final\_analysis

Kyle Monper

March 6, 2020

library(tidyverse)  
  
yearlydata <- read\_csv("yearlydata.csv")  
all\_data <- read\_csv("all\_data.csv") #%>% # In original csv, search and replaced cage column "Small" with "small" and "Big" with "big"  
 #filter(janrain != "NA")  
precip <- read\_csv("water\_year.csv") %>%   
 select(-year)

### model

## exogenous drivers -- should be generated on a microhabitat/treatment basis  
# d <- # death rate of seeds (how can we figure this out!?!)  
# f <- # fecundity - note: need to find a way to model this so we can randomize based on some known mean and sd  
# g <- # germination rate -- (same note^^^)  
  
fecund\_model <- glm(seedpods ~ decrain + janrain + febrain + marchrain + aprilrain + mayrain + cage, data = all\_data, family = poisson)   
summary(fecund\_model)

##   
## Call:  
## glm(formula = seedpods ~ decrain + janrain + febrain + marchrain +   
## aprilrain + mayrain + cage, family = poisson, data = all\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.726 -4.726 -3.714 -0.999 32.284   
##   
## Coefficients: (5 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.622772 0.076123 21.318 < 2e-16 \*\*\*  
## decrain 0.032982 0.003175 10.387 < 2e-16 \*\*\*  
## janrain NA NA NA NA   
## febrain NA NA NA NA   
## marchrain NA NA NA NA   
## aprilrain NA NA NA NA   
## mayrain NA NA NA NA   
## cagenone -13.079616 172.654523 -0.076 0.93961   
## cagesmall 0.153921 0.057386 2.682 0.00731 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 8614.1 on 282 degrees of freedom  
## Residual deviance: 8473.7 on 279 degrees of freedom  
## (79 observations deleted due to missingness)  
## AIC: 8977.2  
##   
## Number of Fisher Scoring iterations: 9

#!!!!!!! NOTE: still need to update this based on new bank data  
germ\_model <- glm(cbind(germination, bank - germination) ~ decrain + janrain, data = yearlydata, family=binomial())  
summary(germ\_model)

##   
## Call:  
## glm(formula = cbind(germination, bank - germination) ~ decrain +   
## janrain, family = binomial(), data = yearlydata)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -6.2251 -1.2786 -1.0455 -0.0531 8.7682   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.0896912 0.1069928 -38.224 < 2e-16 \*\*\*  
## decrain 0.0268804 0.0013475 19.949 < 2e-16 \*\*\*  
## janrain -0.0039320 0.0006877 -5.718 1.08e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 2025.7 on 323 degrees of freedom  
## Residual deviance: 1405.7 on 321 degrees of freedom  
## AIC: 1826.6  
##   
## Number of Fisher Scoring iterations: 5

sim\_lupin <- function(planted, cage\_type) {  
   
   
 N <- vector(length = 5)  
 b <- vector(length = 6)  
 b[1] <- planted  
   
 d <- .17  
   
 ## randomize years to be selected  
 years <- ceiling(runif(5, 0, 13))  
   
 ## get random precip years + add cage type to df to be used w/in predict  
 precip$cage <- cage\_type  
 precip\_year <- precip[years,]  
   
 f <- predict(fecund\_model, precip\_year, type = "response") \* 2 # x2 for number of sees per pod  
 f[f<0] <- 0  
  
   
 ## need to figure out the proper distribution for these ( for each treatment type?)  
 g <- predict(germ\_model, precip\_year, type = "response")  
 g[g<0] <- 0  
  
 for(year in 1:5) {  
 N[year] <- b[year]\* (1 - d) \* g[year]  
 b[year + 1] <- (N[year] \* f[year]) + (b[year] - N[year])  
 }  
   
 res <- data.frame(year = seq(1,5), pop = N[1:5], bank = b[1:5])  
 res$scenario <- planted  
   
 return(res)  
   
}  
  
test <- sim\_lupin(1000, "big")  
test

## year pop bank scenario  
## 1 1 14.42993 1000.000 1000  
## 2 2 46.33867 1227.313 1000  
## 3 3 71.07053 2930.622 1000  
## 4 4 62.87629 4482.847 1000  
## 5 5 75.50325 5232.406 1000

Dependent variable:

Germination

Fecundity

logistic

Poisson

(1)

(2)

Dec Rain

0.027\*\*\* (0.001)

0.033\*\*\* (0.003)

Jan Rain

-0.004\*\*\* (0.001)

Feb Rain

Mar Rain

Apr Rain

May Rain

Big Cage

-13.080 (172.655)

No Cage

0.154\*\*\* (0.057)

Small Cage

-4.090\*\*\* (0.107)

1.623\*\*\* (0.076)

Observations

324

283

Log Likelihood

-910.322

-4,484.584

Akaike Inf. Crit.

