Results - guild abundance

Jesse Wood

Sept 12, 2018

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

Setup

library("unmarked")

## Loading required package: reshape

## Loading required package: lattice

## Loading required package: parallel

## Loading required package: Rcpp

library("AICcmodavg")  
library("VGAM")

## Loading required package: stats4

##   
## Attaching package: 'stats4'

## The following object is masked from 'package:unmarked':  
##   
## mle

## Loading required package: splines

##   
## Attaching package: 'VGAM'

## The following object is masked from 'package:AICcmodavg':  
##   
## AICc

setwd("C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds")

# PC Data Only -

all counts (1-4 in 2017, 1-3 in 2018) analyzed using pcount() function in unmarked

What you can & can’t use: \*as with before, just don’t use Water, or LowDev, or Impervious HWdens\_1050 & NHW\_saplings ok 500m: evergreen & ag ok, scrubs & ag, high &open ok 1km: scrubs & ag, high&open ok

500 # fine to use evergreen & ag # fine to use scrubs & ag 1 #### can’t use evergreen & ag at 1km # fine to use scrubs & ag 5 # - can’t use Evergreen&Ag, #+ can’t use HighDev&OpenDev together #### can’t use open & water #### can’t use evergreen & open #### can’t use Ag & scrubs 30 #### can’t use evergreen & protected together!! #### can’t use evergreen & scrubs together!! #### can’t use ag & water together!! #### can’t use open & scrubs together!! #+ can’t use Water&Protected together  
#+ can’t use Ag&OpenDev together #+ can’t use Grass&Ag together #- can’t use Protected&Ag together #- can’t use Ag&HighDev together #- can’t use HighDev&OpenDev together #- can’t use Evergreen&Ag together #- can’t use Schrubs&OpenDev together #+ fine to use Schrubs&HighDev together

# Nesting guilds, 4 #

Cavity-nesters (n=16)

# covariates: tree height, age, BA, big trees, snags, open space #burns based on Greenberg paper!  
  
cavity.abund<- csvToUMF("Nesting\_cavity\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
obsCovs(cavity.abund)= scale (obsCovs(cavity.abund))  
sc <- siteCovs(cavity.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(cavity.abund) <- sc

#detection covariates first  
det.null.cavity <- pcount(~1 ~1, cavity.abund, mixture="P", K=50)  
det.weather.cavity <- pcount(~ Wind + Sky ~1, cavity.abund, mixture="P", K=50)  
det.global.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, cavity.abund, mixture="P", K=50)  
det.sound.cavity <- pcount(~ Noise + Wind ~1, cavity.abund, mixture="P", K=50)  
det.date.cavity <- pcount(~ Jdate ~1, cavity.abund, mixture="P", K=50)  
det.detect.cavity <- pcount(~ Jdate + Noise + Time ~1, cavity.abund, mixture="P", K=50)  
det.notdate.cavity <-pcount(~ Wind + Sky + Noise ~1, cavity.abund, mixture="P", K=50)  
det.time.cavity <-pcount(~ Time ~1, cavity.abund, mixture="P",K=50)  
det.timing.cavity <-pcount(~ Time + Jdate ~1, cavity.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.cavity, det.weather.cavity, det.global.cavity,  
 det.sound.cavity, det.date.cavity, det.detect.cavity, det.notdate.cavity,  
 det.time.cavity, det.timing.cavity)

## Warning in fitList(det.null.cavity, det.weather.cavity,  
## det.global.cavity, : Your list was unnamed, so model names were added as  
## object names

msDC.cavity <- modSel(fmsDC)  
msDC.cavity

## nPars AIC delta AICwt cumltvWt  
## det.global.cavity 7 1387.56 0.00 0.3648 0.36  
## det.weather.cavity 4 1388.26 0.69 0.2579 0.62  
## det.time.cavity 3 1389.11 1.55 0.1684 0.79  
## det.notdate.cavity 5 1390.09 2.52 0.1033 0.89  
## det.timing.cavity 4 1390.95 3.38 0.0672 0.96  
## det.detect.cavity 5 1392.69 5.13 0.0281 0.99  
## det.null.cavity 2 1395.59 8.03 0.0066 1.00  
## det.date.cavity 3 1397.49 9.92 0.0026 1.00  
## det.sound.cavity 4 1398.96 11.39 0.0012 1.00

#msDC.cavity@Full  
#old summary: weather, time, date, timing, global, null, notdate all under 2.0  
#2018: changed! #summary: global, weather, time

det.global.cavity

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ 1, data = cavity.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.03 0.254 11.9 7.28e-33  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.9804 0.3500 -2.801 0.00509  
## Jdate -0.0257 0.0323 -0.794 0.42740  
## Wind -0.0262 0.0331 -0.792 0.42848  
## Sky -0.0925 0.0333 -2.774 0.00553  
## Noise 0.0143 0.0328 0.436 0.66306  
## Time 0.0681 0.0328 2.076 0.03787  
##   
## AIC: 1387.564

confint(det.global.cavity, type="det", method="normal")

## 0.025 0.975  
## p(Int) -1.666388240 -0.29443394  
## p(Jdate) -0.089014598 0.03770275  
## p(Wind) -0.091139646 0.03869072  
## p(Sky) -0.157873861 -0.02715501  
## p(Noise) -0.050071791 0.07869614  
## p(Time) 0.003812246 0.13229484

confint(det.global.cavity, type="state", method="normal")

## 0.025 0.975  
## lam(Int) 2.534959 3.530589

# neg with sky, pos with time  
  
det.weather.cavity

##   
## Call:  
## pcount(formula = ~Wind + Sky ~ 1, data = cavity.abund, K = 50,   
## mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.17 0.293 10.8 2.91e-27  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.15926 0.3852 -3.0096 0.00262  
## Wind -0.00251 0.0300 -0.0836 0.93338  
## Sky -0.09762 0.0307 -3.1797 0.00147  
##   
## AIC: 1388.257

confint(det.weather.cavity, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.91421090 -0.40429947  
## p(Wind) -0.06133441 0.05631648  
## p(Sky) -0.15779803 -0.03744803

confint(det.weather.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.592675 3.740343

det.time.cavity

##   
## Call:  
## pcount(formula = ~Time ~ 1, data = cavity.abund, K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.13 0.272 11.5 1.42e-30  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.1064 0.3626 -3.05 0.00228  
## Time 0.0843 0.0302 2.79 0.00532  
##   
## AIC: 1389.11

confint(det.time.cavity, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.81712899 -0.3956216  
## p(Time) 0.02500758 0.1435572

confint(det.time.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.594468 3.661193

Cavity-nester detection models: global best (- with sky, + with survey time) weather second best (- with sky) time third best (+ with time)

write.table(msDC.cavity@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_cavity\_top\_models\_msDC.xls",sep="\t")

#run this when have CSV with both years  
null.cavity <- pcount(~1 ~1, cavity.abund, mixture="P", K=80)  
year.cavity <- pcount(~1 ~ YearCat, cavity.abund, mixture="P", K=80)  
fms.year.cavity<- fitList(null.cavity, year.cavity)

## Warning in fitList(null.cavity, year.cavity): Your list was unnamed, so  
## model names were added as object names

year.ms.cavity<-modSel(fms.year.cavity)  
year.ms.cavity

## nPars AIC delta AICwt cumltvWt  
## null.cavity 2 1395.59 0.00 0.59 0.59  
## year.cavity 3 1396.33 0.74 0.41 1.00

^ year is also under d2 but not best fit

# need to update below with global dets Jdate + Wind + Sky + Noise +Time  
# then add YearCat  
  
##site covariates next (global model for detection covariates)  
null.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, cavity.abund, mixture="P", K=80)  
global.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + Herbicide + BA + Nsnags +Ccover  
 + Ldepth + TreeHt + Age + TimeSinceB + TimeSinceT + Nthins + Nburns  
 + HW\_dens\_1050 + NP\_over\_20cm  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , cavity.abund, mixture="P", K=80) #+ FPSiteIndex  
local.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , cavity.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Age + TreeHt + BA + NP\_over\_20cm + Nsnags + Rel\_HW2P\_canopy + YearCat  
 , cavity.abund, mixture="P", K=80)  
#tree height, age, BA, big trees, snags, open space #burns based on Greenberg paper!  
landmetrics.cavity <- pcount (~ Jdate + Wind + Sky + Noise +Time  
 ~ Parea + ShapeIndex + YearCat  
 , cavity.abund, mixture="P",K=80)  
landscape500.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen500m + HighDev500m + Schrubs500m + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape1.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen1km + HighDev1km + Schrubs1km + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape5.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape30.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen30km + HighDev30km + YearCat  
 , cavity.abund, mixture="P", K=80) #had to remove Protected  
treatment.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + Nthins + YearCat  
 , cavity.abund, mixture ="P", K=80)  
management.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , cavity.abund, mixture="P", K=80)  
disturbance.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , cavity.abund, mixture="P", K=80)  
siteprod.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ PISoils + NSoilTypes + YearCat  
 , cavity.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.cavity <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ BA + Nsnags + Nburns + YearCat  
 , cavity.abund, mixture="P", K=80)  
coord.cavity <-pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Latitude + Longitude + YearCat, cavity.abund, mixture="P", K=80)  
  
  
fmsCN <- fitList(null.cavity, global.cavity, local.cavity, lh.cavity, landmetrics.cavity,  
 landscape500.cavity, landscape1.cavity, landscape5.cavity, landscape30.cavity,  
 treatment.cavity, management.cavity, disturbance.cavity,  
 siteprod.cavity, greenberg.cavity, coord.cavity)

## Warning in fitList(null.cavity, global.cavity, local.cavity, lh.cavity, :  
## Your list was unnamed, so model names were added as object names

ms.cavity <- modSel(fmsCN)  
ms.cavity

## nPars AIC delta AICwt cumltvWt  
## local.cavity 11 1384.58 0.00 3.1e-01 0.31  
## siteprod.cavity 10 1385.49 0.91 2.0e-01 0.50  
## landmetrics.cavity 10 1385.67 1.09 1.8e-01 0.68  
## greenberg.cavity 11 1386.45 1.87 1.2e-01 0.80  
## null.cavity 7 1387.56 2.99 6.9e-02 0.87  
## disturbance.cavity 10 1389.30 4.73 2.9e-02 0.90  
## landscape5.cavity 11 1389.75 5.18 2.3e-02 0.93  
## coord.cavity 10 1390.10 5.53 1.9e-02 0.95  
## lh.cavity 14 1390.53 5.96 1.6e-02 0.96  
## treatment.cavity 12 1390.85 6.27 1.3e-02 0.97  
## landscape30.cavity 10 1391.76 7.19 8.5e-03 0.98  
## landscape1.cavity 11 1392.23 7.66 6.7e-03 0.99  
## management.cavity 15 1392.69 8.11 5.4e-03 0.99  
## landscape500.cavity 11 1392.82 8.24 5.0e-03 1.00  
## global.cavity 29 1402.99 18.42 3.1e-05 1.00

