

Appendix 3: Jags model code

Spatio-temporal population change of arctic-breeding waterbirds

Example model notation for a negative binomial generalized linear mixed model implemented in jags (Plummer 2003) through R (R Core Team 2018) to evaluate spatially-explicit abundance and trends of waterbirds on the Arctic Coastal Plain, Alaska.

- Plummer, M. 2003. JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003), March 20–22, Vienna, Austria. ISSN 1609–395X.
<https://www.r-project.org/conferences/DSC-2003/Drafts/Plummer.pdf> (14 March 2018).
- R Core Team. 2018. R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing. <http://www.R-project.org/> (14 March 2018).
- Wood, S. N. 2018. Package mgcv, Available: <https://cran.r-project.org/web/packages/mgcv/mgcv.pdf> (5 MAR 2018).

Data objects include:

ncells = number of unique approximately 6 km by 6 km cells
ncellyrrep = number of unique cell-year-surveys (counts)
nyears = number of years
repyrs = specific years with a later survey replicate
z.repyrs = specific years without a later survey replicate
REP = survey ID (0 for early survey, 1 for later survey)
year = numeric year (1:25)
cell = numeric cell ID (1:ncells)
cellyr = unique cell-yr ID (length: ncells*nyears)
S = smoothing parameters from jagam() in mgcv R package
zero = matrix of zeros from jagam() in mgcv R package
p.surv = proportion of the cell surveyed per cell-year-survey (to weight uncertainty in estimates)
a.surv = area surveyed per cell-year-survey (to adjust counts to density per km²)
sost = normalized start of season day (mean = 0, SD = 1)
counts = observed counts per cell-year-survey

```
model{
# Covariate priors
  b.spring~dnorm(0,0.001)
  b.spring2~dnorm(0,0.001)
  for (y in 1:nyears){
    b0[y]~dnorm(0,tau.yr) # year intercepts
  }
  tau.yr<-pow(sd.yr,-2)
  sd.yr~dunif(0,5)
  mn.b0<-mean(b0[]) # average year effect for smoothed predictions
```

```

for (c in 1:ncells){ # cell effects
  mu.cell[c]~dnorm(0,tau.cell)
}
tau.cell<-pow(sd.cell,-2)
sd.cell~dunif(0,5)
mncell<-mean(mu.cell[]) # mean cell effect for smoothed predictions

# Spatio-temporal autocorrelation (from jagam() in mgcv (Woods 2018)
# S coefficients provided as data
blyr <- X %*% b ## linear predictor
for (i in 1:1) { b[i] ~ dnorm(0,0.05) }
  ## prior for s(long2,dcoast2)...
  K1 <- S1[1:s1[1],1:s1[1]] * lambda[1] + S1[1:s1[1],(s1[1]+1):s1[2]] * lambda[2] +
  S1[1:s1[1],(s1[2]+1):s1[3]] * lambda[3]
  b[2:kk[1]] ~ dnorm(zero[2:kk[1]],K1)

# prior for s(year2)...
for (i in (kk[1]+1):(kk[1]+5)) { b[i] ~ dnorm(0, lambda[4]) }
for (i in (kk[1]+6):(kk[1]+6)) { b[i] ~ dnorm(0, lambda[5]) }
# prior for ti(long2,dcoast2,year2)...
K3 <- S3[1:s3[1],1:s3[1]] * lambda[6] + S3[1:s3[1],(s3[1]+1):s3[2]] * lambda[7] +
S3[1:s3[1],(s3[2]+1):s3[3]] * lambda[8]
b[(kk[1]+7):kk[2]] ~ dnorm(zero[(kk[1]+7):kk[2]],K3)
# smoothing parameter priors CHECK...
for (i in 1:8) { #modified from jagam options – these values can be slow to converge
  lambda[i]<-pow(sd.lambda[i],-2)
  sd.lambda[i]~dunif(0,20)
}

# Rep effects - set early survey as zero or default
# Only had replicate surveys prior to 2007
for (y in c(repyrs)){
  repeff[y]~dnorm(0,tau.rep)
}
for (y in c(z.repyrs)){ #years after 2006
  repeff[y]<-0
}
tau.rep<-pow(sd.rep,-2)
sd.rep~dunif(0.00001,100)

# Overdispersion – r.mid is mean rate
r.mid~dnorm(0,0.01)
for (n in 1:ncellyrrep){
  r[n]<-exp(r.mid - p.surv[n]) # p.surv is proportion of cell surveyed, centered

```

```

counts[n] ~ dnegbin(p[n],r[n])
p[n]<-r[n]/(r[n]+E[n]) # Convert mean to prob
E[n]<-density[n]*s.area[n]
log(density[n])<-blyr[cellyr[n]] + b0[year[n]] + b.spring*sost[n]
+ b.spring2*pow(sost[n],2) +
mu.cell[cell[n]] + repeff[year[n]]*REP[n]

}

# Summaries and predictions
r.eff<-repeff[1:15]

for(c in 1:ncells){
  for (y in 1:nyears){

    bly[c,y]<-blyr[c+ncells*(y-1)] # convert to wide format
    dens [c,y]<-exp(bly[c,y]+b0[y]+mu.cell[c])
    dens2[c,y]<-exp(bly[c,y]+mn.b0+mu.cell[c] +0.5*sd.yr*sd.yr) # smoothed over annual var.,
    abund[c,y]<-dens[c,y]*cellarea[c]
    abund2[c,y]<-dens2[c,y]*cellarea[c]
  }
}

for (y in 1:nyears){
  N.sum[y]<-sum(abund[,y])#
  N.sum2[y]<-sum(abund2[,y])# # smoothed over annual var.
}
T.trend<-pow(N.sum[nyears]/N.sum[1],1/(nyears-1)) # geometric mean rate of change
T.trend2<-pow(N.sum2[nyears]/N.sum2[1],1/(nyears-1)) # smoothed trend (slope)

for (y in 1:(nyears-1)){
  g.growth[y]<-log(N.sum2[y+1]/N.sum2[y]) # Annual growth
}
g.trend<-mean(g.growth[])

for (c in 1:ncells){
  celldens[c]<-mean(dens2[c,])
  c.trend[c]<-pow(abund2[c,nyears]/abund2[c,1],1/(nyears-1))
  c.trend2[c]<-pow(abund[c,nyears]/abund[c,1],1/(nyears-1))
}

# Abundance
mdens<-mean(dens[1:ncells,1:nyears])
}

```