

# Hierarchical and state-space models:

Separating out process variation and observation error

# Hierarchical models

Hierarchical modeling is a generalization of linear (and generalized linear) modeling in which regression coefficients are themselves given a model, whose parameters are also estimated from data

# Hierarchical models

Hierarchical modeling is a generalization of linear (and generalized linear) modeling in which regression coefficients are themselves given a model, whose parameters are also estimated from data

$$y = mx + b + \varepsilon \quad \varepsilon \sim N(0, \sigma)$$

# Hierarchical models

Hierarchical modeling is a generalization of linear (and generalized linear) modeling in which regression coefficients are themselves given a model, whose parameters are also estimated from data

$$y = mx + b_i + \varepsilon \quad \varepsilon \sim N(0, \sigma)$$

# Hierarchical models

Hierarchical modeling is a generalization of linear (and generalized linear) modeling in which regression coefficients are themselves given a model, whose parameters are also estimated from data

$$y = mx + b_i + \varepsilon$$
$$\varepsilon \sim N(0, \sigma)$$
$$b_i \sim N(\mu_b, \sigma_b)$$

# Hierarchical models

Hierarchical modeling is a generalization of linear (and generalized linear) modeling in which regression coefficients are themselves given a model, whose parameters are also estimated from data

Called the hyper-distribution

$$\longrightarrow b_i \sim N(\mu_b, \sigma_b)$$

# Hierarchical models

Does this sound familiar??

This is really just a random effects model!

Technically, all random effects models are  
hierarchical models

# Hierarchical models

In EEBB, hierarchical models are often used to separately model the biological process of interest and the sampling process (although there are many other reasons to use hierarchical models)



# Hierarchical models: sequence of random variables

- HMs powerful manner of building statistical models
- Typically reflect our thinking about mechanistic structure of process studied
- Describe stochastic processes with multiple layers/levels of variability and uncertainty
- HMs can be fitted using classical (likelihood) methods (e.g., R packages **lme4** and **unmarked**)
- Bayesian analysis more straightforward

# Why hierarchical models?

HMs make the fitting of complex models easier (especially bayesianly)

- Foster mechanistic, cleaner thinking about a problem
- Lead to mechanistic modeling
- Foster synthetic understanding of large array of models
- Enable honest accounting for all components of variability and uncertainty in a system
- Ideal for rigorous modeling of measurement-error processes in ecological field data

# State-space models

- State-space models are hierarchical models that decompose an observed time-series of counts (usually) and an observation component.
- Originally developed in industry for time series; analyzed in population dynamics recently (De Valpine & Hastings 2002; Buckland *et al.* 2004; Clark & Björnstad 2004)
- Described as a Markovian process that is observed imperfectly (i.e., the quantity of interest is a latent state)
- Change of population size across time is a Markovian process, because population size in year  $t+1$  depends on population size in year  $t$ .

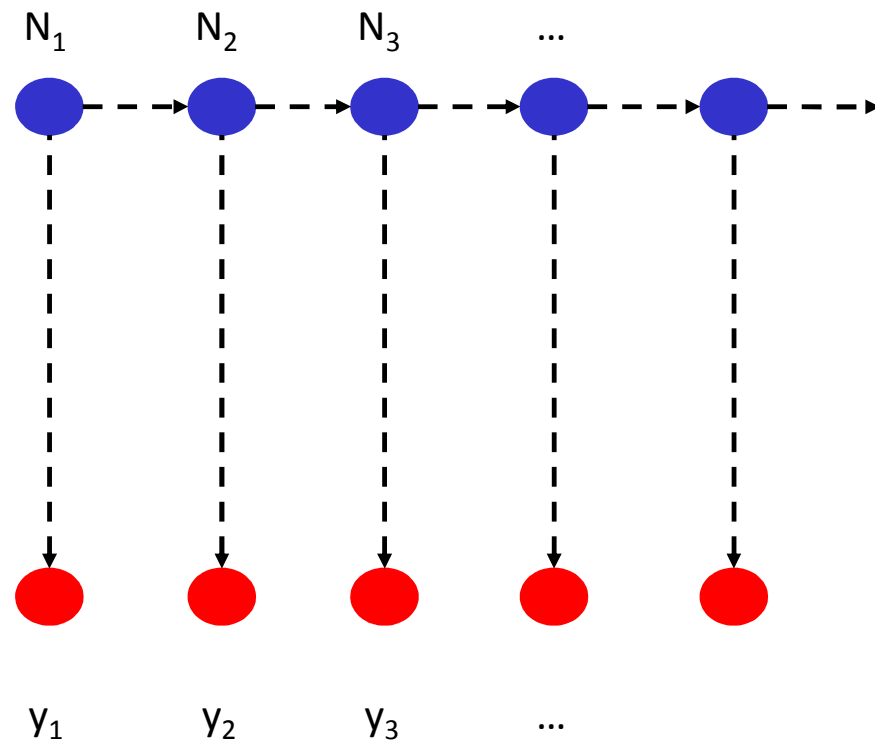
# State-space models

*State process*

**True population size**

*Observation process*

**Observed count**



- - -> Stochastic process

# State-space models

- Any model that describes population growth can be used
- Consider the simplest population model, one with exponential growth (discrete time):

