

Supplementary material

Appendix S1. OpenBUGS code for estimating adult annual survival probabilities, reproduction rates (number of independent young per female per year), and finite rates of increase for North Island robins in a pine plantation and native forest fragments.

Model

```
{
# REPRODUCTION

# Priors for reproduction parameters
a.f ~ dnorm(0, 1.0E-6)           # intercept for log(f), where f = juveniles/female
b.pine.f ~ dnorm(0, 1.0E-6)     # effect of pine vs native on log(f)
b.gr.f ~ dnorm(0, 1.0E-6)      # effect of grazing on log(f)
s.site.f ~ dunif(0,100)         # sd among sites in log(f)
tau.site.f <- pow(s.site.f,-2)  # convert to precision
s.fem ~ dunif(0,100)           # sd among females in log(f)
tau.fem <- pow(s.fem,-2)       # convert to precision
s.year.f ~ dunif(0,100)        # sd among years in log(f)
tau.year.f <- pow(s.year.f,-2) # convert to precision

# Random effects for reproduction parameters
for (i in 1:n.year.f) {
  re.year.f[i] ~ dnorm(0,tau.year.f) # assign random effect to each year
}
for (i in 1:n.site) {
  re.site.f[i] ~ dnorm(0,tau.site.f) # assign random effect to each site
}
for (i in 1:n.fem){
  re.fem[i] ~ dnorm(0,tau.fem)      # assign random effect to each female
}

# Sampling reproduction observations (no. independent young produced by each fem. each year)
for (i in 1:n.obs.f)              # for each reproduction observation...
{
  juvs[i] ~ dpois(mu.f[i])         # sample observed no. juvs from Poisson based on expected no.
  log(mu.f[i]) <- fixed.f[i]+random.f[i] # calculate expected number
  fixed.f[i] <- a.f+b.pine.f*pine[site.f[i]]+b.gr.f*gr[site.f[i]]
  random.f[i] <-re.year.f[year.f[i]]+re.site.f[site.f[i]]+re.fem[fem[i]]
}

# ADULT SURVIVAL

# Adult survival and re-sighting priors
a.phi ~ dnorm(0, 0.001)          # intercept for logit(sa), where sa = adult surv prob over 4 mo.
b.pine.phi ~ dnorm(0, 0.001)     # effect of being in pine on on logit(sa)
b.pine.sex.phi ~ dnorm(0, 0.001) # interaction between pine and sex effect
b.sj.phi ~ dnorm(0, 0.001)      # effect of season (Sep-Jan) on logit(sa)
b.jm.phi ~ dnorm(0, 0.001)      # effect of season (Jan-Sep) on logit(sa)
b.sex.sm.phi ~ dnorm(0, 0.001)  # effect of sex on on logit(sa) over breed.intervals (Sep-May)
s.year.phi ~ dunif(0, 10)        # sd among years in logit(sa)
tau.year.phi <- pow(s.year.phi, -2) # convert to precision
s.site.phi ~ dunif(0, 100)       # sd among sites in logit(sa)
tau.site.phi <- pow(s.site.phi, -2) # convert to precision
a.p ~ dnorm(0, 0.001)           # mean logit (p), where p = re-sighting probability
s.t.p ~ dunif(0, 10)           # sd among surveys in logit(p)
tau.t.p <- pow(s.t.p, -2)       # convert to precision
```

```

# Random year effect for adult survival
for (j in 1:n.year.s) {
  re.year.phi[j] ~ dnorm(0, tau.year.phi) # assign random time effect on survival probability
}
# Random survey effect for re-sighting
for (j in 1:n.surveys) {
  re.t.p[j] ~ dnorm(0, tau.t.p) # assign random time effect on re-sighting probability
}
# Random fragment effects for adult survival
for (k in 1:n.site) {
  re.site.phi[k] ~ dnorm(0, tau.site.phi) # assign random site effect
}
# Sampling encounter history for adult survival
for (i in 1:n.ind) # for each individual (row)...
{
  alive[i, first[i]] <- 1 # 1 = alive the first time observed as settled adult
  for (j in first[i]+1:n.surveys) # for each subsequent survey....
  {
    # observation equation
    seen[i, j] ~ dbern(psight[i, j]) # whether individual was seen is sampled from Bernoulli
    psight[i, j] <- p[i,j]*alive[i, j] # prob of being seen = re-sighting prob x whether it was alive
    logit(p[i,j]) <- a.p+re.t.p[j] # re-sighting prob calculated for that survey

    # state equation
    alive[i, j] ~ dbern(palive[i, j]) # whether individual was alive is sampled from Bernoulli
    palive[i, j] <- phi.int[i, j]*alive[i, j-1] # prob alive = survival prob x whether alive last survey
    phi.int[i, j] <- pow(phi[i, j], int[j]-1) # adjust for interval length
    logit(phi[i, j]) <- fixed.phi[i, j]+random.phi[i, j] # calculate annual survival prob based
    fixed.phi[i, j] <- a.phi+b.sj.phi*sj[j-1]+b.jm.phi*jm[j-1]+b.sex.sm.phi*sex[i]*(1-ms[j]-1)+b.pine.phi*pine[site.s[i]]+b.pine.sex.phi*pine[site.s[i]]*sex[i]*(1-ms[j]-1)
    random.phi[i, j] <- re.year.phi[year[j-1]]+re.site.phi[site.s[i]]
  }
}

# JUVENILE SURVIVAL
logit.sjuv30 ~ dnorm(logit.sjuv30.mean, logit.sjuv30.tau)
logit.sjuv30.tau <- pow(logit.sjuv30.sd,-2)
logit(sjuv30) <- cut(logit.sjuv30)

# CALCULATING DERIVED PARAMETERS

log(mu.fenced) <- a.f # mean reproduction rate in fenced native fragments
log(mu.grazed) <- a.f+b.gr.f # mean reproduction rate in grazed native fragments
log(mu.pine) <- a.f+b.pine.f # mean reproduction rate in pine plantation

logit(s.fem.sj.native) <- a.phi+b.sj.phi # prob female survives Sep-Jan native
logit(s.fem.jm.native) <- a.phi+b.jm.phi # prob female survives Jan-May native
logit(s.ms.native) <- a.phi # prob female survives May-Sep native
s.fem.annual.native <- s.fem.sj.native*s.fem.jm.native*s.ms.natavec

logit(s.fem.sj.pine) <- a.phi+b.pine.phi+b.sj.phi # prob female survives Sep-Jan pines
logit(s.fem.jm.pine) <- a.phi+b.pine.phi+b.jm.phi # prob female survives Jan-May pines
logit(s.ms.pine) <- a.phi+b.pine.phi # prob female survives May-Sep pines
s.fem.annual.pine <- s.fem.sj.pine*s.fem.jm.pine*s.ms.pine # mean annual female survival pines

# Prob of juvenile surviving to adulthood
s.juv.native <- sjuv30*pow(s.fem.jm.native,3/4)*s.ms.native # juvenile survival native
s.juv.pine <- sjuv30*pow(s.fem.jm.pine,3/4)*s.ms.pine # juvenile survival pines
# Note: assumes 8 mo. from independence to adulthood, with first 30d at juvenile survival rate

# Finite rate of increase (lambda)
lambda.fenced <- s.fem.annual.native+0.5*mu.fenced*s.juv.native
lambda.grazed <- s.fem.annual.native+0.5*mu.grazed*s.juv.native
lambda.pine <- s.fem.annual.pine+0.5*mu.pine*s.juv.pine

