Against all odds, a *Gnu*, bright future for Wildebeest: A Bayesian state-space model suggests

Misha Tseitlin || Muntasir Akash

21 November 2023

Assignment 2: State-space model for wildebeest population dynamics

University of
St Andrews

School of Mathematics and Statistics

in partial fulfilment of the requirements for
MT5767: Modelling Wildlife Population Dynamics

Contents

Introduction

The wildebeest (*Connochaetes*) live and move in large herds (Estes 2014). The blue wildebeest (*Connochaetes taurinus*) migrate in a clockwise manner from Serengeti, Tanzania to Masai Mara, Kenya, forming the backbone of the Great Migration in south-eastern Africa (Estes 2014). During this annual crossing of great distances, the blue wildebeest follow annual rain cycles and move along its trajectory. The species also breeds annually, dependent on rainfall (Estes 2014). Due to expansion of agriculture, Rinderpest induced by the spread of cattle-farming, and poaching pressure, this antelope species's abundance steeply declined in the 1960s and 70s. Since then, population monitoring has been a key focus of biologists and conservationists.

This report used 1960–1989 measures of wildebeest populations (sourced from Sutherland 2023) to construct a Bayesian state-space model, analyse species population size and growth including years lacking survey data, and project population trajectories five years into the future.

Methodology

Ecological modelling A state-space model (SSM) consists of two components: an unobserved "true" population process, N_t, and the observation process, y_t, that often entails observation process errors and uncertainties. Here, true wildebeest population N_t depends on the population growth rate, λ_t, that has been accounted for yearly removals from poaching, c_t. Variables can either be deterministic (i.e., $=$) or stochastic (i.e., $\sim$); deterministic values can be perfectly calculated from the right hand of the equation with no variation while stochastic values include natural, statistical variation often represented by known distributions (e.g., normal, poisson, binomial). To account for random variation common in natural systems, large N_{t+1}s take a normal distribution consistent with the millions of wildebeest observed more suitable to a continuous distribution.

$$N_{t+1} \sim N(\lambda_t * (N_t - c_t), sigma_t^2)$$

λ_t, i.e., $\frac{N_{t+1}}{N_t}$, depends on rainfall R_t as suggested by Estes (2014). To test this relation, β_0 and β_1 estimate $log(\lambda_t)$, a logarithmic transformation allowing for (i) only positive λ values, (ii) estimation of the per-capita growth rate $r_t = log_e(\lambda_t) = \frac{\Delta N}{N \Delta t}$, and (iii) more easy interpretation of the relationship with βs.

$$log(\lambda_t) = \beta_0 + \beta_1 * R_t$$

Finally, population measures y_t are treated as normally distributed and unbiased with an associated known spread of values se_t. Thus, they correspond to a simple stochastic normal distribution linked to the underlying true state N_t.

$$y_t \sim N(N_t, se_t^2)$$

These three equations are the likelihoods estimated by Bayesian methods.

Bayesian analysis Implemented in RStudio Environment (RStudio 2023) using a JAGS user interface package (Kellner 2021), Bayesian analysis updates pre-existing knowledge about parameters, the prior distribution $p(\theta)$, with estimates derived from observed data, a likelihood function $p(y|\theta)$. These generate parameter estimates as a posterior distribution, $p(\theta|y)$ (Schoot et al. 2021). In contrast to classical statistics generating estimates as single values, Bayesian analyses describe parameters as full distributions (Kéry and Schaub 2011). These all derive from Bayes' rule.

$$p(\theta|y) = \frac{p(y|\theta) * p(\theta)}{p(y)}$$

$p(y)$ does not dependent on any parameter θ and is treated as an omitted normalising factor (Schoot et al. 2021). Thus, the Bayesian approach simplies to deriving the posterior from existing priors and an estimated likelihood $p(\theta|y) \propto p(y|\theta) * p(\theta)$.

