Results - guild abundance

Jesse Wood

Sept 25, 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 together (correlations): \*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

det.date.cavity <- pcount(~ Jdate ~1, cavity.abund, mixture="P", K=15)  
det.date2.cavity <- pcount(~ Jdate + I(Jdate^2) ~1, cavity.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.cavity, det.date2.cavity)

## Warning in fitList(det.date.cavity, det.date2.cavity): Your list was  
## unnamed, so model names were added as object names

DATEtest.cavity <- modSel(mstestDATE)  
DATEtest.cavity

## nPars AIC delta AICwt cumltvWt  
## det.date2.cavity 4 1440.37 0.00 0.72 0.72  
## det.date.cavity 3 1442.22 1.85 0.28 1.00

det.time.cavity <-pcount(~ Time ~1, cavity.abund, mixture="P",K=15)  
det.time2.cavity <-pcount(~ Time + I(Time^2) ~1, cavity.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.cavity, det.time2.cavity)

## Warning in fitList(det.time.cavity, det.time2.cavity): Your list was  
## unnamed, so model names were added as object names

TIMEtest.cavity <- modSel(mstestTIME)  
TIMEtest.cavity

## nPars AIC delta AICwt cumltvWt  
## det.time2.cavity 4 1424.55 0.00 0.72 0.72  
## det.time.cavity 3 1426.48 1.92 0.28 1.00

#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 + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2) ~1, cavity.abund, mixture="P", K=50)  
det.sound.cavity <- pcount(~ Noise + Wind ~1, cavity.abund, mixture="P", K=50)  
det.date2.cavity <- pcount(~ Jdate + I(Jdate^2) ~1, cavity.abund, mixture="P", K=50)  
det.detect2.cavity <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, cavity.abund, mixture="P", K=50)  
det.notdate.cavity <-pcount(~ Wind + Sky + Noise ~1, cavity.abund, mixture="P", K=50)  
det.time2.cavity <-pcount(~ Time + I(Time^2) ~1, cavity.abund, mixture="P",K=50)  
det.timing2.cavity <-pcount(~ Time + I(Time^2) + Jdate + I(Jdate^2) ~1, cavity.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.cavity, det.weather.cavity, det.global.cavity,  
 det.sound.cavity, det.date2.cavity, det.detect2.cavity, det.notdate.cavity,  
 det.time2.cavity, det.timing2.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 9 1385.63 0.00 0.54177 0.54  
## det.time2.cavity 4 1388.02 2.38 0.16461 0.71  
## det.weather.cavity 4 1388.26 2.62 0.14600 0.85  
## det.timing2.cavity 6 1390.06 4.43 0.05925 0.91  
## det.notdate.cavity 5 1390.09 4.45 0.05850 0.97  
## det.detect2.cavity 7 1391.88 6.25 0.02380 0.99  
## det.null.cavity 2 1395.59 9.96 0.00373 1.00  
## det.date2.cavity 4 1397.25 11.61 0.00163 1.00  
## det.sound.cavity 4 1398.96 13.32 0.00069 1.00

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

det.global.cavity

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Wind + Sky + Noise + Time +   
## I(Time^2) ~ 1, data = cavity.abund, K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.96 0.234 12.6 1.21e-36  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.7838 0.3448 -2.273 0.02301  
## Jdate -0.0269 0.0333 -0.808 0.41918  
## I(Jdate^2) -0.0541 0.0313 -1.727 0.08417  
## Wind -0.0340 0.0353 -0.963 0.33563  
## Sky -0.0993 0.0344 -2.891 0.00384  
## Noise 0.0129 0.0340 0.379 0.70472  
## Time 0.0875 0.0354 2.469 0.01354  
## I(Time^2) -0.0445 0.0330 -1.346 0.17821  
##   
## AIC: 1385.635

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

## 0.025 0.975  
## p(Int) -1.45962558 -0.108047932  
## p(Jdate) -0.09224501 0.038398027  
## p(I(Jdate^2)) -0.11545085 0.007294434  
## p(Wind) -0.10307326 0.035163669  
## p(Sky) -0.16666402 -0.031981581  
## p(Noise) -0.05382193 0.079623253  
## p(Time) 0.01805190 0.156982402  
## p(I(Time^2)) -0.10924251 0.020276880

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

## 0.025 0.975  
## lam(Int) 2.501985 3.419956

# neg with sky, pos with time  
  
#det.time2.cavity  
#confint(det.time2.cavity, type="det",method="normal")  
#confint(det.time2.cavity, type="state",method="normal")  
  
#det.weather.cavity  
#confint(det.weather.cavity, type="det",method="normal")  
#confint(det.weather.cavity, type="state",method="normal")

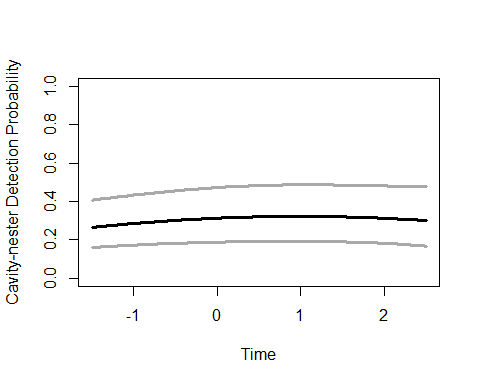
Cavity-nester detection models: global best (- with sky, + with survey time) weather second best (- with sky) time third best (+ with time)

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")

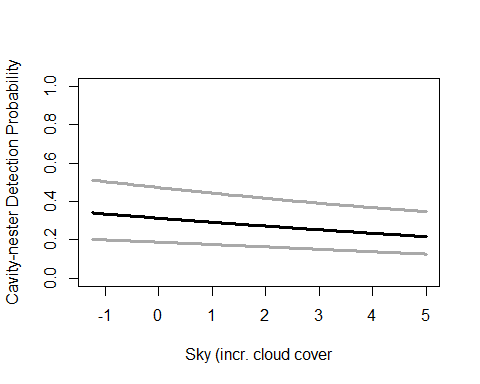
#best model is global, two variables sig (time, sky)  
summary(obsCovs(cavity.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

det.global.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2) ~1, cavity.abund, mixture="P", K=50)  
  
NewData.CNd1 <-data.frame(Time=seq(-1.5,2.5,length=100),Sky=0,Wind=0,Noise=0, Jdate=0)  
cn.est.det1 <- predict(det.global.cavity, type="det",  
 newdata=NewData.CNd1,appendData=TRUE)  
  
plot(Predicted~ Time, data=cn.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Time", ylab="Cavity-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Time, data=cn.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Time, data=cn.est.det1, type="l", lwd=3, col="darkgray")



#best model is global, two variables sig (date, sky)  
#summary(obsCovs(cavity.abund))  
#det.global.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2) ~1, cavity.abund, mixture="P", K=50)  
  
NewData.CNd2 <-data.frame(Sky=seq(-1.25,5,length=100),Jdate=0,Wind=0,Noise=0,Time=0)  
cn.est.det2 <- predict(det.global.cavity, type="det",  
 newdata=NewData.CNd2,appendData=TRUE)  
  
plot(Predicted~ Sky, data=cn.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Sky (incr. cloud cover", ylab="Cavity-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Sky, data=cn.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Sky, data=cn.est.det2, type="l", lwd=3, col="darkgray")



#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

#not better for most, except for FG\_herb  
testR.cavity <- pcount(~1 ~BA, cavity.abund, mixture="P", K=30)  
testQ.cavity <- pcount(~1 ~BA + I(BA^2), cavity.abund, mixture="P", K=30)  
msBAtest <- fitList(testR.cavity, testQ.cavity)

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

BAtest.cavity <- modSel(msBAtest)  
BAtest.cavity

## nPars AIC delta AICwt cumltvWt  
## testR.cavity 3 1396.70 0.00 0.60 0.60  
## testQ.cavity 4 1397.49 0.79 0.40 1.00

testR.cavity <- pcount(~1 ~HW\_dens\_1050, cavity.abund, mixture="P", K=30)  
testQ.cavity <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), cavity.abund, mixture="P", K=30)  
msHW1050test <- fitList(testR.cavity, testQ.cavity)

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

HW1050test.cavity <- modSel(msHW1050test)  
HW1050test.cavity

## nPars AIC delta AICwt cumltvWt  
## testR.cavity 3 1400.04 0.00 0.56 0.56  
## testQ.cavity 4 1400.54 0.50 0.44 1.00

#quadratic better for FG\_herb!  
testR.cavity <- pcount(~1 ~FG\_herb, cavity.abund, mixture="P", K=30)  
testQ.cavity <- pcount(~1 ~FG\_herb + I(FG\_herb^2), cavity.abund, mixture="P", K=30)  
msFGHtest <- fitList(testR.cavity, testQ.cavity)

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

FGHtest.cavity <- modSel(msFGHtest)  
FGHtest.cavity

## nPars AIC delta AICwt cumltvWt  
## testQ.cavity 4 1399.15 0.00 0.61 0.61  
## testR.cavity 3 1400.03 0.89 0.39 1.00

testR.cavity <- pcount(~1 ~FG\_shrub, cavity.abund, mixture="P", K=30)  
testQ.cavity <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), cavity.abund, mixture="P", K=30)  
msFGStest <- fitList(testR.cavity, testQ.cavity)

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

FGStest.cavity <- modSel(msFGStest)  
FGStest.cavity

## nPars AIC delta AICwt cumltvWt  
## testR.cavity 3 1399.56 0.00 0.54 0.54  
## testQ.cavity 4 1399.88 0.32 0.46 1.00

testR.cavity <- pcount(~1 ~NHW\_saplings, cavity.abund, mixture="P", K=30)  
testQ.cavity <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), cavity.abund, mixture="P", K=30)  
msHWStest <- fitList(testR.cavity, testQ.cavity)

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

HWStest.cavity <- modSel(msHWStest)  
HWStest.cavity

## nPars AIC delta AICwt cumltvWt  
## testR.cavity 3 1396.30 0.00 0.73 0.73  
## testQ.cavity 4 1398.25 1.95 0.27 1.00

# need to update below with global dets Jdate + Wind + Sky + Noise +Time, and quadratics  
# then add YearCat  
  
##site covariates next (global model for detection covariates)  
null.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2) ~1, cavity.abund, mixture="P", K=80)  
global.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ 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 + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , cavity.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ 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 + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Parea + ShapeIndex + YearCat  
 , cavity.abund, mixture="P",K=80)  
landscape500.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Evergreen500m + HighDev500m + Schrubs500m + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape1.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Evergreen1km + HighDev1km + Schrubs1km + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape5.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , cavity.abund, mixture="P", K=80)  
landscape30.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Evergreen30km + HighDev30km + YearCat  
 , cavity.abund, mixture="P", K=80) #had to remove Protected  
treatment.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Treatment + Nthins + YearCat  
 , cavity.abund, mixture ="P", K=80)  
management.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , cavity.abund, mixture="P", K=80)  
disturbance.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , cavity.abund, mixture="P", K=80)  
siteprod.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ PISoils + NSoilTypes + YearCat  
 , cavity.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ BA + Nsnags + Nburns + YearCat  
 , cavity.abund, mixture="P", K=80)  
coord.cavity <-pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ 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  
## landmetrics.cavity 12 1382.40 0.000 2.6e-01 0.26  
## local.cavity 13 1382.48 0.074 2.5e-01 0.51  
## siteprod.cavity 12 1383.39 0.990 1.6e-01 0.66  
## greenberg.cavity 13 1383.92 1.520 1.2e-01 0.78  
## null.cavity 9 1385.63 3.233 5.1e-02 0.84  
## disturbance.cavity 12 1385.79 3.391 4.7e-02 0.88  
## landscape5.cavity 13 1386.74 4.343 2.9e-02 0.91  
## coord.cavity 12 1387.39 4.988 2.1e-02 0.93  
## treatment.cavity 14 1387.57 5.172 1.9e-02 0.95  
## lh.cavity 16 1388.28 5.874 1.4e-02 0.97  
## landscape30.cavity 12 1388.62 6.213 1.2e-02 0.98  
## landscape1.cavity 13 1389.18 6.779 8.7e-03 0.99  
## landscape500.cavity 13 1389.55 7.152 7.2e-03 0.99  
## management.cavity 17 1390.28 7.878 5.0e-03 1.00  
## global.cavity 31 1402.25 19.847 1.3e-05 1.00

#ms.cavity@Full

#local, siteprod, landmetrics, greenberg all under 2.0  
#changed order with quadratics included: landmetrics, local, siteprod, greenberg  
  
landmetrics.cavity

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Wind + Sky + Noise + Time +   
## I(Time^2) ~ Parea + ShapeIndex + YearCat, data = cavity.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.1180 0.2867 10.876 1.50e-27  
## Parea -0.0831 0.0332 -2.501 1.24e-02  
## ShapeIndex 0.0223 0.0305 0.732 4.64e-01  
## YearCatB -0.1118 0.0635 -1.761 7.82e-02  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.92761 0.3975 -2.3338 0.01961  
## Jdate -0.02859 0.0325 -0.8786 0.37960  
## I(Jdate^2) -0.06776 0.0312 -2.1715 0.02989  
## Wind -0.05915 0.0360 -1.6425 0.10048  
## Sky -0.09285 0.0339 -2.7410 0.00613  
## Noise 0.00246 0.0334 0.0737 0.94125  
## Time 0.08892 0.0348 2.5561 0.01059  
## I(Time^2) -0.03568 0.0323 -1.1060 0.26871  
##   
## AIC: 1382.402

confint(landmetrics.cavity, type="state",method="normal") #- with patch area

## 0.025 0.975  
## lam(Int) 2.55611432 3.67987513  
## lam(Parea) -0.14826369 -0.01798291  
## lam(ShapeIndex) -0.03746173 0.08214055  
## lam(YearCatB) -0.23630155 0.01261646

local.cavity

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Wind + Sky + Noise + Time +   
## I(Time^2) ~ Ccover + TreeHt + Ldepth + YearCat, data = cavity.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.2386 0.3458 9.367 7.49e-21  
## Ccover -0.0296 0.0309 -0.959 3.38e-01  
## TreeHt 0.0242 0.0329 0.738 4.61e-01  
## Ldepth -0.0757 0.0336 -2.251 2.44e-02  
## YearCatB -0.1399 0.0699 -2.002 4.53e-02  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.0836 0.4619 -2.346 0.0190  
## Jdate -0.0257 0.0319 -0.806 0.4202  
## I(Jdate^2) -0.0634 0.0306 -2.072 0.0383  
## Wind -0.0541 0.0352 -1.534 0.1250  
## Sky -0.0839 0.0332 -2.529 0.0114  
## Noise 0.0304 0.0325 0.935 0.3496  
## Time 0.0811 0.0342 2.370 0.0178  
## I(Time^2) -0.0281 0.0321 -0.875 0.3814  
##   
## AIC: 1382.476

