at(vars(react), ~1 \* (. > 0)) #changed all values back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("both\_react\_preempt" = sum(react, preempt)) %>% #rowSums with mutate (need rowwise)  
 mutate\_at(vars(both\_react\_preempt), ~1 \* (. >= 2)) # change all specified mutate functions back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("react\_only" = react - both\_react\_preempt) #%>% #rowSums with mutate (need rowwise)  
# mutate\_at(vars(react\_only), ~"react\_only" \* (. > 0)) #changed all values back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("preempt\_only" = preempt - both\_react\_preempt)  
  
#outcomes %>% mutate(vars(both\_react\_preempt:preempt\_only), funs(ifelse(. == 1, deparse(substitute(.)), .)))  
  
## trying to get a single column with preempt, protect, or both as cell values for ggplot ease  
outcomes$actionresults <- apply(outcomes[,30:32], 1, function(x) max(names(which(x >0))))

## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA

## Bar chart Preempt: proactive, protective, both  
ggplot(outcomes, aes(f0)) +   
 geom\_histogram(mapping = aes(x = actionresults), stat = "count", fill = "red", alpha = 0.6, position = "dodge")

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

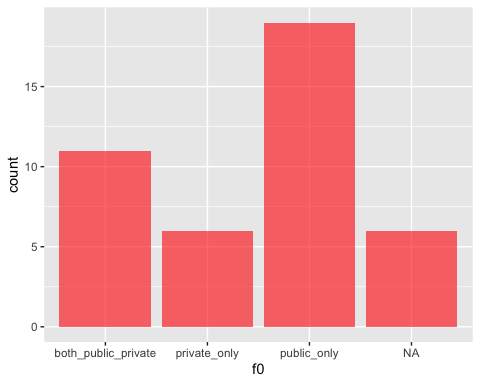


### ### Land  
outcomes <- outcomes %>% rowwise() %>% mutate("public" = sum(preempt\_public, public\_unclear, respon\_public)) %>% #rowSums with mutate (need rowwise)   
 mutate\_at(vars(public), ~1 \* (. > 0)) #changed all values back to ones and zeros ## this doesn't work because changes whole df   
  
outcomes <- outcomes %>% rowwise() %>% mutate("private" = sum(respon\_private, private\_unclear, preempt\_private)) %>% #rowSums with mutate (need rowwise)  
 mutate\_at(vars(private), ~1 \* (. > 0)) #changed all values back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("both\_public\_private" = sum(public, private)) %>% #rowSums with mutate (need rowwise)  
 mutate\_at(vars(both\_public\_private), ~1 \* (. >= 2)) # change all specified mutate functions back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("public\_only" = public - both\_public\_private) #%>% #rowSums with mutate (need rowwise)  
# mutate\_at(vars(react\_only), ~"react\_only" \* (. > 0)) #changed all values back to ones and zeros   
  
outcomes <- outcomes %>% rowwise() %>% mutate("private\_only" = private - both\_public\_private)  
  
#outcomes %>% mutate(vars(both\_react\_preempt:preempt\_only), funs(ifelse(. == 1, deparse(substitute(.)), .)))  
  
## trying to get a single column with preempt, protect, or both as cell values for ggplot ease  
outcomes$landresults <- apply(outcomes[,36:38], 1, function(x) max(names(which(x >0))))

## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA  
  
## Warning in max(names(which(x > 0))): no non-missing arguments, returning NA

## Barchart: Land: public, private, both  
ggplot(outcomes, aes(f0)) +   
 geom\_histogram(mapping = aes(x = landresults), stat = "count", fill = "red", alpha = 0.6, position = "dodge")

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



### ### ### ### ### ### Taxa   
  
taxa <- outcomes  
taxa <- taxa %>% select(scientific\_name, common\_name, preempt\_private:species\_group, preempt:landresults)  
  
## Filling in missing data   
taxa[which(is.na(taxa$vert\_vs\_invert)),12] <- c("I")  
taxa[which(is.na(taxa$species\_group)),13] <- "Insects"   
  
### ### ### Taxa -- Actions  
 ### all species, plant species, animal species (outcome: preempt)  
  
taxa\_small <- taxa %>% select(scientific\_name, actionresults, vert\_vs\_invert )  
  
## trying to get the following data frame   
#| | both| preempt | protect|  
#|:------|-----|---------|-------:|  
#|all sp | | | |  
#|plant | | | |  
#|animal | | | |  
  
# Doing so by subsetting by animal type and then will recombine   
  
plants <- taxa\_small %>% filter(vert\_vs\_invert == "P") %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
plants[,c(1:5)] <- sapply(plants[ ,c(1:5)], as.numeric)  
plants <- colSums(plants[,c(2:5)], na.rm = T)  
plants <- as.data.frame(plants) %>% rownames\_to\_column()  
  
invert <- taxa\_small %>% filter(vert\_vs\_invert == "I") %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
invert[,c(1:4)] <- sapply(invert[ ,c(1:4)], as.numeric)  
invert <- colSums(invert[,c(2:4)], na.rm = T)   
invert <- as.data.frame(invert) %>% rownames\_to\_column()  
 #rownames\_to\_column()   
  
vert <- taxa\_small %>% filter(vert\_vs\_invert == "V") %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
vert[,c(1:3)] <- sapply(vert[ ,c(1:3)], as.numeric)  
vert <- colSums(vert[,c(2:3)], na.rm = T)   
vert <- as.data.frame(vert) %>% rownames\_to\_column()  
  
species <- full\_join(plants, vert, by = "rowname")  
species <- full\_join(species, invert, by = "rowname")  
species <- species %>% rowwise() %>% mutate("all\_sp" = sum(plants,vert, invert, na.rm = T))  
   