#ms.cavity@Full

#local, siteprod, landmetrics, greenberg all under 2.0  
local.cavity

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Ccover +   
## TreeHt + Ldepth + YearCat, data = cavity.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.3002 0.3672 8.987 2.53e-19  
## Ccover -0.0334 0.0302 -1.104 2.70e-01  
## TreeHt 0.0228 0.0325 0.701 4.83e-01  
## Ldepth -0.0773 0.0331 -2.333 1.97e-02  
## YearCatB -0.1114 0.0673 -1.654 9.81e-02  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.2701 0.4682 -2.713 0.00667  
## Jdate -0.0264 0.0310 -0.854 0.39315  
## Wind -0.0393 0.0329 -1.194 0.23244  
## Sky -0.0788 0.0322 -2.445 0.01450  
## Noise 0.0284 0.0313 0.906 0.36482  
## Time 0.0676 0.0315 2.147 0.03178  
##   
## AIC: 1384.576

confint(local.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.58051726 4.01994879  
## lam(Ccover) -0.09260950 0.02586735  
## lam(TreeHt) -0.04086895 0.08639537  
## lam(Ldepth) -0.14226207 -0.01235770  
## lam(YearCatB) -0.24332757 0.02059570

siteprod.cavity

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ PISoils +   
## NSoilTypes + YearCat, data = cavity.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.17e+00 0.3042 10.4053 2.34e-25  
## PISoils 4.08e-05 0.0291 0.0014 9.99e-01  
## NSoilTypes -7.67e-02 0.0300 -2.5536 1.07e-02  
## YearCatB -7.51e-02 0.0608 -1.2359 2.16e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.11423 0.4016 -2.774 0.00553  
## Jdate -0.02446 0.0318 -0.770 0.44138  
## Wind -0.03424 0.0335 -1.021 0.30715  
## Sky -0.08459 0.0330 -2.567 0.01026  
## Noise 0.00361 0.0322 0.112 0.91071  
## Time 0.06877 0.0323 2.132 0.03305  
##   
## AIC: 1385.486

confint(siteprod.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.56902140 3.76143624  
## lam(PISoils) -0.05690271 0.05698426  
## lam(NSoilTypes) -0.13550601 -0.01782276  
## lam(YearCatB) -0.19427736 0.04401346

landmetrics.cavity

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Parea +   
## ShapeIndex + YearCat, data = cavity.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.1557 0.2978 10.596 3.10e-26  
## Parea -0.0816 0.0329 -2.484 1.30e-02  
## ShapeIndex 0.0228 0.0302 0.754 4.51e-01  
## YearCatB -0.0786 0.0610 -1.288 1.98e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.1003 0.3945 -2.7890 0.00529  
## Jdate -0.0285 0.0317 -0.8991 0.36860  
## Wind -0.0428 0.0337 -1.2716 0.20350  
## Sky -0.0876 0.0331 -2.6487 0.00808  
## Noise 0.0011 0.0325 0.0339 0.97299  
## Time 0.0726 0.0323 2.2490 0.02451  
##   
## AIC: 1385.666

confint(landmetrics.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.57196411 3.73936375  
## lam(Parea) -0.14605059 -0.01723184  
## lam(ShapeIndex) -0.03644699 0.08205011  
## lam(YearCatB) -0.19813886 0.04098364

greenberg.cavity

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ BA + Nsnags +   
## Nburns + YearCat, data = cavity.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.2488 0.3397 9.563 1.15e-21  
## BA -0.0614 0.0331 -1.855 6.36e-02  
## Nsnags 0.0383 0.0282 1.360 1.74e-01  
## Nburns 0.0273 0.0309 0.884 3.77e-01  
## YearCatB -0.0930 0.0607 -1.534 1.25e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.2147 0.4384 -2.771 0.00559  
## Jdate -0.0282 0.0312 -0.903 0.36642  
## Wind -0.0390 0.0332 -1.177 0.23900  
## Sky -0.0804 0.0324 -2.480 0.01314  
## Noise 0.0266 0.0316 0.843 0.39933  
## Time 0.0686 0.0317 2.167 0.03026  
##   
## AIC: 1386.449

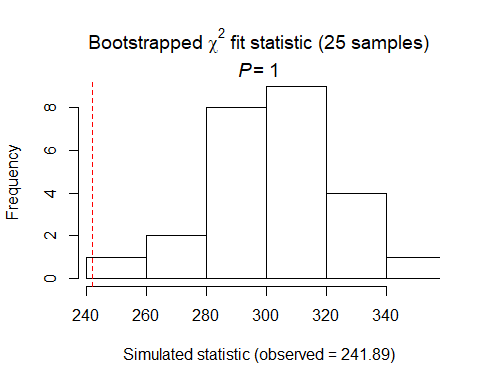
confint(greenberg.cavity, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.58290292 3.91460040  
## lam(BA) -0.12635205 0.00346835  
## lam(Nsnags) -0.01688697 0.09347241  
## lam(Nburns) -0.03329677 0.08795682  
## lam(YearCatB) -0.21192328 0.02582380

write.table(ms.cavity@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_cavity\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(local.cavity, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 241.8878   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 244 289 304 314 353   
##   
## Estimate of c-hat = 0.8

#?Nmix.gof.test()  
Nmix.chisq(local.cavity)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 241.8878

# Cavity-nesters (n=16) Summary:

P distribution DCs: global, weather, time all under 2.0 - with sky code, + with time, SCs using global model: \*changed with 2018 data - Local best (- with litter depth) Site Productivity next best (- with # soil types) Land Metrics next best (- with patch area) Greenberg study next best (none sig)

Tree-nesters (n=25)

#covariates: tree ht, age, density?, canopy? idk else Greenberg: shrubs stem density, maybe burns  
tree.abund<- csvToUMF("Nesting\_tree\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
  
obsCovs(tree.abund)= scale (obsCovs(tree.abund))  
sc <- siteCovs(tree.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(tree.abund) <- sc

#run this when have CSV with both years  
null.tree<- pcount(~1 ~1, tree.abund, mixture="P", K=80)  
year.tree <- pcount(~1 ~ YearCat, tree.abund, mixture="P", K=80)  
fms.year.tree<- fitList(null.tree, year.tree)

## Warning in fitList(null.tree, year.tree): Your list was unnamed, so model  
## names were added as object names

year.ms.tree<-modSel(fms.year.tree)  
year.ms.tree

## nPars AIC delta AICwt cumltvWt  
## year.tree 3 1394.43 0.00 0.979 0.98  
## null.tree 2 1402.10 7.67 0.021 1.00

^ definite year effect! will be important

#detection covariates first  
det.null.tree <- pcount(~1 ~1, tree.abund, mixture="P", K=50)  
det.weather.tree <- pcount(~ Wind + Sky ~1, tree.abund, mixture="P", K=50)  
det.global.tree <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, tree.abund, mixture="P", K=50)  
det.sound.tree <- pcount(~ Noise + Wind ~1, tree.abund, mixture="P", K=50)  
det.date.tree <- pcount(~ Jdate ~1, tree.abund, mixture="P", K=50)  
det.detect.tree <- pcount(~ Jdate + Noise + Time ~1, tree.abund, mixture="P", K=50)  
det.notdate.tree <-pcount(~ Wind + Sky + Noise ~1, tree.abund, mixture="P", K=50)  
det.time.tree <-pcount(~ Time ~1, tree.abund, mixture="P",K=50)  
det.timing.tree <-pcount(~ Time + Jdate ~1, tree.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.tree, det.weather.tree, det.global.tree,  
 det.sound.tree, det.date.tree, det.detect.tree, det.notdate.tree,  
 det.time.tree, det.timing.tree)

## Warning in fitList(det.null.tree, det.weather.tree, det.global.tree,  
## det.sound.tree, : Your list was unnamed, so model names were added as  
## object names

msDC.tree <- modSel(fmsDC)  
msDC.tree

## nPars AIC delta AICwt cumltvWt  
## det.detect.tree 5 1399.77 0.000 0.224 0.22  
## det.date.tree 3 1399.81 0.038 0.220 0.44  
## det.sound.tree 4 1401.02 1.251 0.120 0.56  
## det.notdate.tree 5 1401.24 1.469 0.108 0.67  
## det.timing.tree 4 1401.46 1.692 0.096 0.77  
## det.weather.tree 4 1401.94 2.168 0.076 0.84  
## det.null.tree 2 1402.10 2.327 0.070 0.91  
## det.global.tree 7 1402.40 2.624 0.060 0.97  
## det.time.tree 3 1404.09 4.320 0.026 1.00

#msDC.tree@Full  
#summary now: detect, date, sound, notdate, timing all under 2.0

detect (+ with Date, close with noise)

det.detect.tree

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ 1, data = tree.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.14 0.281 11.2 4.27e-29  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.9349 0.3918 -2.39 0.0170  
## Jdate 0.0667 0.0289 2.31 0.0209  
## Noise -0.0605 0.0323 -1.87 0.0610  
## Time 0.0196 0.0279 0.70 0.4836  
##   
## AIC: 1399.772

confint(det.detect.tree, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.70279307 -0.167075855  
## p(Jdate) 0.01008129 0.123282229  
## p(Noise) -0.12376907 0.002780148  
## p(Time) -0.03517911 0.074311517

confint(det.detect.tree, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.592538 3.692859

det.date.tree

##   
## Call:  
## pcount(formula = ~Jdate ~ 1, data = tree.abund, K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.17 0.29 10.9 7.92e-28  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.9784 0.4001 -2.45 0.0145  
## Jdate 0.0555 0.0274 2.03 0.0427  
##   
## AIC: 1399.81

confint(det.date.tree, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.762513367 -0.1943032  
## p(Jdate) 0.001826213 0.1091546

confint(det.date.tree, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.605185 3.743136

write.table(msDC.tree@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_tree\_top\_models\_msDC.xls",sep="\t")