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

## 0.025 0.975  
## lam(Int) 2.56091603 3.916262734  
## lam(Ccover) -0.09017678 0.030940910  
## lam(TreeHt) -0.04015703 0.088625685  
## lam(Ldepth) -0.14165416 -0.009798513  
## lam(YearCatB) -0.27689203 -0.002919225

siteprod.cavity

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Wind + Sky + Noise + Time +   
## I(Time^2) ~ PISoils + NSoilTypes + YearCat, data = cavity.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.10326 0.2816 11.0206 3.04e-28  
## PISoils -0.00226 0.0295 -0.0764 9.39e-01  
## NSoilTypes -0.07122 0.0305 -2.3354 1.95e-02  
## YearCatB -0.10539 0.0634 -1.6626 9.64e-02  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.91767 0.3934 -2.333 0.01965  
## Jdate -0.02533 0.0327 -0.774 0.43916  
## I(Jdate^2) -0.06231 0.0314 -1.986 0.04704  
## Wind -0.04943 0.0360 -1.372 0.17000  
## Sky -0.09081 0.0339 -2.678 0.00742  
## Noise 0.00584 0.0333 0.175 0.86104  
## Time 0.08476 0.0350 2.424 0.01536  
## I(Time^2) -0.03356 0.0325 -1.033 0.30179  
##   
## AIC: 1383.392

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

## 0.025 0.975  
## lam(Int) 2.55136561 3.65516428  
## lam(PISoils) -0.06014899 0.05563835  
## lam(NSoilTypes) -0.13098725 -0.01144836  
## lam(YearCatB) -0.22962874 0.01884742

greenberg.cavity

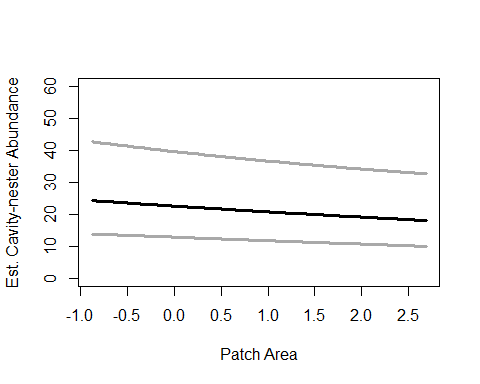
##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Wind + Sky + Noise + Time +   
## I(Time^2) ~ BA + Nsnags + Nburns + YearCat, data = cavity.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.1824 0.3160 10.071 7.41e-24  
## BA -0.0548 0.0338 -1.619 1.05e-01  
## Nsnags 0.0384 0.0286 1.345 1.79e-01  
## Nburns 0.0310 0.0314 0.986 3.24e-01  
## YearCatB -0.1235 0.0634 -1.948 5.14e-02  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.0137 0.4291 -2.362 0.01817  
## Jdate -0.0276 0.0322 -0.857 0.39151  
## I(Jdate^2) -0.0648 0.0308 -2.103 0.03547  
## Wind -0.0543 0.0356 -1.524 0.12761  
## Sky -0.0862 0.0334 -2.577 0.00998  
## Noise 0.0280 0.0328 0.852 0.39424  
## Time 0.0841 0.0344 2.441 0.01464  
## I(Time^2) -0.0320 0.0323 -0.991 0.32181  
##   
## AIC: 1383.922

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

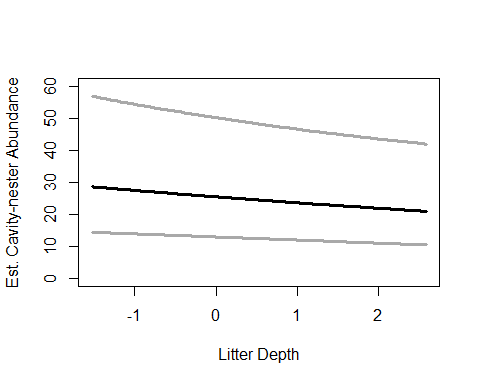
## 0.025 0.975  
## lam(Int) 2.56303575 3.8016871765  
## lam(BA) -0.12107964 0.0115295342  
## lam(Nsnags) -0.01757045 0.0943954258  
## lam(Nburns) -0.03059635 0.0925134101  
## lam(YearCatB) -0.24775988 0.0007323323

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")

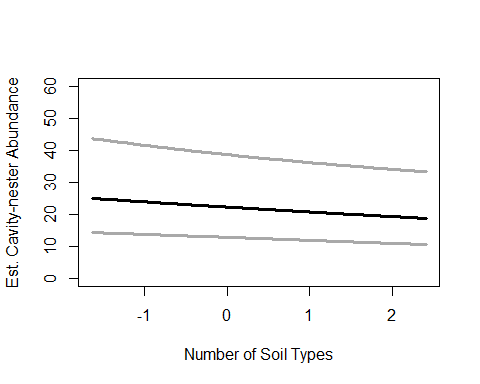
#for figures: landmetrics with patch area  
  
landmetrics.cavity <- pcount (~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Parea + ShapeIndex + YearCat  
 , cavity.abund, mixture="P",K=80)  
  
NewData.CNPA <-data.frame(Parea=seq(min(sc$Parea),max(sc$Parea),length=100),ShapeIndex=0,YearCat=0)  
cn.est.parea <- predict(landmetrics.cavity, type="state",  
 newdata=NewData.CNPA,appendData=TRUE)  
  
plot(Predicted~ Parea, data=cn.est.parea, ylim=c(0,60), type="l", lwd=3,  
 xlab="Patch Area", ylab="Est. Cavity-nester Abundance")  
##95% confidence intervals  
lines(lower~ Parea, data=cn.est.parea, type="l", lwd=3, col="darkgray")  
lines(upper~ Parea, data=cn.est.parea, type="l", lwd=3, col="darkgray")



#for figures: local with litter depth, local with Year  
local.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , cavity.abund, mixture="P", K=80)  
  
NewData.CNLD <-data.frame(Ldepth=seq(min(sc$Ldepth),max(sc$Ldepth),length=100),Ccover=0,TreeHt=0,YearCat=0)  
cn.est.ldepth <- predict(local.cavity, type="state",  
 newdata=NewData.CNLD,appendData=TRUE)  
  
plot(Predicted~ Ldepth, data=cn.est.ldepth, ylim=c(0,60), type="l", lwd=3,  
 xlab="Litter Depth", ylab="Est. Cavity-nester Abundance")  
##95% confidence intervals  
lines(lower~ Ldepth, data=cn.est.ldepth, type="l", lwd=3, col="darkgray")  
lines(upper~ Ldepth, data=cn.est.ldepth, type="l", lwd=3, col="darkgray")

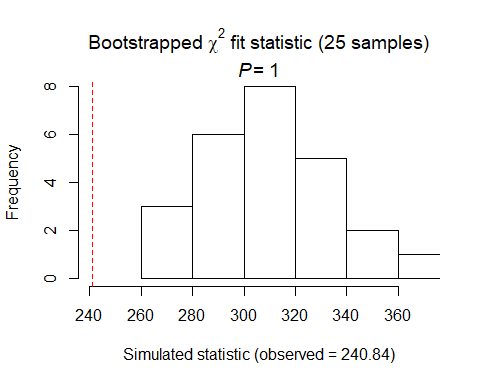


#placeholder one for year!  
  
  
#site productivity with # soil types  
siteprod.cavity <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2)  
 ~ PISoils + NSoilTypes + YearCat  
 , cavity.abund, mixture="P", K=80) #FPSiteIndex  
  
NewData.CNS <-data.frame(NSoilTypes=seq(min(sc$NSoilTypes),max(sc$NSoilTypes),length=100),PISoils=0,YearCat=0)  
cn.est.soils <- predict(siteprod.cavity, type="state",  
 newdata=NewData.CNS,appendData=TRUE)  
  
plot(Predicted~ NSoilTypes, data=cn.est.soils, ylim=c(0,60), type="l", lwd=3,  
 xlab="Number of Soil Types", ylab="Est. Cavity-nester Abundance")  
##95% confidence intervals  
lines(lower~ NSoilTypes, data=cn.est.soils, type="l", lwd=3, col="darkgray")  
lines(upper~ NSoilTypes, data=cn.est.soils, type="l", lwd=3, col="darkgray")



#put species' top model in place of "landscape5.prwa" & up nsim  
Nmix.gof.test(landmetrics.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 = 240.8437   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 270 295 308 327 371   
##   
## Estimate of c-hat = 0.77

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 237.9438

# 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 - Landmetrics best (- with patch area) Local second (- with litter depth & - with yearB) Site Productivity next best (- with # soil types) 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

det.date.tree <- pcount(~ Jdate ~1, tree.abund, mixture="P", K=15)  
det.date2.tree <- pcount(~ Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.tree, det.date2.tree)

## Warning in fitList(det.date.tree, det.date2.tree): Your list was unnamed,  
## so model names were added as object names

DATEtest.tree <- modSel(mstestDATE)  
DATEtest.tree

## nPars AIC delta AICwt cumltvWt  
## det.date2.tree 4 1436.90 0.00 0.88 0.88  
## det.date.tree 3 1440.98 4.08 0.12 1.00

det.time.tree <-pcount(~ Time ~1, tree.abund, mixture="P",K=15)  
det.time2.tree <-pcount(~ Time + I(Time^2) ~1, tree.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.tree, det.time2.tree)

## Warning in fitList(det.time.tree, det.time2.tree): Your list was unnamed,  
## so model names were added as object names

TIMEtest.tree <- modSel(mstestTIME)  
TIMEtest.tree

## nPars AIC delta AICwt cumltvWt  
## det.time2.tree 4 1447.88 0.00 0.56 0.56  
## det.time.tree 3 1448.39 0.51 0.44 1.00

#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.global2.tree <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time + I(Time^2) ~1, tree.abund, mixture="P", K=50)  
det.sound.tree <- pcount(~ Noise + Wind ~1, tree.abund, mixture="P", K=50)  
det.date2.tree <- pcount(~ Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=50)  
det.detect2.tree <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, tree.abund, mixture="P", K=50)  
det.notdate.tree <-pcount(~ Wind + Sky + Noise ~1, tree.abund, mixture="P", K=50)  
det.time2.tree <-pcount(~ Time + I(Time^2) ~1, tree.abund, mixture="P",K=50)  
det.timing2.tree <-pcount(~ Time + I(Time^2) + Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.tree, det.weather.tree, det.global2.tree,  
 det.sound.tree, det.date2.tree, det.detect2.tree, det.notdate.tree,  
 det.time2.tree, det.timing2.tree)

## Warning in fitList(det.null.tree, det.weather.tree, det.global2.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.date2.tree 4 1397.41 0.00 0.414 0.41  
## det.detect2.tree 7 1398.89 1.48 0.198 0.61  
## det.timing2.tree 6 1400.57 3.16 0.085 0.70  
## det.global2.tree 9 1400.69 3.28 0.080 0.78  
## det.sound.tree 4 1401.02 3.61 0.068 0.85  
## det.notdate.tree 5 1401.24 3.83 0.061 0.91  
## det.weather.tree 4 1401.94 4.53 0.043 0.95  
## det.null.tree 2 1402.10 4.69 0.040 0.99  
## det.time2.tree 4 1404.63 7.22 0.011 1.00

#msDC.tree@Full  
#summary now: date2, detect2 all under 2.0  
# after quadratic: timing, sound, notdate no longer under 2.0

^ date2 best (date^2 sig), detect2 second best (none sig)

det.date2.tree

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) ~ 1, data = tree.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.18 0.301 10.6 4.78e-26  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.9265 0.4189 -2.21 0.0270  
## Jdate 0.0650 0.0280 2.32 0.0203  
## I(Jdate^2) -0.0566 0.0273 -2.07 0.0384  
##   
## AIC: 1397.41

confint(det.date2.tree, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.74759675 -0.105433038  
## p(Jdate) 0.01010262 0.119944075  
## p(I(Jdate^2)) -0.11009339 -0.003031264

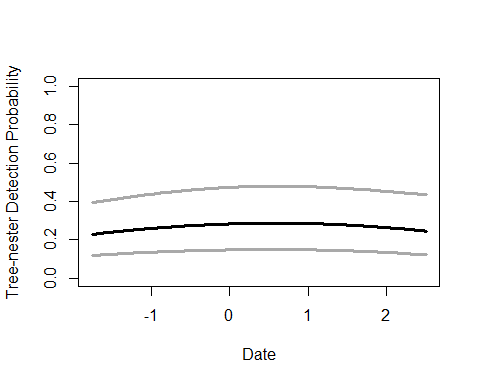
confint(det.date2.tree, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.586098 3.765437

#best model is date2,  
summary(obsCovs(tree.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

det.date2.tree <- pcount(~ Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=50)  
  
NewData.TNd1 <-data.frame(Jdate=seq(-1.75,2.5,length=100))  
tn.est.det1 <- predict(det.date2.tree, type="det",  
 newdata=NewData.TNd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=tn.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Tree-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=tn.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=tn.est.det1, type="l", lwd=3, col="darkgray")



det.detect2.tree

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~   
## 1, data = tree.abund, K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.11 0.274 11.4 7.12e-30  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.8137 0.3970 -2.050 0.0404  
## Jdate 0.0731 0.0298 2.453 0.0142  
## I(Jdate^2) -0.0523 0.0280 -1.870 0.0614  
## Noise -0.0614 0.0329 -1.867 0.0619  
## Time 0.0301 0.0303 0.993 0.3207  
## I(Time^2) -0.0266 0.0308 -0.865 0.3871  
##   
## AIC: 1398.889

confint(det.detect2.tree, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.59173011 -0.035647969  
## p(Jdate) 0.01468944 0.131412423  
## p(I(Jdate^2)) -0.10706214 0.002507738  
## p(Noise) -0.12592579 0.003050461  
## p(Time) -0.02929781 0.089463035  
## p(I(Time^2)) -0.08699428 0.033722538

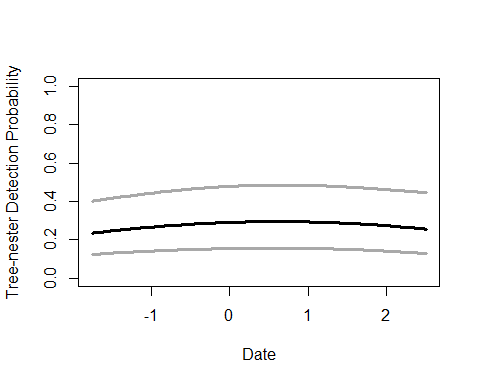
confint(det.detect2.tree, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.573769 3.64779

det.date2.tree <- pcount(~ Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=50)  
det.detect2.tree <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, tree.abund, mixture="P", K=50)  
  
#second best model is detect2 - also date --- MODEL-AVERAGED DATE WITH BOTH MODELS  
dms\_top.tree <- fitList(det.date2.tree,det.detect2.tree)

## Warning in fitList(det.date2.tree, det.detect2.tree): Your list was  
## unnamed, so model names were added as object names

ND.TNd2 <-data.frame(Jdate=seq(-1.75,2.5,length=100),Noise=0,Time=0)  
tn.est.det2 <- predict(dms\_top.tree, type="det",  
 newdata=ND.TNd2,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=tn.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Tree-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=tn.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=tn.est.det2, type="l", lwd=3, col="darkgray")