For all estimated parameters, Bayesian methods require specifying prior beliefs about the variables of interest (commonly as statistical distributions). $log(\lambda_t)$ deterministically (Kellner 2021) depended on β_0 and β_1, feeding into an N_{t+1} including variation $sigma_t$. β priors centred on 0 (suggesting no relationship) with a large, normal spread in values to minimally influence estimates. Thus, non-zero estimates of growth rates clearly derive from the data.

$$\beta_0, \beta_1 \sim N(0, 1000)$$

σ priors appear small but represent the millions scale (the same as observations y_t); they must be positive due to statistical principles. Thus, the true population may vary uniformly anywhere between 0 and 1 million wildebeest.

$$\sigma_t \sim U(0, 1)$$

Finally, initial population size N_1 also used a prior consistent with the observations uniformly between 0 and 0.7 (e.g., 700 000 wildebeest)—larger than any observed abundance before 1970.

$$N_1 \sim U(0, 0.7)$$

All variance terms (i.e., σ_t^2, and se_t^2) were input as precision, $\tau = \frac{1}{\sigma^2}$, to JAGS code.

Model estimation Bayesian analysis consisted of six steps: writing models in JAGS, packaging data, setting initial values, defining parameters of interest, simulating Markov Chain Monte Carlo (MCMC) values, and posterior predictive checking. Our priors of interest were β_0, β_1, and $sigma_t^2$, and our latent variables were growth rate and estimated population size (in millions). Markov chains are a iterative process where a value in $t + 1$ is only related to the value of its t step. Monte Carlo is a stochastic simulation procedure to determine integrals by simulating random numbers from a given distribution. In MCMC settings, we ran three different chains. To deal with the value autocorrelation, we dropped the every sixth value in the MCMC run. MCMC used 200 000 total iterations, half of which were discarded to "burn-in" the model. Posterior checks used trace plots to visually detect convergence supported with Bruce-Gelman-Rubin (BGR) $\hat{R}$ statistics. Given assumed unbiased observations with known observation error, posterior predictive checking was deemed unnecessary.

To deal with N/A values (i.e., years without survey), Zeileis and Grothendieck (2005) back-filled initial N_t values derived from y_t to initialise the model. However, only years with observations were used in likelihood settings to baseline against y_t in the observation process. Youngflesh (2018) was followed for MCMC checks and Wickham (2016) for graphs.

Results and Discussion

First, model convergence from BGR statistics of three main parameters illustrated high confidence in results (Table 1). Mean σ_t, estimated around 0.04, seems small but pertains to a typical spread of 40 000 wildebeest that can be equivalent to the maximum annual removal. As the 95% credible interval (CI) excluded 0, the model concluded a 95% chance of some non-zero variation in the true population, and data supported the inclusion of a stochastic term in the true population. In contrast, mean β_0 suggested 9.6% population growth in years with no rain that decreased by mean β_1 0.02% with each additional mm of rainfall R_t. However, the 95% CIs included zero indicating that rainfall may actually not influence population growth rates. The current model opted for density-independent growth lacking terms like carrying capacity K. This alternative approach may capture the true effect of rainfall more strongly.

Figure 1 visually confirms convergence and graphically represents the distributions of the three parameters. βs appeared broadly normal following from the specified priors, and both centred quite close to zero: chains explored possible values well including extreme options after 130 000 iterations. Notably, chain starting values were tightly constrained, and the model was quite sensitive to prior specification. The Jenkins prior, a type of diffuse prior, and other uninformative options may be good to test the suitability of other formulations for prior predictive checking. Additionally, priors around zero closely matched analogous classical inference strategies using null hypotheses. In contrast, σ_t

more consistently traversed possible values given its limitation to positive values and provided an apparently log-normal posterior distribution with a narrow left tail excluding zero; the diffuse uniform prior suggests strong confidence in these results. The selected 100 000 burn-in seemed excessive with relatively quick chain convergence. But, the presence of large jumps implied the converse: parameter variance may actually increase with longer chains. Thus, the selected formation was a reasonable middle ground between these two considerations.

