eBird Occupancy Model Testing

Laura Graham

July 21, 2016

1 Data preparation

Currently working with the data for hummingbirds from Colorado. The dataset has been generated in Python and I have filtered it such that only locations which have greater than 3 observations in at least one month are included. This reduces the number of locations from 7381 to 39.

The final data output from Python is a 4-dimensional array where the dimensions are: Year, Location, Species and Replicate.

```
load("data/hummingbirds_colorado.rda")

# size of array
dim(wide_dat)
```

```
## [1]  7 39  8 82
```

2 Initial models

2.1 Basic dynamic occupancy model (no covariates)

Firstly, we want to fit a model with no covariates and get some initial parameter estimates out.

```
# Code modified from Kery and Schaub 2012
# Bundle data
wide_dat[is.nan(wide_dat)] <- NA
dat <- list(y = wide_dat, nyear = dim(wide_dat)[1], nsite = dim(wide_dat)[2], nspecies = dim(wide_dat)[3

# Initial values
zst <- apply(wide_dat, c(1, 2, 3), max, na.rm=TRUE) # Observed occurrence as inits for z
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
## -Inf
```

```
zst[is.infinite(zst)] <- NA
inits <- function(){ list(z = zst)}
# Parameters monitored
params <- c("psi", "phi", "gamma", "p", "n.occ", "growthr", "turnover")

# MCMC settings
ni <- 2500
nt <- 4
nb <- 500
nc <- 3

# Call JAGS from R
strt <- Sys.time()

source("code/JAGSParallel.R")
out <- JAGSParallel(n.cores = 3, data = dat, inits = inits, params = params, model.file = "code/dynocc..
```

```
## R Version:  R version 3.3.0 (2016-05-03)

## snowfall 1.84-6.1 initialized (using snow 0.4-1): parallel execution on 3 CPUs.

## Library R2jags loaded.

## Library R2jags loaded in cluster.
```

```
## Library snowfall loaded.

## Library snowfall loaded in cluster.

##
## Stopping cluster
```

```r
save(out, file="basic_model.Rda")

print(Sys.time() - strt)
```

```
## Time difference of 1.675253 mins
```

```r
print(out$JAGSoutput, digits=3)
```

```
## Inference for Bugs model at "code/dynocc.JAGS.R", fit using jags,
##  3 chains, each with 2500 iterations (first 500 discarded), n.thin = 4
##  n.sims = 1500 iterations saved
##             mean    sd  2.5%   25%   50%   75% 97.5%  Rhat n.eff
## gamma[1,1] 0.232 0.073 0.102 0.180 0.226 0.279 0.395 1.002   870
## gamma[2,1] 0.065 0.043 0.010 0.033 0.056 0.088 0.175 1.003   720
## gamma[3,1] 0.061 0.042 0.007 0.030 0.051 0.080 0.165 1.000  1500
## gamma[4,1] 0.146 0.062 0.048 0.102 0.137 0.184 0.277 1.000  1500
## gamma[5,1] 0.090 0.048 0.020 0.054 0.082 0.117 0.206 1.002  1000
## gamma[6,1] 0.114 0.052 0.034 0.076 0.108 0.145 0.234 1.002   840
## gamma[1,2] 0.073 0.044 0.012 0.042 0.064 0.101 0.174 1.001  1500
## gamma[2,2] 0.052 0.034 0.008 0.026 0.046 0.072 0.135 1.001  1500
## gamma[3,2] 0.096 0.178 0.001 0.014 0.041 0.090 0.843 1.011   250
## gamma[4,2] 0.136 0.241 0.002 0.015 0.038 0.100 0.940 1.100    30
## gamma[5,2] 0.080 0.098 0.005 0.031 0.057 0.090 0.391 1.040   100
## gamma[6,2] 0.086 0.153 0.001 0.012 0.032 0.082 0.628 1.028    98
## gamma[1,3] 0.251 0.084 0.105 0.190 0.246 0.304 0.428 1.002   850
## gamma[2,3] 0.231 0.090 0.078 0.167 0.224 0.290 0.426 1.003   720