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")

#none of these better quadratic  
testR.tree <- pcount(~1 ~BA, tree.abund, mixture="P", K=20)  
testQ.tree <- pcount(~1 ~BA + I(BA^2), tree.abund, mixture="P", K=20)  
msBAtest <- fitList(testR.tree, testQ.tree)

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

BAtest.tree <- modSel(msBAtest)  
BAtest.tree

## nPars AIC delta AICwt cumltvWt  
## testR.tree 3 1417.81 0.00 0.72 0.72  
## testQ.tree 4 1419.73 1.91 0.28 1.00

testR.tree <- pcount(~1 ~HW\_dens\_1050, tree.abund, mixture="P", K=20)  
testQ.tree <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), tree.abund, mixture="P", K=20)  
msHW1050test <- fitList(testR.tree, testQ.tree)

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

HW1050test.tree <- modSel(msHW1050test)  
HW1050test.tree

## nPars AIC delta AICwt cumltvWt  
## testR.tree 3 1417.76 0.00 0.70 0.70  
## testQ.tree 4 1419.46 1.70 0.30 1.00

testR.tree <- pcount(~1 ~FG\_herb, tree.abund, mixture="P", K=20)  
testQ.tree <- pcount(~1 ~FG\_herb + I(FG\_herb^2), tree.abund, mixture="P", K=20)  
msFGHtest <- fitList(testR.tree, testQ.tree)

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

FGHtest.tree <- modSel(msFGHtest)  
FGHtest.tree

## nPars AIC delta AICwt cumltvWt  
## testR.tree 3 1416.97 0.00 0.73 0.73  
## testQ.tree 4 1418.96 1.99 0.27 1.00

testR.tree <- pcount(~1 ~FG\_shrub, tree.abund, mixture="P", K=20)  
testQ.tree <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), tree.abund, mixture="P", K=20)  
msFGStest <- fitList(testR.tree, testQ.tree)

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

FGStest.tree <- modSel(msFGStest)  
FGStest.tree

## nPars AIC delta AICwt cumltvWt  
## testR.tree 3 1417.49 0.00 0.73 0.73  
## testQ.tree 4 1419.48 1.98 0.27 1.00

testR.tree <- pcount(~1 ~NHW\_saplings, tree.abund, mixture="P", K=20)  
testQ.tree <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), tree.abund, mixture="P", K=20)  
msHWStest <- fitList(testR.tree, testQ.tree)

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

HWStest.tree <- modSel(msHWStest)  
HWStest.tree

## nPars AIC delta AICwt cumltvWt  
## testR.tree 3 1417.82 0.00 0.68 0.68  
## testQ.tree 4 1419.29 1.47 0.32 1.00

##site covariates next  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
#with DATE2 covariates now - Jdate + Jdate^2  
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~  
  
null.tree <- pcount(~ Jdate + I(Jdate^2) ~1, tree.abund, mixture="P", K=80)  
global.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ 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 + I(Jdate^2)  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , tree.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ 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 + I(Jdate^2)  
 ~ Parea + ShapeIndex + YearCat  
 , tree.abund, mixture="P",K=80)  
landscape500.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Evergreen500m + HighDev500m + Schrubs500m + OpenDev500m  
 + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape1.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape5.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , tree.abund, mixture="P", K=80)  
landscape30.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Evergreen30km + HighDev30km + YearCat  
 , tree.abund, mixture="P", K=80) #removed Protected  
treatment.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Treatment + Nthins + YearCat  
 , tree.abund, mixture ="P", K=80)  
management.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , tree.abund, mixture="P", K=80)  
disturbance.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , tree.abund, mixture="P", K=80)  
siteprod.tree <- pcount(~ Jdate + I(Jdate^2)  
 ~ PISoils + NSoilTypes + YearCat  
 , tree.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.tree <- pcount(~ Jdate + I(Jdate^2)   
 ~ Ccover + NHW\_saplings + HW\_dens\_1050 + Nburns  
 + YearCat, tree.abund, mixture="P", K=80)  
coord.tree <-pcount(~ Jdate + I(Jdate^2)  
 ~ 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 7 1387.94 0.00 0.57713 0.58  
## siteprod.tree 7 1391.04 3.10 0.12266 0.70  
## disturbance.tree 7 1391.82 3.88 0.08276 0.78  
## local.tree 8 1391.86 3.92 0.08143 0.86  
## landscape30.tree 7 1393.40 5.46 0.03764 0.90  
## coord.tree 7 1394.12 6.18 0.02623 0.93  
## landscape500.tree 9 1394.67 6.73 0.01995 0.95  
## management.tree 12 1395.55 7.61 0.01283 0.96  
## landscape5.tree 8 1395.58 7.64 0.01263 0.97  
## lh.tree 10 1396.08 8.14 0.00987 0.98  
## landscape1.tree 9 1397.18 9.24 0.00569 0.99  
## null.tree 4 1397.41 9.47 0.00507 0.99  
## treatment.tree 9 1398.11 10.17 0.00356 1.00  
## greenberg.tree 9 1398.87 10.93 0.00244 1.00  
## global.tree 24 1405.06 17.12 0.00011 1.00

#ms.tree@Full

#land metrics only top model! - didn't change  
landmetrics.tree

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) ~ Parea + ShapeIndex + YearCat,   
## data = tree.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.6091 0.7248 4.980 6.37e-07  
## Parea 0.0763 0.0254 3.002 2.68e-03  
## ShapeIndex -0.0230 0.0270 -0.851 3.95e-01  
## YearCatB 0.1523 0.0522 2.916 3.55e-03  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.6008 0.8733 -1.83 0.0668  
## Jdate 0.0564 0.0266 2.12 0.0340  
## I(Jdate^2) -0.0347 0.0265 -1.31 0.1910  
##   
## AIC: 1387.94

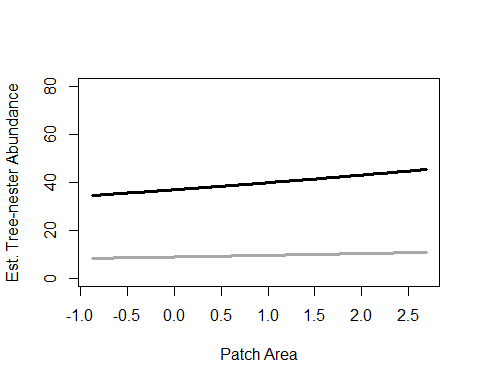
confint(landmetrics.tree, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.18855889 5.02964561  
## lam(Parea) 0.02649468 0.12617916  
## lam(ShapeIndex) -0.07595492 0.02994716  
## lam(YearCatB) 0.04991303 0.25463390

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")

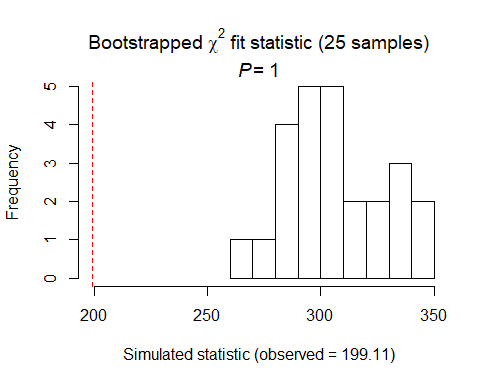
#landmetrics with patch area  
landmetrics.tree <- pcount (~ Jdate + I(Jdate^2)  
 ~ Parea + ShapeIndex + YearCat  
 , tree.abund, mixture="P",K=80)  
  
NewData.TN <-data.frame(Parea=seq(min(sc$Parea),max(sc$Parea),length=100),ShapeIndex=0,YearCat=0)  
tn.est.area <- predict(landmetrics.tree, type="state",newdata=NewData.TN,appendData=TRUE)  
  
plot(Predicted~ Parea, data=tn.est.area, ylim=c(0,80), type="l", lwd=3,  
xlab="Patch Area", ylab="Est. Tree-nester Abundance")  
##95% confidence intervals  
lines(lower~ Parea, data=tn.est.area, type="l", lwd=3, col="darkgray")  
lines(upper~ Parea, data=tn.est.area, type="l", lwd=3, col="darkgray")



#placeholder for year effect figure? not necessary...

#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 = 199.1124   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 264 293 300 326 346   
##   
## 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 = 199.1124

Tree-nesters (n=25) Summary: P distribution DCs: date2, detect2 top two date2 (- with date2?), detect2 (none sig except date?) SCs using date2: 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.date.shrub <- pcount(~ Jdate ~1, shrub.abund, mixture="P", K=15)  
det.date2.shrub <- pcount(~ Jdate + I(Jdate^2) ~1, shrub.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.shrub, det.date2.shrub)

## Warning in fitList(det.date.shrub, det.date2.shrub): Your list was unnamed,  
## so model names were added as object names

DATEtest.shrub <- modSel(mstestDATE)  
DATEtest.shrub

## nPars AIC delta AICwt cumltvWt  
## det.date2.shrub 4 1239.81 0.00 0.943 0.94  
## det.date.shrub 3 1245.43 5.62 0.057 1.00

det.time.shrub <-pcount(~ Time ~1, shrub.abund, mixture="P",K=15)  
det.time2.shrub <-pcount(~ Time + I(Time^2) ~1, shrub.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.shrub, det.time2.shrub)

## Warning in fitList(det.time.shrub, det.time2.shrub): Your list was unnamed,  
## so model names were added as object names

TIMEtest.shrub <- modSel(mstestTIME)  
TIMEtest.shrub

## nPars AIC delta AICwt cumltvWt  
## det.time.shrub 3 1266.07 0.00 0.71 0.71  
## det.time2.shrub 4 1267.83 1.76 0.29 1.00

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.global2.shrub <- pcount(~ Jdate + I(Jdate^2) + 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.date2.shrub <- pcount(~ Jdate + I(Jdate^2) ~1, shrub.abund, mixture="P", K=50)  
det.detect2.shrub <- pcount(~ Jdate + I(Jdate^2) + 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.timing2.shrub <-pcount(~ Time + Jdate + I(Jdate^2) ~1, shrub.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.shrub, det.weather.shrub, det.global2.shrub,  
 det.sound.shrub, det.date2.shrub, det.detect2.shrub, det.notdate.shrub,  
 det.time.shrub, det.timing2.shrub)

## Warning in fitList(det.null.shrub, det.weather.shrub, det.global2.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.detect2.shrub 6 1232.66 0.00 7.6e-01 0.76  
## det.global2.shrub 8 1235.70 3.05 1.6e-01 0.92  
## det.timing2.shrub 5 1238.36 5.71 4.4e-02 0.96  
## det.date2.shrub 4 1238.77 6.11 3.6e-02 1.00  
## det.sound.shrub 4 1258.15 25.50 2.2e-06 1.00  
## det.notdate.shrub 5 1259.57 26.92 1.1e-06 1.00  
## det.null.shrub 2 1260.42 27.76 7.1e-07 1.00  
## det.time.shrub 3 1262.13 29.48 3.0e-07 1.00  
## det.weather.shrub 4 1263.40 30.74 1.6e-07 1.00

#msDC.shrub@Full  
#summary: detection is best model and only under 2.0 #didn't change in 2018 or w/ date^2

det.detect2.shrub

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time ~ 1, data = shrub.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.23 0.129 17.3 3.71e-67  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.0581 0.2489 -0.233 8.16e-01  
## Jdate 0.2158 0.0460 4.691 2.72e-06  
## I(Jdate^2) -0.1002 0.0405 -2.477 1.32e-02  
## Noise -0.1337 0.0482 -2.772 5.57e-03  
## Time 0.0720 0.0411 1.750 8.02e-02  
##   
## AIC: 1232.656

confint(det.detect2.shrub, type="det",method="normal")

## 0.025 0.975  
## p(Int) -0.545974204 0.42981884  
## p(Jdate) 0.125658485 0.30602333  
## p(I(Jdate^2)) -0.179501052 -0.02091815  
## p(Noise) -0.228167310 -0.03916719  
## p(Time) -0.008646344 0.15254846

confint(det.detect2.shrub, type="state",method="normal")

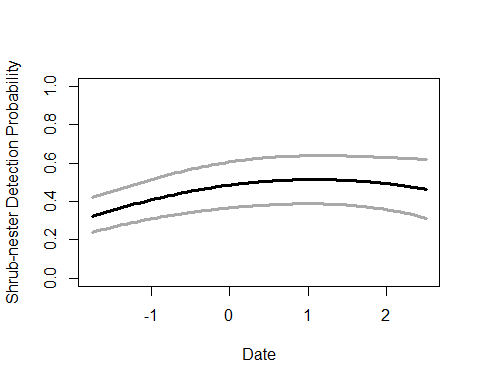
## 0.025 0.975  
## lam(Int) 1.976955 2.48169

Detect was only top model (+ with date (-with date^2), - with noise, non-sig with time)

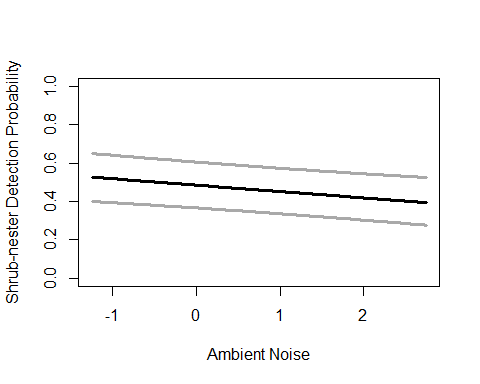
#best model is detect2 - date, noise  
summary(obsCovs(shrub.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

det.detect2.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, shrub.abund, mixture="P", K=50)  
  
NewData.SNd1 <-data.frame(Jdate=seq(-1.75,2.5,length=100),Noise=0,Time=0)  
sn.est.det1 <- predict(det.detect2.shrub, type="det",  
 newdata=NewData.SNd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=sn.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Shrub-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=sn.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=sn.est.det1, type="l", lwd=3, col="darkgray")



#best model is detect2 - date, noise  
#summary(obsCovs(shrub.abund))  
#det.detect2.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, shrub.abund, mixture="P", K=50)  
  
NewData.SNd2 <-data.frame(Noise=seq(-1.25,2.75,length=100),Jdate=0,Time=0)  
sn.est.det2 <- predict(det.detect2.shrub, type="det",  
 newdata=NewData.SNd2,appendData=TRUE)  
  
plot(Predicted~ Noise, data=sn.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Ambient Noise", ylab="Shrub-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Noise, data=sn.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Noise, data=sn.est.det2, type="l", lwd=3, col="darkgray")



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")

testR.shrub <- pcount(~1 ~BA, shrub.abund, mixture="P", K=20)  
testQ.shrub <- pcount(~1 ~BA + I(BA^2), shrub.abund, mixture="P", K=20)  
msBAtest <- fitList(testR.shrub, testQ.shrub)

