Introduction to univariate AR lag-1 state-space models

Eli Holmes

FISH 507 – Applied Time Series Analysis

17 January 2017

Weeks 1-3: building blocks for analysis of multivariate time-series data with observation error, structure, and missing values

- Matrix math (multivariate)
- Properties of time series data (AR and MA models)
  $x(t) = b_1\, x(t-1) + b_2\, x(t-2) + e(t)$
- Fitting models and model selection (analysis)
  - Bayesian models (non-gaussian errors, non-linearity, zeros)
- State-space models (observation error and missing values)

Starting next week: putting this all together to start analyzing ecological data sets

univariate linear state-space model

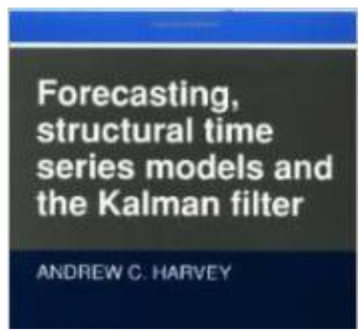
$$x_t = x_{t-1} + u + w_t, \quad w_t \sim Normal(0, q)$$
$$y_t = x_t + v_t, \quad v_t \sim Normal(0, r)$$

The x model is the classic “random walk”.
This model is a random walk observed with error.

univariate linear state-space model

$$x_t = x_{t-1} + u + w_t, \quad w_t \sim Normal(0, q)$$

$$y_t = x_t + v_t, \quad v_t \sim Normal(0, r)$$



Many textbooks on this class of model. Used in extensively in economics and engineering



Definition: AR-1 or AR lag-1

Value at time t is the value at time t-1 plus random error

$$x_t = x_{t-1} + u + w_t$$

$$x_{t+1} = x_t + w_t$$

$$x_t = bx_{t-1} + u + w_t$$

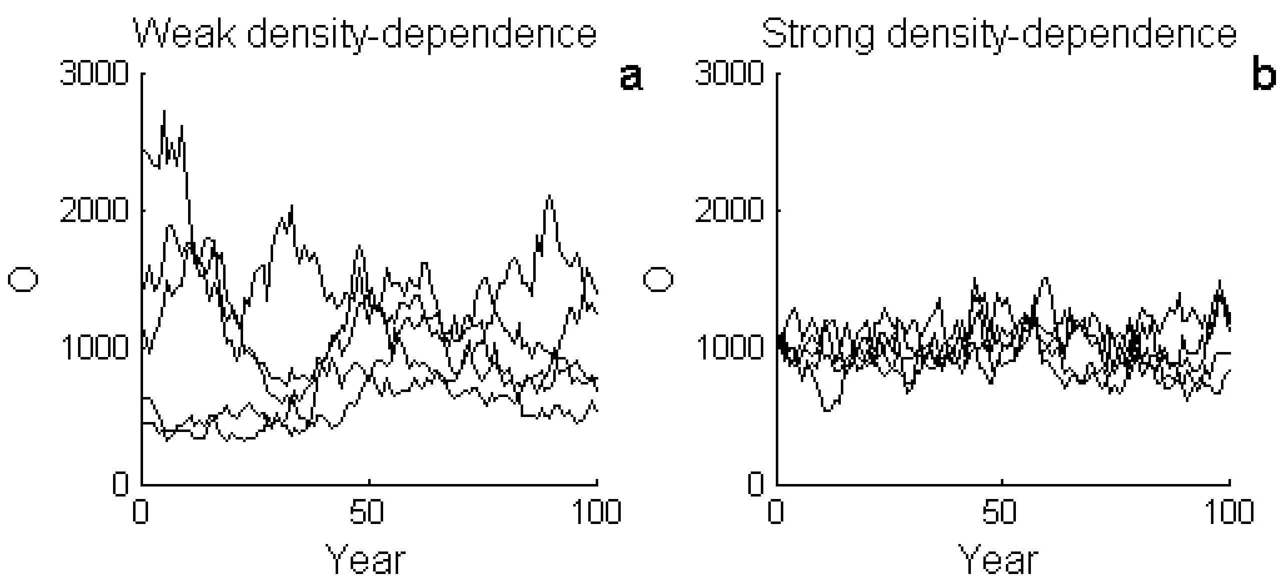
Addition of “b” (<1) leads to process model with mean-reversion,

$$N_t = \exp(u + e_t) N_{t-1}^b$$

$$x_t = b x_{t-1} + u + e_t$$

Log-space

$$e_t \sim Normal(0, q)$$



b<1: Gompertz density-dependent process

This model is quite hard to fit

$$N_t = \exp(u + e_t) N_{t-1}^b$$

$$x_t = b x_{t-1} + u + e_t$$

Log-space

$$e_t \sim Normal(0, q)$$

b and u are confounded = ridge likelihood = many b/u combinations that fit the data

If you have observation error, you need either long times or replication to estimate this model.