### make bar graph   
  
s <- species  
rownames(s) <- c("NA", "react\_only", "both\_react\_preempt", "preempt\_only" )

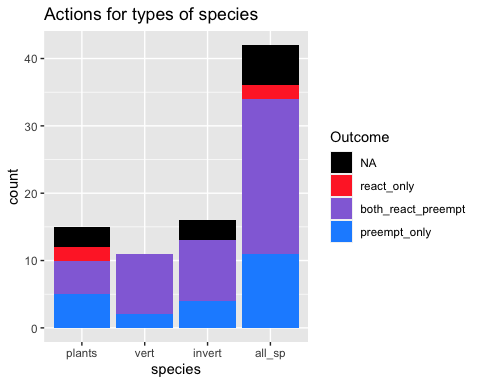
## Warning: Setting row names on a tibble is deprecated.

s <- as.data.frame(s)  
#s <- s[,-c(1)]  
smelt <- s %>% melt()

## Using rowname as id variables

ggplot(smelt) + aes(x=variable, y=value, fill = factor(rowname, levels=c("NA", "react\_only", "both\_react\_preempt", "preempt\_only"))) +  
 geom\_bar(stat="identity") +   
 scale\_x\_discrete(name ="species") +   
 ggtitle("Actions for types of species") +   
 scale\_y\_continuous(name = "count")+  
 labs(fill = "Outcome")+   
 scale\_fill\_manual(values=c("black", "firebrick1", "mediumpurple", "dodgerblue"))

## Warning: Removed 3 rows containing missing values (position\_stack).



### ### ### Taxa -- Land   
 ### all species, plant species, animal species (outcome: land)  
  
taxa\_land <- taxa %>% select(scientific\_name, landresults, vert\_vs\_invert )  
  
plants <- taxa\_land %>% filter(vert\_vs\_invert == "P") %>% ## Select which species type to subset   
pivot\_wider(names\_from = landresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
plants[,c(1:5)] <- sapply(plants[ ,c(1:5)], as.numeric)  
plants <- colSums(plants[,c(2:5)], na.rm = T)  
plants <- as.data.frame(plants) %>% rownames\_to\_column()  
  
invert <- taxa\_land %>% filter(vert\_vs\_invert == "I") %>% ## Select which species type to subset   
pivot\_wider(names\_from = landresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
invert[,c(1:4)] <- sapply(invert[ ,c(1:4)], as.numeric)  
invert <- colSums(invert[,c(2:4)], na.rm = T)   
invert <- as.data.frame(invert) %>% rownames\_to\_column()  
 #rownames\_to\_column()   
  
vert <- taxa\_land %>% filter(vert\_vs\_invert == "V") %>% ## Select which species type to subset   
pivot\_wider(names\_from = landresults, values\_from = vert\_vs\_invert) %>% ## may action types col names  
 as.data.frame() %>% # turn to dataframe  
 replace(!is.na(.), 1) # convert all values to 1s so can col sum below   
vert[,c(1:5)] <- sapply(vert[ ,c(1:5)], as.numeric)  
vert <- colSums(vert[,c(2:5)], na.rm = T)   
vert <- as.data.frame(vert) %>% rownames\_to\_column()  
  
species\_actions <- full\_join(plants, vert, by = "rowname")  
species\_actions <- full\_join(species\_actions, invert, by = "rowname")  
species\_actions <- species\_actions %>% rowwise() %>% mutate("all\_sp" = sum(plants,vert, invert, na.rm = T))  
   
### make bar graph   
  
sactions <- species\_actions  
rownames(sactions) <- c("public\_only", "NA", "both\_public\_private", "private\_only" )

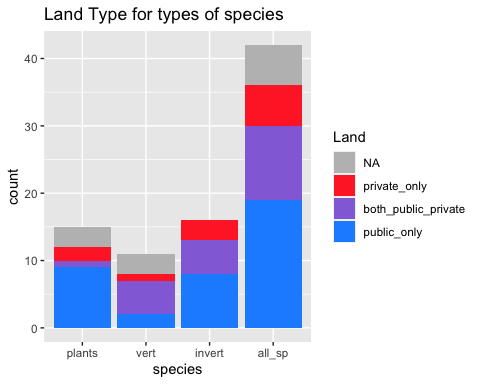
## Warning: Setting row names on a tibble is deprecated.

sactions <- as.data.frame(sactions)  
#s <- s[,-c(1)]  
sactions\_melt <- sactions %>% melt()

## Using rowname as id variables

ggplot(sactions\_melt) + aes(x=variable, y=value, fill = factor(rowname, levels=c("NA", "private\_only", "both\_public\_private", "public\_only"))) +  
 geom\_bar(stat="identity") +   
 scale\_x\_discrete(name ="species") +   
 ggtitle("Land Type for types of species") +   
 scale\_y\_continuous(name = "count")+   
 labs(fill = "Land")+   
 scale\_fill\_manual(values=c("grey", "firebrick1", "mediumpurple", "dodgerblue"))

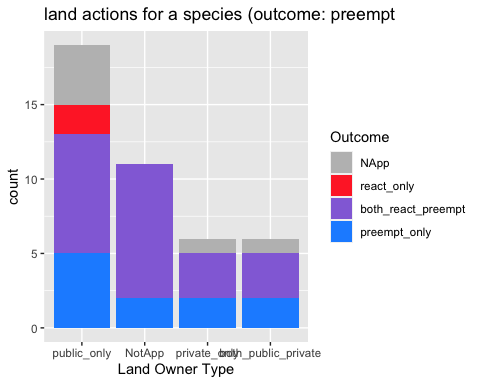
## Warning: Removed 1 rows containing missing values (position\_stack).