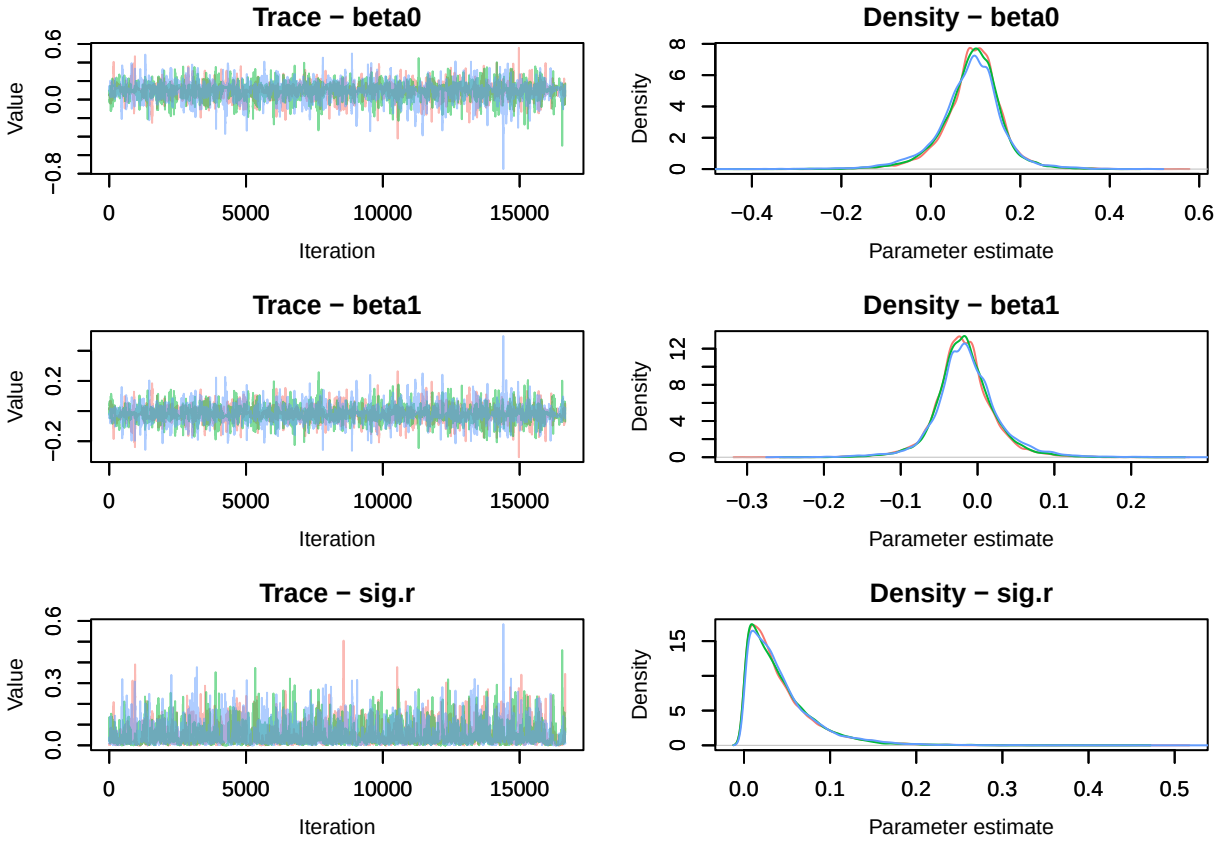

Figure 1: Primary parameter trace plots for β_0, β_1, and σ_t; derived latent parameters are not shown, but all converged

The initial population size, N_1, strongly depended on sensible starting values to initialise simulations. Thus, an explicit specification of biological realism is imperative. Such simulation outputs would benefit conservation of wildebeest populations. Autocorrelation remains a consistent issue with these data: parameters strongly correlate across iterations. The thinning approach undertaken could not solve the issue. Future potential solutions include more aggressive thinning or other hierarchical approaches (e.g., model reformulation).

Looking into future projections (1990-1994), the SSM matched well with observed data under the constraints of a exponential population growth (Figure 2). The true range of N_t did not perfectly match estimated y_t or its spread se_t. Mean estimated N_t fell within the 95% confidence interval of the observed data, as expected given imperfect biological surveys. The coherence of mean estimated N_t with observation error se_t confirmed SSM output values correctly. y_t and N_t matched well before 1972 but the former more freely approximated the latter in the final decade of surveying. Projections assumed illegal poaching remains at 1989 values and 1.49 mm dry season rainfall; under these conditions, wildebeest populations were projected to consistently exceed 1.3 million and approach 2 million individuals by 1994. Given missing observations, the 95% CI was quite wide and suggested a potential 2.5 million wildebeest in 1994.

Population growth rates were consistent to varying degrees (mean estimates 1.05–1.10) (Figure 3). Understandably, the 95% CI showed higher uncertainty during un-surveyed years compared to years surveyed. The CIs occasionally and only slightly crossed the threshold of decreased growth (< 1).