## gamma[3,3] 0.312 0.098 0.142 0.243 0.306 0.378 0.516 1.000  1500
## gamma[4,3] 0.218 0.088 0.074 0.152 0.208 0.273 0.419 1.001  1500
## gamma[5,3] 0.268 0.099 0.094 0.199 0.263 0.332 0.475 1.000  1500
## gamma[6,3] 0.171 0.087 0.032 0.107 0.161 0.226 0.358 1.000  1500
## gamma[1,4] 0.506 0.196 0.153 0.361 0.506 0.644 0.879 1.002  1500
## gamma[2,4] 0.709 0.184 0.299 0.586 0.732 0.859 0.980 1.001  1500
## gamma[3,4] 0.299 0.227 0.013 0.110 0.245 0.449 0.838 1.006   330
## gamma[4,4] 0.591 0.209 0.175 0.441 0.603 0.758 0.935 1.001  1500
## gamma[5,4] 0.666 0.235 0.149 0.504 0.698 0.866 0.986 1.002  1400
## gamma[6,4] 0.503 0.292 0.019 0.247 0.505 0.761 0.974 1.001  1300
## gamma[1,5] 0.357 0.084 0.203 0.295 0.354 0.413 0.533 1.000  1500
## gamma[2,5] 0.125 0.066 0.026 0.075 0.115 0.163 0.274 1.000  1500
## gamma[3,5] 0.128 0.055 0.044 0.087 0.122 0.162 0.253 1.001  1500
## gamma[4,5] 0.090 0.047 0.020 0.055 0.083 0.117 0.205 1.001  1500
## gamma[5,5] 0.153 0.057 0.058 0.114 0.147 0.188 0.284 1.001  1500
## gamma[6,5] 0.055 0.036 0.007 0.028 0.047 0.075 0.145 1.004   620
## gamma[1,6] 0.299 0.209 0.021 0.143 0.258 0.401 0.862 1.009   220
## gamma[2,6] 0.224 0.084 0.080 0.161 0.221 0.278 0.402 1.001  1500
## gamma[3,6] 0.187 0.077 0.061 0.133 0.180 0.230 0.361 1.000  1500
```

```
## gamma[4,6] 0.409 0.110 0.206 0.334 0.404 0.482 0.634 1.007   360
## gamma[5,6] 0.174 0.097 0.018 0.102 0.164 0.233 0.387 1.002  1200
## gamma[6,6] 0.311 0.113 0.123 0.225 0.304 0.386 0.548 1.001  1500
## gamma[1,7] 0.161 0.092 0.026 0.094 0.145 0.210 0.387 1.003   790
## gamma[2,7] 0.094 0.054 0.014 0.054 0.086 0.127 0.214 1.002  1500
## gamma[3,7] 0.183 0.091 0.051 0.116 0.170 0.232 0.400 1.005   480
## gamma[4,7] 0.077 0.061 0.002 0.032 0.064 0.109 0.236 1.002  1500
## gamma[5,7] 0.198 0.108 0.045 0.121 0.181 0.256 0.455 1.006   320
## gamma[6,7] 0.145 0.113 0.008 0.061 0.119 0.199 0.436 1.006   360
## gamma[1,8] 0.049 0.038 0.004 0.022 0.042 0.068 0.142 1.003   630
## gamma[2,8] 0.137 0.280 0.001 0.011 0.026 0.062 0.983 1.252    14
## gamma[3,8] 0.056 0.099 0.001 0.010 0.023 0.055 0.385 1.071    41
## gamma[4,8] 0.031 0.038 0.001 0.008 0.020 0.041 0.124 1.008   260
## gamma[5,8] 0.032 0.035 0.001 0.009 0.020 0.041 0.128 1.001  1500
## gamma[6,8] 0.062 0.097 0.001 0.010 0.025 0.064 0.387 1.008   440
## p[1,1]     0.915 0.057 0.768 0.883 0.927 0.957 0.990 1.001  1500
## p[2,1]     0.910 0.033 0.835 0.891 0.914 0.935 0.961 1.001  1500
## p[3,1]     0.978 0.021 0.922 0.969 0.984 0.993 1.000 1.009   320
## p[4,1]     0.772 0.057 0.654 0.736 0.774 0.812 0.873 1.002  1300
## p[5,1]     0.978 0.023 0.914 0.969 0.985 0.994 0.999 1.008   770
## p[6,1]     0.978 0.022 0.918 0.969 0.984 0.993 0.999 1.002  1500
## p[7,1]     0.796 0.047 0.702 0.765 0.797 0.828 0.883 1.000  1500
## p[1,2]     0.891 0.097 0.655 0.841 0.919 0.966 0.997 1.005   890
## p[2,2]     0.965 0.032 0.880 0.951 0.974 0.989 0.999 1.002  1200
## p[3,2]     0.959 0.039 0.858 0.943 0.970 0.987 0.999 1.006  1500
## p[4,2]     0.087 0.073 0.004 0.034 0.069 0.124 0.279 1.023   160
## p[5,2]     0.246 0.298 0.001 0.017 0.094 0.409 0.953 1.174    18
## p[6,2]     0.943 0.053 0.800 0.920 0.957 0.983 0.999 