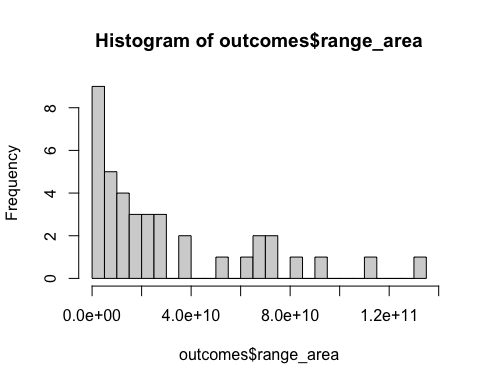
### ### ### ### ### ### ### ### ### ### ### ### ### Land  
  
### ### ### ### ### Land -- Preempt   
 ### public land actions for a species, private land actions for a species (outcome: preempt)  
  
land\_preempt <- outcomes  
land\_preempt <- land\_preempt %>% select(scientific\_name, actionresults, landresults)  
  
land\_preempt$number <- rownames(land\_preempt)  
  
  
land\_preempt\_trial <- land\_preempt %>% pivot\_wider(values\_from = number, names\_from = landresults)  
  
land\_preempt\_trial[,c(3:6)] <- sapply(land\_preempt\_trial[ ,c(3:6)], as.numeric)  
  
land\_preempt\_trial <- land\_preempt\_trial %>% mutate\_if(is.numeric, ~1 \* (. > 0)) #changed all values back to ones and zeros  
  
land\_preempt\_trial <- as.data.table(land\_preempt\_trial)  
names(land\_preempt\_trial) <- c( "scientific\_name" , "actionresults" , "public\_only" , "NotApp" , "private\_only" , "both\_public\_private" )  
  
land\_preempt\_trial[which(is.na(land\_preempt\_trial$actionresults)),2] <- "NApp"   
  
#Want to use equivalent of group by and then colsum (tidyr version didn't work but the datatable method does)  
count <- land\_preempt\_trial[,list(public\_only=sum(public\_only, na.rm = T), NotApp=sum(NotApp, na.rm = T), private\_only = sum(private\_only, na.rm = T), both\_public\_private = sum(both\_public\_private, na.rm = T)), by='actionresults']  
  
## Couldn't get this to work  
#land\_preempt\_trial <- land\_preempt\_trial %>% select(-scientific\_name) %>% group\_by(actionresults) %>%   
#summarize\_at(var(react\_only, NApp, preempt\_only, both\_react\_preempt), fun = colSums)  
# summarise\_if(is.integer, sum, na.rm = T)  
  
act\_land <- count %>% melt()

## Using actionresults as id variables

ggplot(act\_land) + aes(x=variable, y=value, fill = factor(actionresults, levels=c("NApp", "react\_only", "both\_react\_preempt", "preempt\_only"))) +  
 geom\_bar(stat="identity") +   
 scale\_x\_discrete(name ="Land Owner Type") +   
 ggtitle("land actions for a species (outcome: preempt") +   
 scale\_y\_continuous(name = "count")+  
 labs(fill = "Outcome")+   
 scale\_fill\_manual(values=c("grey", "firebrick1", "mediumpurple", "dodgerblue"))



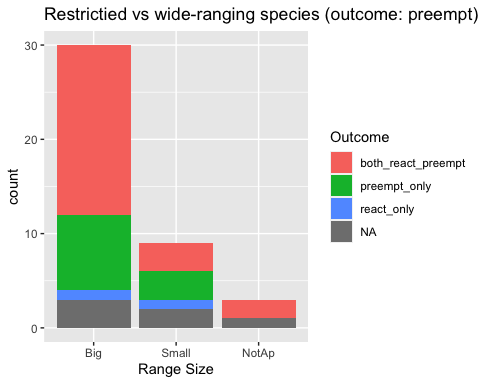
### ### ### ### ### Range size -- Preempt   
 ### restricted versus wide-ranging species? (outcome: preempt)  
## Note - this seems to really depend on what you set the area cutoff to  
  
hist(outcomes$range\_area, breaks = 40)



## let's do 2 categories of small v large range (split point = 2.0e + 10)  
  
range <- outcomes  
range <- range %>% select(scientific\_name, range\_area, actionresults)  
range <- range %>% mutate(size = range\_area < (.5\*10^10))  
#Want to use equivalent of group by and then colsum (tidyr version didn't work but the datatable method does)  
  
range$number <- 1  
range <- range %>% pivot\_wider(values\_from = number, names\_from = size)  
names(range) <- c("scientific\_name" ,"range\_area" , "actionresults" , "Big", "Small" , "NotAp" )  
range <- as.data.table(range)  
count\_range <- range[,list(Big=sum(Big, na.rm = T), Small=sum(Small, na.rm = T), NotAp = sum(NotAp, na.rm = T)), by='actionresults']  
  
count\_range <- count\_range %>% melt()

## Using actionresults as id variables

ggplot(count\_range) + aes(x=variable, y=value, fill = actionresults) +  
 geom\_bar(stat="identity") +   
 scale\_x\_discrete(name ="Range Size") +   
 ggtitle("Restrictied vs wide-ranging species (outcome: preempt)") +   
 scale\_y\_continuous(name = "count") +  
 labs(fill = "Outcome")