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

BAtest.shrub <- modSel(msBAtest)  
BAtest.shrub

## nPars AIC delta AICwt cumltvWt  
## testR.shrub 3 1248.31 0.00 0.69 0.69  
## testQ.shrub 4 1249.87 1.56 0.31 1.00

testR.shrub <- pcount(~1 ~HW\_dens\_1050, shrub.abund, mixture="P", K=20)  
testQ.shrub <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), shrub.abund, mixture="P", K=20)  
msHW1050test <- fitList(testR.shrub, testQ.shrub)

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

HW1050test.shrub <- modSel(msHW1050test)  
HW1050test.shrub

## nPars AIC delta AICwt cumltvWt  
## testR.shrub 3 1248.78 0.00 0.73 0.73  
## testQ.shrub 4 1250.76 1.98 0.27 1.00

testR.shrub <- pcount(~1 ~FG\_herb, shrub.abund, mixture="P", K=20)  
testQ.shrub <- pcount(~1 ~FG\_herb + I(FG\_herb^2), shrub.abund, mixture="P", K=20)  
msFGHtest <- fitList(testR.shrub, testQ.shrub)

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

FGHtest.shrub <- modSel(msFGHtest)  
FGHtest.shrub

## nPars AIC delta AICwt cumltvWt  
## testR.shrub 3 1252.07 0.00 0.73 0.73  
## testQ.shrub 4 1254.01 1.94 0.27 1.00

testR.shrub <- pcount(~1 ~FG\_shrub, shrub.abund, mixture="P", K=20)  
testQ.shrub <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), shrub.abund, mixture="P", K=20)  
msFGStest <- fitList(testR.shrub, testQ.shrub)

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

FGStest.shrub <- modSel(msFGStest)  
FGStest.shrub

## nPars AIC delta AICwt cumltvWt  
## testR.shrub 3 1261.49 0.00 0.63 0.63  
## testQ.shrub 4 1262.55 1.05 0.37 1.00

testR.shrub <- pcount(~1 ~NHW\_saplings, shrub.abund, mixture="P", K=20)  
testQ.shrub <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), shrub.abund, mixture="P", K=20)  
msHWStest <- fitList(testR.shrub, testQ.shrub)

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

HWStest.shrub <- modSel(msHWStest)  
HWStest.shrub

## nPars AIC delta AICwt cumltvWt  
## testR.shrub 3 1261.45 0.00 0.65 0.65  
## testQ.shrub 4 1262.68 1.23 0.35 1.00

##site covariates next  
null.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, shrub.abund, mixture="P", K=80)  
global.shrub <- pcount(~ Jdate + I(Jdate^2) + 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 + I(Jdate^2) + Noise + Time   
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , shrub.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.shrub <- pcount(~ Jdate + I(Jdate^2) + 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 + I(Jdate^2) + Noise + Time   
 ~ Parea + ShapeIndex + YearCat  
 , shrub.abund, mixture="P",K=80)  
landscape500.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ Evergreen500m + HighDev500m + OpenDev500m  
 + Schrubs500m + Ag500m + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape1.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ Ag1km + HighDev1km + OpenDev1km  
 + Schrubs1km + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape5.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ OpenDev5km + Schrubs5km + YearCat  
 , shrub.abund, mixture="P", K=80)  
landscape30.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ Schrubs30km + HighDev30km + Protected30km + YearCat  
 , shrub.abund, mixture="P", K=80)  
treatment.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ Treatment + Nthins + YearCat  
 , shrub.abund, mixture ="P", K=80)  
management.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , shrub.abund, mixture="P", K=80)  
disturbance.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time   
 ~ TimeSinceB + TimeSinceT + YearCat  
 , shrub.abund, mixture="P", K=80)  
siteprod.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~ PISoils + NSoilTypes + YearCat  
 , shrub.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.shrub <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~ BA + HW\_dens\_1050 + Nburns + YearCat  
 , shrub.abund, mixture="P", K=80)  
coord.shrub <-pcount (~Jdate + I(Jdate^2) + 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 10 1225.04 0.00 6.9e-01 0.69  
## lh.shrub 11 1227.29 2.26 2.2e-01 0.92  
## landscape5.shrub 9 1231.42 6.38 2.9e-02 0.95  
## null.shrub 6 1232.66 7.62 1.5e-02 0.96  
## local.shrub 10 1232.80 7.76 1.4e-02 0.98  
## management.shrub 14 1234.47 9.43 6.2e-03 0.98  
## disturbance.shrub 9 1234.78 9.74 5.3e-03 0.99  
## landscape30.shrub 10 1235.99 10.96 2.9e-03 0.99  
## landscape1.shrub 11 1236.37 11.33 2.4e-03 0.99  
## treatment.shrub 11 1236.89 11.86 1.8e-03 1.00  
## coord.shrub 9 1236.90 11.86 1.8e-03 1.00  
## landmetrics.shrub 9 1237.58 12.54 1.3e-03 1.00  
## landscape500.shrub 12 1238.50 13.47 8.3e-04 1.00  
## siteprod.shrub 9 1238.58 13.55 7.9e-04 1.00  
## global.shrub 27 1255.69 30.65 1.5e-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 + I(Jdate^2) + Noise + Time ~ BA + HW\_dens\_1050 +   
## Nburns + YearCat, data = shrub.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.2915 0.1605 14.276 3.08e-46  
## BA -0.0870 0.0447 -1.944 5.19e-02  
## HW\_dens\_1050 0.0668 0.0433 1.541 1.23e-01  
## Nburns 0.0273 0.0403 0.678 4.98e-01  
## YearCatB -0.0302 0.0765 -0.394 6.94e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.1671 0.2894 -0.577 5.64e-01  
## Jdate 0.2012 0.0460 4.379 1.19e-05  
## I(Jdate^2) -0.0994 0.0399 -2.493 1.27e-02  
## Noise -0.1053 0.0477 -2.205 2.75e-02  
## Time 0.0696 0.0402 1.734 8.29e-02  
##   
## AIC: 1225.038

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

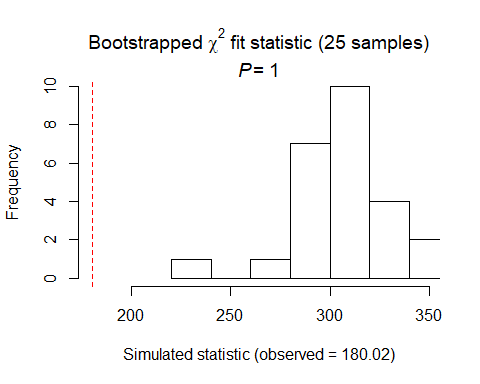
## 0.025 0.975  
## lam(Int) 1.97693452 2.6061461399  
## lam(BA) -0.17468299 0.0007046259  
## lam(HW\_dens\_1050) -0.01817429 0.1517412786  
## lam(Nburns) -0.05169140 0.1063607142  
## lam(YearCatB) -0.18019767 0.1198666125

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")

# no figure bc technically none sig

#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 = 180.0246   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 226 289 303 316 348   
##   
## Estimate of c-hat = 0.59

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 180.0246

Shrub-nesters (n=17) Summary: P distribution DCs: detect2 best and only <2.0 + with Date (- with date^2), - with Noise (non-sig with time) SCs using Date^2+Noise+Time (detect) Best model is based on Greenberg et al paper in Southern Apps on this guild technically none sig

# 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)

det.date.ground <- pcount(~ Jdate ~1, ground.abund, mixture="P", K=15)  
det.date2.ground <- pcount(~ Jdate + I(Jdate^2) ~1, ground.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.ground, det.date2.ground)

## Warning in fitList(det.date.ground, det.date2.ground): Your list was  
## unnamed, so model names were added as object names

DATEtest.ground <- modSel(mstestDATE)  
DATEtest.ground

## nPars AIC delta AICwt cumltvWt  
## det.date.ground 3 1084.09 0.00 0.71 0.71  
## det.date2.ground 4 1085.89 1.80 0.29 1.00

det.time.ground <-pcount(~ Time ~1, ground.abund, mixture="P",K=15)  
det.time2.ground <-pcount(~ Time + I(Time^2) ~1, ground.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.ground, det.time2.ground)

## Warning in fitList(det.time.ground, det.time2.ground): Your list was  
## unnamed, so model names were added as object names

TIMEtest.ground <- modSel(mstestTIME)  
TIMEtest.ground

## nPars AIC delta AICwt cumltvWt  
## det.time2.ground 4 1087.23 0.00 0.61 0.61  
## det.time.ground 3 1088.13 0.90 0.39 1.00

#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.global2.ground <- pcount(~ Jdate + Wind + Sky + Noise +Time + I(Time^2) ~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.detect2.ground <- pcount(~ Jdate + Noise + Time + I(Time^2) ~1, ground.abund, mixture="P", K=50)  
det.notdate.ground <-pcount(~ Wind + Sky + Noise ~1, ground.abund, mixture="P", K=50)  
det.time2.ground <-pcount(~ Time + I(Time^2) ~1, ground.abund, mixture="P",K=50)  
det.timing2.ground <-pcount(~ Time + I(Time^2) + Jdate ~1, ground.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.ground, det.weather.ground, det.global2.ground,  
 det.sound.ground, det.date.ground, det.detect2.ground, det.notdate.ground,  
 det.time2.ground, det.timing2.ground)

## Warning in fitList(det.null.ground, det.weather.ground,  
## det.global2.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.detect2.ground 6 1078.27 0.00 5.0e-01 0.50  
## det.timing2.ground 5 1079.36 1.10 2.9e-01 0.79  
## det.global2.ground 8 1080.48 2.21 1.7e-01 0.96  
## det.date.ground 3 1083.78 5.51 3.2e-02 0.99  
## det.time2.ground 4 1087.00 8.73 6.4e-03 1.00  
## det.null.ground 2 1092.33 14.06 4.5e-04 1.00  
## det.sound.ground 4 1093.20 14.94 2.9e-04 1.00  
## det.notdate.ground 5 1094.11 15.85 1.8e-04 1.00  
## det.weather.ground 4 1095.48 17.22 9.2e-05 1.00

#msDC.ground@Full  
#summary: detect, then global, then timing (date+time) all under 2.0  
#WITH quadratic time: detect2, then timing2 (global above 2.0)

^detect2 best (+ with date, - with time), timing2 (- with time, + with date) second best

det.detect2.ground

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time + I(Time^2) ~ 1, data = ground.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 1.89 0.166 11.4 4.48e-30  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.815 0.2384 -3.42 0.000629  
## Jdate 0.167 0.0527 3.16 0.001569  
## Noise -0.103 0.0582 -1.77 0.075902  
## Time -0.136 0.0575 -2.36 0.018337  
## I(Time^2) 0.104 0.0566 1.84 0.065731  
##   
## AIC: 1078.266

confint(det.detect2.ground, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.282284346 -0.34776751  
## p(Jdate) 0.063361519 0.27004100  
## p(Noise) -0.217414185 0.01076826  
## p(Time) -0.248322818 -0.02292900  
## p(I(Time^2)) -0.006777946 0.21516807

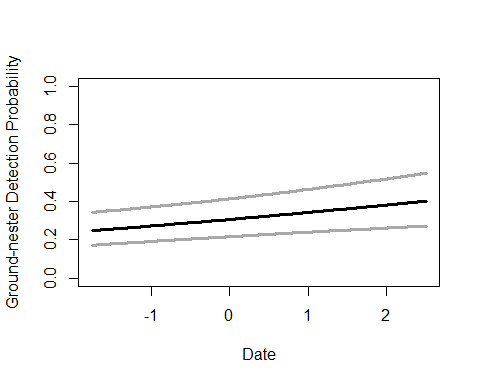
confint(det.detect2.ground, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.567509 2.21882

#best model is detect2 ... date, time - THIS ONE NOT MODEL-AVERAGED  
summary(obsCovs(ground.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

det.detect2.ground <- pcount(~ Jdate + Noise + Time + I(Time^2) ~1, ground.abund, mixture="P", K=50)  
  
NewData.GNd1 <-data.frame(Jdate=seq(-1.75,2.5,length=100),Noise=0,Time=0)  
gn.est.det1 <- predict(det.detect2.ground, type="det",  
 newdata=NewData.GNd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=gn.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Ground-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=gn.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=gn.est.det1, type="l", lwd=3, col="darkgray")



dms\_top.ground <- fitList(det.detect2.ground, det.timing2.ground)

## Warning in fitList(det.detect2.ground, det.timing2.ground): Your list was  
## unnamed, so model names were added as object names

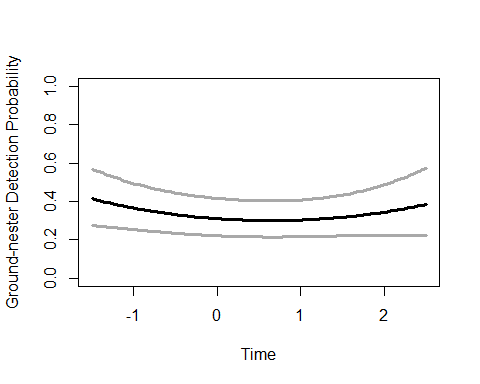
ND.GNd1 <- data.frame(Jdate=seq(-1.75,2.5,length=100),Noise=0,Time=0)  
gn.est.det11 <- predict(dms\_top.ground, type="det",  
 newdata=NewData.GNd1,appendData=TRUE)  
  
#plot Date across all top models  
plot(Predicted~ Jdate, data=gn.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Ground-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=gn.est.det11, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=gn.est.det11, type="l", lwd=3, col="darkgray")



#best model is detect2 ... date, time - this one model-averaged  
#summary(obsCovs(ground.abund))  
#det.detect2.ground <- pcount(~ Jdate + Noise + Time + I(Time^2) ~1, ground.abund, mixture="P", K=50)  
  
dms\_top.ground <- fitList(det.detect2.ground, det.timing2.ground)

## Warning in fitList(det.detect2.ground, det.timing2.ground): Your list was  
## unnamed, so model names were added as object names

ND.GNd2 <- data.frame(Time=seq(-1.5,2.5,length=100),Noise=0,Jdate=0)  
gn.est.det22 <- predict(dms\_top.ground, type="det",  
 newdata=ND.GNd2,appendData=TRUE)  
  
plot(Predicted~ Time, data=gn.est.det22, ylim=c(0,1), type="l", lwd=3,  
 xlab="Time", ylab="Ground-nester Detection Probability")  
##95% confidence intervals  
lines(lower~ Time, data=gn.est.det22, type="l", lwd=3, col="darkgray")  
lines(upper~ Time, data=gn.est.det22, type="l", lwd=3, col="darkgray")



det.timing2.ground

##   
## Call:  
## pcount(formula = ~Time + I(Time^2) + Jdate ~ 1, data = ground.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 1.86 0.156 11.9 7.83e-33  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.766 0.2279 -3.36 0.000778  
## Time -0.148 0.0578 -2.56 0.010582  
## I(Time^2) 0.116 0.0566 2.06 0.039810  
## Jdate 0.159 0.0528 3.00 0.002663  
##   
## AIC: 1079.362

confint(det.timing2.ground, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.212497063 -0.31918394  
## p(Time) -0.261105155 -0.03447225  
## p(I(Time^2)) 0.005414675 0.22709445  
## p(Jdate) 0.055112065 0.26199036

confint(det.timing2.ground, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.551486 2.161212