##site covariates next  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
#with detect covariates - Jdate + Noise + Time  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
null.tree <- pcount(~ Jdate + Noise + Time ~1, tree.abund, mixture="P", K=80)  
global.tree <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + Herbicide + BA +Ccover  
 + Ldepth + TreeHt + Age + TimeSinceB + TimeSinceT + Nthins  
 + HW\_dens\_1050 + NHW\_saplings  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , tree.abund, mixture="P", K=80) #FPSiteIndex, middle row, snags  
local.tree <- pcount(~ Jdate + Noise + Time  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , tree.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.tree <- pcount(~ Jdate + Noise + Time  
 ~ Age + TreeHt + BA + NHW\_saplings + Rel\_HW2P\_canopy + YearCat  
 , tree.abund, mixture="P", K=80)  
#covariates: tree ht, age, density?, canopy? idk else Greenberg: shrubs stem density, maybe burns  
landmetrics.tree <- pcount (~ Jdate + Noise + Time  
 ~ Parea + ShapeIndex + YearCat  
 , tree.abund, mixture="P",K=80)  
landscape500.tree <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen500m + HighDev500m + Schrubs500m + OpenDev500m  
 + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape1.tree <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape5.tree <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape30.tree <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen30km + HighDev30km + YearCat  
 , tree.abund, mixture="P", K=80) #removed Protected  
treatment.tree <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + Nthins + YearCat  
 , tree.abund, mixture ="P", K=80)  
management.tree <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , tree.abund, mixture="P", K=80)  
disturbance.tree <- pcount(~ Jdate + Noise + Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , tree.abund, mixture="P", K=80)  
siteprod.tree <- pcount(~ Jdate + Noise + Time ~ PISoils + NSoilTypes + YearCat  
 , tree.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.tree <- pcount(~ Jdate + Noise + Time ~ Ccover + NHW\_saplings + HW\_dens\_1050 + Nburns  
 + YearCat, tree.abund, mixture="P", K=80)  
coord.tree <-pcount(~ Jdate + Noise + Time  
 ~ Latitude + Longitude + YearCat, tree.abund, mixture="P", K=80)  
  
fmsTN <- fitList(null.tree, global.tree, local.tree, lh.tree, landmetrics.tree,  
 landscape500.tree, landscape1.tree, landscape5.tree, landscape30.tree,  
 treatment.tree, management.tree, disturbance.tree,  
 siteprod.tree, greenberg.tree, coord.tree)

## Warning in fitList(null.tree, global.tree, local.tree, lh.tree,  
## landmetrics.tree, : Your list was unnamed, so model names were added as  
## object names

ms.tree <- modSel(fmsTN)  
ms.tree

## nPars AIC delta AICwt cumltvWt  
## landmetrics.tree 8 1389.79 0.00 4.3e-01 0.43  
## disturbance.tree 8 1391.89 2.10 1.5e-01 0.58  
## siteprod.tree 8 1392.48 2.69 1.1e-01 0.69  
## landscape30.tree 8 1393.16 3.37 7.9e-02 0.77  
## local.tree 9 1393.32 3.53 7.3e-02 0.84  
## coord.tree 8 1394.06 4.27 5.1e-02 0.89  
## landscape5.tree 9 1395.30 5.51 2.7e-02 0.92  
## landscape500.tree 10 1395.38 5.59 2.6e-02 0.94  
## management.tree 13 1396.22 6.43 1.7e-02 0.96  
## landscape1.tree 10 1396.59 6.80 1.4e-02 0.98  
## lh.tree 11 1397.16 7.37 1.1e-02 0.99  
## treatment.tree 10 1398.31 8.51 6.1e-03 0.99  
## greenberg.tree 10 1399.07 9.28 4.1e-03 1.00  
## null.tree 5 1399.77 9.98 2.9e-03 1.00  
## global.tree 25 1408.08 18.29 4.6e-05 1.00

#ms.tree@Full

#land metrics only top model! same as 2017 alone - didn't change  
landmetrics.tree

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ Parea + ShapeIndex +   
## YearCat, data = tree.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.5703 0.7035 5.075 3.87e-07  
## Parea 0.0699 0.0262 2.667 7.66e-03  
## ShapeIndex -0.0223 0.0271 -0.821 4.12e-01  
## YearCatB 0.1690 0.0514 3.288 1.01e-03  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.59787 0.8461 -1.888 0.0590  
## Jdate 0.05653 0.0273 2.068 0.0386  
## Noise -0.03910 0.0299 -1.307 0.1911  
## Time 0.00984 0.0260 0.379 0.7045  
##   
## AIC: 1389.793

confint(landmetrics.tree, type="state",method="normal")

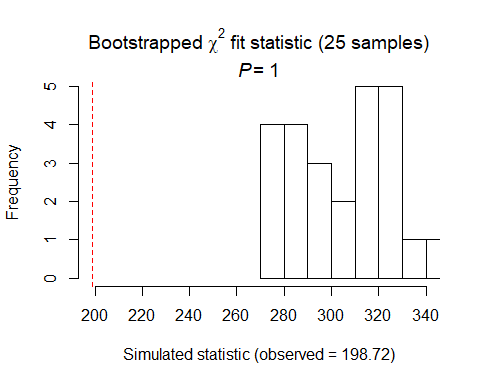
## 0.025 0.975  
## lam(Int) 2.19158382 4.94911169  
## lam(Parea) 0.01853747 0.12135137  
## lam(ShapeIndex) -0.07542678 0.03090339  
## lam(YearCatB) 0.06827130 0.26971942

Shape Metrics (+ with year cat & + with patch area)

write.table(ms.tree@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_tree\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(landmetrics.tree, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 198.7167   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 274 285 307 321 340   
##   
## Estimate of c-hat = 0.65

#?Nmix.gof.test()  
Nmix.chisq(landmetrics.tree)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 198.7167

Tree-nesters (n=25) Summary: P distribution DCs: detect, date, sound, notdate, timing all under 2.0 detect (+ with date, close to - sig with noise but not) SCs using detect: Land Metrics (Area + Shape + Year) only top model! + with Area and with YearCat \*sig

Shrub-nesters (n=17)

#covariates: midstory, HW saplings, BA, greenberg - Nburns, - tree density, + shrub stem density  
shrub.abund<- csvToUMF("Nesting\_shrub\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
  
obsCovs(shrub.abund)= scale (obsCovs(shrub.abund))  
sc <- siteCovs(shrub.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(shrub.abund) <- sc

#run this when have CSV with both years  
null.shrub<- pcount(~1 ~1, shrub.abund, mixture="P", K=80)  
year.shrub <- pcount(~1 ~ YearCat, shrub.abund, mixture="P", K=80)  
fms.year.shrub<- fitList(null.shrub, year.shrub)

## Warning in fitList(null.shrub, year.shrub): Your list was unnamed, so model  
## names were added as object names

year.ms.shrub<-modSel(fms.year.shrub)  
year.ms.shrub

## nPars AIC delta AICwt cumltvWt  
## null.shrub 2 1260.42 0.00 0.73 0.73  
## year.shrub 3 1262.40 1.99 0.27 1.00

^ null better, but year @ delta 1.99 so maybe

det.null.shrub <- pcount(~1 ~1, shrub.abund, mixture="P", K=50)  
det.weather.shrub <- pcount(~ Wind + Sky ~1, shrub.abund, mixture="P", K=50)  
det.global.shrub <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, shrub.abund, mixture="P", K=50)  
det.sound.shrub <- pcount(~ Noise + Wind ~1, shrub.abund, mixture="P", K=50)  
det.date.shrub <- pcount(~ Jdate ~1, shrub.abund, mixture="P", K=50)  
det.detect.shrub <- pcount(~ Jdate + Noise + Time ~1, shrub.abund, mixture="P", K=50)  
det.notdate.shrub <-pcount(~ Wind + Sky + Noise ~1, shrub.abund, mixture="P", K=50)  
det.time.shrub <-pcount(~ Time ~1, shrub.abund, mixture="P",K=50)  
det.timing.shrub <-pcount(~ Time + Jdate ~1, shrub.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.shrub, det.weather.shrub, det.global.shrub,  
 det.sound.shrub, det.date.shrub, det.detect.shrub, det.notdate.shrub,  
 det.time.shrub, det.timing.shrub)

## Warning in fitList(det.null.shrub, det.weather.shrub, det.global.shrub, :  
## Your list was unnamed, so model names were added as object names

msDC.shrub <- modSel(fmsDC)  
msDC.shrub

## nPars AIC delta AICwt cumltvWt  
## det.detect.shrub 5 1237.02 0.00 7.5e-01 0.75  
## det.global.shrub 7 1239.63 2.61 2.0e-01 0.95  
## det.timing.shrub 4 1243.70 6.68 2.7e-02 0.98  
## det.date.shrub 3 1243.98 6.95 2.3e-02 1.00  
## det.sound.shrub 4 1258.15 21.13 1.9e-05 1.00  
## det.notdate.shrub 5 1259.57 22.55 9.5e-06 1.00  
## det.null.shrub 2 1260.42 23.39 6.2e-06 1.00  
## det.time.shrub 3 1262.13 25.11 2.6e-06 1.00  
## det.weather.shrub 4 1263.40 26.37 1.4e-06 1.00

#msDC.shrub@Full  
#summary: detection is best model and only under 2.0 #didn't change in 2018

det.detect.shrub

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ 1, data = shrub.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.26 0.136 16.6 9.94e-62  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.2105 0.2450 -0.859 3.90e-01  
## Jdate 0.1954 0.0451 4.332 1.48e-05  
## Noise -0.1397 0.0476 -2.935 3.34e-03  
## Time 0.0697 0.0405 1.721 8.52e-02  
##   
## AIC: 1237.024

confint(det.detect.shrub, type="det",method="normal")

## 0.025 0.975  
## p(Int) -0.690683786 0.26973194  
## p(Jdate) 0.106986228 0.28376390  
## p(Noise) -0.232951165 -0.04639324  
## p(Time) -0.009671836 0.14901455

confint(det.detect.shrub, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.992724 2.527063

Detect was only top model (+ with date, - with noise, non-sig with time)

write.table(msDC.shrub@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_shrub\_top\_models\_msDC.xls",sep="\t")