## Try again later

### ### ### ### ### Range size -- Preempt   
 ### presence of each threat type (outcome: preempt)  
  
threat\_type <- outcomes  
threat\_type <- threat\_type %>% select(scientific\_name, habitat\_modification:total\_threats, actionresults)  
  
# note - going to do this independently (so for each threat type not by species.. slightly different than other ones)  
  
habitat <- threat\_type %>% filter(habitat\_modification == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = habitat\_modification) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
habitat[,c(8:11)] <- sapply(habitat[ ,c(8:11)], as.numeric)  
habitat <- colSums(habitat[,c(8:11)], na.rm = T)   
habitat <- as.data.frame(habitat) %>% rownames\_to\_column()  
  
#only one instance for this one  
overut <- threat\_type %>% filter(overutilization == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = overutilization) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
overut[,c(8)] <- sapply(overut[ ,c(8)], as.numeric)  
overut <- colSums(overut[,c(7:8)], na.rm = T)   
overut <- as.data.frame(overut) %>% rownames\_to\_column()  
  
polu <- threat\_type %>% filter(pollution == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = pollution) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
polu[,c(7:8)] <- sapply(polu[ ,c(7:8)], as.numeric)  
polu <- colSums(polu[,c(7:8)], na.rm = T)   
polu <- as.data.frame(polu) %>% rownames\_to\_column()  
  
spsp <- threat\_type %>% filter(species\_species\_interactions == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = species\_species\_interactions) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
spsp[,c(8:11)] <- sapply(spsp[ ,c(8:11)], as.numeric)  
spsp <- colSums(spsp[,c(8:11)], na.rm = T)   
spsp <- as.data.frame(spsp) %>% rownames\_to\_column()  
  
env\_st <- threat\_type %>% filter(env\_stochasticity == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = env\_stochasticity) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
env\_st[,c(8:10)] <- sapply(env\_st[ ,c(8:10)], as.numeric)  
env\_st <- colSums(env\_st[,c(8:10)], na.rm = T)   
env\_st <- as.data.frame(env\_st) %>% rownames\_to\_column()  
  
demo\_st <- threat\_type %>% filter(demographic\_stochasticity == 1) %>% ## Select which species type to subset   
pivot\_wider(names\_from = actionresults, values\_from = demographic\_stochasticity) %>% ## may action types col names  
 as.data.frame() # turn to dataframe  
demo\_st[,c(8:9)] <- sapply(demo\_st[ ,c(8:9)], as.numeric)  
demo\_st <- colSums(demo\_st[,c(8:9)], na.rm = T)   
demo\_st <- as.data.frame(demo\_st) %>% rownames\_to\_column()  
  
threats\_actions <- full\_join(habitat, overut) %>% full\_join(spsp) %>% full\_join(env\_st) %>% full\_join(demo\_st)

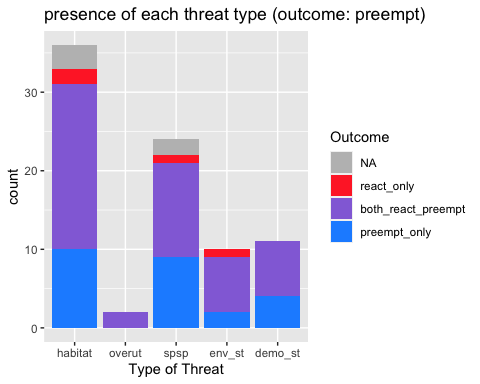
## Joining, by = "rowname"  
## Joining, by = "rowname"  
## Joining, by = "rowname"  
## Joining, by = "rowname"

threats\_actions <- threats\_actions[-which(is.na(threats\_actions$habitat)),]  
  
threats\_actions <- threats\_actions %>% melt()

## Using rowname as id variables

ggplot(threats\_actions) + aes(x=variable, y=value, fill = factor(rowname, levels=c("NA", "react\_only", "both\_react\_preempt", "preempt\_only"))) +  
 geom\_bar(stat="identity") +   
 scale\_x\_discrete(name ="Type of Threat") +   
 ggtitle("presence of each threat type (outcome: preempt)") +   
 scale\_y\_continuous(name = "count") +  
 labs(fill = "Outcome")+   
 scale\_fill\_manual(values=c("grey", "firebrick1", "mediumpurple", "dodgerblue"))

## Warning: Removed 6 rows containing missing values (position\_stack).



### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

Chi squares

# assumption: 5 or more in at least 80% of the cells, and no cell should have an expected of less than one (3).  
  
  
### ### ### Differences in frequency of proactive and protective measures on public versus private lands (Chi-square)  
  
differences <- outcomes  
differences <- as.data.frame(differences)  
differences[,c(3:10)] <- sapply(differences[ ,c(3:10)], as.numeric)  
count\_diff <- colSums(differences[,c(3:10)], na.rm = T)  
count\_diff <- as.data.frame(count\_diff)  
  
## manually entering into table   
#### note numbers have changes  
diff\_tb <- data.frame('preempt' = c(15,21,22), 'respon' = c(2,14,17), 'unc' = c(7,13,0))  
rownames(diff\_tb) <- c("private", "public", "unc")  
  
kable(diff\_tb)

|  |  |  |  |
| --- | --- | --- | --- |
|  | preempt | respon | unc |
| private | 15 | 2 | 7 |
| public | 21 | 14 | 13 |
| unc | 22 | 17 | 0 |