```

```

for (i in 1:n.site) { # for each site calculate...

  # Site-specific estimates of reproduction rate (Fig. 2)
  log(mu[i]) <- a.f+b.pine.f*pine[i]+b.gr.f*gr[i]+re.site.f[i] # mean reproduction rate

  # Site-specific estimates of annual adult female survival probability (Fig. 2)
  logit(s.fem.sj[i]) <- a.phi+b.pine.phi*pine[i]+re.site.phi[i]+b.sj.phi # prob female survives Sep-Jan
  logit(s.fem.jm[i]) <- a.phi+b.pine.phi*pine[i]+re.site.phi[i]+b.jm.phi # prob fem. survives Jan-May
  logit(s.ms[i]) <- a.phi+b.pine.phi*pine[i]+re.site.phi[i] # prob female survives May-Sep
  s.fem.annual[i] <- s.fem.sj[i]*s.fem.jm[i]*s.ms[i] # prob female survives the whole year

  # Prob of juvenile surviving to adulthood
  s.juv[i] <- sjuv30*pow(s.fem.jm[i],3/4)*s.ms[i]
  # Note: assumes 8 mo. from independence to adulthood, with first 30d at juvenile survival rate

  # Site-specific estimates of finite rate of increase (Fig. 2)
  lambda[i] <- s.fem.annual[i]+0.5*mu[i]*s.juv[i]

}
}

```

Appendix S2. OpenBUGS code for estimating nest survival probabilities for North Island robins in a pine plantation and native forest fragments.