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")

# quadratic better for BA, HW\_dens\_1050, FG\_shrub (close but not better FG\_herb,saplings)  
testR.ground <- pcount(~1 ~BA, ground.abund, mixture="P", K=20)  
testQ.ground <- pcount(~1 ~BA + I(BA^2), ground.abund, mixture="P", K=20)  
msBAtest <- fitList(testR.ground, testQ.ground)

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

BAtest.ground <- modSel(msBAtest)  
BAtest.ground

## nPars AIC delta AICwt cumltvWt  
## testQ.ground 4 1085.66 0.00 0.75 0.75  
## testR.ground 3 1087.82 2.17 0.25 1.00

testR.ground <- pcount(~1 ~HW\_dens\_1050, ground.abund, mixture="P", K=20)  
testQ.ground <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), ground.abund, mixture="P", K=20)  
msHW1050test <- fitList(testR.ground, testQ.ground)

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

HW1050test.ground <- modSel(msHW1050test)  
HW1050test.ground

## nPars AIC delta AICwt cumltvWt  
## testQ.ground 4 1086.77 0.00 0.89 0.89  
## testR.ground 3 1091.05 4.28 0.11 1.00

testR.ground <- pcount(~1 ~FG\_herb, ground.abund, mixture="P", K=20)  
testQ.ground <- pcount(~1 ~FG\_herb + I(FG\_herb^2), ground.abund, mixture="P", K=20)  
msFGHtest <- fitList(testR.ground, testQ.ground)

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

FGHtest.ground <- modSel(msFGHtest)  
FGHtest.ground

## nPars AIC delta AICwt cumltvWt  
## testR.ground 3 1082.02 0.000 0.50 0.50  
## testQ.ground 4 1082.04 0.025 0.50 1.00

testR.ground <- pcount(~1 ~FG\_shrub, ground.abund, mixture="P", K=20)  
testQ.ground <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), ground.abund, mixture="P", K=20)  
msFGStest <- fitList(testR.ground, testQ.ground)

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

FGStest.ground <- modSel(msFGStest)  
FGStest.ground

## nPars AIC delta AICwt cumltvWt  
## testQ.ground 4 1090.38 0.00 0.52 0.52  
## testR.ground 3 1090.57 0.19 0.48 1.00

testR.ground <- pcount(~1 ~NHW\_saplings, ground.abund, mixture="P", K=20)  
testQ.ground <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), ground.abund, mixture="P", K=20)  
msHWStest <- fitList(testR.ground, testQ.ground)

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

HWStest.ground <- modSel(msHWStest)  
HWStest.ground

## nPars AIC delta AICwt cumltvWt  
## testR.ground 3 1093.79 0.00 0.56 0.56  
## testQ.ground 4 1094.27 0.48 0.44 1.00

##site covariates next (detect2 - time, not date, quadratic)  
null.ground <- pcount(~ Jdate + Noise + Time + I(Time^2) ~1, ground.abund, mixture="P", K=80)  
global.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Treatment + Herbicide + BA + I(BA^2) + Nsnags +Ccover  
 + Ldepth + TreeHt + TimeSinceB + TimeSinceT + Nthins + Nburns  
 + HW\_dens\_1050 + I(HW\_dens\_1050^2) + FG\_herb + FG\_shrub  
 + I(FG\_shrub^2) + Rel\_HW2P\_canopy + PISoils + NSoilTypes  
 + Parea + ShapeIndex + YearCat  
 , ground.abund, mixture="P", K=80) #FPSiteIndex  
local.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , ground.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ FG\_herb + FG\_shrub + I(FG\_shrub^2)  
 + HW\_dens\_1050 + I(HW\_dens\_1050^2) + Ldepth +  
 Rel\_HW2P\_canopy + BA + I(BA^2) + YearCat  
 , ground.abund, mixture="P", K=80)  
landmetrics.ground <- pcount (~ Jdate + Noise + Time + I(Time^2)  
 ~ Parea + ShapeIndex + YearCat  
 , ground.abund, mixture="P",K=80)  
landscape500.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen500m + HighDev500m + Schrubs500m + OpenDev500m  
 + Ag500m + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape1.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape5.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen5km + HighDev5km + Schrubs5km + YearCat  
 , ground.abund, mixture="P", K=80)  
landscape30.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen30km + HighDev30km + YearCat  
 , ground.abund, mixture="P", K=80) #removed Protected  
treatment.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Treatment + Nthins + YearCat  
 , ground.abund, mixture ="P", K=80)  
management.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Treatment + BA + I(BA^2) + TimeSinceB  
 + TimeSinceT + Herbicide + YearCat  
 , ground.abund, mixture="P", K=80)  
disturbance.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , ground.abund, mixture="P", K=80)  
siteprod.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ PISoils + NSoilTypes + YearCat  
 , ground.abund, mixture="P", K=80) #FPSiteIndex  
greenberg.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Ccover + Nsnags +  
 Nburns + TimeSinceB + Ldepth + YearCat  
 , ground.abund, mixture="P", K=80)  
coord.ground <- pcount (~Jdate + Noise + Time + I(Time^2)  
 ~ 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 11 1061.31 0.00 9.7e-01 0.97  
## lh.ground 16 1071.34 10.04 6.4e-03 0.98  
## local.ground 10 1071.88 10.57 4.9e-03 0.98  
## landscape500.ground 12 1072.29 10.99 4.0e-03 0.99  
## landmetrics.ground 9 1072.35 11.05 3.9e-03 0.99  
## disturbance.ground 9 1072.82 11.51 3.1e-03 1.00  
## greenberg.ground 12 1073.27 11.96 2.5e-03 1.00  
## management.ground 15 1075.39 14.08 8.5e-04 1.00  
## landscape5.ground 10 1077.18 15.88 3.5e-04 1.00  
## null.ground 6 1078.27 16.96 2.0e-04 1.00  
## landscape30.ground 9 1079.20 17.90 1.3e-04 1.00  
## global.ground 31 1080.20 18.89 7.7e-05 1.00  
## treatment.ground 11 1081.68 20.37 3.7e-05 1.00  
## siteprod.ground 9 1081.94 20.63 3.2e-05 1.00  
## coord.ground 9 1082.61 21.30 2.3e-05 1.00

#ms.ground@Full  
#summary: changed 2018: now only landscape 1km top model (same with quadratics)

landscape1.ground

##   
## Call:  
## pcount(formula = ~Jdate + Noise + Time + I(Time^2) ~ Evergreen1km +   
## HighDev1km + Schrubs1km + OpenDev1km + YearCat, data = ground.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 1.93300 0.2295 8.421 3.73e-17  
## Evergreen1km 0.19510 0.0492 3.965 7.35e-05  
## HighDev1km -0.02009 0.0606 -0.332 7.40e-01  
## Schrubs1km 0.00796 0.0501 0.159 8.74e-01  
## OpenDev1km -0.11255 0.0540 -2.086 3.70e-02  
## YearCatB 0.14275 0.0961 1.486 1.37e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.0441 0.3048 -3.426 0.000613  
## Jdate 0.1419 0.0516 2.749 0.005980  
## Noise -0.0563 0.0573 -0.983 0.325508  
## Time -0.1435 0.0560 -2.564 0.010362  
## I(Time^2) 0.1248 0.0550 2.267 0.023367  
##   
## AIC: 1061.305

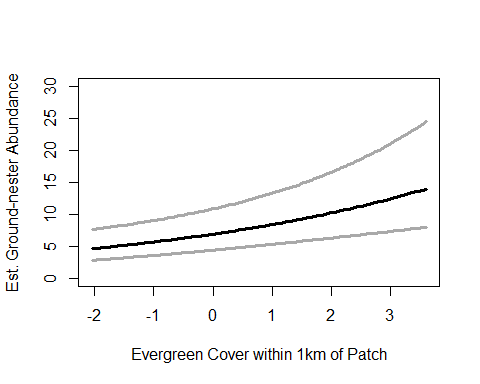
confint(landscape1.ground, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 1.48310340 2.38289810  
## lam(Evergreen1km) 0.09865131 0.29155475  
## lam(HighDev1km) -0.13889730 0.09870811  
## lam(Schrubs1km) -0.09021946 0.10614292  
## lam(OpenDev1km) -0.21831724 -0.00678766  
## lam(YearCatB) -0.04558141 0.33108666

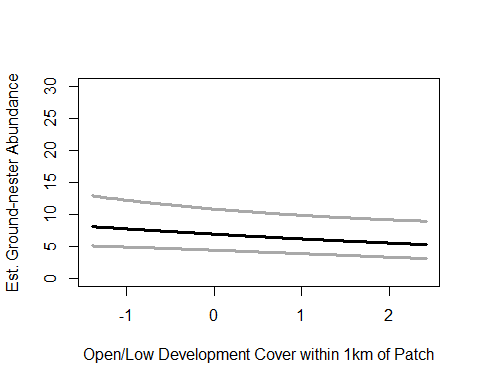
Landscape1 only top model (+ with evergreen1km, - with opendev1km)

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")

#for figures: landscape 1 model  
  
landscape1.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , ground.abund, mixture="P", K=80)  
  
NewData.ground1 <-data.frame(Evergreen1km=seq(min(sc$Evergreen1km),max(sc$Evergreen1km),length=100), OpenDev1km=0, HighDev1km=0, Schrubs1km=0, YearCat=0)  
ground.est.ev1 <- predict(landscape1.ground, type="state",  
 newdata=NewData.ground1,appendData=TRUE)  
  
plot(Predicted~ Evergreen1km, data=ground.est.ev1, ylim=c(0,30), type="l", lwd=3,  
 xlab="Evergreen Cover within 1km of Patch", ylab="Est. Ground-nester Abundance")  
##95% confidence intervals  
lines(lower~ Evergreen1km, data=ground.est.ev1, type="l", lwd=3, col="darkgray")  
lines(upper~ Evergreen1km, data=ground.est.ev1, type="l", lwd=3, col="darkgray")

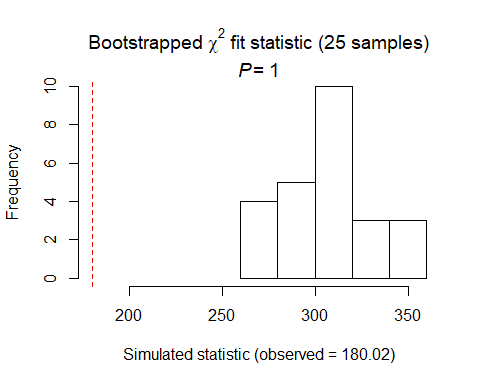


landscape1.ground <- pcount(~ Jdate + Noise + Time + I(Time^2)  
 ~ Evergreen1km + HighDev1km + Schrubs1km + OpenDev1km  
 + YearCat  
 , ground.abund, mixture="P", K=80)  
  
NewData.ground1 <-data.frame(OpenDev1km=seq(min(sc$OpenDev1km),max(sc$OpenDev1km),length=100), Evergreen1km=0, HighDev1km=0, Schrubs1km=0, YearCat=0)  
ground.est.odev1 <- predict(landscape1.ground, type="state",  
 newdata=NewData.ground1,appendData=TRUE)  
  
plot(Predicted~ OpenDev1km, data=ground.est.odev1, ylim=c(0,30), type="l", lwd=3,  
 xlab="Open/Low Development Cover within 1km of Patch", ylab="Est. Ground-nester Abundance")  
##95% confidence intervals  
lines(lower~ OpenDev1km, data=ground.est.odev1, type="l", lwd=3, col="darkgray")  
lines(upper~ OpenDev1km, data=ground.est.odev1, type="l", lwd=3, col="darkgray")



#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 = 180.0246   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 263 297 308 318 360   
##   
## Estimate of c-hat = 0.58

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 180.0246

Ground-nesters (n=10) Summary: P distribution DCs: Detect is best, then timing (date+time) both under 2.0 Detect2 (+ with date, - with time, time2) Timing2 second best (- with time, + with date) SCs using detect2: 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

#neither date nor time better quadratic  
det.date.bf <- pcount(~ Jdate ~1, bf.abund, mixture="P", K=15)  
det.date2.bf <- pcount(~ Jdate + I(Jdate^2) ~1, bf.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.bf, det.date2.bf)

## Warning in fitList(det.date.bf, det.date2.bf): Your list was unnamed, so  
## model names were added as object names

DATEtest.bf <- modSel(mstestDATE)  
DATEtest.bf

## nPars AIC delta AICwt cumltvWt  
## det.date.bf 3 1176.39 0.00 0.52 0.52  
## det.date2.bf 4 1176.59 0.20 0.48 1.00

det.time.bf <-pcount(~ Time ~1, bf.abund, mixture="P",K=15)  
det.time2.bf <-pcount(~ Time + I(Time^2) ~1, bf.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.bf, det.time2.bf)

## Warning in fitList(det.time.bf, det.time2.bf): Your list was unnamed, so  
## model names were added as object names

TIMEtest.bf <- modSel(mstestTIME)  
TIMEtest.bf

## nPars AIC delta AICwt cumltvWt  
## det.time.bf 3 1172.64 0.00 0.67 0.67  
## det.time2.bf 4 1174.03 1.38 0.33 1.00

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 only 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")

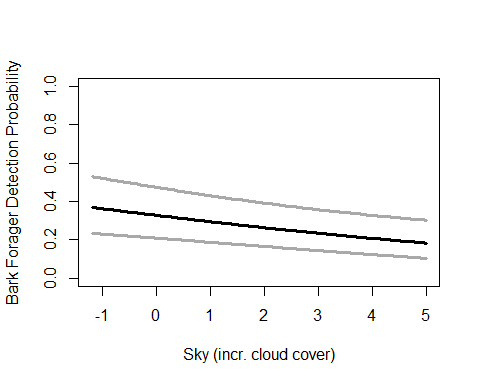
#best model is global, two variables sig (date, sky)  
det.global.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, bf.abund, mixture="P", K=50)  
  
NewData.BFd1 <-data.frame(Jdate=seq(-2,2.5,length=100),Sky=0,Wind=0,Noise=0, Time=0)  
bf.est.det1 <- predict(det.global.bf, type="det",  
 newdata=NewData.BFd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=bf.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Bark Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=bf.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=bf.est.det1, type="l", lwd=3, col="darkgray")



#best model is global, two variables sig (date, sky)  
summary(obsCovs(bf.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

det.global.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time ~1, bf.abund, mixture="P", K=50)  
  
NewData.BFd2 <-data.frame(Sky=seq(-1.2,5,length=100),Jdate=0,Wind=0,Noise=0, Time=0)  
bf.est.det2 <- predict(det.global.bf, type="det",  
 newdata=NewData.BFd2,appendData=TRUE)  
  
plot(Predicted~ Sky, data=bf.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Sky (incr. cloud cover)", ylab="Bark Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Sky, data=bf.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Sky, data=bf.est.det2, type="l", lwd=3, col="darkgray")