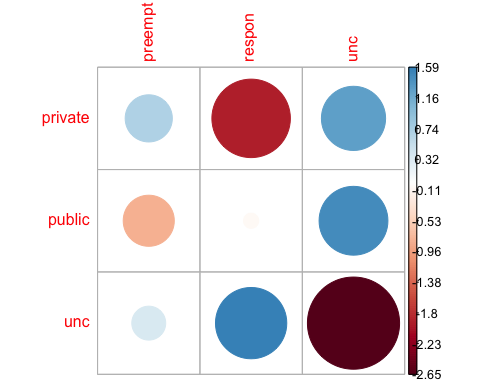
chsq <- chisq.test(diff\_tb)

## Warning in stats::chisq.test(x, y, ...): Chi-squared approximation may be  
## incorrect

chsq$residuals

## preempt respon unc  
## private 0.6945149 -1.92243106 1.286691  
## public -0.8148958 -0.07154548 1.479619  
## unc 0.3592235 1.58745158 -2.650854

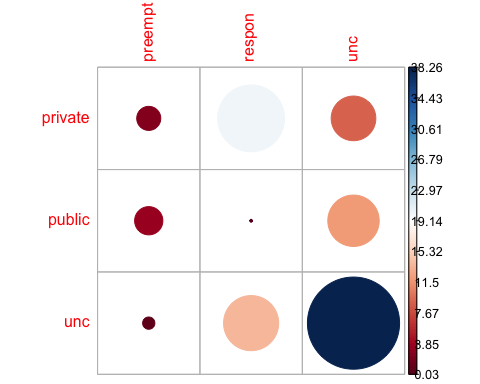
corrplot(chsq$residuals, is.cor = FALSE)



contrib <- 100\*chsq$residuals^2/chsq$statistic  
round(contrib, 3)

## preempt respon unc  
## private 2.626 20.120 9.013  
## public 3.615 0.028 11.919  
## unc 0.703 13.719 38.257

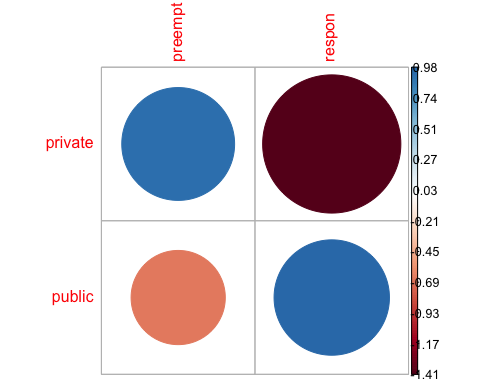
corrplot(contrib, is.cor = FALSE)



### ### ### Removing UNC from - Differences in frequency of proactive and protective measures on public versus private lands (Chi-square)   
  
diff\_tb <- data.frame('preempt' = c(15,21), 'respon' = c(2,14))  
rownames(diff\_tb) <- c("private", "public")  
  
chsq <- chisq.test(diff\_tb)  
chsq$residuals

## preempt respon  
## private 0.9417419 -1.4126129  
## public -0.6563301 0.9844952

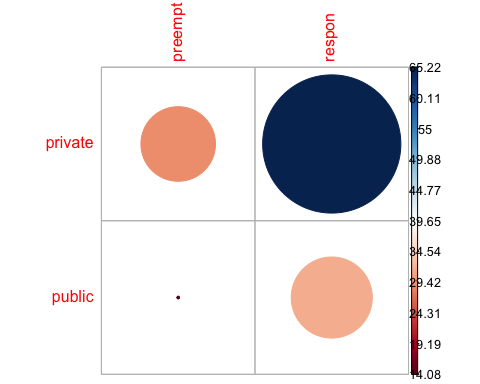
corrplot(chsq$residuals, is.cor = FALSE)



contrib <- 100\*chsq$residuals^2/chsq$statistic  
round(contrib, 3)

## preempt respon  
## private 28.988 65.224  
## public 14.080 31.680

corrplot(contrib, is.cor = FALSE)



### I think I need to use fisher's exact here because 1 out of 4 values is less than 5...   
  
fish <- fisher.test(diff\_tb)  
fish

##   
## Fisher's Exact Test for Count Data  
##   
## data: diff\_tb  
## p-value = 0.05541  
## alternative hypothesis: true odds ratio is not equal to 1  
## 95 percent confidence interval:  
## 0.9004998 50.4253804  
## sample estimates:  
## odds ratio   
## 4.86386

### ### ### Difference in protective versus proactive for plants versus animals  
  
act\_taxa <- s  
act\_taxa <- act\_taxa %>% select(-rowname)  
  
#add vert and invert to see if can get values up for chi square  
act\_taxa <- act\_taxa %>% mutate(allvert = vert + invert)  
act\_taxa <- act\_taxa %>% select(-vert, -invert)  
  
## \*\*\* issue here - values are still too low   
  
### ### ### Difference in public versus private lands actions for plant versus animal  
  
land\_taxa <- sactions  
land\_taxa <- land\_taxa %>% select(-rowname)  
rname <- rownames(land\_taxa)  
  
#add vert and invert to see if can get values up for chi square  
land\_taxa <- land\_taxa %>% mutate(allvert = vert + invert)  
land\_taxa <- land\_taxa %>% select(-vert, -invert)  
rownames(land\_taxa) <- rname  
  
## \*\*\* issue here - values are still too low

## 3. Who are the organizations/players involved in conserving the conservation of the species (eg in an agreement of effort)?