Model

```
{
  # Priors
  a ~ dnorm(3, 2)           # intercept for logit(DSR), where DSR = daily survival probability
  b.pine ~ dnorm(0, 1)     # effect of pine vs native on logit(DSR)
  b.gr ~ dnorm(0, 1)      # effect of grazing on logit(DSR)
  s.site ~ dunif(0, 1)    # sd among sites
  tau.site <- pow(s.site, -2) # convert to precision
  s.fem ~ dunif(0, 1)    # sd among females
  tau.fem <- pow(s.fem, -2) # convert to precision
  s.year ~ dunif(0, 1)   # sd among years
  tau.year <- pow(s.year, -2) # convert to precision

  # Random effects
  for (i in 1:n.year) {
    re.year[i] ~ dnorm(0, tau.year) # assign random effect to each year
  }
  for (i in 1:n.site) {
    re.site[i] ~ dnorm(0, tau.site) # assign random effect to each site
  }
  for (i in 1:n.fem){
    re.fem[i] ~ dnorm(0, tau.fem) # assign random effect to each female
  }

  # Sample nest success data
  for (i in 1:n.nests) { # for each nest...
    p.succ[i] <- pow(p.day[i], last.alive[i]-first[i]) # calculate pr. surviving interval observed active
    succ[i] ~ dbern(p.succ[i]) # model success
    p.fail[i] <- 1-pow(p.day[i], last.check[i]-last.alive[i]) # calculate pr. failure over interval when failed
    fail[i] ~ dbern(p.fail[i]) # model failure
    logit(p.day[i]) <- fixed[i]+random[i] # calculate daily nest survival probability
    fixed[i] <- a+b.pine*pine[site[i]]+b.gr*gr[site[i]]
    random[i] <- re.year[year[i]]+re.site[site[i]]+re.fem[fem[i]]
  }

  # CALCULATING DERIVED PARAMETERS

  # Average nest survival rates for fenced, grazed, and pines (quoted in text)
  logit(p.day.site.fenced) <- a
  logit(p.day.site.grazed) <- a+b.gr
  logit(p.day.site.pine) <- a+b.pine
  p.40.site.fenced <- pow(p.day.site.fenced, 40)
  p.40.site.grazed <- pow(p.day.site.grazed, 40)
  p.40.site.pine <- pow(p.day.site.pine, 40)

  for (i in 1:n.site) { # for each site calculate...

    # Fragment-specific nest survival rates shown in Fig. 3
    logit(p.day.site[i]) <- a+b.pine*pine[i]+b.gr*gr[i]+re.site[i]
    p.40.site[i] <- pow(p.day.site[i], 40)
  }
}
```

Appendix S3. OpenBUGS code for estimating rat tracking rates (probability that at least one rat passes through a baited tracking tunnel in 24 h) in a pine plantation and native forest fragments.

Model

```
{
  # Priors
  a ~ dnorm(0, 1.0E-6)           # intercept logit(p), where p = prob of tunnel being tracked
  b.pine ~ dnorm(0, 1.0E-6)      # effect of pine vs native on logit(p)
  b.gr ~ dnorm(0, 1.0E-6)       # effect of grazing on logit(p)
  s.grid ~ dunif(0,100)         # sd among grids in logit(p)
  tau.grid <- pow(s.grid,-2)     # convert to precision
  s.tun ~ dunif(0,100)         # sd among individual tracking tunnel sites
  tau.tun <- pow(s.tun,-2)     # convert to precision
  s.year ~ dunif(0,100)        # sd among years
  tau.year <- pow(s.year,-2)   # convert to precision
  s.time ~ dunif(0,100)       # sd among sampling occasions
  tau.time <- pow(s.time,-2)  # convert to precision

  # Assigning random effects
  for (i in 1:n.grid) {
    re.grid[i] ~ dnorm(0,tau.grid) # grid effect on logit(p)
  }
  for (i in 1:n.tun) {
    re.tun[i] ~ dnorm(0,tau.tun)  # individual tunnel effect
  }
  for (i in 1:n.time) {
    re.time[i] ~ dnorm(0,tau.time) # time effect (sampling occasion)
  }
  for (i in 1:n.year) {
    re.year[i] ~ dnorm(0,tau.year) # year effect
  }

  # Sampling tracking data (either rat tracks or mouse tracks)
  for (i in 1:n.obs) { # for each individual tracking tunnel on each sampling occasion...
    x[i] ~ dbern(p[i]) # whether tunnel was tracked is sampled from Bernoulli distribution
    logit(p[i]) <- fixed[i]+random[i] # calculate probability of it being tracked
    fixed[i] <- a+b.pine*pine[grid[i]]+b.gr*gr[grid[i]]
    random[i] <- re.grid[grid[i]]+re.tun[tun[i]]+re.year[year[i]]+re.time[time[i]]
  }

  # Calculating derived parameters
  for (i in 1:n.grid) { # for each grid, calculate...
    logit(p.grid[i]) <- a+b.pine*pine[i]+b.gr*gr[i]+re.grid[i] # mean tracking probability
  }
}
```