Why is the AR-1 model so important in analysis of ecological data?

Additive random walks

- Movement, changes in gene frequency, somatic growth if growth is by fixed amounts

$$x_t = x_{t-1} + u + w_t, \quad w_t \sim Normal(0, q)$$

Why normal? The average of many small perturbations, regardless of their distribution, is normal

Multiplicative random walks

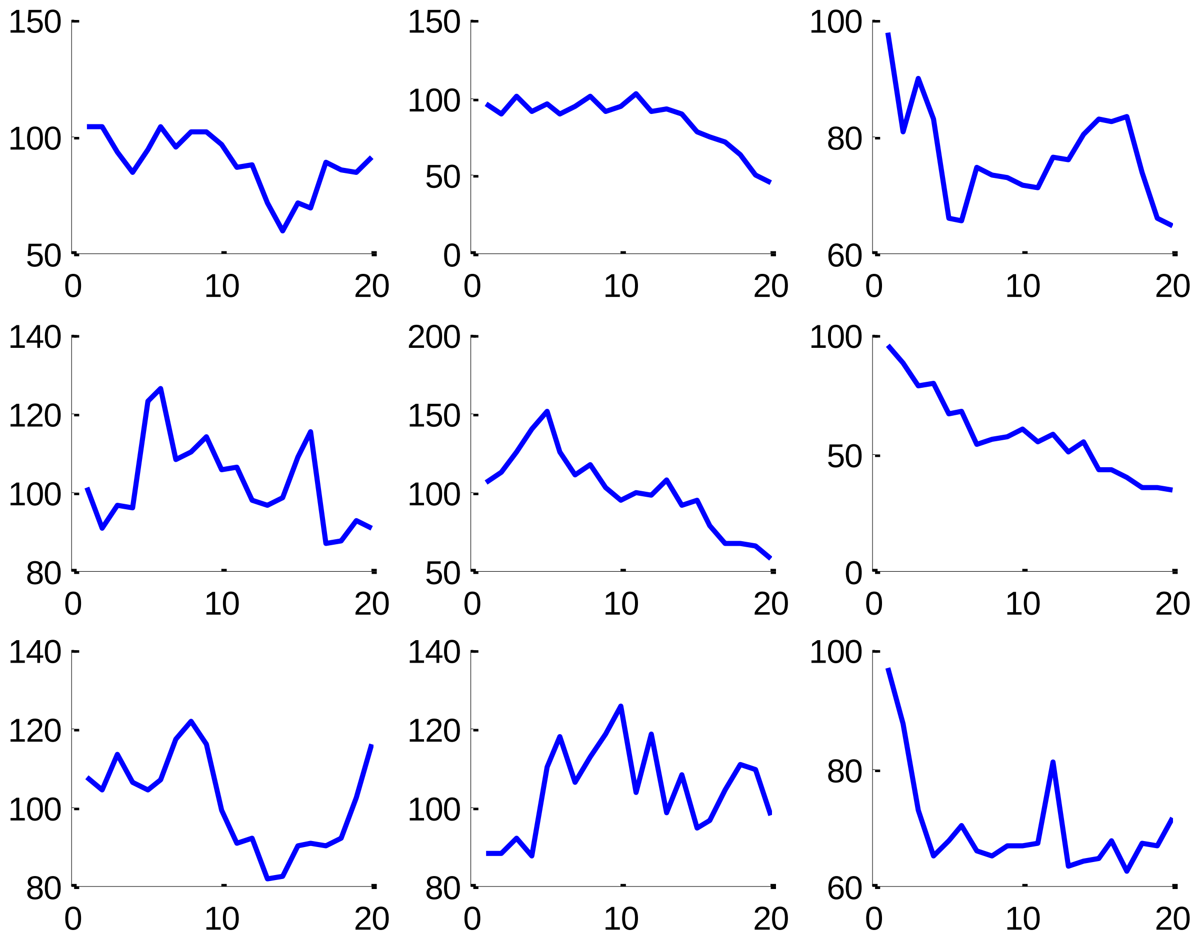
- Population growth, somatic growth if growth is by percentage