##site covariates next  
null.shrub <- pcount(~ Jdate + Noise + Time ~1, shrub.abund, mixture="P", K=80)  
global.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Treatment + Herbicide + BA + Ccover  
 + Ldepth + TreeHt + Age + TimeSinceB + TimeSinceT + Nthins + Nburns  
 + HW\_dens\_1050 + FG\_shrub + NHW\_saplings  
 + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , shrub.abund, mixture="P", K=80) #FPSiteIndex  
local.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , shrub.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.shrub <- pcount(~ Jdate + Noise + Time   
 ~ BA + NHW\_saplings + FG\_shrub + HW\_dens\_1050 + YearCat  
 , shrub.abund, mixture="P", K=80)  
#covariates: midstory, HW saplings, BA, greenberg - Nburns, - tree density, + shrub stem density  
landmetrics.shrub <- pcount (~ Jdate + Noise + Time   
 ~ Parea + ShapeIndex + YearCat  
 , shrub.abund, mixture="P",K=80)  
landscape500.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Evergreen500m + HighDev500m + OpenDev500m  
 + Schrubs500m + Ag500m + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape1.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Ag1km + HighDev1km + OpenDev1km  
 + Schrubs1km + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape5.shrub <- pcount(~ Jdate + Noise + Time   
 ~ OpenDev5km + Schrubs5km + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape30.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Schrubs30km + HighDev30km + Protected30km + YearCat  
 , shrub.abund, mixture="P", K=80)  
treatment.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Treatment + Nthins + YearCat  
 , shrub.abund, mixture ="P", K=80)  
management.shrub <- pcount(~ Jdate + Noise + Time   
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , shrub.abund, mixture="P", K=80)  
disturbance.shrub <- pcount(~ Jdate + Noise + Time   
 ~ TimeSinceB + TimeSinceT + YearCat  
 , shrub.abund, mixture="P", K=80)  
siteprod.shrub <- pcount(~ Jdate + Noise + Time ~ PISoils + NSoilTypes + YearCat  
 , shrub.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.shrub <- pcount(~ Jdate + Noise + Time ~ BA + HW\_dens\_1050 + Nburns + YearCat  
 , shrub.abund, mixture="P", K=80)  
coord.shrub <-pcount (~Jdate + Noise + Time ~ Latitude + Longitude + YearCat  
 , shrub.abund, mixture="P", K=80)  
  
fmsSN <- fitList(null.shrub, global.shrub, local.shrub, lh.shrub, landmetrics.shrub,  
 landscape500.shrub, landscape1.shrub, landscape5.shrub, landscape30.shrub,  
 treatment.shrub, management.shrub, disturbance.shrub,  
 siteprod.shrub, greenberg.shrub, coord.shrub)

## Warning in fitList(null.shrub, global.shrub, local.shrub, lh.shrub,  
## landmetrics.shrub, : Your list was unnamed, so model names were added as  
## object names

ms.shrub <- modSel(fmsSN)  
ms.shrub

## nPars AIC delta AICwt cumltvWt  
## greenberg.shrub 9 1229.54 0.00 6.8e-01 0.68  
## lh.shrub 10 1231.65 2.11 2.4e-01 0.91  
## landscape5.shrub 8 1235.81 6.27 3.0e-02 0.94  
## local.shrub 9 1236.87 7.33 1.7e-02 0.96  
## null.shrub 5 1237.02 7.48 1.6e-02 0.98  
## management.shrub 13 1238.75 9.21 6.8e-03 0.98  
## disturbance.shrub 8 1239.39 9.84 4.9e-03 0.99  
## landscape30.shrub 9 1240.35 10.80 3.1e-03 0.99  
## landscape1.shrub 10 1240.67 11.12 2.6e-03 0.99  
## coord.shrub 8 1241.19 11.65 2.0e-03 1.00  
## treatment.shrub 10 1241.45 11.91 1.8e-03 1.00  
## landmetrics.shrub 8 1241.93 12.38 1.4e-03 1.00  
## landscape500.shrub 11 1242.91 13.36 8.5e-04 1.00  
## siteprod.shrub 8 1242.94 13.39 8.4e-04 1.00  
## global.shrub 26 1259.87 30.32 1.8e-07 1.00

#ms.shrub@Full  
#summary: greenberg model best and only under 2.0 (others from 2017 dropped down)

greenberg.shrub

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ BA + HW\_dens\_1050 +   
## Nburns + YearCat, data = shrub.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.31878 0.1727 13.4264 4.24e-41  
## BA -0.08901 0.0444 -2.0060 4.49e-02  
## HW\_dens\_1050 0.06277 0.0430 1.4599 1.44e-01  
## Nburns 0.02727 0.0400 0.6821 4.95e-01  
## YearCatB -0.00396 0.0754 -0.0525 9.58e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.3322 0.2921 -1.14 2.55e-01  
## Jdate 0.1793 0.0449 3.99 6.54e-05  
## Noise -0.1115 0.0470 -2.37 1.76e-02  
## Time 0.0673 0.0395 1.71 8.80e-02  
##   
## AIC: 1229.544

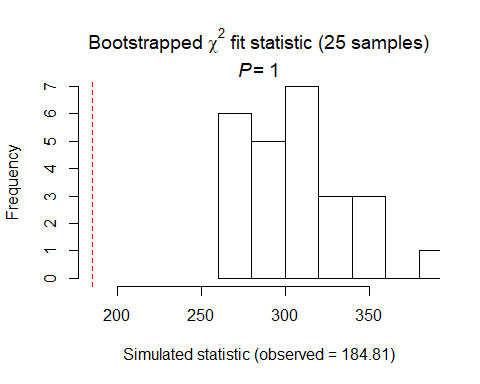
confint(greenberg.shrub, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.98028427 2.657269143  
## lam(BA) -0.17598690 -0.002042734  
## lam(HW\_dens\_1050) -0.02150001 0.147048982  
## lam(Nburns) -0.05109098 0.105630763  
## lam(YearCatB) -0.15170464 0.143786335

write.table(ms.shrub@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_shrub\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(greenberg.shrub, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 184.8143   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 261 288 308 324 384   
##   
## Estimate of c-hat = 0.6

#?Nmix.gof.test()  
Nmix.chisq(greenberg.shrub)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 184.8143

Shrub-nesters (n=17) Summary: P distribution DCs: detect best and only <2.0 (didnt change with year) + with Date, - with Noise (non-sig with time) SCs using Date+Noise+Time (detect) Best model is based on Greenberg et al paper in Southern Apps on this guild - with BA

Ground-nesters (n=10)

#covariates: forbs & grasses at 2 low heights, HW\_dens\_1050, leaf litter depth,  
# Greenberg: -Nburns, -TimeSinceB, leaf litter depth, - Nsnags  
  
ground.abund<- csvToUMF("Nesting\_ground\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
  
obsCovs(ground.abund)= scale (obsCovs(ground.abund))  
sc <- siteCovs(ground.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(ground.abund) <- sc

#run this when have CSV with both years  
null.ground<- pcount(~1 ~1, ground.abund, mixture="P", K=80)  
year.ground <- pcount(~1 ~ YearCat, ground.abund, mixture="P", K=80)  
fms.year.ground<- fitList(null.ground, year.ground)

## Warning in fitList(null.ground, year.ground): Your list was unnamed, so  
## model names were added as object names

year.ms.ground<-modSel(fms.year.ground)  
year.ms.ground

## nPars AIC delta AICwt cumltvWt  
## null.ground 2 1092.33 0.00 0.59 0.59  
## year.ground 3 1093.01 0.69 0.41 1.00

^ null ranked higher but year 0.69 (well under 2)

#detection covariates first  
det.null.ground <- pcount(~1 ~1, ground.abund, mixture="P", K=50)  
det.weather.ground <- pcount(~ Wind + Sky ~1, ground.abund, mixture="P", K=50)  
det.global.ground <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, ground.abund, mixture="P", K=50)  
det.sound.ground <- pcount(~ Noise + Wind ~1, ground.abund, mixture="P", K=50)  
det.date.ground <- pcount(~ Jdate ~1, ground.abund, mixture="P", K=50)  
det.detect.ground <- pcount(~ Jdate + Noise + Time ~1, ground.abund, mixture="P", K=50)  
det.notdate.ground <-pcount(~ Wind + Sky + Noise ~1, ground.abund, mixture="P", K=50)  
det.time.ground <-pcount(~ Time ~1, ground.abund, mixture="P",K=50)  
det.timing.ground <-pcount(~ Time + Jdate ~1, ground.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.ground, det.weather.ground, det.global.ground,  
 det.sound.ground, det.date.ground, det.detect.ground, det.notdate.ground,  
 det.time.ground, det.timing.ground)

## Warning in fitList(det.null.ground, det.weather.ground,  
## det.global.ground, : Your list was unnamed, so model names were added as  
## object names

msDC.ground <- modSel(fmsDC)  
msDC.ground

## nPars AIC delta AICwt cumltvWt  
## det.detect.ground 5 1079.79 0.00 0.49366 0.49  
## det.global.ground 7 1081.19 1.40 0.24495 0.74  
## det.timing.ground 4 1081.78 1.98 0.18302 0.92  
## det.date.ground 3 1083.78 3.99 0.06715 0.99  
## det.time.ground 3 1087.78 7.99 0.00911 1.00  
## det.null.ground 2 1092.33 12.53 0.00094 1.00  
## det.sound.ground 4 1093.20 13.41 0.00060 1.00  
## det.notdate.ground 5 1094.11 14.32 0.00038 1.00  
## det.weather.ground 4 1095.48 15.69 0.00019 1.00

#msDC.ground@Full  
#summary: detect, then global, then timing (date+time) all under 2.0

det.detect.ground

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ 1, data = ground.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 1.96 0.184 10.7 1.43e-26  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.8198 0.2661 -3.08 0.00206  
## Jdate 0.1512 0.0510 2.97 0.00301  
## Noise -0.1136 0.0565 -2.01 0.04436  
## Time -0.0939 0.0513 -1.83 0.06721  
##   
## AIC: 1079.792

confint(det.detect.ground, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.34136072 -0.298287909  
## p(Jdate) 0.05128384 0.251062249  
## p(Noise) -0.22434625 -0.002863419  
## p(Time) -0.19441170 0.006653733

confint(det.detect.ground, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.60305 2.324623

write.table(msDC.ground@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_ground\_top\_models\_msDC.xls",sep="\t")