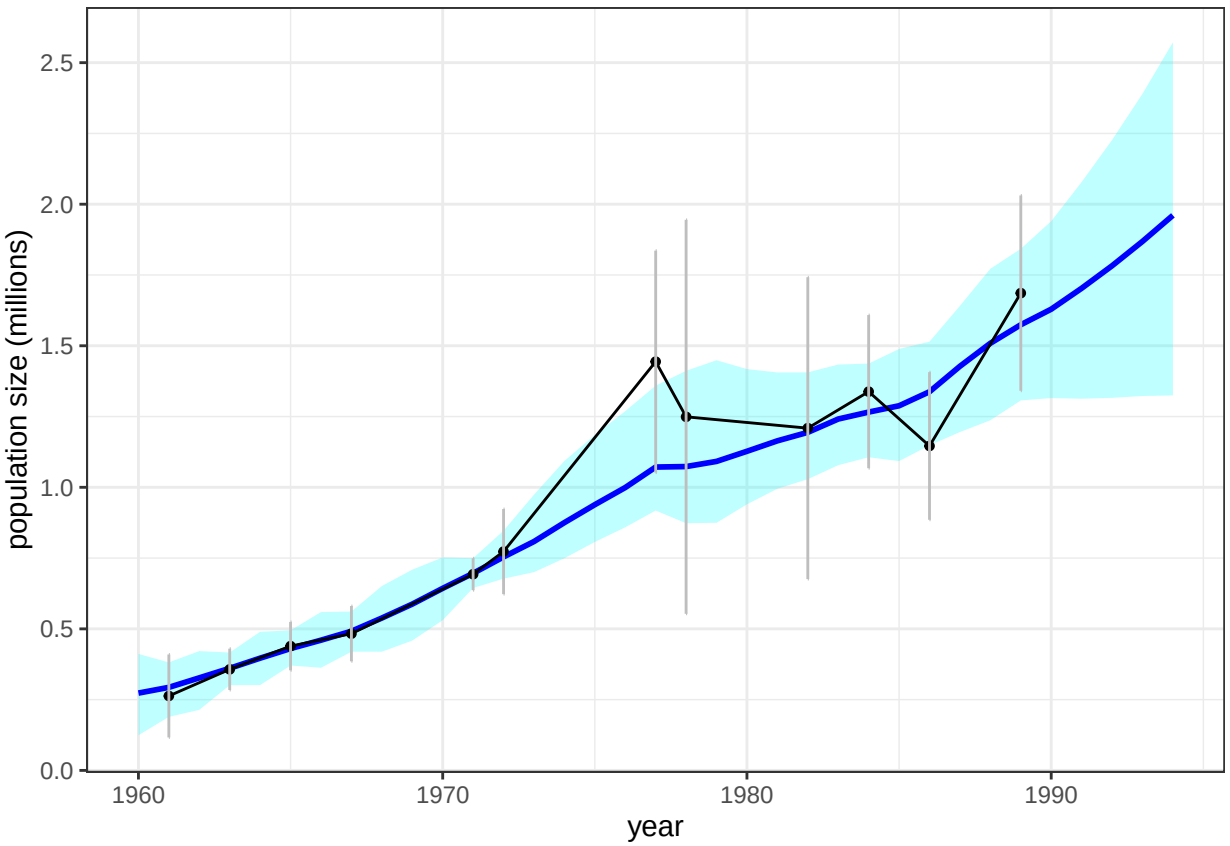

Figure 2: Projected values for wildebeest population. State process denoted by blue line with CIs (cyan). Observation process depicted with black line and grey error bars.

Table 1: SSM primary parameter summary for β_0, β_1, and σ_t

Parameter	Mean	Standard Deviation	2.5%	Median	97.5%	BGR Stat	ESS
beta0	0.0916493	0.0684968	-0.0620578	0.0963729	0.2195532	1.01	747
beta1	-0.0155740	0.0401444	-0.0941256	-0.0175536	0.0715563	1.01	777
sig.r	0.0422150	0.0392684	0.0016340	0.0313277	0.1460716	1.00	4044

Estimates were constant for projected years due to no reference survey data: Wildebeest populations may grow by 7.1% [3.2%, 9.7%] starting in 1990. In short, the high λs suggest little threat of population extinction over the survey period or during short-run projections and confirm that wildebeest will remain stable.