$$n_t = \lambda n_{t-1} w_t, \quad w_t \sim \log-Normal(0, q)$$

- take log and you get the linear additive model above. log-normal means that 10% increase is as likely as 10% decrease

An AR-1 random walk can show a wide-range of trajectories, even for the same parameter values

All trajectories came from the same rw model: $x_t = x_{t-1} - 0.02 + e_t$, $e_t \sim$ Normal(mean=0.0, var=0.01)
same as the “stochastic exponential growth model”: $N_t = N_{t-1} \exp(-0.02+e_t)$



Definition: state-space

The "state", the x, is a hidden (dynamical) variable. In this class, it is a **hidden random walk.**

Our data, y, are observations of this.

Often state-space models include inputs (explanatory variables). and typically at least the x is multivariate, and often also y.

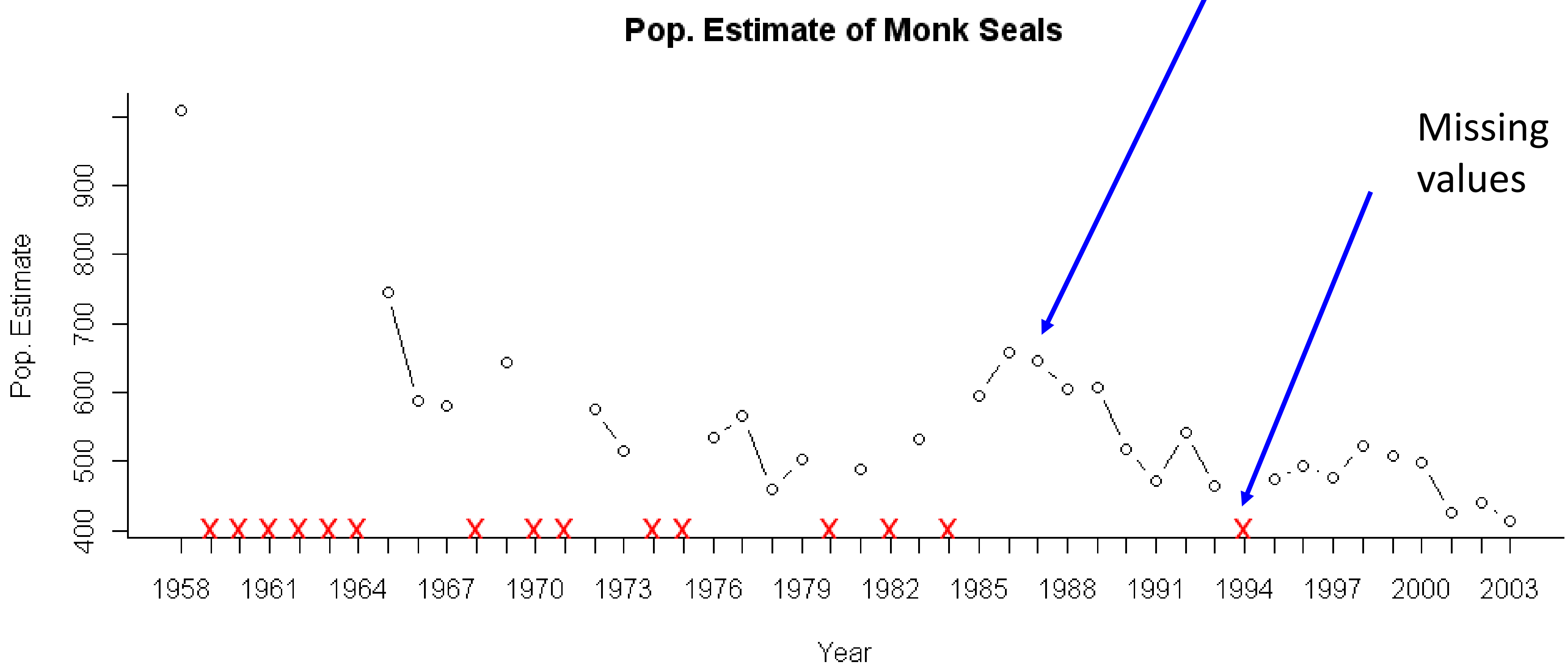
The model you are seeing today is a simple univariate state-space model with no inputs.