##site covariates next  
null.ground <- pcount(~ Jdate + Noise + Time ~1, ground.abund, mixture="P", K=80)  
global.ground <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + Herbicide + BA + Nsnags +Ccover  
 + Ldepth + TreeHt + TimeSinceB + TimeSinceT + Nthins + Nburns  
 + HW\_dens\_1050 + FG\_herb + FG\_shrub  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , ground.abund, mixture="P", K=80) #FPSiteIndex  
local.ground <- pcount(~ Jdate + Noise + Time  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , ground.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.ground <- pcount(~ Jdate + Noise + Time  
 ~ FG\_herb + FG\_shrub + HW\_dens\_1050 + Ldepth +  
 Rel\_HW2P\_canopy + BA + YearCat  
 , ground.abund, mixture="P", K=80)  
landmetrics.ground <- pcount (~ Jdate + Noise + Time  
 ~ Parea + ShapeIndex + YearCat  
 , ground.abund, mixture="P",K=80)  
landscape500.ground <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen500m + HighDev500m + Schrubs500m + OpenDev500m  
 + Ag500m + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape1.ground <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape5.ground <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape30.ground <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen30km + HighDev30km + YearCat  
 , ground.abund, mixture="P", K=80) #removed Protected  
treatment.ground <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + Nthins + YearCat  
 , ground.abund, mixture ="P", K=80)  
management.ground <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , ground.abund, mixture="P", K=80)  
disturbance.ground <- pcount(~ Jdate + Noise + Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , ground.abund, mixture="P", K=80)  
siteprod.ground <- pcount(~ Jdate + Noise + Time ~ PISoils + NSoilTypes + YearCat  
 , ground.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.ground <- pcount(~ Jdate + Noise + Time ~ Ccover + Nsnags +  
 Nburns + TimeSinceB + Ldepth + YearCat  
 , ground.abund, mixture="P", K=80)  
coord.ground <- pcount (~Jdate + Noise + Time ~ Latitude + Longitude + YearCat  
 , ground.abund, mixture="P", K=80)  
  
  
fmsGN <- fitList(null.ground, global.ground, local.ground, lh.ground, landmetrics.ground,  
 landscape500.ground, landscape1.ground, landscape5.ground, landscape30.ground,  
 treatment.ground, management.ground, disturbance.ground,  
 siteprod.ground, greenberg.ground, coord.ground)

## Warning in fitList(null.ground, global.ground, local.ground, lh.ground, :  
## Your list was unnamed, so model names were added as object names

ms.ground <- modSel(fmsGN)  
ms.ground

## nPars AIC delta AICwt cumltvWt  
## landscape1.ground 10 1064.74 0.00 9.6e-01 0.96  
## landmetrics.ground 8 1073.42 8.69 1.3e-02 0.98  
## landscape500.ground 11 1074.55 9.81 7.1e-03 0.98  
## disturbance.ground 8 1074.62 9.88 6.9e-03 0.99  
## local.ground 9 1075.00 10.27 5.7e-03 1.00  
## greenberg.ground 11 1076.97 12.23 2.1e-03 1.00  
## landscape5.ground 9 1079.39 14.65 6.3e-04 1.00  
## null.ground 5 1079.79 15.06 5.2e-04 1.00  
## management.ground 13 1080.27 15.54 4.1e-04 1.00  
## landscape30.ground 8 1081.13 16.39 2.7e-04 1.00  
## lh.ground 12 1081.54 16.80 2.2e-04 1.00  
## siteprod.ground 8 1083.23 18.49 9.3e-05 1.00  
## treatment.ground 10 1083.51 18.78 8.1e-05 1.00  
## global.ground 27 1084.14 19.40 5.9e-05 1.00  
## coord.ground 8 1084.24 19.50 5.6e-05 1.00

#ms.ground@Full  
#summary: changed 2018: now only landscape 1km top model

landscape1.ground

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ Evergreen1km + HighDev1km +   
## Schrubs1km + OpenDev1km + YearCat, data = ground.abund, K = 80,   
## mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.0412 0.2662 7.668 1.74e-14  
## Evergreen1km 0.1800 0.0478 3.763 1.68e-04  
## HighDev1km -0.0219 0.0595 -0.367 7.13e-01  
## Schrubs1km 0.0103 0.0491 0.209 8.34e-01  
## OpenDev1km -0.1095 0.0531 -2.062 3.92e-02  
## YearCatB 0.1386 0.0945 1.467 1.42e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.0669 0.3550 -3.01 0.00265  
## Jdate 0.1250 0.0499 2.51 0.01213  
## Noise -0.0686 0.0554 -1.24 0.21542  
## Time -0.0945 0.0499 -1.89 0.05857  
##   
## AIC: 1064.735

confint(landscape1.ground, type="state",method="normal")

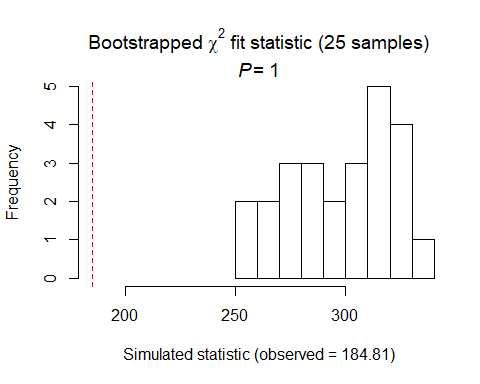
## 0.025 0.975  
## lam(Int) 1.51946811 2.562912147  
## lam(Evergreen1km) 0.08625604 0.273750016  
## lam(HighDev1km) -0.13849463 0.094787626  
## lam(Schrubs1km) -0.08604437 0.106583451  
## lam(OpenDev1km) -0.21365718 -0.005427956  
## lam(YearCatB) -0.04655988 0.323760942

Landscape only top model (+ with evergreen1km, - with opendev1km)

write.table(ms.ground@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Nest\_ground\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(greenberg.shrub, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 184.8143   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 256 277 303 315 336   
##   
## Estimate of c-hat = 0.62

#?Nmix.gof.test()  
Nmix.chisq(greenberg.shrub)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 184.8143

Ground-nesters (n=10) Summary: P distribution DCs: Detect is best, then global, then timing (date+time) all under 2.0 Detect (+ with date, - with noise, close to - with time) SCs using detect: Changed - from 4 to 1 top model Landscape 1km (+ with evergreen1km, - with opendev1km)

# Behavior (foraging) guilds, 3 #

Bark foragers (n=10)

# covariates: tree height, age, BA, big trees, snags, open space #burns based on Greenberg paper!  
# LH: tree height, age, BA, big trees, snags, open space #burns based on Greenberg paper!  
#note: these are same as cavity-nesters right now  
  
bf.abund<- csvToUMF("Behavior\_bf\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
##type may need to change for occupancy (occuRN, pcountOpen, or whichever used) ##  
  
obsCovs(bf.abund)= scale (obsCovs(bf.abund))  
sc <- siteCovs(bf.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(bf.abund) <- sc

#run this when have CSV with both years  
null.bf<- pcount(~1 ~1, bf.abund, mixture="P", K=80)  
year.bf <- pcount(~1 ~ YearCat, bf.abund, mixture="P", K=80)  
fms.year.bf<- fitList(null.bf, year.bf)

## Warning in fitList(null.bf, year.bf): Your list was unnamed, so model names  
## were added as object names

year.ms.bf<-modSel(fms.year.bf)  
year.ms.bf

## nPars AIC delta AICwt cumltvWt  
## null.bf 2 1178.67 0.00 0.72 0.72  
## year.bf 3 1180.56 1.89 0.28 1.00

^ null ranked higher but year 1.89

det.null.bf <- pcount(~1 ~1, bf.abund, mixture="P", K=50)  
det.weather.bf <- pcount(~ Wind + Sky ~1, bf.abund, mixture="P", K=50)  
det.global.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, bf.abund, mixture="P", K=50)  
det.sound.bf <- pcount(~ Noise + Wind ~1, bf.abund, mixture="P", K=50)  
det.date.bf <- pcount(~ Jdate ~1, bf.abund, mixture="P", K=50)  
det.detect.bf <- pcount(~ Jdate + Noise + Time ~1, bf.abund, mixture="P", K=50)  
det.notdate.bf <-pcount(~ Wind + Sky + Noise ~1, bf.abund, mixture="P", K=50)  
det.time.bf <-pcount(~ Time ~1, bf.abund, mixture="P",K=50)  
det.timing.bf <-pcount(~ Time + Jdate ~1, bf.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.bf, det.weather.bf, det.global.bf,  
 det.sound.bf, det.date.bf, det.detect.bf, det.notdate.bf,  
 det.time.bf, det.timing.bf)

## Warning in fitList(det.null.bf, det.weather.bf, det.global.bf,  
## det.sound.bf, : Your list was unnamed, so model names were added as object  
## names

msDC.bf <- modSel(fmsDC)  
msDC.bf

## nPars AIC delta AICwt cumltvWt  
## det.global.bf 7 1162.45 0.00 0.92194 0.92  
## det.timing.bf 4 1168.90 6.45 0.03663 0.96  
## det.detect.bf 5 1170.77 8.32 0.01440 0.97  
## det.time.bf 3 1170.80 8.35 0.01420 0.99  
## det.weather.bf 4 1172.28 9.83 0.00677 0.99  
## det.date.bf 3 1173.76 11.30 0.00324 1.00  
## det.notdate.bf 5 1174.26 11.81 0.00252 1.00  
## det.null.bf 2 1178.67 16.22 0.00028 1.00  
## det.sound.bf 4 1182.54 20.08 0.00004 1.00

#msDC.bf@Full  
#old: time+date best, date, global, detect all under 2.0  
#changed in 2018: global best

det.global.bf

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ 1, data = bf.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.23 0.209 10.7 1.47e-26  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.7166 0.3125 -2.293 0.02183  
## Jdate -0.1431 0.0477 -2.998 0.00272  
## Wind -0.0316 0.0487 -0.649 0.51612  
## Sky -0.1545 0.0475 -3.250 0.00115  
## Noise 0.0092 0.0486 0.189 0.84987  
## Time 0.0746 0.0457 1.633 0.10249  
##   
## AIC: 1162.453

confint(det.global.bf, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.32910537 -0.10417750  
## p(Jdate) -0.23671798 -0.04954873  
## p(Wind) -0.12704475 0.06381307  
## p(Sky) -0.24761953 -0.06131577  
## p(Noise) -0.08610219 0.10451122  
## p(Time) -0.01494551 0.16419159

confint(det.global.bf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.818073 2.636652

global best model (- with Date, - with Sky)

write.table(msDC.bf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_bf\_top\_models\_msDC.xls",sep="\t")