1,826.644

8,977.167

Note:

*p<0.1;* ***p<0.05;*** p<0.01

[1] “”  
[2] “

Dependent variable:

"  
[3] “

"  
[4] “

Germination

Fecundity

"  
[5] “

logistic

Poisson

"  
[6] “

(1)

(2)

"  
[7] “

Dec Rain

0.027\*\*\* (0.001)

0.033\*\*\* (0.003)

"  
[8] “

Jan Rain

-0.004\*\*\* (0.001)

"  
[9] “

Feb Rain

"  
[10] “

Mar Rain

"  
[11] “

Apr Rain

"  
[12] “

May Rain

"  
[13] “

Big Cage

-13.080 (172.655)

"  
[14] “

No Cage

0.154\*\*\* (0.057)

"  
[15] “

Small Cage

-4.090\*\*\* (0.107)

1.623\*\*\* (0.076)

"  
[16] “

Observations

324

283

"  
[17] “

Log Likelihood

-910.322

-4,484.584

"  
[18] “

Akaike Inf. Crit.

1,826.644

8,977.167

"  
[19] “

Note:

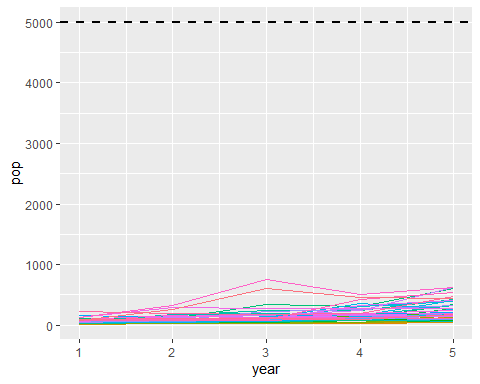
*p<0.1;* ***p<0.05;*** p<0.01

" [20] “

"

# single run - big cage

### how to choose necessary planting  
  
planted <- seq(1000, 6000, 100)  
  
sim\_list <- lapply(planted, sim\_lupin, "big")  
  
sim <- bind\_rows(sim\_list)  
seeds\_needed <- sim %>%   
 filter(year == 5 & pop > 5000) %>%   
 filter(scenario == min(scenario)) %>% ### do we want to select the min or average here?  
 select(scenario) %>%   
 as.numeric()  
  
  
## next to do: monte-carlo simulation to see how randomness effects each of these  
  
  
  
# plot the trend  
ggplot(sim, aes(year, pop, group = as.factor(scenario), color = as.factor(scenario))) +  
 geom\_line() +  
 geom\_hline(yintercept = 5000, linetype = "dashed", size = 1) +  
 theme(legend.position = "none")



# montecarlo run big cage

### this is for if we know the mean and sd for death, fecundity, and germination  
## if any of these are unknown, see the MonteCarlo simulation in ./MonteCarlo.Rmd  
  
num\_trials <- 1000  
  
trial <- vector(length = num\_trials)  
  
planted <- seq(1000, 6000, 100)  
  
for (i in 1:length(trial)) {  
 sim\_list <- lapply(planted, sim\_lupin, "big")  
  
 sim <- bind\_rows(sim\_list)  
   
 seeds\_needed <- sim %>%   
 filter(year == 5 & pop > 5000) %>%   
 filter(scenario == min(scenario)) %>% ### do we want to select the min or average here?  
 select(scenario) %>%   
 as.numeric()  
   
 trial[i] <- seeds\_needed  
   
 # mark progress (for sanity's sake)  
 if(i%%100 == 0) {  
 print(paste(i/length(trial)))  
 }  
  
   
}

## [1] "0.1"  
## [1] "0.2"  
## [1] "0.3"  
## [1] "0.4"  
## [1] "0.5"  
## [1] "0.6"  
## [1] "0.7"  
## [1] "0.8"  
## [1] "0.9"  
## [1] "1"

#hist(trial, main = "big cage treatment")  
  