1.000  1500
## p[7,2]     0.284 0.310 0.001 0.021 0.140 0.524 0.954 1.028   140
## p[1,3]     0.858 0.038 0.774 0.834 0.862 0.885 0.922 1.001  1500
## p[2,3]     0.470 0.038 0.400 0.444 0.469 0.495 0.548 1.000  1500
## p[3,3]     0.769 0.029 0.712 0.749 0.770 0.788 0.824 1.003   630
## p[4,3]     0.431 0.035 0.365 0.406 0.430 0.454 0.504 1.003   620
## p[5,3]     0.298 0.029 0.243 0.279 0.298 0.317 0.358 1.002   880
## p[6,3]     0.302 0.026 0.251 0.283 0.302 0.320 0.353 1.003   630
## p[7,3]     0.435 0.027 0.384 0.417 0.435 0.453 0.486 1.000  1500
## p[1,4]     0.784 0.026 0.731 0.766 0.784 0.802 0.835 1.000  1500
## p[2,4]     0.883 0.018 0.843 0.872 0.884 0.895 0.916 1.001  1500
## p[3,4]     0.934 0.013 0.906 0.926 0.935 0.943 0.957 1.004   530
## p[4,4]     0.868 0.018 0.831 0.857 0.869 0.881 0.901 1.003   820
## p[5,4]     0.915 0.013 0.887 0.906 0.916 0.925 0.939 1.006   340
## p[6,4]     0.917 0.012 0.892 0.910 0.918 0.925 0.939 1.001  1500
## p[7,4]     0.913 0.012 0.889 0.905 0.914 0.922 0.936 1.002  1100
## p[1,5]     0.909 0.060 0.760 0.877 0.921 0.954 0.988 1.001  1500
## p[2,5]     0.661 0.040 0.582 0.635 0.661 0.689 0.735 1.000  1500
## p[3,5]     0.976 0.022 0.921 0.967 0.983 0.992 0.999 1.002  1500
## p[4,5]     0.968 0.030 0.889 0.955 0.978 0.991 0.999 1.000  1500
## p[5,5]     0.472 0.092 0.297 0.407 0.472 0.536 0.651 1.002   990
## p[6,5]     0.710 0.064 0.578 0.668 0.712 0.752 0.825 1.003  1500
## p[7,5]     0.932 0.061 0.778 0.902 0.950 0.980 0.998 1.004  1400
## p[1,6]     0.112 0.025 0.069 0.094 0.110 0.128 0.167 1.002   950
## p[2,6]     0.324 0.038 0.247 0.300 0.324 0.349 0.397 1.001  1500
## p[3,6]     0.561 0.036 0.494 0.535 0.561 0.585 0.629 1.000  1500
## p[4,6]     0.411 0.043 0.329 0.382 0.411 0.441 0.499 1.001  1500
```

```
## p[5,6]      0.168 0.027 0.118 0.149 0.167 0.185 0.223 1.002  1500
## p[6,6]      0.422 0.031 0.361 0.402 0.422 0.443 0.484 1.002  1200
## p[7,6]      0.142 0.025 0.097 0.125 0.140 0.158 0.194 1.001  1500
## p[1,7]      0.105 0.052 0.025 0.067 0.098 0.135 0.228 1.005   440
## p[2,7]      0.210 0.044 0.129 0.179 0.207 0.237 0.304 1.001  1400
## p[3,7]      0.665 0.047 0.573 0.633 0.666 0.697 0.754 1.002   920
## p[4,7]      0.094 0.030 0.044 0.072 0.091 0.112 0.159 1.001  1500
## p[5,7]      0.203 0.042 0.128 0.173 0.201 0.229 0.294 1.003  1000
## p[6,7]      0.077 0.024 0.036 0.060 0.076 0.092 0.126 1.008   250
## p[7,7]      0.057 0.023 0.022 0.040 0.054 0.071 0.105 1.003   830
## p[1,8]      0.386 0.302 0.004 0.108 0.330 0.636 0.964 1.028   140
## p[2,8]      0.522 0.128 0.282 0.433 0.522 0.608 0.769 1.005   690
## p[3,8]      0.374 0.324 0.001 0.055 0.303 0.659 0.965 1.306    13
## p[4,8]      0.433 0.306 0.005 0.149 0.401 0.693 0.971 1.007  1500
## p[5,8]      0.443 0.305 0.009 0.161 0.411 0.711 0.979 1.006   890
## p[6,8]      0.446 0.305 0.006 0.164 0.436 0.698 0.972 1.007   600
## p[7,8]      0.357 0.312 0.001 0.050 0.298 0.621 0.955 1.025   130
## phi[1,1]    0.448 0.157 0.164 0.331 0.444 0.564 0.745 1.005  1500
## phi[2,1]    0.523 0.142 0.257 0.420 0.521 0.622 0.792 1.002   910
## phi[3,1]    0.729 0.151 0.395 0.633 0.747 0.847 0.961 1.002  1400
## phi[4,1]    0.617 