##site covariates next  
# Timing (Time+Date from best model)  
null.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, bf.abund, mixture="P", K=80)  
global.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + Herbicide + BA + Nsnags +Ccover  
 + Ldepth + TreeHt + Age + TimeSinceB + TimeSinceT + Nthins  
 + NP\_over\_20cm  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , bf.abund, mixture="P", K=80) #+ FPSiteIndex  
local.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , bf.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Age + BA + NP\_over\_20cm + Nsnags + Rel\_HW2P\_canopy + YearCat  
 , bf.abund, mixture="P", K=80)  
landmetrics.bf <- pcount (~ Jdate + Wind + Sky + Noise +Time  
 ~ Parea + ShapeIndex + YearCat  
 , bf.abund, mixture="P",K=80)  
landscape500.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen500m + HighDev500m + OpenDev500m + Schrubs500m + YearCat  
 , bf.abund, mixture="P", K=80)  
landscape1.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen1km + HighDev1km + OpenDev500m + Schrubs1km + YearCat  
 , bf.abund, mixture="P", K=80)  
landscape5.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , bf.abund, mixture="P", K=80)  
landscape30.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen30km + HighDev30km + YearCat  
 , bf.abund, mixture="P", K=80) #removed Protected  
treatment.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + Nthins + YearCat  
 , bf.abund, mixture ="P", K=80)  
management.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , bf.abund, mixture="P", K=80)  
disturbance.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , bf.abund, mixture="P", K=80)  
siteprod.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ PISoils + NSoilTypes + YearCat  
 , bf.abund, mixture="P", K=80) #FPSiteIndex  
#greenberg.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time ~ BA + Nsnags + Nburns, bf.abund, mixture="P", K=80)  
coord.bf <- pcount (~Jdate + Wind + Sky + Noise +Time ~ Latitude + Longitude + YearCat  
 , bf.abund, mixture="P", K=80)  
  
  
fmsBF <- fitList(null.bf, global.bf, local.bf, lh.bf, landmetrics.bf,  
 landscape500.bf, landscape1.bf, landscape5.bf, landscape30.bf,  
 treatment.bf, management.bf, disturbance.bf,  
 siteprod.bf, coord.bf) #no greenberg

## Warning in fitList(null.bf, global.bf, local.bf, lh.bf, landmetrics.bf, :  
## Your list was unnamed, so model names were added as object names

ms.bf <- modSel(fmsBF)  
ms.bf

## nPars AIC delta AICwt cumltvWt  
## null.bf 7 1162.45 0.00 3.1e-01 0.31  
## landscape30.bf 10 1163.40 0.95 1.9e-01 0.50  
## coord.bf 10 1163.47 1.02 1.9e-01 0.69  
## local.bf 11 1164.58 2.13 1.1e-01 0.80  
## treatment.bf 12 1166.20 3.75 4.8e-02 0.84  
## landmetrics.bf 10 1166.71 4.25 3.7e-02 0.88  
## landscape1.bf 12 1166.98 4.53 3.2e-02 0.91  
## landscape500.bf 12 1167.78 5.33 2.2e-02 0.94  
## siteprod.bf 10 1167.89 5.43 2.1e-02 0.96  
## disturbance.bf 10 1168.23 5.78 1.7e-02 0.97  
## landscape5.bf 11 1168.25 5.79 1.7e-02 0.99  
## management.bf 15 1170.17 7.72 6.5e-03 1.00  
## lh.bf 13 1171.73 9.28 3.0e-03 1.00  
## global.bf 27 1187.83 25.37 9.6e-07 1.00

#ms.bf@Full  
# null is best, then landscape30km (0.95), then coord (1.02)

landscape30.bf

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Evergreen30km +   
## HighDev30km + YearCat, data = bf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.24812 0.2298 9.7835 1.33e-22  
## Evergreen30km 0.00362 0.0445 0.0814 9.35e-01  
## HighDev30km -0.09768 0.0473 -2.0641 3.90e-02  
## YearCatB 0.02325 0.0864 0.2692 7.88e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.76951 0.3307 -2.3267 0.01998  
## Jdate -0.12632 0.0479 -2.6376 0.00835  
## Wind -0.01673 0.0500 -0.3348 0.73778  
## Sky -0.14551 0.0475 -3.0611 0.00221  
## Noise -0.00268 0.0483 -0.0554 0.95583  
## Time 0.07461 0.0453 1.6468 0.09960  
##   
## AIC: 1163.402

confint(landscape30.bf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.79774441 2.698493204  
## lam(Evergreen30km) -0.08362343 0.090869217  
## lam(HighDev30km) -0.19044294 -0.004926807  
## lam(YearCatB) -0.14601209 0.192515486

coord.bf

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Latitude +   
## Longitude + YearCat, data = bf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.2531 0.2342 9.620 6.57e-22  
## Latitude 0.0665 0.0439 1.516 1.29e-01  
## Longitude -0.0529 0.0420 -1.258 2.08e-01  
## YearCatB 0.0331 0.0860 0.385 7.00e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.78337 0.3357 -2.333 0.01962  
## Jdate -0.12630 0.0479 -2.638 0.00834  
## Wind -0.01689 0.0500 -0.338 0.73542  
## Sky -0.14418 0.0476 -3.030 0.00245  
## Noise -0.00309 0.0482 -0.064 0.94899  
## Time 0.07724 0.0452 1.707 0.08776  
##   
## AIC: 1163.47

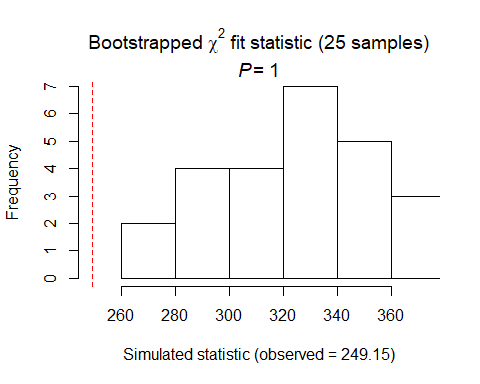
confint(coord.bf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.79407712 2.71215539  
## lam(Latitude) -0.01946541 0.15245385  
## lam(Longitude) -0.13522586 0.02947401  
## lam(YearCatB) -0.13540383 0.20156036

write.table(ms.bf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_bf\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(null.bf, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



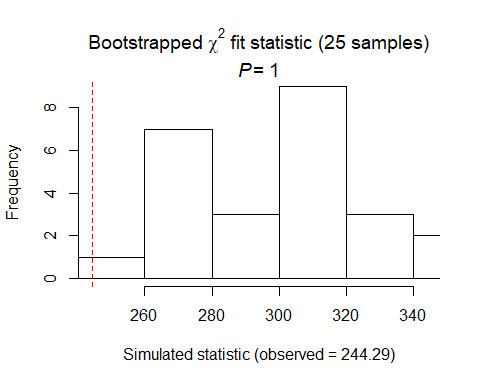
##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 249.1549   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 264 304 325 349 373   
##   
## Estimate of c-hat = 0.77

#?Nmix.gof.test()  
Nmix.chisq(null.bf)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 249.1549

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(landscape30.bf, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 244.2875   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 247 275 303 315 344   
##   
## Estimate of c-hat = 0.82

#?Nmix.gof.test()  
Nmix.chisq(landscape30.bf)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 244.2875

Bark foragers (n=10) Summary: P distribution DCs: Global is best and only under d2.0 (- with Date, - with Sky) SCs using global: Null is still best, second best if landscape30km at d0.95 (- with high development) third best is coord (lat/long) is d1.02 (none sig)

Foliage gleaners (n=21)

# covariates: tree height, age, BA, big trees, snags, open space #burns based on Greenberg paper!  
#LH: tree height, ccover, NOT snags, hardwood ratio (tree diversity)  
#note: similar to tree nesters  
  
fg.abund<- csvToUMF("Behavior\_fg\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
  
obsCovs(fg.abund)= scale (obsCovs(fg.abund))  
sc <- siteCovs(fg.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(fg.abund) <- sc

#run this when have CSV with both years  
null.fg<- pcount(~1 ~1, fg.abund, mixture="P", K=80)  
year.fg <- pcount(~1 ~ YearCat, fg.abund, mixture="P", K=80)  
fms.year.fg<- fitList(null.fg, year.fg)

## Warning in fitList(null.fg, year.fg): Your list was unnamed, so model names  
## were added as object names

year.ms.fg<-modSel(fms.year.fg)  
year.ms.fg

## nPars AIC delta AICwt cumltvWt  
## null.fg 2 1401.85 0.00 0.69 0.69  
## year.fg 3 1403.42 1.57 0.31 1.00

^ null ranked higher but year within 1.57 delta

#detection covariates first  
det.null.fg <- pcount(~1 ~1, fg.abund, mixture="P", K=50)  
det.weather.fg <- pcount(~ Wind + Sky ~1, fg.abund, mixture="P", K=50)  
det.global.fg <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, fg.abund, mixture="P", K=50)  
det.sound.fg <- pcount(~ Noise + Wind ~1, fg.abund, mixture="P", K=50)  
det.date.fg <- pcount(~ Jdate ~1, fg.abund, mixture="P", K=50)  
det.detect.fg <- pcount(~ Jdate + Noise + Time ~1, fg.abund, mixture="P", K=50)  
det.notdate.fg <-pcount(~ Wind + Sky + Noise ~1, fg.abund, mixture="P", K=50)  
det.time.fg <-pcount(~ Time ~1, fg.abund, mixture="P",K=50)  
det.timing.fg <-pcount(~ Time + Jdate ~1, fg.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.fg, det.weather.fg, det.global.fg,  
 det.sound.fg, det.date.fg, det.detect.fg, det.notdate.fg,  
 det.time.fg, det.timing.fg)