#### 3 - Data wrangling

partners <- sub\_initiatves\_partners  
  
# remove rows where there is an NA for partner  
##partners <- partners[-which(is.na(partners$RowSpecInfo)),] ## this might be redudent and cause issues later but for now needed (delete later if run into NA issues)  
  
#need to remove multi partner strings  
partners <- partners[-which(partners$Type\_of\_Org == "multi"),]  
  
#remove black "NA" partners  
partners <- partners[-which(is.na(partners$Partners)),]  
  
# also want to do a check of spelling to see if sort partner names alphabetically, do any different verions show up (might not even be user error - some entities had name changes that were flagged in FR)  
# check complete - ran:   
 #mod\_partner\_count <- sub\_partners %>% group\_by(scientific\_name) %>% distinct(Partners)  
# and looks good   
  
  
## count number of partners for each species  
## make two versions   
 # one normal  
 # the other where remove text after - (eg BLM-Suprise Feild office would be BLM)  
  
normal\_partner\_count <- partners %>% group\_by(scientific\_name) %>%   
 distinct(Partners) %>% summarise(n())   
#nothing removed or changed   
  
names(normal\_partner\_count) <- c("scientific\_name", "normal\_count" )  
  
#remove flattailed outlier   
  
no\_lizard <- normal\_partner\_count  
  
no\_lizard <- no\_lizard %>% filter(!grepl("Phrynosoma", scientific\_name))  
names(no\_lizard) <- c("scientific\_name", "no\_liz" )  
  
#remove offices  
sub\_partners <- partners  
sub\_partners$Partners <- gsub("(.\*)-.\*", "\\1", sub\_partners$Partners) ## Altered text from https://stackoverflow.com/questions/25307899/r-remove-anything-after-comma-from-columnn   
  
mod\_partner\_count <- sub\_partners %>% group\_by(scientific\_name) %>%   
 distinct(Partners) %>% summarise(n())   
  
names(mod\_partner\_count) <- c("scientific\_name", "no\_offices" )  
  
### addressing issue that Flattailed Horned lizard has so many partners -   
## after combine Federal agency offices, remove all remaining partners with "district" (otherwise will remove federal district offices if do so before - one work around might be to only remove district for this partner) issue is that the partner has 23 partners that are a subsection of county and getting repeated multiple times   
  
no\_district <- sub\_partners %>%   
 mutate(district = case\_when(grepl("district", Partners, ignore.case = TRUE) ~ 1)) # Finding all cases of district and replacing with NA  
  
#remove black the now "NA" district partners  
no\_district\_part <- no\_district[-which(no\_district$district == 1),]  
  
no\_district\_part\_count <- no\_district\_part %>% group\_by(scientific\_name) %>%   
 distinct(Partners) %>% summarise(n()) #count modified number of partners here now  
  
names(no\_district\_part\_count) <- c("scientific\_name", "no\_distinct")  
  
# note - getting a bug when use duplicate so next time try match\_df from plyr   
  
### ### So need to decide on bins

o Summary stats… histograms/bar charts… (perhaps stacked bar charts where color code stacks for low, medium, high confidence in having collected most of partners)?

o To account for the uncertainty in number of partners, we put this information into bins and then “then use an ordered probit to look at how well variables predict the rough estimate of numbers of partners.”

o We use logit regression models to predict participation by different “types” of partners

#### Set up model

## Add number of partners   
  
partners <- full\_join(normal\_partner\_count, mod\_partner\_count) %>% full\_join(no\_district\_part\_count) %>% full\_join(no\_lizard)

## Joining, by = "scientific\_name"  
## Joining, by = "scientific\_name"  
## Joining, by = "scientific\_name"

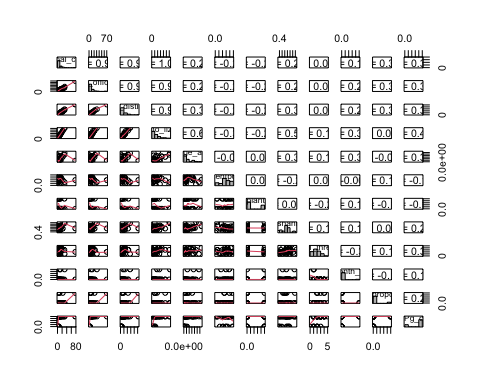
reg\_prdctr <- reg\_prdctr %>% left\_join(partners)

## Joining, by = "scientific\_name"

#note - this variable was made in a chunk below so won't be able to knit script without bringing that information forward   
  
  
## rearrange order  
  
reg\_prdctr <- reg\_prdctr %>%   
 select(-no\_iniv) %>%   
 select(scientific\_name, common\_name, normal\_count:no\_liz, range\_area: Big\_Pg\_count)

### Base model

# Correlation Matrix of Predictors vs Responses ##  
## modified some code from http://handlesman.blogspot.com/2011/03/matrix-plot-with-confidence-intervals.html  
#Not working because of NAs in data - getting error that x is not numeric   
  
  
#set up  
PredictorsOnlyPixel <- reg\_prdctr[,c(3:7)]  
PredictAndResponsePixel <- reg\_prdctr[,-c(1:2)]  
PredictAndResponseGrid <- reg\_prdctr[,-c(1:2)]  
   
# put histograms on the diagonal panel   
panel.hist <- function (x,...) # define a function that says what we want to plot in the diagonal  
{  
 usr <- par("usr"); on.exit(par(usr)) # not sure what usr is for?  
 par(usr = c(usr[1:2],0,1.5))  
 h <- hist(x, plot = FALSE)  
 breaks <- h$breaks; nB <- length(breaks) # make the hist   
 y <- h$counts; y <- y/max(y)  
 rect(breaks[-nB], 0, breaks[-1], y, col="grey", ...) # defines what the histogram is going to look like  
}  
  