# mean and sd of seeds needed (is the mode more appropriate here?)  
mean(trial)

## [1] NA

sd(trial)

## [1] NA

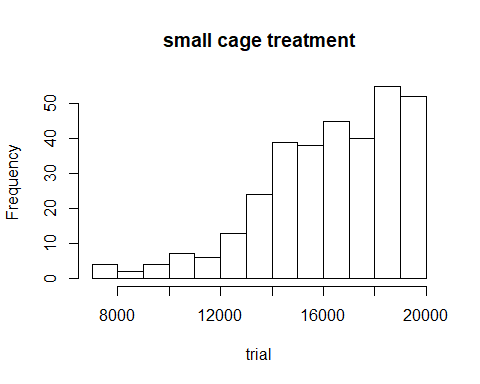
### once we get this mean, we can then frame the question in another way:  
## if we plant X# of seeds, what is the probability of getting to N=5000 by year 5

# montecarlo run small cage

### this is for if we know the mean and sd for death, fecundity, and germination  
## if any of these are unknown, see the MonteCarlo simulation in ./MonteCarlo.Rmd  
  
suppressWarnings({   
 num\_trials <- 1000  
  
trial <- vector(length = num\_trials)  
  
planted <- seq(5000, 20000, 100)  
  
for (i in 1:length(trial)) {  
 sim\_list <- lapply(planted, sim\_lupin, "small")  
  
 sim <- bind\_rows(sim\_list)  
   
 seeds\_needed <- sim %>%   
 filter(year == 5 & pop > 5000) %>%   
 filter(scenario == min(scenario)) %>% ### do we want to select the min or average here?  
 select(scenario) %>%   
 as.numeric()  
   
 trial[i] <- seeds\_needed  
   
 # mark progress (for sanity's sake)  
 if(i%%100 == 0) {  
 print(paste(i/length(trial)))  
 }  
  
   
}})

## [1] "0.1"  
## [1] "0.2"  
## [1] "0.3"  
## [1] "0.4"  
## [1] "0.5"  
## [1] "0.6"  
## [1] "0.7"  
## [1] "0.8"  
## [1] "0.9"  
## [1] "1"

hist(trial, main = "small cage treatment")



# mean and sd of seeds needed (is the mode more appropriate here?)  
mean(trial)

## [1] NA

sd(trial)

## [1] NA

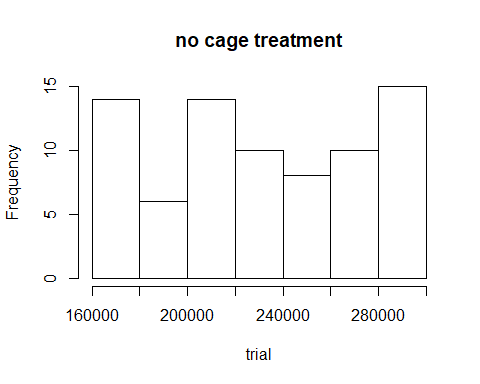
### once we get this mean, we can then frame the question in another way:  
## if we plant X# of seeds, what is the probability of getting to N=5000 by year 5

# montecarlo run no cage

### this is for if we know the mean and sd for death, fecundity, and germination  
## if any of these are unknown, see the MonteCarlo simulation in ./MonteCarlo.Rmd  
  
num\_trials <- 100  
  
trial <- vector(length = num\_trials)  
  
planted <- seq(100000, 300000, 20000)  
  
suppressWarnings({   
 for (i in 1:length(trial)) {  
 sim\_list <- lapply(planted, sim\_lupin, "none")  
  
 sim <- bind\_rows(sim\_list)  
   
 seeds\_needed <- sim %>%   
 filter(year == 5 & pop > 5000) %>%   
 filter(scenario == min(scenario)) %>% ### do we want to select the min or average here?  
 select(scenario) %>%   
 as.numeric()  
   
 trial[i] <- seeds\_needed  
   
 # mark progress (for sanity's sake)  
 if(i%%100 == 0) {  
 print(paste(i/length(trial)))  
 }  
  
   
}})

## [1] "1"

hist(trial, main = "no cage treatment")



# mean and sd of seeds needed (is the mode more appropriate here?)  
mean(trial)

## [1] NA

sd(trial)

## [1] NA

### once we get this mean, we can then frame the question in another way:  
## if we plant X# of seeds, what is the probability of getting to N=5000 by year 5