## Warning in fitList(det.null.fg, det.weather.fg, det.global.fg,  
## det.sound.fg, : Your list was unnamed, so model names were added as object  
## names

msDC.fg <- modSel(fmsDC)  
msDC.fg

## nPars AIC delta AICwt cumltvWt  
## det.detect.fg 5 1376.60 0.00 6.9e-01 0.69  
## det.global.fg 7 1379.50 2.90 1.6e-01 0.85  
## det.timing.fg 4 1379.88 3.29 1.3e-01 0.98  
## det.date.fg 3 1383.45 6.86 2.2e-02 1.00  
## det.notdate.fg 5 1395.49 18.89 5.4e-05 1.00  
## det.weather.fg 4 1396.26 19.66 3.7e-05 1.00  
## det.sound.fg 4 1399.33 22.74 7.9e-06 1.00  
## det.null.fg 2 1401.85 25.25 2.3e-06 1.00  
## det.time.fg 3 1402.29 25.70 1.8e-06 1.00

#msDC.fg@Full  
#old: notdate (Wind, Sky, Noise) best, weather, sound all under 2.0  
#changed, summary: detect is only one under 2.0

det.detect.fg

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ 1, data = fg.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.65 0.122 21.6 6.2e-104  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.2016 0.2212 -0.912 3.62e-01  
## Jdate 0.1735 0.0370 4.687 2.77e-06  
## Noise -0.0888 0.0389 -2.284 2.24e-02  
## Time 0.0840 0.0340 2.467 1.36e-02  
##   
## AIC: 1376.595

confint(det.detect.fg, type="det",method="normal")

## 0.025 0.975  
## p(Int) -0.63509157 0.23188108  
## p(Jdate) 0.10092816 0.24599487  
## p(Noise) -0.16500461 -0.01260327  
## p(Time) 0.01727597 0.15074732

confint(det.detect.fg, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.409082 2.888708

write.table(msDC.fg@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_fg\_top\_models\_msDC.xls",sep="\t")

#more appropriate detection covariates (Date+Noise+Time from best model)  
null.fg <- pcount(~ Jdate + Noise + Time ~1, fg.abund, mixture="P", K=80)  
global.fg <- pcount(~ Jdate + Noise + Time   
 ~ Treatment + Herbicide + BA + Nsnags +Ccover  
 + Ldepth + TreeHt + Age + TimeSinceB + TimeSinceT + Nthins + Nburns  
 + HW\_dens\_1050 + NP\_over\_20cm  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , fg.abund, mixture="P", K=80) #+ FPSiteIndex  
local.fg <- pcount(~ Jdate + Noise + Time   
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , fg.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.fg <- pcount(~ Jdate + Noise + Time  
 ~ TreeHt + Ccover + Nsnags + Rel\_HW2P\_canopy + NHW\_saplings + FG\_herb  
 + HW\_dens\_1050 + YearCat  
 , fg.abund, mixture="P", K=80)  
landmetrics.fg <- pcount (~ Jdate + Noise + Time  
 ~ Parea + ShapeIndex + YearCat  
 , fg.abund, mixture="P",K=80)  
landscape500.fg <- pcount(~ Jdate + Noise + Time  
 ~ Evergreen500m + OpenDev500m + Schrubs500m + Ag500m + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape1.fg <- pcount(~ Jdate + Noise + Time  
 ~ OpenDev1km + Schrubs1km + Ag1km + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape5.fg <- pcount(~ Jdate + Noise + Time  
 ~ OpenDev5km + Ag5km + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape30.fg <- pcount(~ Jdate + Noise + Time  
 ~ HighDev30km + Protected30km + Schrubs30km + YearCat  
 , fg.abund, mixture="P", K=80)  
treatment.fg <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + Nthins + YearCat  
 , fg.abund, mixture ="P", K=80)  
management.fg <- pcount(~ Jdate + Noise + Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , fg.abund, mixture="P", K=80)  
disturbance.fg <- pcount(~ Jdate + Noise + Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , fg.abund, mixture="P", K=80)  
siteprod.fg <- pcount(~ Jdate + Noise + Time ~ PISoils + NSoilTypes + YearCat  
 , fg.abund, mixture="P", K=80) #FPSiteIndex  
#greenberg.fg <- pcount(~ Jdate + Noise + Time ~ BA + Nsnags + Nburns, fg.abund, mixture="P", K=80)  
coord.ground <- pcount (~Jdate + Noise + Time ~ Latitude + Longitude + YearCat  
 , fg.abund, mixture="P", K=80)  
  
  
fmsFG <- fitList(null.fg, global.fg, local.fg, lh.fg, landmetrics.fg,  
 landscape500.fg, landscape1.fg, landscape5.fg, landscape30.fg,  
 treatment.fg, management.fg, disturbance.fg,  
 siteprod.fg) #no greenberg

## Warning in fitList(null.fg, global.fg, local.fg, lh.fg, landmetrics.fg, :  
## Your list was unnamed, so model names were added as object names

ms.fg <- modSel(fmsFG)  
ms.fg

## nPars AIC delta AICwt cumltvWt  
## landscape1.fg 9 1372.01 0.00 5.0e-01 0.50  
## landscape500.fg 10 1373.68 1.67 2.2e-01 0.72  
## disturbance.fg 8 1374.37 2.36 1.5e-01 0.87  
## null.fg 5 1376.60 4.58 5.1e-02 0.92  
## lh.fg 13 1378.02 6.00 2.5e-02 0.95  
## landscape5.fg 8 1379.10 7.09 1.4e-02 0.96  
## local.fg 9 1380.02 8.00 9.1e-03 0.97  
## management.fg 13 1380.20 8.19 8.3e-03 0.98  
## siteprod.fg 8 1380.33 8.32 7.8e-03 0.99  
## treatment.fg 10 1380.53 8.52 7.1e-03 0.99  
## landmetrics.fg 8 1382.06 10.05 3.3e-03 1.00  
## landscape30.fg 9 1382.49 10.48 2.7e-03 1.00  
## global.fg 27 1392.65 20.63 1.7e-05 1.00

#ms.fg@Full  
#slight change  
#summary: landscape at 1km best, landscape @ 500m second best

landscape1.fg

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ OpenDev1km + Schrubs1km +   
## Ag1km + YearCat, data = fg.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.69570 0.1523 17.705 3.81e-70  
## OpenDev1km -0.01969 0.0320 -0.615 5.38e-01  
## Schrubs1km 0.00713 0.0315 0.227 8.21e-01  
## Ag1km -0.10730 0.0355 -3.021 2.52e-03  
## YearCatB 0.04347 0.0613 0.710 4.78e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.3346 0.2581 -1.30 1.95e-01  
## Jdate 0.1593 0.0366 4.35 1.36e-05  
## Noise -0.0655 0.0384 -1.71 8.82e-02  
## Time 0.0757 0.0332 2.28 2.27e-02  
##   
## AIC: 1372.013

confint(landscape1.fg, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.39729366 2.99411568  
## lam(OpenDev1km) -0.08242703 0.04303882  
## lam(Schrubs1km) -0.05450963 0.06877462  
## lam(Ag1km) -0.17691105 -0.03769324  
## lam(YearCatB) -0.07659870 0.16353974

landscape500.fg

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time ~ Evergreen500m + OpenDev500m +   
## Schrubs500m + Ag500m + YearCat, data = fg.abund, K = 80,   
## mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.69303 0.1512 17.815 5.39e-71  
## Evergreen500m -0.02749 0.0369 -0.745 4.56e-01  
## OpenDev500m -0.01299 0.0335 -0.388 6.98e-01  
## Schrubs500m -0.00908 0.0335 -0.271 7.87e-01  
## Ag500m -0.12552 0.0397 -3.164 1.56e-03  
## YearCatB 0.03657 0.0613 0.596 5.51e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.3236 0.2568 -1.26 2.08e-01  
## Jdate 0.1629 0.0366 4.45 8.60e-06  
## Noise -0.0702 0.0385 -1.82 6.84e-02  
## Time 0.0758 0.0333 2.28 2.29e-02  
##   
## AIC: 1373.681

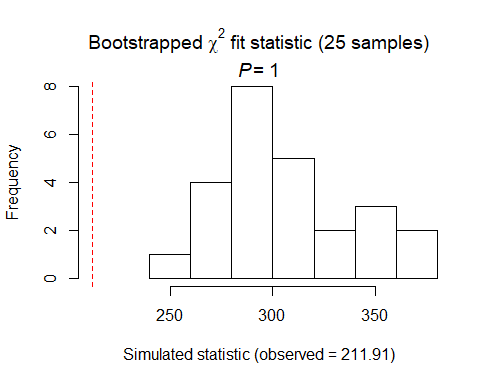
confint(landscape500.fg, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.39674720 2.98930353  
## lam(Evergreen500m) -0.09981518 0.04483222  
## lam(OpenDev500m) -0.07861980 0.05263262  
## lam(Schrubs500m) -0.07480942 0.05664430  
## lam(Ag500m) -0.20327859 -0.04776371  
## lam(YearCatB) -0.08363011 0.15677734

write.table(ms.fg@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_fg\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(landscape1.fg, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



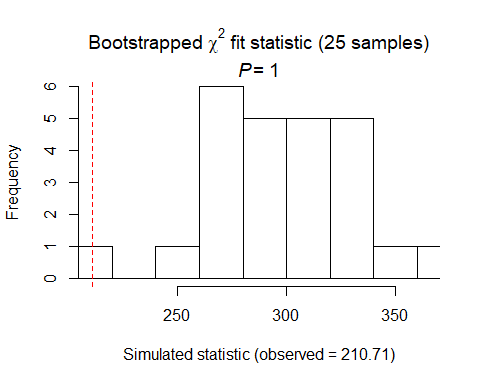
##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 211.9064   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 245 285 299 332 375   
##   
## Estimate of c-hat = 0.69

#?Nmix.gof.test()  
Nmix.chisq(landscape1.fg)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 211.9064

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(landscape500.fg, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 210.7075   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 212 269 293 324 364   
##   
## Estimate of c-hat = 0.71

#?Nmix.gof.test()  
Nmix.chisq(landscape500.fg)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 210.7075

Foliage gleaners (n=21) Summary: P distribution DCs: detect (date, noise, time) is only best + with date and + time, - with noise SCs using Date+Noise+Time: Landscape at 1km is best: Only Ag significant (-) Landscape @ 500m is second best Only Agis significant (-)

Ground foragers (n=27)

#covariates: midstory, HW saplings, BA, greenberg - Nburns, - tree density, + shrub stem density  
  
gf.abund<- csvToUMF("Behavior\_gf\_pcount.csv", long = FALSE, type = "unmarkedFramePCount")  
  
obsCovs(gf.abund)= scale (obsCovs(gf.abund))  
sc <- siteCovs(gf.abund)  
sc[,c(6:77)] <- scale(sc[, c(6:77)])  
siteCovs(gf.abund) <- sc