# put correlations on the upper panels,  
panel.cor <- function(x, y, digits=2, prefix="", cex.cor, ...)  
{  
 usr <- par("usr"); on.exit(par(usr))  
 par(usr = c(0, 1, 0, 1))  
 r <- cor(x, y,use="complete.obs")   
 txt <- format(c(r, 0.123456789), digits=digits)[1]  
 prefix <- "r = "  
 rc <- cor.test(x,y,method = c("pearson")) ## calculate pearsons rho for upper grid  
 txt <- paste(prefix,txt,sep="")  
 text(0.5, 0.5, txt, cex = 1)  
}  
  
## plot a correlation matrix plot that uses the functions specified above to say what to plot where  
 ## this was taken directly from website and still not plotting r values for all   
#pairs(PredictAndResponsePixel[1:6], lower.panel=panel.smooth, cex = .8, diag.panel=panel.hist, cex.labels = 1.2, font.labels=2, upper.panel=panel.cor)  
  
##Pixel level  
  
pairs(PredictAndResponsePixel,lower.panel = panel.smooth, diag.panel=panel.hist,upper.panel=panel.cor)



### Predictors only --> only have one response so not going to run this one  
##pairs(PredictorsOnlyPixel,lower.panel = panel.smooth, diag.panel=panel.hist,upper.panel=panel.cor)  
  
  
#test correlations to look for covariation between predictors  
cor(reg\_prdctr[,-c(1:2)], use = "complete.obs") # matrix seems fine

## normal\_count no\_offices no\_distinct no\_liz  
## normal\_count 1.0000000 0.9801463 0.9803687 1.0000000  
## no\_offices 0.9801463 1.0000000 0.9969452 0.9801463  
## no\_distinct 0.9803687 0.9969452 1.0000000 0.9803687  
## no\_liz 1.0000000 0.9801463 0.9803687 1.0000000  
## range\_area 0.6903805 0.6758389 0.6682025 0.6903805  
## percentpublic -0.3233285 -0.3172476 -0.3345227 -0.3233285  
## plant -0.1751231 -0.2663458 -0.2520721 -0.1751231  
## public\_mang\_shannons\_diversity 0.6376294 0.5967571 0.5984130 0.6376294  
## total\_threats 0.1962556 0.2022437 0.2036186 0.1962556  
## month\_12 0.3871240 0.4388148 0.4264433 0.3871240  
## wd\_proposed 0.1655464 0.2046008 0.2283094 0.1655464  
## Big\_Pg\_count 0.4574510 0.4758349 0.4814419 0.4574510  
## range\_area percentpublic plant  
## normal\_count 0.69038046 -0.323328474 -0.17512312  
## no\_offices 0.67583890 -0.317247573 -0.26634582  
## no\_distinct 0.66820248 -0.334522686 -0.25207205  
## no\_liz 0.69038046 -0.323328474 -0.17512312  
## range\_area 1.00000000 -0.027258762 -0.05221807  
## percentpublic -0.02725876 1.000000000 0.02291764  
## plant -0.05221807 0.022917639 1.00000000  
## public\_mang\_shannons\_diversity 0.52591974 -0.248459091 0.15240607  
## total\_threats 0.20432319 0.074353386 -0.36397151  
## month\_12 0.37700410 -0.009307553 0.17440555  
## wd\_proposed 0.01428200 0.220661642 -0.28539090  
## Big\_Pg\_count 0.41340125 -0.101875709 -0.26468319  
## public\_mang\_shannons\_diversity total\_threats  
## normal\_count 0.6376294 0.19625560  
## no\_offices 0.5967571 0.20224373  
## no\_distinct 0.5984130 0.20361855  
## no\_liz 0.6376294 0.19625560  
## range\_area 0.5259197 0.20432319  
## percentpublic -0.2484591 0.07435339  
## plant 0.1524061 -0.36397151  
## public\_mang\_shannons\_diversity 1.0000000 0.12098223  
## total\_threats 0.1209822 1.00000000  
## month\_12 0.1746740 -0.37176014  
## wd\_proposed -0.1007297 0.02753779  
## Big\_Pg\_count 0.1708481 0.40342256  
## month\_12 wd\_proposed Big\_Pg\_count  
## normal\_count 0.387123980 0.16554644 0.4574510  
## no\_offices 0.438814830 0.20460078 0.4758349  
## no\_distinct 0.426443277 0.22830942 0.4814419  
## no\_liz 0.387123980 0.16554644 0.4574510  
## range\_area 0.377004097 0.01428200 0.4134013  
## percentpublic -0.009307553 0.22066164 -0.1018757  
## plant 0.174405548 -0.28539090 -0.2646832  
## public\_mang\_shannons\_diversity 0.174674003 -0.10072970 0.1708481  
## total\_threats -0.371760141 0.02753779 0.4034226  
## month\_12 1.000000000 -0.11538462 0.2096570  
## wd\_proposed -0.115384615 1.00000000 0.2096570  
## Big\_Pg\_count 0.209656967 0.20965697 1.0000000

#no na.rm in this one instead it's "use"

* Note: percent public variable is still in matrix but was removed from model

VIFS

vif(lm(log(normal\_count) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr))

## range\_area plant   
## 1.216819 1.083752   
## public\_mang\_shannons\_diversity total\_threats   
## 1.245754 1.125229

vif(lm(log(no\_offices) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr))

## range\_area plant   
## 1.216819 1.083752   
## public\_mang\_shannons\_diversity total\_threats   
## 1.245754 1.125229

vif(lm(log(no\_distinct) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr))