#none of these better as quadratic  
testR.bf <- pcount(~1 ~BA, bf.abund, mixture="P", K=10)  
testQ.bf <- pcount(~1 ~BA + I(BA^2), bf.abund, mixture="P", K=10)  
msBAtest <- fitList(testR.bf, testQ.bf)

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

BAtest.bf <- modSel(msBAtest)  
BAtest.bf

## nPars AIC delta AICwt cumltvWt  
## testR.bf 3 1201.26 0.00 0.64 0.64  
## testQ.bf 4 1202.42 1.16 0.36 1.00

testR.bf <- pcount(~1 ~HW\_dens\_1050, bf.abund, mixture="P", K=10)  
testQ.bf <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), bf.abund, mixture="P", K=10)  
msHW1050test <- fitList(testR.bf, testQ.bf)

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

HW1050test.bf <- modSel(msHW1050test)  
HW1050test.bf

## nPars AIC delta AICwt cumltvWt  
## testR.bf 3 1200.63 0.00 0.68 0.68  
## testQ.bf 4 1202.13 1.50 0.32 1.00

testR.bf <- pcount(~1 ~FG\_herb, bf.abund, mixture="P", K=10)  
testQ.bf <- pcount(~1 ~FG\_herb + I(FG\_herb^2), bf.abund, mixture="P", K=10)  
msFGHtest <- fitList(testR.bf, testQ.bf)

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

FGHtest.bf <- modSel(msFGHtest)  
FGHtest.bf

## nPars AIC delta AICwt cumltvWt  
## testR.bf 3 1200.87 0.00 0.70 0.70  
## testQ.bf 4 1202.53 1.66 0.30 1.00

testR.bf <- pcount(~1 ~FG\_shrub, bf.abund, mixture="P", K=10)  
testQ.bf <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), bf.abund, mixture="P", K=10)  
msFGStest <- fitList(testR.bf, testQ.bf)

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

FGStest.bf <- modSel(msFGStest)  
FGStest.bf

## nPars AIC delta AICwt cumltvWt  
## testR.bf 3 1200.71 0.00 0.73 0.73  
## testQ.bf 4 1202.71 2.00 0.27 1.00

testR.bf <- pcount(~1 ~NHW\_saplings, bf.abund, mixture="P", K=10)  
testQ.bf <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), bf.abund, mixture="P", K=10)  
msHWStest <- fitList(testR.bf, testQ.bf)

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

HWStest.bf <- modSel(msHWStest)  
HWStest.bf

## nPars AIC delta AICwt cumltvWt  
## testR.bf 3 1201.42 0.00 0.71 0.71  
## testQ.bf 4 1203.24 1.82 0.29 1.00

##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

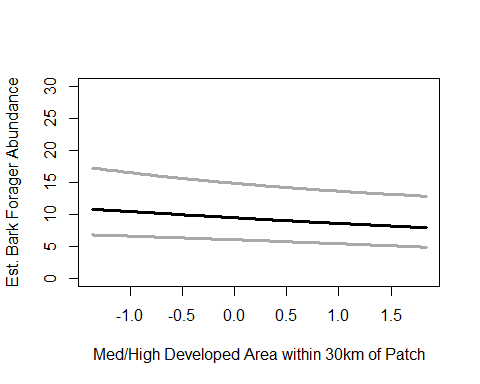
##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ Latitude +   
## Longitude + YearCat, data = bf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.2531 0.2342 9.620 6.57e-22  
## Latitude 0.0665 0.0439 1.516 1.29e-01  
## Longitude -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

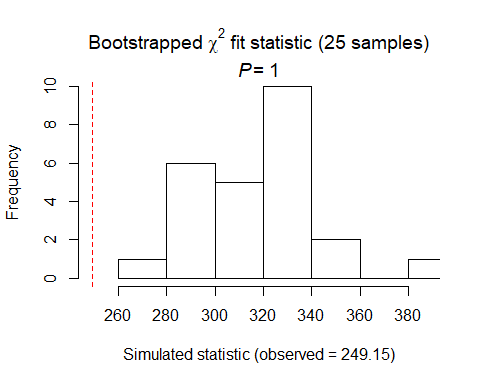
write.table(ms.bf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_bf\_top\_models\_ms.xls",sep="\t")

#for figures: landscape 30 model (-highdev30km)  
  
landscape30.bf <- pcount(~ Jdate + Wind + Sky + Noise +Time  
 ~ Evergreen30km + HighDev30km + YearCat  
 , bf.abund, mixture="P", K=80)  
  
NewData.BF <-data.frame(HighDev30km=seq(min(sc$HighDev30km),max(sc$HighDev30km),length=100),Evergreen30km=0, YearCat=0)  
bf.est.hdev <- predict(landscape30.bf, type="state",  
 newdata=NewData.BF,appendData=TRUE)  
  
plot(Predicted~ HighDev30km, data=bf.est.hdev, ylim=c(0,30), type="l", lwd=3,  
 xlab="Med/High Developed Area within 30km of Patch", ylab="Est. Bark Forager Abundance")  
##95% confidence intervals  
lines(lower~ HighDev30km, data=bf.est.hdev, type="l", lwd=3, col="darkgray")  
lines(upper~ HighDev30km, data=bf.est.hdev, type="l", lwd=3, col="darkgray")



#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



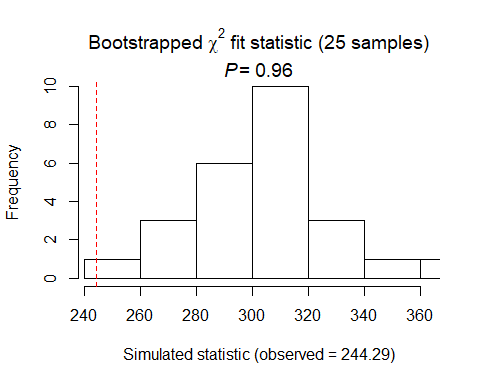
##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 249.1549   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 263 296 321 335 387   
##   
## Estimate of c-hat = 0.78

#?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 = 0.96  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 243 284 310 316 362   
##   
## Estimate of c-hat = 0.81

#?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

det.date.fg <- pcount(~ Jdate ~1, fg.abund, mixture="P", K=15)  
det.date2.fg <- pcount(~ Jdate + I(Jdate^2) ~1, fg.abund, mixture="P", K=15)  
mstestDATE <- fitList(det.date.fg, det.date2.fg)

## Warning in fitList(det.date.fg, det.date2.fg): Your list was unnamed, so  
## model names were added as object names

DATEtest.fg <- modSel(mstestDATE)  
DATEtest.fg

## nPars AIC delta AICwt cumltvWt  
## det.date2.fg 4 1387.48 0.00 0.99987 1.00  
## det.date.fg 3 1405.35 17.87 0.00013 1.00

det.time.fg <-pcount(~ Time ~1, fg.abund, mixture="P",K=15)  
det.time2.fg <-pcount(~ Time + I(Time^2) ~1, fg.abund, mixture="P",K=15)  
mstestTIME <- fitList(det.time.fg, det.time2.fg)

## Warning in fitList(det.time.fg, det.time2.fg): Your list was unnamed, so  
## model names were added as object names

TIMEtest.fg <- modSel(mstestTIME)  
TIMEtest.fg

## nPars AIC delta AICwt cumltvWt  
## det.time2.fg 4 1429.80 0.00 0.80 0.80  
## det.time.fg 3 1432.52 2.72 0.20 1.00

#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.date2.fg <- pcount(~ Jdate + I(Jdate^2) ~1, fg.abund, mixture="P", K=50)  
det.detect2.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, fg.abund, mixture="P", K=50)  
det.notdate.fg <-pcount(~ Wind + Sky + Noise ~1, fg.abund, mixture="P", K=50)  
det.time2.fg <-pcount(~ Time + I(Time^2) ~1, fg.abund, mixture="P",K=50)  
det.timing2.fg <-pcount(~ Time + I(Time^2) + Jdate + I(Jdate^2) ~1, fg.abund, mixture="P", K=50)  
  
fmsDC <- fitList(det.null.fg, det.weather.fg, det.global.fg,  
 det.sound.fg, det.date2.fg, det.detect2.fg, det.notdate.fg,  
 det.time2.fg, det.timing2.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.detect2.fg 7 1364.53 0.00 7.7e-01 0.77  
## det.timing2.fg 6 1367.39 2.86 1.9e-01 0.96  
## det.date2.fg 4 1370.41 5.88 4.1e-02 1.00  
## det.global.fg 7 1379.50 14.97 4.3e-04 1.00  
## det.notdate.fg 5 1395.49 30.96 1.5e-07 1.00  
## det.weather.fg 4 1396.26 31.73 1.0e-07 1.00  
## det.sound.fg 4 1399.33 34.81 2.1e-08 1.00  
## det.time2.fg 4 1400.70 36.18 1.1e-08 1.00  
## det.null.fg 2 1401.85 37.32 6.1e-09 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  
#with quadratic - same - detect2 is only best.

det.detect2.fg

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~   
## 1, data = fg.abund, K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 2.59 0.112 23.1 1.4e-117  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) 0.0607 0.2292 0.265 7.91e-01  
## Jdate 0.1962 0.0388 5.050 4.43e-07  
## I(Jdate^2) -0.1261 0.0351 -3.593 3.27e-04  
## Noise -0.0881 0.0402 -2.190 2.85e-02  
## Time 0.1046 0.0376 2.784 5.37e-03  
## I(Time^2) -0.0339 0.0366 -0.926 3.54e-01  
##   
## AIC: 1364.526

confint(det.detect2.fg, type="det",method="normal")

## 0.025 0.975  
## p(Int) -0.38856301 0.510023112  
## p(Jdate) 0.12003168 0.272320374  
## p(I(Jdate^2)) -0.19483287 -0.057294983  
## p(Noise) -0.16694693 -0.009244067  
## p(Time) 0.03095415 0.178216365  
## p(I(Time^2)) -0.10560955 0.037820734

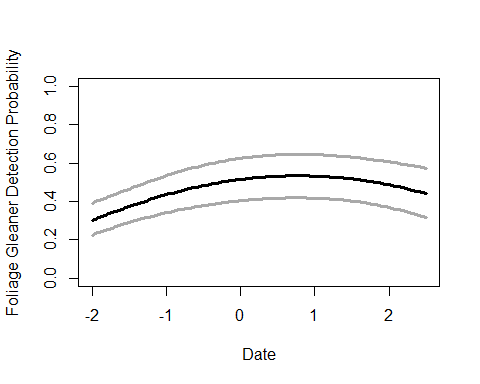
confint(det.detect2.fg, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.370484 2.811032

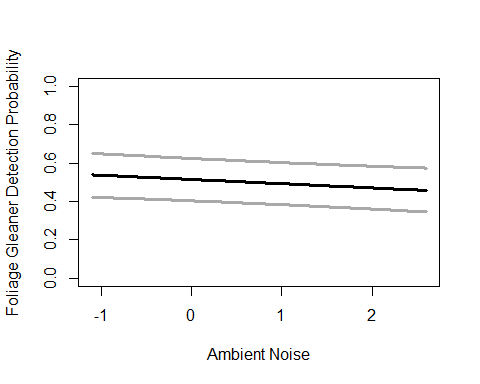
detect model (- Jdate2, - noise, time?)

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")

#best model, one variable (date)  
det.detect2.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, fg.abund, mixture="P", K=50)  
  
NewData.FGd1 <-data.frame(Jdate=seq(-2,2.5,length=100),Noise=0, Time=0)  
fg.est.det1 <- predict(det.detect2.fg, type="det",  
 newdata=NewData.FGd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=fg.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Foliage Gleaner Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=fg.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=fg.est.det1, type="l", lwd=3, col="darkgray")



#best model next variable (noise)  
det.detect2.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~1, fg.abund, mixture="P", K=50)  
  
NewData.FGd2 <-data.frame(Noise=seq(-1.1,2.6,length=100),Jdate=0, Time=0)  
fg.est.det2 <- predict(det.detect2.fg, type="det",  
 newdata=NewData.FGd2,appendData=TRUE)  
  
plot(Predicted~ Noise, data=fg.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Ambient Noise", ylab="Foliage Gleaner Detection Probability")  
##95% confidence intervals  
lines(lower~ Noise, data=fg.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Noise, data=fg.est.det2, type="l", lwd=3, col="darkgray")



#none of these quad better  
testR.fg <- pcount(~1 ~BA, fg.abund, mixture="P", K=20)  
testQ.fg <- pcount(~1 ~BA + I(BA^2), fg.abund, mixture="P", K=20)  
msBAtest <- fitList(testR.fg, testQ.fg)

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

BAtest.fg <- modSel(msBAtest)  
BAtest.fg

## nPars AIC delta AICwt cumltvWt  
## testR.fg 3 1403.54 0.00 0.73 0.73  
## testQ.fg 4 1405.53 2.00 0.27 1.00

testR.fg <- pcount(~1 ~HW\_dens\_1050, fg.abund, mixture="P", K=20)  
testQ.fg <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), fg.abund, mixture="P", K=20)  
msHW1050test <- fitList(testR.fg, testQ.fg)

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

HW1050test.fg <- modSel(msHW1050test)  
HW1050test.fg

## nPars AIC delta AICwt cumltvWt  
## testR.fg 3 1401.21 0.00 0.72 0.72  
## testQ.fg 4 1403.08 1.87 0.28 1.00

testR.fg <- pcount(~1 ~FG\_herb, fg.abund, mixture="P", K=20)  
testQ.fg <- pcount(~1 ~FG\_herb + I(FG\_herb^2), fg.abund, mixture="P", K=20)  
msFGHtest <- fitList(testR.fg, testQ.fg)

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

FGHtest.fg <- modSel(msFGHtest)  
FGHtest.fg

## nPars AIC delta AICwt cumltvWt  
## testR.fg 3 1401.27 0.00 0.67 0.67  
## testQ.fg 4 1402.69 1.43 0.33 1.00

testR.fg <- pcount(~1 ~FG\_shrub, fg.abund, mixture="P", K=20)  
testQ.fg <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), fg.abund, mixture="P", K=20)  
msFGStest <- fitList(testR.fg, testQ.fg)

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

FGStest.fg <- modSel(msFGStest)  
FGStest.fg

## nPars AIC delta AICwt cumltvWt  
## testR.fg 3 1408.71 0.00 0.66 0.66  
## testQ.fg 4 1410.05 1.34 0.34 1.00

testR.fg <- pcount(~1 ~NHW\_saplings, fg.abund, mixture="P", K=20)  
testQ.fg <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), fg.abund, mixture="P", K=20)  
msHWStest <- fitList(testR.fg, testQ.fg)