#run this when have CSV with both years  
null.gf<- pcount(~1 ~1, gf.abund, mixture="P", K=80)  
year.gf <- pcount(~1 ~ YearCat, gf.abund, mixture="P", K=80)  
fms.year.gf<- fitList(null.gf, year.gf)

## Warning in fitList(null.gf, year.gf): Your list was unnamed, so model names  
## were added as object names

year.ms.gf<-modSel(fms.year.gf)  
year.ms.gf

## nPars AIC delta AICwt cumltvWt  
## null.gf 2 1505.07 0.00 0.65 0.65  
## year.gf 3 1506.30 1.23 0.35 1.00

^ null is ranked first but year is |z|) ## 3.42 0.196 17.4 8.34e-68 ## ## Detection: ## Estimate SE z P(>|z|) ## (Intercept) -1.0221 0.2678 -3.816 0.000136 ## Jdate 0.0998 0.0282 3.539 0.000402 ## Wind 0.0487 0.0278 1.753 0.079580 ## Sky -0.0320 0.0264 -1.214 0.224856 ## Noise -0.0602 0.0278 -2.161 0.030724 ## Time -0.0203 0.0261 -0.778 0.436561 ## ## AIC: 1492.058 ```

confint(det.global.gf, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.547026931 -0.497130480  
## p(Jdate) 0.044548028 0.155133080  
## p(Wind) -0.005747544 0.103179048  
## p(Sky) -0.083756002 0.019694086  
## p(Noise) -0.114748667 -0.005588011  
## p(Time) -0.071452704 0.030845231

confint(det.global.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 3.033718 3.803951

write.table(msDC.gf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_gf\_top\_models\_msDC.xls",sep="\t")

##site covariates next  
# detection covariates (global!)  
null.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time ~1, gf.abund, mixture="P", K=80)  
global.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Treatment + Herbicide + BA + Nsnags +Ccover  
 + Ldepth + TreeHt + TimeSinceB + TimeSinceT + Nthins  
 + HW\_dens\_1050 + NP\_over\_20cm + FG\_herb + FG\_shrub  
 + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , gf.abund, mixture="P", K=80) #+ FPSiteIndex  
local.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Ccover + TreeHt + Ldepth+ YearCat  
 , gf.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ FG\_herb + FG\_shrub + HW\_dens\_1050 + Ldepth + Rel\_HW2P\_canopy  
 + BA + NHW\_saplings+ YearCat  
 , gf.abund, mixture="P", K=80)  
#note: these are similar to ground NESTERS now (kinda to shrubs too)  
#covariates: forbes & grasses at 2 low heights, HW\_dens\_1050, leaf litter depth,  
# Greenberg: -Nburns, -TimeSinceB, leaf litter depth, - Nsnags  
landmetrics.gf <- pcount (~ Jdate + Wind + Sky + Noise + Time  
 ~ Parea + ShapeIndex+ YearCat  
 , gf.abund, mixture="P",K=80)  
landscape500.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Evergreen500m + HighDev500m + Schrubs500m + Ag500m+ YearCat  
 , gf.abund, mixture="P", K=80)  
landscape1.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ HighDev1km + Schrubs1km + Ag1km + YearCat  
 , gf.abund, mixture="P", K=80)  
landscape5.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Ag5km + HighDev5km + YearCat  
 , gf.abund, mixture="P", K=80)  
landscape30.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ HighDev30km + Protected30km + YearCat  
 , gf.abund, mixture="P", K=80)  
treatment.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Treatment + Nthins + YearCat  
 , gf.abund, mixture ="P", K=80)  
management.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , gf.abund, mixture="P", K=80)  
disturbance.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , gf.abund, mixture="P", K=80)  
siteprod.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time  
 ~ PISoils + NSoilTypes + YearCat  
 , gf.abund, mixture="P", K=80) #FPSiteIndex  
#greenberg.gf <- pcount(~ Jdate + Wind + Sky + Noise + Time ~ BA + Nsnags + Nburns, gf.abund, mixture="P", K=80)  
# Greenberg: -Nburns, -TimeSinceB, leaf litter depth, - Nsnags  
coord.gf <- pcount (~Jdate + Wind + Sky + Noise + Time  
 ~ Latitude + Longitude + YearCat  
 , gf.abund, mixture="P", K=80)  
  
fmsGF <- fitList(null.gf, global.gf, local.gf, lh.gf, landmetrics.gf,  
 landscape500.gf, landscape1.gf, landscape5.gf, landscape30.gf,  
 treatment.gf, management.gf, disturbance.gf,  
 siteprod.gf, coord.gf) #no greenberg

## Warning in fitList(null.gf, global.gf, local.gf, lh.gf, landmetrics.gf, :  
## Your list was unnamed, so model names were added as object names

ms.gf <- modSel(fmsGF)  
ms.gf

## nPars AIC delta AICwt cumltvWt  
## local.gf 11 1486.67 0.00 5.5e-01 0.55  
## lh.gf 15 1488.59 1.92 2.1e-01 0.77  
## coord.gf 10 1491.18 4.52 5.8e-02 0.82  
## treatment.gf 12 1491.64 4.98 4.6e-02 0.87  
## null.gf 7 1492.00 5.34 3.8e-02 0.91  
## disturbance.gf 10 1492.71 6.04 2.7e-02 0.94  
## landmetrics.gf 10 1493.04 6.37 2.3e-02 0.96  
## siteprod.gf 10 1494.24 7.57 1.3e-02 0.97  
## landscape30.gf 10 1494.47 7.80 1.1e-02 0.98  
## landscape5.gf 10 1495.19 8.52 7.8e-03 0.99  
## management.gf 15 1496.30 9.63 4.5e-03 1.00  
## landscape1.gf 11 1497.30 10.63 2.7e-03 1.00  
## landscape500.gf 12 1497.94 11.28 2.0e-03 1.00  
## global.gf 29 1507.52 20.86 1.6e-05 1.00

#ms.gf@Full  
#summary: local best, life history second

local.gf

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Ccover +   
## TreeHt + Ldepth + YearCat, data = gf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.71324 0.4161 8.924 4.50e-19  
## Ccover -0.07659 0.0245 -3.128 1.76e-03  
## TreeHt 0.00644 0.0268 0.240 8.10e-01  
## Ldepth -0.00729 0.0269 -0.271 7.86e-01  
## YearCatB 0.04764 0.0550 0.866 3.86e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.4381 0.5147 -2.794 0.00521  
## Jdate 0.0905 0.0278 3.259 0.00112  
## Wind 0.0466 0.0273 1.707 0.08776  
## Sky -0.0275 0.0250 -1.101 0.27078  
## Noise -0.0488 0.0262 -1.864 0.06226  
## Time -0.0170 0.0248 -0.685 0.49321  
##   
## AIC: 1486.666

confint(local.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.89770774 4.52877684  
## lam(Ccover) -0.12458050 -0.02860447  
## lam(TreeHt) -0.04611676 0.05899232  
## lam(Ldepth) -0.05999388 0.04542324  
## lam(YearCatB) -0.06013775 0.15541985

lh.gf

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ FG\_herb +   
## FG\_shrub + HW\_dens\_1050 + Ldepth + Rel\_HW2P\_canopy + BA +   
## NHW\_saplings + YearCat, data = gf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.834660 0.2270 16.8945 4.94e-64  
## FG\_herb 0.000879 0.0293 0.0300 9.76e-01  
## FG\_shrub 0.032879 0.0257 1.2803 2.00e-01  
## HW\_dens\_1050 0.008219 0.0307 0.2674 7.89e-01  
## Ldepth 0.001086 0.0274 0.0397 9.68e-01  
## Rel\_HW2P\_canopy -0.015105 0.0289 -0.5222 6.01e-01  
## BA -0.041468 0.0293 -1.4139 1.57e-01  
## NHW\_saplings -0.069467 0.0307 -2.2636 2.36e-02  
## YearCatB 0.053333 0.0536 0.9947 3.20e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.5904 0.2715 -5.858 4.68e-09  
## Jdate 0.0939 0.0258 3.633 2.80e-04  
## Wind 0.0488 0.0266 1.831 6.72e-02  
## Sky -0.0271 0.0246 -1.099 2.72e-01  
## Noise -0.0596 0.0261 -2.279 2.26e-02  
## Time -0.0123 0.0246 -0.502 6.16e-01  
##   
## AIC: 1488.589

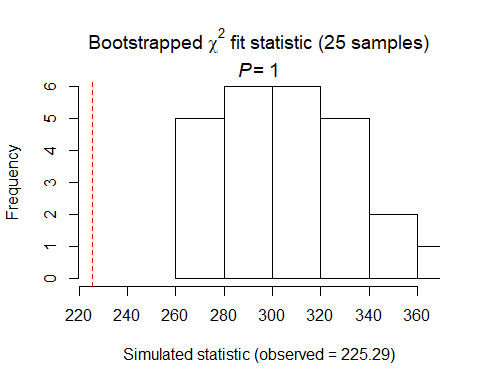
confint(lh.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 3.38979254 4.279526546  
## lam(FG\_herb) -0.05652831 0.058287063  
## lam(FG\_shrub) -0.01745281 0.083210827  
## lam(HW\_dens\_1050) -0.05201648 0.068454926  
## lam(Ldepth) -0.05252003 0.054692131  
## lam(Rel\_HW2P\_canopy) -0.07179406 0.041583639  
## lam(BA) -0.09895333 0.016016952  
## lam(NHW\_saplings) -0.12961725 -0.009317132  
## lam(YearCatB) -0.05175778 0.158424724

write.table(ms.gf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_gf\_top\_models\_ms.xls",sep="\t")

#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(local.gf, nsim = 25, plot.hist = TRUE, report = NULL) #increase nsim

## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced  
  
## Warning in rbinom(M \* J, size = rep(N, each = J), prob = pvec): NAs  
## produced



##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 225.2857   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 262 285 309 321 364   
##   
## Estimate of c-hat = 0.74

#?Nmix.gof.test()  
Nmix.chisq(local.gf)

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 225.2857

Ground foragers (n=27) Summary: P distribution DCs: global best (+ with date, - with noise, close with wind) then detect () then date () under 2.0 SCs using Global: Local best (- with canopy cover) Life history second best (- with # hardwood saplings)