The SSM imperfectly supports other empirical results like a rainfall relationship and suggests that exponential growth may be a sub-optimal assumption. Even so, SSMs can approximate observed populations well and serve a strong tool in conservation ecology.

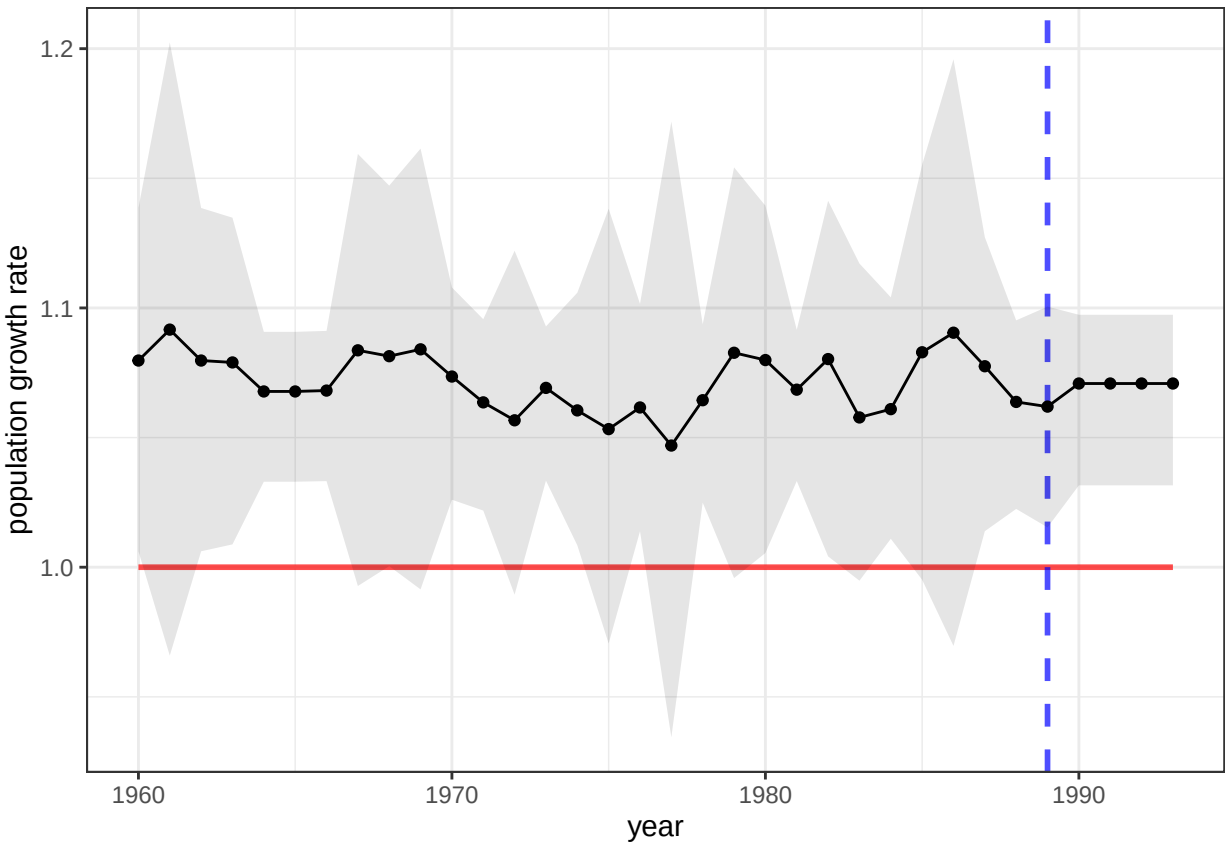

Figure 3: Estimated wildebeest population growth rates with 95% credible interval compared against the no-change line of $\lambda = 1$

Authors' Contributions

To master understanding Bayesian analysis and writing in JAGS environment, MT and MA wrote and completed running their own separate models. MT coded four models, and MA coded one model. Authors agreed upon one and built their report around it. For report writing, the sections were distributed equally (MA = introduction and methodology; MT = result and discussion) in the first draft compilation. Following these steps and beyond, collaborative discussion and editing was done at each step. So much so, to select the writing media for the report (i.e., Rmarkdown or Word doc), an unbiased coin mediated decisions. Both authors reviewed the materials and agreed to the submitted version. For more detail on individual contributions, please reference GitHub.