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

HWStest.fg <- modSel(msHWStest)  
HWStest.fg

## nPars AIC delta AICwt cumltvWt  
## testR.fg 3 1406.47 0.00 0.70 0.70  
## testQ.fg 4 1408.12 1.65 0.30 1.00

#more appropriate detection covariates (Date+Noise+Time from best model)(now quad)  
null.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~1, fg.abund, mixture="P", K=80)  
global.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ 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 + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Ccover + TreeHt + Ldepth + YearCat  
 , fg.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ TreeHt + Ccover + Nsnags + Rel\_HW2P\_canopy + NHW\_saplings + FG\_herb  
 + HW\_dens\_1050 + YearCat  
 , fg.abund, mixture="P", K=80)  
landmetrics.fg <- pcount (~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Parea + ShapeIndex + YearCat  
 , fg.abund, mixture="P",K=80)  
landscape500.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Evergreen500m + OpenDev500m + Schrubs500m + Ag500m + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape1.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ OpenDev1km + Schrubs1km + Ag1km + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape5.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ OpenDev5km + Ag5km + YearCat  
 , fg.abund, mixture="P", K=80)  
landscape30.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ HighDev30km + Protected30km + Schrubs30km + YearCat  
 , fg.abund, mixture="P", K=80)  
treatment.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Treatment + Nthins + YearCat  
 , fg.abund, mixture ="P", K=80)  
management.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , fg.abund, mixture="P", K=80)  
disturbance.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ TimeSinceB + TimeSinceT + YearCat  
 , fg.abund, mixture="P", K=80)  
siteprod.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ PISoils + NSoilTypes + YearCat  
 , fg.abund, mixture="P", K=80) #FPSiteIndex  
#greenberg.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)  
#~ BA + Nsnags + Nburns, fg.abund, mixture="P", K=80)  
coord.ground <- pcount (~Jdate + I(Jdate^2) + Noise + Time + I(Time^2)  
 ~ 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 11 1360.94 0.00 4.8e-01 0.48  
## landscape500.fg 12 1362.80 1.86 1.9e-01 0.67  
## disturbance.fg 10 1363.04 2.11 1.7e-01 0.84  
## null.fg 7 1364.53 3.59 8.0e-02 0.92  
## lh.fg 15 1367.08 6.15 2.2e-02 0.94  
## landscape5.fg 10 1367.95 7.02 1.4e-02 0.96  
## treatment.fg 12 1368.67 7.74 1.0e-02 0.97  
## siteprod.fg 10 1368.87 7.94 9.1e-03 0.98  
## local.fg 11 1368.98 8.04 8.6e-03 0.98  
## management.fg 15 1369.10 8.16 8.1e-03 0.99  
## landmetrics.fg 10 1370.39 9.45 4.3e-03 1.00  
## landscape30.fg 11 1371.02 10.08 3.1e-03 1.00  
## global.fg 29 1383.50 22.57 6.1e-06 1.00

#ms.fg@Full  
#slight change  
#summary: landscape at 1km best, landscape @ 500m second best (order unchanged w quadratic)

landscape1.fg

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~   
## OpenDev1km + Schrubs1km + Ag1km + YearCat, data = fg.abund,   
## K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.6352 0.1357 19.414 5.87e-84  
## OpenDev1km -0.0183 0.0326 -0.563 5.74e-01  
## Schrubs1km 0.0068 0.0320 0.212 8.32e-01  
## Ag1km -0.1065 0.0360 -2.954 3.14e-03  
## YearCatB 0.0109 0.0628 0.174 8.62e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.0532 0.2574 -0.207 8.36e-01  
## Jdate 0.1836 0.0386 4.752 2.01e-06  
## I(Jdate^2) -0.1207 0.0347 -3.483 4.95e-04  
## Noise -0.0652 0.0400 -1.629 1.03e-01  
## Time 0.0959 0.0369 2.601 9.28e-03  
## I(Time^2) -0.0308 0.0356 -0.866 3.87e-01  
##   
## AIC: 1360.936

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

## 0.025 0.975  
## lam(Int) 2.36914284 2.90121620  
## lam(OpenDev1km) -0.08213103 0.04550087  
## lam(Schrubs1km) -0.05596486 0.06956268  
## lam(Ag1km) -0.17712262 -0.03583030  
## lam(YearCatB) -0.11206569 0.13393795

landscape500.fg

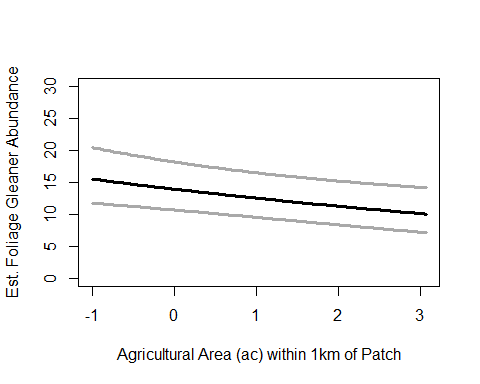
##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time + I(Time^2) ~   
## Evergreen500m + OpenDev500m + Schrubs500m + Ag500m + YearCat,   
## data = fg.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 2.63256 0.1347 19.5489 4.22e-85  
## Evergreen500m -0.02482 0.0377 -0.6592 5.10e-01  
## OpenDev500m -0.01113 0.0340 -0.3273 7.43e-01  
## Schrubs500m -0.00677 0.0341 -0.1984 8.43e-01  
## Ag500m -0.12264 0.0403 -3.0433 2.34e-03  
## YearCatB 0.00496 0.0628 0.0789 9.37e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.0421 0.2563 -0.164 8.70e-01  
## Jdate 0.1865 0.0387 4.824 1.41e-06  
## I(Jdate^2) -0.1199 0.0348 -3.442 5.77e-04  
## Noise -0.0702 0.0401 -1.749 8.03e-02  
## Time 0.0966 0.0370 2.614 8.96e-03  
## I(Time^2) -0.0320 0.0358 -0.894 3.71e-01  
##   
## AIC: 1362.799

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

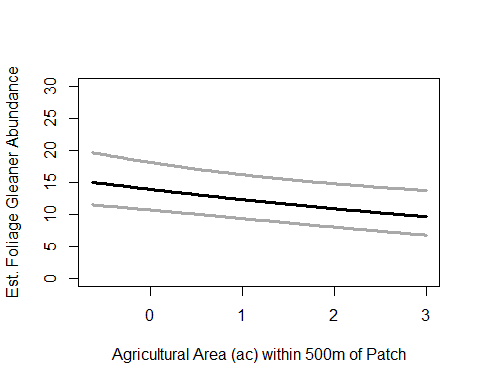
## 0.025 0.975  
## lam(Int) 2.36862183 2.89650078  
## lam(Evergreen500m) -0.09863549 0.04898711  
## lam(OpenDev500m) -0.07777710 0.05551557  
## lam(Schrubs500m) -0.07364973 0.06011130  
## lam(Ag500m) -0.20161615 -0.04365386  
## lam(YearCatB) -0.11815059 0.12806856

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")

#for figures: landscape 1 model (Ag1) landscape 500m model (Ag500)  
  
landscape1.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ OpenDev1km + Schrubs1km + Ag1km + YearCat  
 , fg.abund, mixture="P", K=80)  
  
NewData.FG1 <-data.frame(Ag1km=seq(min(sc$Ag1km),max(sc$Ag1km),length=100), OpenDev1km=0, Schrubs1km=0, YearCat=0)  
fg.est.ag1 <- predict(landscape1.fg, type="state",  
 newdata=NewData.FG1,appendData=TRUE)  
  
plot(Predicted~ Ag1km, data=fg.est.ag1, ylim=c(0,30), type="l", lwd=3,  
 xlab="Agricultural Area (ac) within 1km of Patch", ylab="Est. Foliage Gleaner Abundance")  
##95% confidence intervals  
lines(lower~ Ag1km, data=fg.est.ag1, type="l", lwd=3, col="darkgray")  
lines(upper~ Ag1km, data=fg.est.ag1, type="l", lwd=3, col="darkgray")

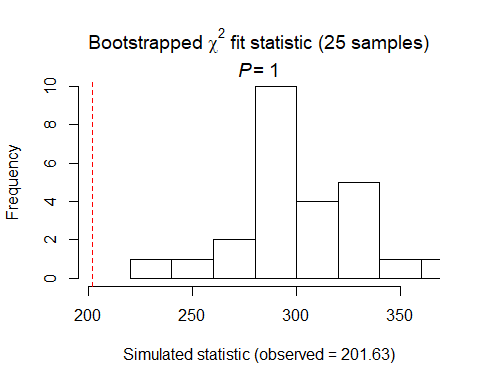


landscape500.fg <- pcount(~ Jdate + I(Jdate^2) + Noise + Time + I(Time^2)   
 ~ Evergreen500m + OpenDev500m + Schrubs500m + Ag500m + YearCat  
 , fg.abund, mixture="P", K=80)  
  
NewData.FG500 <-data.frame(Ag500m=seq(min(sc$Ag500m),max(sc$Ag500m),length=100), Evergreen500m=0, OpenDev500m=0, Schrubs500m=0, YearCat=0)  
fg.est.ag500 <- predict(landscape500.fg, type="state",  
 newdata=NewData.FG500,appendData=TRUE)  
  
plot(Predicted~ Ag500m, data=fg.est.ag500, ylim=c(0,30), type="l", lwd=3,  
 xlab="Agricultural Area (ac) within 500m of Patch", ylab="Est. Foliage Gleaner Abundance")  
##95% confidence intervals  
lines(lower~ Ag500m, data=fg.est.ag500, type="l", lwd=3, col="darkgray")  
lines(upper~ Ag500m, data=fg.est.ag500, type="l", lwd=3, col="darkgray")



#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



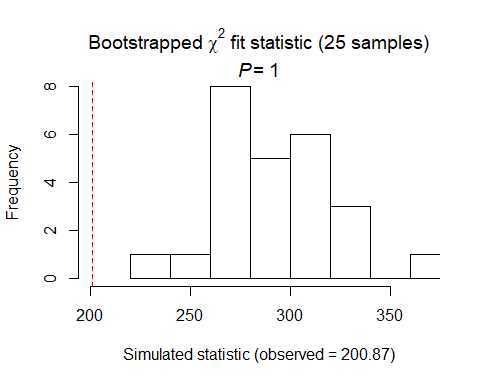
##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 201.6255   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 239 284 297 322 362   
##   
## Estimate of c-hat = 0.67

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 201.6255

#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 = 200.868   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 237 274 293 311 368   
##   
## Estimate of c-hat = 0.69

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 200.868

Foliage gleaners (n=21) Summary: P distribution DCs: detect (date, noise, time) is only best - with noise, ? with date2 & time2 SCs using Date+Noise+Time: Landscape at 1km is best: Only Ag significant (-) Landscape @ 500m is second best Only Ag is 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|) ## 3.41 0.195 17.5 1.75e-68 ## ## Detection: ## Estimate SE z P(>|z|) ## (Intercept) -0.9696 0.2698 -3.59 3.26e-04 ## Jdate 0.1015 0.0252 4.02 5.73e-05 ## I(Jdate^2) -0.0477 0.0241 -1.98 4.82e-02 ## ## AIC: 1490.75 ```

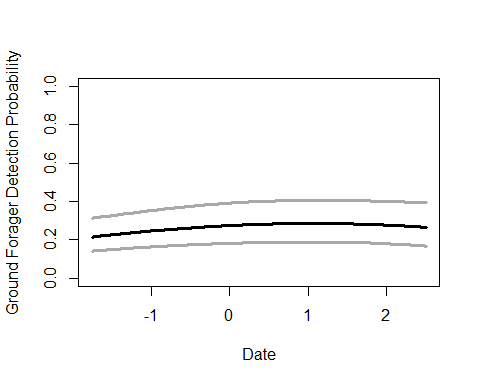
confint(det.date2.gf, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.49846152 -0.4407944429  
## p(Jdate) 0.05206256 0.1509537137  
## p(I(Jdate^2)) -0.09503399 -0.0003769477

confint(det.date2.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 3.031947 3.797316

#best model is date^2 - date plotted here  
#summary(obsCovs(gf.abund))  
det.date2.gf <- pcount(~ Jdate + I(Jdate^2) ~1, gf.abund, mixture="P", K=50)  
  
NewData.GFd1 <-data.frame(Jdate=seq(-1.75,2.5,length=100))  
gf.est.det1 <- predict(det.date2.gf, type="det",  
 newdata=NewData.GFd1,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=gf.est.det1, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Ground Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=gf.est.det1, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=gf.est.det1, type="l", lwd=3, col="darkgray")



det.detect2.gf

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) + Noise + Time ~ 1, data = gf.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.42 0.193 17.8 1.25e-70  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.98176 0.2654 -3.699 2.17e-04  
## Jdate 0.10466 0.0260 4.026 5.68e-05  
## I(Jdate^2) -0.04468 0.0242 -1.849 6.44e-02  
## Noise -0.05282 0.0277 -1.906 5.67e-02  
## Time -0.00577 0.0248 -0.233 8.16e-01  
##   
## AIC: 1490.954

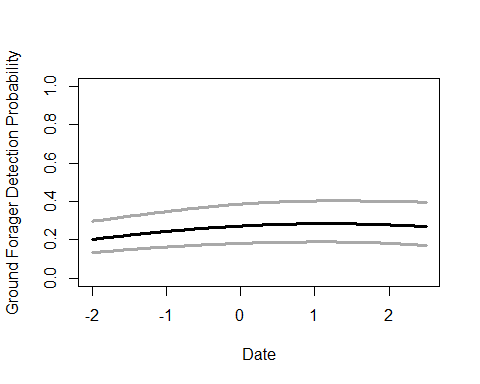
confint(det.detect2.gf, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.50199208 -0.461534462  
## p(Jdate) 0.05370538 0.155623208  
## p(I(Jdate^2)) -0.09203623 0.002677484  
## p(Noise) -0.10714466 0.001496601  
## p(Time) -0.05433240 0.042784500

confint(det.detect2.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 3.043619 3.798346

#second best model is detect2 - date shown here only sig one  
det.detect22.gf <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, gf.abund, mixture="P", K=50)  
  
NewData.GFd2 <-data.frame(Jdate=seq(-2,2.5,length=100),Noise=0, Time=0)  
gf.est.det2 <- predict(det.detect22.gf, type="det",  
 newdata=NewData.GFd2,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=gf.est.det2, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Ground Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=gf.est.det2, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=gf.est.det2, type="l", lwd=3, col="darkgray")



det.global.gf #noise as well as date!