state process $x_t = x_{t-1} + u + w_t, \quad w_t \sim Normal(0, q)$

obs process $y_t = x_t + v_t, \quad v_t \sim Normal(0, r)$

univariate example: population count data

Yearly (usually) population (or subpopulation) counts

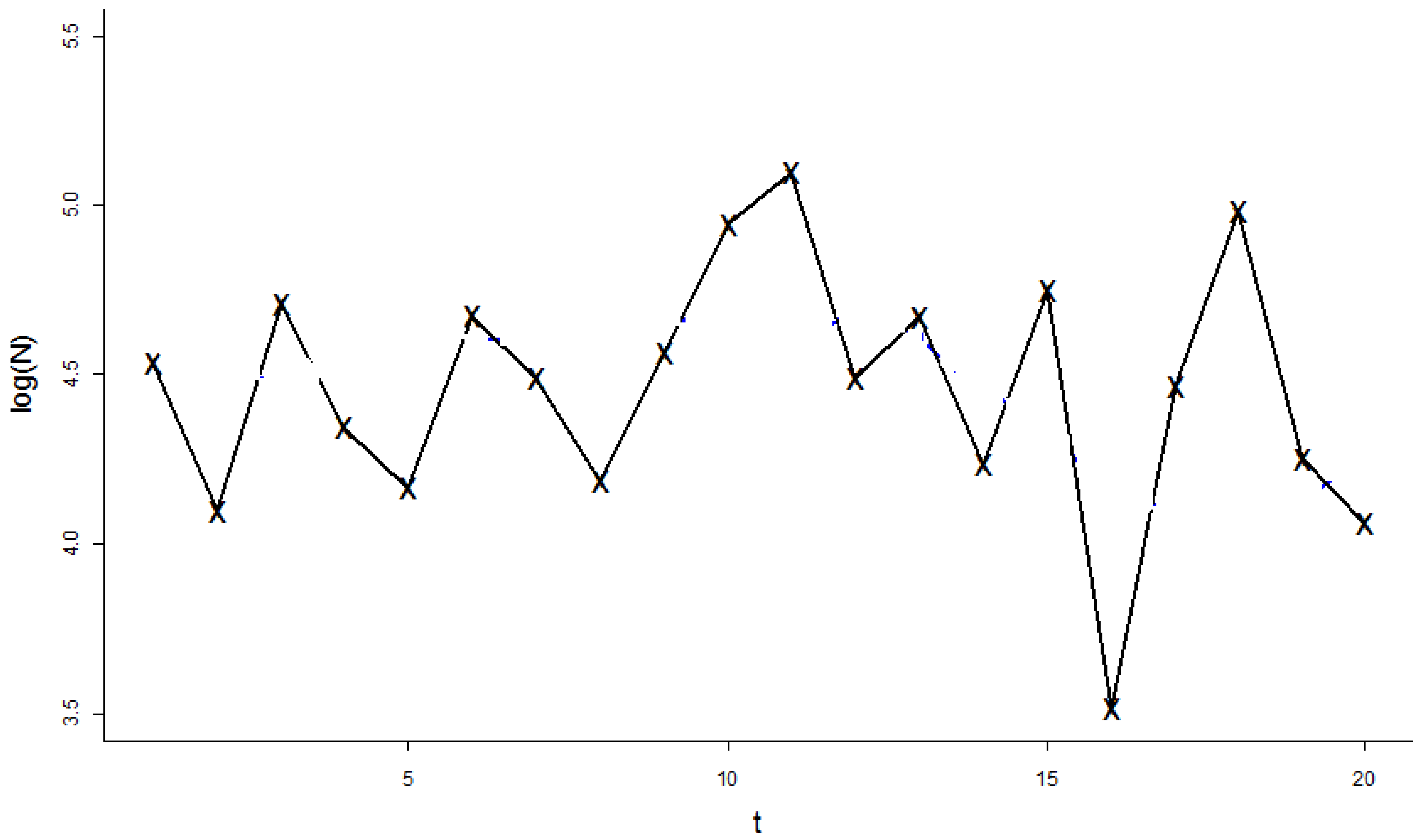


Missing values

Observation error

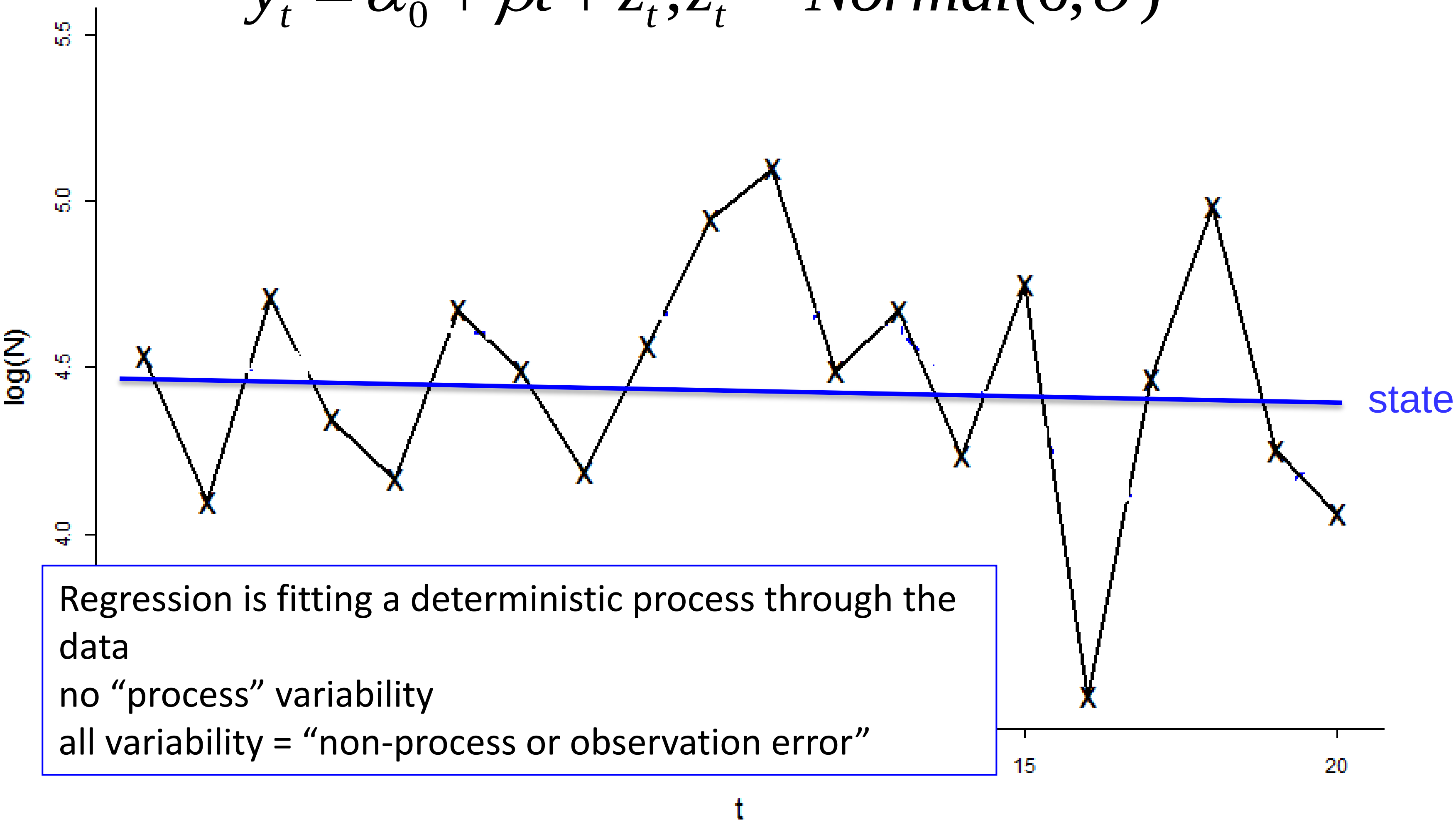
There IS some number of sea lions in our population in year x, but we don't know that number precisely. It is "hidden".