Code Supplement

```
# N0 ~ uniform(0, U)

# Nt | Nt-1 ~ normal [lambda*(Nt-1 - ct-1), sigmaN]

# yt | Nt ~ normal (Nt, sigmaY)

# writing the model in BUGS

# model specification

cat("
model{
  # priors
  # this prior is quite irrelevant to the final data spread: tested between 0.5 and 3
  n1 ~ dunif(0,0.7)   # 0.7 is the most suitable option theoretorically
  N.est[1] <- n1
  beta0 ~ dnorm(0,0.001)
  beta1 ~ dnorm(0,0.001)
  sig.r ~ dunif(0, 1)
  sig2.r <- pow(sig.r, 2)
  tau.r <- pow(sig.r, -2)

  # likelihood - state process
  for(t in 1:(nyrs-1)){
    log.lambda[t] <- beta0 + beta1*R[t]
    log(lambda[t]) <- log.lambda[t]
    N.est[t+1] ~ dnorm(lambda[t]*(N.est[t] - c[t]), tau.r)
  }

  # likelihood - observation process
  for (t in validYrs) {
    y[t] ~ dnorm(N.est[t], obs.tau[t])
  }

}
",fill = TRUE, file='wildessmBasic1.txt')

# JAGS package data

wildedata <- list(y = wildebeest$Nhat,

                  nyrs = nrow(wildebeest),

                  validYrs = validObs,

                  R = wildebeest$rain,

                  obs.tau = wildebeest$sehat^-2,

                  c = wildebeest$Catch)
```

```r
# set initial values for the unknown parameters

wildeinits <- function(){

  list(beta0 = rnorm(1),

       beta1 = rnorm(1),

       sig.r = runif(1),

       N = wildebeestImpute$Nhat)
}

# parameters monitoring '

wildeparms <- c("beta0", "beta1", "sig.r", "lambda", "N.est")

# MCMC settings

nt <- 6 # thinning rate to reduce autocorrelation

nc <- 3 # number of chains

ni <- 200000 # number of iteration

nb <- 100000 # number of burn-ins / warm-ups

#conduct the MCMC analysis
wildeout1 <- jags(data = wildedata,
                  inits = wildeinits,
                  parameters.to.save = wildeparms,
                  model.file = "wildessmBasic1.txt",
                  n.chains = nc,
                  n.iter = ni,
                  n.burnin = nb,
                  n.thin = nt)

#pull out relevant data from JAGS output
wilde_traj1 <- data.frame(Year = wildebeest$year,
                          Mean = wildeout1$mean$N.est,
                          Lower = wildeout1$q2.5$N.est,
                          Upper = wildeout1$q97.5$N.est,
                          Obs = wildebeest$Nhat,
                          LowerObs = wildebeest$lci,
                          UpperObs = wildebeest$uci)

#plot estimated population trajectories against observed data
ggplot(data = wilde_traj1) +
  geom_ribbon(aes(x=Year, y=Mean, ymin=Lower, ymax=Upper),
              fill="cyan", alpha = 0.25) +
  geom_line(aes(x=Year, y=Mean), linewidth=1, color="blue") +
  geom_point(aes(x=Year, y=Obs), size=1.2) +
  geom_line(data = na.omit(wilde_traj1), aes(x=Year, y=Obs)) +
```

```
  geom_errorbar(aes(x=Year,
                    y=Obs,
                    ymin=LowerObs,
                    ymax=UpperObs), width=0, color="grey") +
  theme_bw()

#project values forward 5 years from 1990-1994
nproj <- 5

#assume that illegal harvesting continues at current levels
#use average observed rainfall for future projections
#impute last year values for Nhat and sehat; they aren't referenced though
wildedata_proj1 <- list(y = c(wildebeest$Nhat,
                              rep(wildebeest$Nhat[nrow(wildebeest)], nproj)),
                        nyrs = nrow(wildebeest) + nproj,
                        validYrs = validObs,
                        R = c(wildebeest$rain,
                              rep(mean(wildebeest$rain), nproj)),
                        obs.tau = c(wildebeest$sehat^-2,
                                    rep(wildebeest$sehat[nrow(wildebeest)]^-2, nproj)),
                        c = c(wildebeest$Catch,
                              rep(wildebeest$Catch[nrow(wildebeest)], nproj)))

#re-conduct MCMC into future years with specified data
wildeproj1 <- jags(data = wildedata_proj1,
                   inits = wildeinits,
                   parameters.to.save = wildeparms,
                   model.file = "wildessmBasic1.txt",
                   n.chains = nc,
                   n.iter = ni,
                   n.burnin = nb,
                   n.thin = nt)

#compile the new projected JAGS output
wilde_proj1 <- data.frame(Year = c(wildebeest$year, 1990:1994),
                          Mean = wildeproj1$mean$N.est,
                          Lower = wildeproj1$q2.5$N.est,
                          Upper = wildeproj1$q97.5$N.est,
                          Obs = c(wildebeest$Nhat,rep(NA,nproj)),
                          LowerObs = c(wildebeest$lci,rep(NA,nproj)),
                          UpperObs = c(wildebeest$uci,rep(NA,nproj)))

#plot population trajectories as earlier extended into the future
ggplot(data = wilde_proj1) +
  geom_ribbon(aes(x=Year, y=Mean, ymin=Lower, ymax=Upper),
              fill="cyan", alpha = 0.25) +
  geom_line(aes(x=Year, y=Mean), linewidth=1, color="blue") +
  geom_point(aes(x=Year, y=Obs), size=1.2) +
  geom_line(data = na.omit(wilde_proj1), aes(x=Year, y=Obs)) +
  geom_errorbar(aes(x=Year,
                    y=Obs,
                    ymin=LowerObs,
                    ymax=UpperObs), width=0, color="grey") +
```

```
  theme_bw()

#import estimates for Nhat
Nhat <- wildeproj1$sims.list$N.est
#import lambda estimates
sig.lambda <- wildeproj1$sims.list$lambda

#compile data together
lambda_df <- data.frame(Year = c(wildebeest$year, 1990:1993),
                        Mean = wildeproj1$mean$lambda,
                        Lower = wildeproj1$q2.5$lambda,
                        Upper = wildeproj1$q97.5$lambda)

#plot the lambda values
ggplot(data=lambda_df) +
  geom_line(aes(x=Year, y=1), color="red", alpha = 0.7, linewidth=1) +
  geom_vline(xintercept=1989, color="blue", alpha = 0.7, linetype = 2, linewidth=1) +
  geom_ribbon(aes(x=Year, y=Mean, ymin=Lower, ymax = Upper),
              fill="black", alpha=0.1) +
  geom_line(aes(x=Year, y=Mean)) +
  geom_point(aes(x=Year, y=Mean)) +
  ylab("population growth rate") +
  xlab("year") +
  theme_bw()

citation("statsecol")
```