##   
## Call:  
## pcount(formula = ~Jdate + Wind + Sky + Noise + Time ~ 1, data = gf.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|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

det.timing2.gf

##   
## Call:  
## pcount(formula = ~Time + Jdate + I(Jdate^2) ~ 1, data = gf.abund,   
## K = 50, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## 3.41 0.196 17.4 3.91e-68  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -0.96946 0.2705 -3.584 0.000339  
## Time -0.00886 0.0248 -0.358 0.720525  
## Jdate 0.09943 0.0259 3.846 0.000120  
## I(Jdate^2) -0.04751 0.0242 -1.967 0.049168  
##   
## AIC: 1492.622

confint(det.timing2.gf, type="det",method="normal")

## 0.025 0.975  
## p(Int) -1.49964444 -0.4392711170  
## p(Time) -0.05742978 0.0397003692  
## p(Jdate) 0.04875813 0.1501062441  
## p(I(Jdate^2)) -0.09483999 -0.0001731725

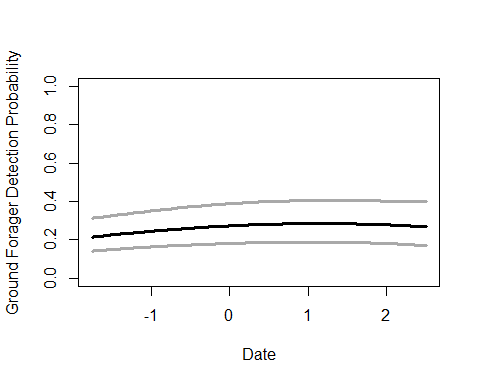
confint(det.timing2.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 3.030699 3.798009

#FINALLY, now the model-averaged one for DATE among all top 4 models:  
det.date2.gf <- pcount(~ Jdate + I(Jdate^2) ~1, gf.abund, mixture="P", K=50)  
det.detect2.gf <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, gf.abund, mixture="P", K=50)  
det.global2.gf <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time ~1, gf.abund, mixture="P", K=50)  
det.timing2.gf <-pcount(~ Time + Jdate + I(Jdate^2) ~1, gf.abund, mixture="P", K=50)  
#top fit list  
dms\_top.gf<- fitList(det.date2.gf,det.detect2.gf,det.global2.gf,det.timing2.gf)

## Warning in fitList(det.date2.gf, det.detect2.gf, det.global2.gf,  
## det.timing2.gf): Your list was unnamed, so model names were added as object  
## names

ND.GFd3 <-data.frame(Jdate=seq(-1.75,2.5,length=100),Wind=0,Sky=0,Noise=0,Time=0)  
gf.est.det3 <- predict(dms\_top.gf, type="det",  
 newdata=ND.GFd3,appendData=TRUE)  
  
plot(Predicted~ Jdate, data=gf.est.det3, ylim=c(0,1), type="l", lwd=3,  
 xlab="Date", ylab="Ground Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Jdate, data=gf.est.det3, type="l", lwd=3, col="darkgray")  
lines(upper~ Jdate, data=gf.est.det3, type="l", lwd=3, col="darkgray")



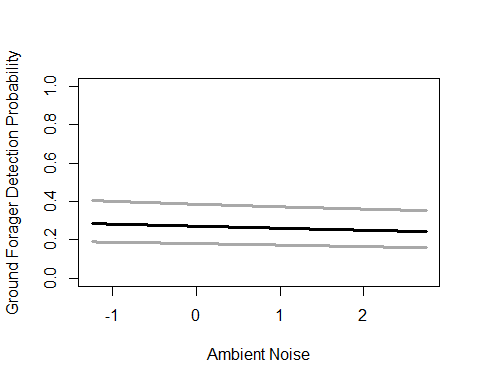
summary(obsCovs(gf.abund))

## Noise Wind Sky Jdate   
## Min. :-1.0298 Min. :-1.0690 Min. :-1.17271 Min. :-1.61739   
## 1st Qu.:-1.0298 1st Qu.:-1.0690 1st Qu.:-1.17271 1st Qu.:-0.95863   
## Median : 0.1654 Median : 0.2385 Median : 0.02255 Median :-0.05061   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.00000   
## 3rd Qu.: 1.0618 3rd Qu.: 0.2385 3rd Qu.: 1.21782 3rd Qu.: 0.80399   
## Max. : 2.5558 Max. : 2.8535 Max. : 4.80361 Max. : 2.22834   
## NA's :86 NA's :86 NA's :86 NA's :86   
## Time   
## Min. :-1.4515   
## 1st Qu.:-0.9236   
## Median :-0.1419   
## Mean : 0.0000   
## 3rd Qu.: 0.7543   
## Max. : 2.4936   
## NA's :86

# now the model-averaged one for NOISE among 2 of the 4 models:  
det.detect2.gf <- pcount(~ Jdate + I(Jdate^2) + Noise + Time ~1, gf.abund, mixture="P", K=50)  
det.global2.gf <- pcount(~ Jdate + I(Jdate^2) + Wind + Sky + Noise +Time ~1, gf.abund, mixture="P", K=50)  
#top fit list  
dms\_top2.gf<- fitList(det.detect2.gf,det.global2.gf)

## Warning in fitList(det.detect2.gf, det.global2.gf): Your list was unnamed,  
## so model names were added as object names

ND.GFd4 <-data.frame(Noise=seq(-1.25,2.75,length=100),Wind=0,Sky=0,Jdate=0,Time=0)  
gf.est.det4 <- predict(dms\_top2.gf, type="det",  
 newdata=ND.GFd4,appendData=TRUE)  
  
plot(Predicted~ Noise, data=gf.est.det4, ylim=c(0,1), type="l", lwd=3,  
 xlab="Ambient Noise", ylab="Ground Forager Detection Probability")  
##95% confidence intervals  
lines(lower~ Noise, data=gf.est.det4, type="l", lwd=3, col="darkgray")  
lines(upper~ Noise, data=gf.est.det4, type="l", lwd=3, col="darkgray")



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")

testR.gf <- pcount(~1 ~BA, gf.abund, mixture="P", K=20)  
testQ.gf <- pcount(~1 ~BA + I(BA^2), gf.abund, mixture="P", K=20)  
msBAtest <- fitList(testR.gf, testQ.gf)

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

BAtest.gf <- modSel(msBAtest)  
BAtest.gf

## nPars AIC delta AICwt cumltvWt  
## testR.gf 3 1556.14 0.00 0.71 0.71  
## testQ.gf 4 1557.89 1.75 0.29 1.00

testR.gf <- pcount(~1 ~HW\_dens\_1050, gf.abund, mixture="P", K=20)  
testQ.gf <- pcount(~1 ~HW\_dens\_1050 + I(HW\_dens\_1050^2), gf.abund, mixture="P", K=20)  
msHW1050test <- fitList(testR.gf, testQ.gf)

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

HW1050test.gf <- modSel(msHW1050test)  
HW1050test.gf

## nPars AIC delta AICwt cumltvWt  
## testR.gf 3 1558.34 0.00 0.73 0.73  
## testQ.gf 4 1560.34 2.00 0.27 1.00

testR.gf <- pcount(~1 ~FG\_herb, gf.abund, mixture="P", K=20)  
testQ.gf <- pcount(~1 ~FG\_herb + I(FG\_herb^2), gf.abund, mixture="P", K=20)  
msFGHtest <- fitList(testR.gf, testQ.gf)

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

FGHtest.gf <- modSel(msFGHtest)  
FGHtest.gf

## nPars AIC delta AICwt cumltvWt  
## testR.gf 3 1556.09 0.00 0.63 0.63  
## testQ.gf 4 1557.14 1.05 0.37 1.00

testR.gf <- pcount(~1 ~FG\_shrub, gf.abund, mixture="P", K=20)  
testQ.gf <- pcount(~1 ~FG\_shrub + I(FG\_shrub^2), gf.abund, mixture="P", K=20)  
msFGStest <- fitList(testR.gf, testQ.gf)

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

FGStest.gf <- modSel(msFGStest)  
FGStest.gf

## nPars AIC delta AICwt cumltvWt  
## testR.gf 3 1556.14 0.00 0.69 0.69  
## testQ.gf 4 1557.75 1.61 0.31 1.00

testR.gf <- pcount(~1 ~NHW\_saplings, gf.abund, mixture="P", K=20)  
testQ.gf <- pcount(~1 ~NHW\_saplings + I(NHW\_saplings^2), gf.abund, mixture="P", K=20)  
msHWStest <- fitList(testR.gf, testQ.gf)

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

HWStest.gf <- modSel(msHWStest)  
HWStest.gf

## nPars AIC delta AICwt cumltvWt  
## testR.gf 3 1555.52 0.00 0.73 0.73  
## testQ.gf 4 1557.49 1.97 0.27 1.00

##site covariates next  
# detection covariates (no longer global, now date^2)  
null.gf <- pcount(~ Jdate + I(Jdate^2) ~1, gf.abund, mixture="P", K=80)  
global.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ 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 + I(Jdate^2)  
 ~ Ccover + TreeHt + Ldepth+ YearCat  
 , gf.abund, mixture="P", K=80) #can only include BA OR CCover  
lh.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ 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 + I(Jdate^2)  
 ~ Parea + ShapeIndex+ YearCat  
 , gf.abund, mixture="P",K=80)  
landscape500.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ Evergreen500m + HighDev500m + Schrubs500m + Ag500m+ YearCat  
 , gf.abund, mixture="P", K=80)  
landscape1.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ HighDev1km + Schrubs1km + Ag1km + YearCat  
 , gf.abund, mixture="P", K=80)  
landscape5.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ Ag5km + HighDev5km + YearCat  
 , gf.abund, mixture="P", K=80)  
landscape30.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ HighDev30km + Protected30km + YearCat  
 , gf.abund, mixture="P", K=80)  
treatment.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ Treatment + Nthins + YearCat  
 , gf.abund, mixture ="P", K=80)  
management.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ Treatment + BA + TimeSinceB + TimeSinceT + Herbicide + YearCat  
 , gf.abund, mixture="P", K=80)  
disturbance.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ TimeSinceB + TimeSinceT + YearCat  
 , gf.abund, mixture="P", K=80)  
siteprod.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ 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 + I(Jdate^2)  
 ~ 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 8 1484.76 0.00 7.6e-01 0.76  
## lh.gf 12 1489.02 4.27 9.0e-02 0.85  
## null.gf 4 1490.70 5.94 3.9e-02 0.89  
## treatment.gf 9 1491.14 6.38 3.1e-02 0.92  
## coord.gf 7 1491.41 6.65 2.7e-02 0.95  
## disturbance.gf 7 1492.49 7.73 1.6e-02 0.96  
## landmetrics.gf 7 1492.53 7.77 1.6e-02 0.98  
## landscape30.gf 7 1494.23 9.47 6.7e-03 0.98  
## siteprod.gf 7 1494.75 9.99 5.1e-03 0.99  
## management.gf 12 1494.77 10.01 5.1e-03 0.99  
## landscape5.gf 7 1495.68 10.92 3.2e-03 1.00  
## landscape500.gf 9 1497.22 12.46 1.5e-03 1.00  
## landscape1.gf 8 1498.19 13.43 9.2e-04 1.00  
## global.gf 26 1505.69 20.93 2.2e-05 1.00

#ms.gf@Full  
#summary: local best & only top model ... now life history bumped past 2.0

local.gf

##   
## Call:  
## pcount(formula = ~Jdate + I(Jdate^2) ~ Ccover + TreeHt + Ldepth +   
## YearCat, data = gf.abund, K = 80, mixture = "P")  
##   
## Abundance:  
## Estimate SE z P(>|z|)  
## (Intercept) 3.71405 0.3899 9.5251 1.65e-21  
## Ccover -0.08386 0.0243 -3.4491 5.62e-04  
## TreeHt 0.01207 0.0266 0.4532 6.50e-01  
## Ldepth -0.00964 0.0269 -0.3584 7.20e-01  
## YearCatB 0.00506 0.0535 0.0946 9.25e-01  
##   
## Detection:  
## Estimate SE z P(>|z|)  
## (Intercept) -1.3743 0.4881 -2.82 0.004872  
## Jdate 0.0912 0.0248 3.67 0.000238  
## I(Jdate^2) -0.0400 0.0235 -1.70 0.089362  
##   
## AIC: 1484.758

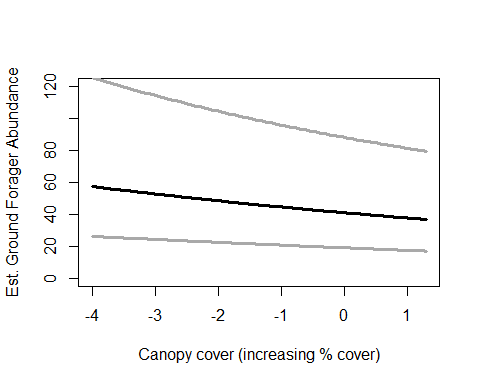
confint(local.gf, type="state",method="normal")

## 0.025 0.975  
## lam(Int) 2.94982079 4.47828313  
## lam(Ccover) -0.13151382 -0.03620666  
## lam(TreeHt) -0.04014470 0.06429272  
## lam(Ldepth) -0.06236603 0.04308468  
## lam(YearCatB) -0.09984416 0.10997087

#no longer top model  
#lh.gf  
#confint(lh.gf, type="state",method="normal")

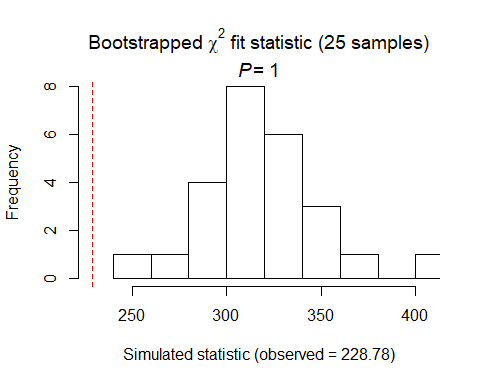
write.table(ms.gf@Full, file="C:/Users/woodj/Documents/GRAD SCHOOL - CLEMSON/Project-Specific/R work/USDA-songbirds/USDA-songbirds/Behavior\_gf\_top\_models\_ms.xls",sep="\t")

#for figures: local model only & canopy cover only variable  
  
local.gf <- pcount(~ Jdate + I(Jdate^2)  
 ~ Ccover + TreeHt + Ldepth+ YearCat  
 , gf.abund, mixture="P", K=80)  
  
NewData.GF <-data.frame(Ccover=seq(min(sc$Ccover),max(sc$Ccover),length=100),TreeHt=0, Ldepth=0, YearCat=0)  
gf.est.ccover <- predict(local.gf, type="state",  
 newdata=NewData.GF,appendData=TRUE)  
  
plot(Predicted~ Ccover, data=gf.est.ccover, ylim=c(0,120), type="l", lwd=3,  
 xlab="Canopy cover (increasing % cover)", ylab="Est. Ground Forager Abundance")  
##95% confidence intervals  
lines(lower~ Ccover, data=gf.est.ccover, type="l", lwd=3, col="darkgray")  
lines(upper~ Ccover, data=gf.est.ccover, type="l", lwd=3, col="darkgray")

 ##why so so high?

#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 = 228.7844   
## Number of bootstrap samples = 25  
## P-value = 1  
##   
## Quantiles of bootstrapped statistics:  
## 0% 25% 50% 75% 100%   
## 257 302 315 333 405   
##   
## Estimate of c-hat = 0.72

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

##   
## Chi-square goodness-of-fit for N-mixture model of 'unmarkedFitPCount' class  
##   
## Observed chi-square statistic = 228.7844

Ground foragers (n=27) Summary: P distribution DCs: date2 (+ with date, - with date^2) then detect2 (only date sig) then global (date, noise sig) then timing2 () under 2.0 SCs using date2: Local best (- with canopy cover)