$$N_{t+1} = N_t \lambda_t$$

$\lambda_t$  = population growth rate in year t

$N_t$  = population size in year t

# State-space models

Can model  $\lambda_t$  as follows:

$$\lambda_t \sim \text{Normal}(\bar{\lambda}, \sigma_{\lambda}^2)$$

- Each  $\lambda_t$  is generated as realization of a Normal process with mean  $\bar{\lambda}$  and variance  $\sigma_{\lambda}^2$
- $\bar{\lambda}$  is the long-term growth rate of the population
- $\sigma_{\lambda}^2$  is a measure of the environmental variability affecting this growth rate (i.e., environmental stochasticity)

# State-space models

The second part of the model maps the true state of the process (only partially observable) to the observed data

- This part is conditional on the process model equations:

$$y_t = N_t + \varepsilon_t$$

$$\varepsilon_t \sim \text{Normal}(0, \sigma_y^2)$$

- Assume that in each year  $t$  we make an error while counting the population ( $\varepsilon_t$ ), in one year we may miss more individuals than we double-count, in another year this may be opposite.

# State-space models

- We also assume that these counting errors can be described by a Normal distribution with mean zero and variance ( $\sigma_y^2$ ), referred to as the observation error.
- These equations are called observation equations



# State-space models

## Some things to consider about basic SSMs:

- Simple state-space performs well in the presence of random observation errors (Random means constant rates of double counting and of nondetection; false positives  $\approx$  false negatives)
- Accuracy depends on the magnitude of the observation error and on sample size
- NOT able to recover unbiased estimates of true population size if there are systematic errors in detection

# State-space models

## Some things to consider about basic SSMs:

- State-space models produce unbiased estimates of population indices (i.e., of  $Np$ ) if detection probability is  $<1$  (and similarly for false positive rates) *iff* there is no systematic temporal pattern in either error rates.
- State-space models do not yield unbiased estimates of population size (nor of indices) if there are temporal (or spatial!) patterns either in detection probability or in the false positive rates (unless of course the two cancel each other out).

# State-space models

## *Lab: SSM with simulated data*

- It is difficult to fit SSM using MLE in R
- Code for this – using the Kalman Filter – is available in the Base repo as is a paper explaining this approach in detail
- We are instead going to start learning about JAGS today for Bayesian analyses.

## *Homework: SSM with simulated data in JAGS*

Simulate the data for a state-space model.

**\*\*Run the model in JAGS and produce a figure of the results.**

# Bayesian analysis of a model

Bayes rule for statistical inference:

$$P(\theta|y) = \frac{P(y|\theta)P(\theta)}{P(y)} = \frac{P(\theta, y)}{P(y)}$$

Posterior distribution:

$$P(\theta|y)$$

Likelihood function:

$$P(y|\theta)$$

Prior distribution:

$$P(\theta)$$

Prob. of data:

$$P(y) = \int P(y|\theta)P(\theta)d\theta$$

- **NOTE:** Use probability to express imperfect knowledge

# Bayesian computation

BUGS: Flexible, generic software, does:

1. Simple and intuitive model description language (BUGS language)
2. Automatic development of MCMC algorithms (algorithmic black box)
3. Run algorithm: produce posterior samples

Three variants:

- **WinBUGS:** [www.mrcbsu.cam.ac.uk/bugs/winbugs/contents.shtml](http://www.mrcbsu.cam.ac.uk/bugs/winbugs/contents.shtml)
- **OpenBUGS:** [www.openbugs.info/w/](http://www.openbugs.info/w/)
- **JAGS:** [mcmc-jags.sourceforge.net/](http://mcmc-jags.sourceforge.net/)

# Bayesian computation in JAGS

Run through R with the library "R2jags"

Need to specify the following things

1. A text file with the model (must include the likelihood and the priors)
2. The data that will be used to estimate parameters in the model (data)
3. Initial values for the Markov chains (inits)
4. The parameters that we want to monitor (params)
5. The number of chains (nc)
6. The number of iterations for each chain to run (ni)
7. The number of iterations that should be discarded as a burn in (nb)
8. The number by which to thin the chains (nt)