References

Estes, Richard D. 2014. *The Gnu's World: Serengeti Wildebeest Ecology and Life History*. University of California Press.

Kellner, Ken. 2021. *jagsUI: A Wrapper Around 'Rjags' to Streamline 'JAGS' Analyses*. https://CRAN.R-project.org/package=jagsUI.

Kéry, Marc, and Michael Schaub. 2011. *Bayesian Population Analysis Using WinBUGS: A Hierarchical Perspective*. Academic Press.

RStudio. 2023. *RStudio: Integrated Development Environment for r*. Posit Software, PBC, Boston, MA: R Foundation for Statistical Computing. http://www.posit.co/.

Schoot, Rens van de, Sarah Depaoli, Ruth King, Bianca Kramer, Kaspar Märtens, Mahlet G Tadesse, Marina Vannucci, et al. 2021. "Bayesian Statistics and Modelling." *Nature Reviews Methods Primers* 1 (1): 1.

Sutherland, Chris. 2023. *Statsecol: A Package Assocaited with the Statistical Ecology MSc*.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.

Youngflesh, Casey. 2018. "MCMCvis: Tools to Visualize, Manipulate, and Summarize MCMC Output." *Journal of Open Source Software* 3 (24): 640. https://doi.org/10.21105/joss.00640.

Zeileis, Achim, and Gabor Grothendieck. 2005. "Zoo: S3 Infrastructure for Regular and Irregular Time Series." *Journal of Statistical Software* 14 (6): 1–27. https://doi.org/10.18637/jss.v014.i06.