## range\_area plant   
## 1.216819 1.083752   
## public\_mang\_shannons\_diversity total\_threats   
## 1.245754 1.125229

vif(lm(log(no\_liz) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr))

## range\_area plant   
## 1.216108 1.089914   
## public\_mang\_shannons\_diversity total\_threats   
## 1.247302 1.130427

lma <- lm(log(normal\_count) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr)  
  
lmb <- lm(log(no\_offices) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr)  
  
lmc <- lm(log(no\_distinct) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr)  
  
lmd <- lm(log(no\_liz) ~ range\_area + plant + public\_mang\_shannons\_diversity + total\_threats,data = reg\_prdctr)  
  
summary(lma)

##   
## Call:  
## lm(formula = log(normal\_count) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1119 -0.5050 0.1168 0.4204 2.5318   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.712e-01 5.138e-01 1.890 0.0678 .  
## range\_area 8.955e-12 4.802e-12 1.865 0.0714 .  
## plant -5.086e-01 3.109e-01 -1.636 0.1117   
## public\_mang\_shannons\_diversity 7.558e-01 4.896e-01 1.544 0.1325   
## total\_threats 3.370e-02 1.388e-01 0.243 0.8098   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8919 on 32 degrees of freedom  
## (5 observations deleted due to missingness)  
## Multiple R-squared: 0.2935, Adjusted R-squared: 0.2052   
## F-statistic: 3.323 on 4 and 32 DF, p-value: 0.02196

summary(lmb)

##   
## Call:  
## lm(formula = log(no\_offices) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.9769 -0.4856 0.1305 0.3691 2.5232   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.750e-01 5.033e-01 1.738 0.0918 .  
## range\_area 8.908e-12 4.704e-12 1.894 0.0674 .  
## plant -6.778e-01 3.046e-01 -2.225 0.0332 \*  
## public\_mang\_shannons\_diversity 6.970e-01 4.796e-01 1.453 0.1559   
## total\_threats 6.391e-02 1.360e-01 0.470 0.6416   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8737 on 32 degrees of freedom  
## (5 observations deleted due to missingness)  
## Multiple R-squared: 0.3353, Adjusted R-squared: 0.2523   
## F-statistic: 4.036 on 4 and 32 DF, p-value: 0.009233

summary(lmc)

##   
## Call:  
## lm(formula = log(no\_distinct) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.9419 -0.4546 0.1126 0.3703 2.2594   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.446e-01 4.860e-01 1.738 0.0919 .  
## range\_area 8.748e-12 4.542e-12 1.926 0.0630 .  
## plant -6.414e-01 2.941e-01 -2.181 0.0366 \*  
## public\_mang\_shannons\_diversity 6.969e-01 4.631e-01 1.505 0.1422   
## total\_threats 6.355e-02 1.313e-01 0.484 0.6317   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8436 on 32 degrees of freedom  
## (5 observations deleted due to missingness)  
## Multiple R-squared: 0.3399, Adjusted R-squared: 0.2573   
## F-statistic: 4.119 on 4 and 32 DF, p-value: 0.008373

summary(lmd)

##   
## Call:  
## lm(formula = log(no\_liz) ~ range\_area + plant + public\_mang\_shannons\_diversity +   
## total\_threats, data = reg\_prdctr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.9119 -0.3432 0.2018 0.4481 1.3741   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.712e-01 4.506e-01 1.712 0.0969 .  
## range\_area 1.014e-11 4.189e-12 2.420 0.0216 \*  
## plant -3.643e-01 2.736e-01 -1.331 0.1927   
## public\_mang\_shannons\_diversity 6.984e-01 4.260e-01 1.639 0.1112   
## total\_threats 6.861e-02 1.211e-01 0.566 0.5752   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7753 on 31 degrees of freedom  
## (6 observations deleted due to missingness)  
## Multiple R-squared: 0.3662, Adjusted R-squared: 0.2844   
## F-statistic: 4.478 on 4 and 31 DF, p-value: 0.005684

#par(mfrow = c(1, 1))  
#plot(lm2)  
  
#qqPlot(residuals(lm2))  
#par(mfrow = c(1, 1))  
  
#plot(predicted(lm2), residuals(lm2))  
#hist(residuals(lm2))  
# how add CI equivalent around q-q plot?   
AIC(lma)

## [1] 103.1644

AIC(lmb)

## [1] 101.6394

AIC(lmc) ## C has lowest value

## [1] 99.04553

AIC(lmd)

## [1] 90.45817

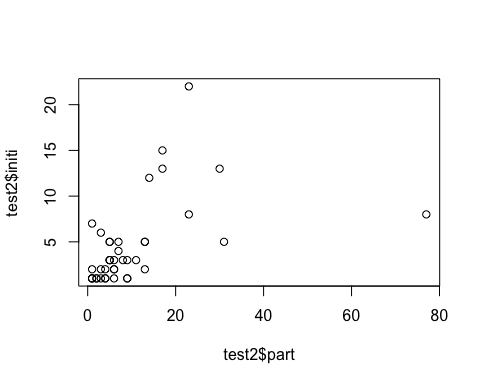
#check\_overdispersion(lm2)

test relationship between number of initiatives and number of partners

test <- full\_join(normal\_partner\_count, names)

## Joining, by = "scientific\_name"

test2 <- test %>% full\_join(no\_init\_per\_sp\_2, by = "common\_name")  
names(test2) <- c("scientific\_name", "part" , "common\_name" , "initi")  
  
plot(test2$initi ~ test2$part)



Q - how many agreement documents are we missing