0.161 0.285 0.506 0.630 0.734 0.898 1.002   990
## phi[5,1]    0.493 0.148 0.211 0.391 0.494 0.593 0.788 1.002  1500
## phi[6,1]    0.624 0.159 0.298 0.512 0.639 0.738 0.904 1.001  1500
## phi[1,2]    0.227 0.176 0.007 0.083 0.195 0.321 0.655 1.001  1500
## phi[2,2]    0.751 0.191 0.299 0.636 0.795 0.908 0.990 1.001  1500
## phi[3,2]    0.637 0.241 0.143 0.457 0.670 0.840 0.984 1.002  1100
## phi[4,2]    0.361 0.271 0.009 0.124 0.306 0.557 0.933 1.023   140
## phi[5,2]    0.429 0.296 0.010 0.162 0.403 0.670 0.969 1.100    34
## phi[6,2]    0.374 0.278 0.009 0.136 0.314 0.587 0.947 1.000  1500
## phi[1,3]    0.909 0.081 0.701 0.869 0.931 0.971 0.998 1.002   980
## phi[2,3]    0.747 0.104 0.522 0.681 0.756 0.826 0.914 1.003  1500
## phi[3,3]    0.528 0.118 0.303 0.444 0.529 0.611 0.755 1.001  1500
## phi[4,3]    0.808 0.108 0.565 0.738 0.820 0.892 0.973 1.006   850
## phi[5,3]    0.803 0.097 0.586 0.741 0.812 0.875 0.957 1.007   790
## phi[6,3]    0.807 0.086 0.623 0.750 0.818 0.872 0.941 1.001  1500
## phi[1,4]    0.927 0.045 0.817 0.901 0.936 0.961 0.990 1.002  1200
## phi[2,4]    0.951 0.035 0.869 0.931 0.960 0.978 0.997 1.003  1500
## phi[3,4]    0.941 0.042 0.836 0.917 0.950 0.973 0.996 1.002  1500
## phi[4,4]    0.973 0.026 0.900 0.961 0.980 0.991 0.999 1.002  1100
## phi[5,4]    0.976 0.023 0.910 0.966 0.982 0.993 0.999 1.001  1500
## phi[6,4]    0.976 0.024 0.912 0.966 0.983 0.993 0.999 1.004   940
## phi[1,5]    0.568 0.160 0.250 0.454 0.576 0.686 0.853 1.000  1500
## phi[2,5]    0.176 0.092 0.042 0.109 0.161 0.226 0.392 1.000  1500
## phi[3,5]    0.367 0.188 0.065 0.220 0.345 0.494 0.768 1.001  1500
## phi[4,5]    0.153 0.132 0.004 0.052 0.116 0.218 0.490 1.001  1500
## phi[5,5]    0.251 0.198 0.007 0.091 0.205 0.374 0.716 1.005   860
## phi[6,5]    0.145 0.124 0.004 0.048 0.108 0.213 0.457 1.007   510
## phi[1,6]    0.392 0.152 0.137 0.281 0.376 0.494 0.728 1.006   380
## phi[2,6]    0.584 0.144 0.297 0.481 0.593 0.686 0.851 1.001  1500
## phi[3,6]    0.374 0.127 0.149 0.279 0.365 0.466 0.631 1.000  1500
## phi[4,6]    0.658 0.165 0.312 0.550 0.669 0.792 0.928 1.000  1500
## phi[5,6]    0.658 0.118 0.417 0.579 0.663 0.743 0.862 1.000  1500
## phi[6,6]    0.621 0.133 0.355 0.532 0.624 0.717 0.870 1.000  1500
## phi[1,7]    0.512 0.258 0.092 0.292 0.495 0.719 0.974 1.011   340
```

```
## phi[2,7]   0.253 0.146 0.040 0.139 0.231 0.343 0.587 1.001  1500
## phi[3,7]   0.485 0.220 0.095 0.319 0.477 0.644 0.905 1.003  1500
## phi[4,7]   0.603 0.201 0.197 0.463 0.615 0.758 0.943 1.002  1500
## phi[5,7]   0.575 0.225 0.147 0.419 0.581 0.752 0.963 1.001  1500
## phi[6,7]   0.613 0.224 0.184 0.449 0.615 0.797 0.974 1.003  1500
## phi[1,8]   0.460 0.289 0.011 0.213 0.459 0.694 0.969 1.009   460
## phi[2,8]   0.383 0.266 0.014 0.150 0.345 0.578 0.929 1.011   240
## phi[3,8]   0.400 0.301 0.006 0.125 0.348 0.653 0.963 1.185    20
## phi[4,8]   0.463 0.292 0.018 0.203 0.444 0.718 0.964 1.002  1100
## phi[5,8]   0.474 0.288 0.024 0.225 0.457 0.717 0.969 1.001  1500
## phi[6,8]   0.481 0.295 0.019 0.211 0.476 0.740 0.973 1.008   900
##
## For each parameter, n.eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor (at convergence, Rhat=1).
```