Suppose we have the following data (population counts logged)



A linear regression model

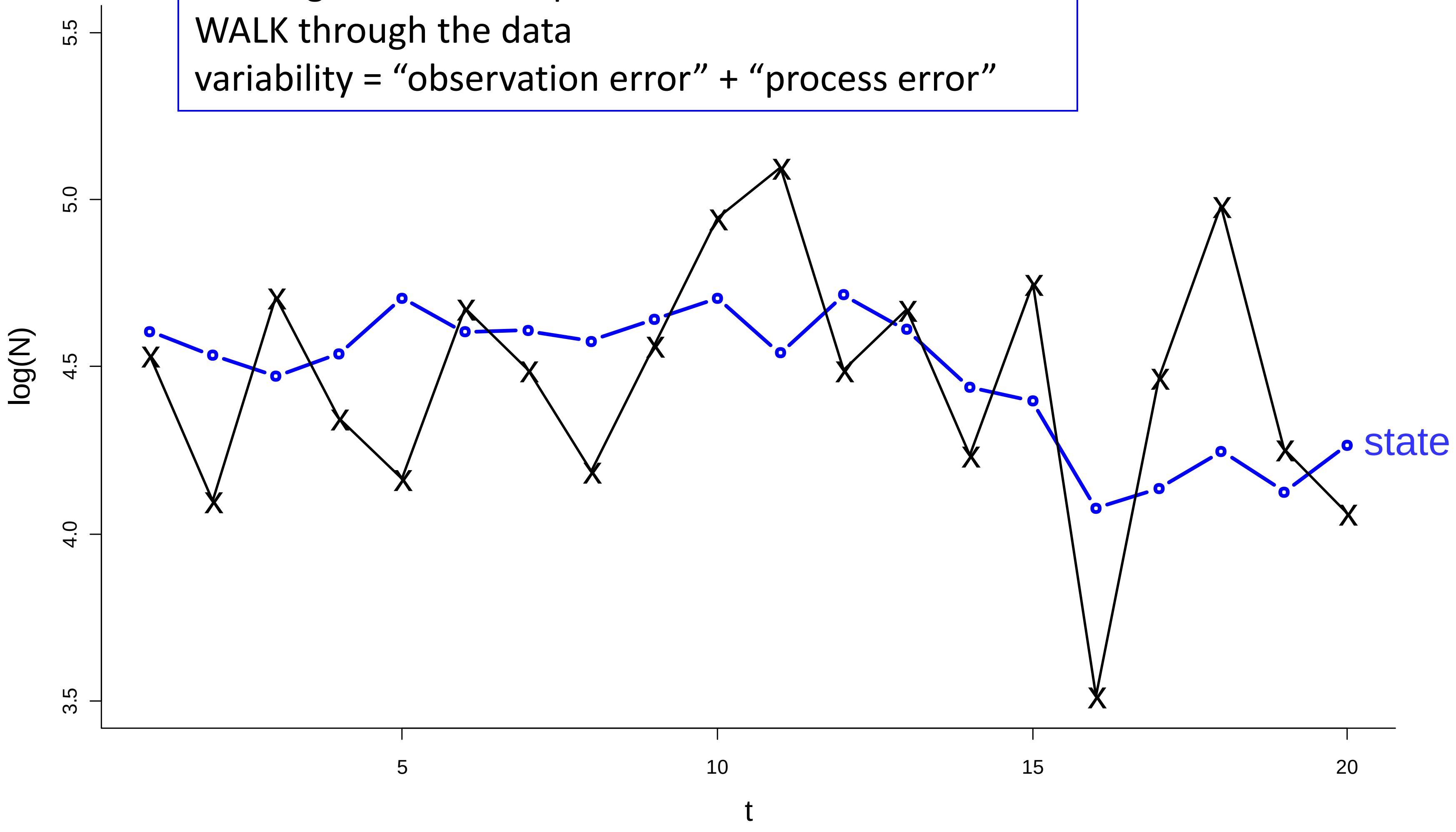
$$y_t = \alpha_0 + \beta t + z_t ; z_t \sim Normal(0, \sigma)$$



Regression is fitting a deterministic process through the data
no "process" variability
all variability = "non-process or observation error"

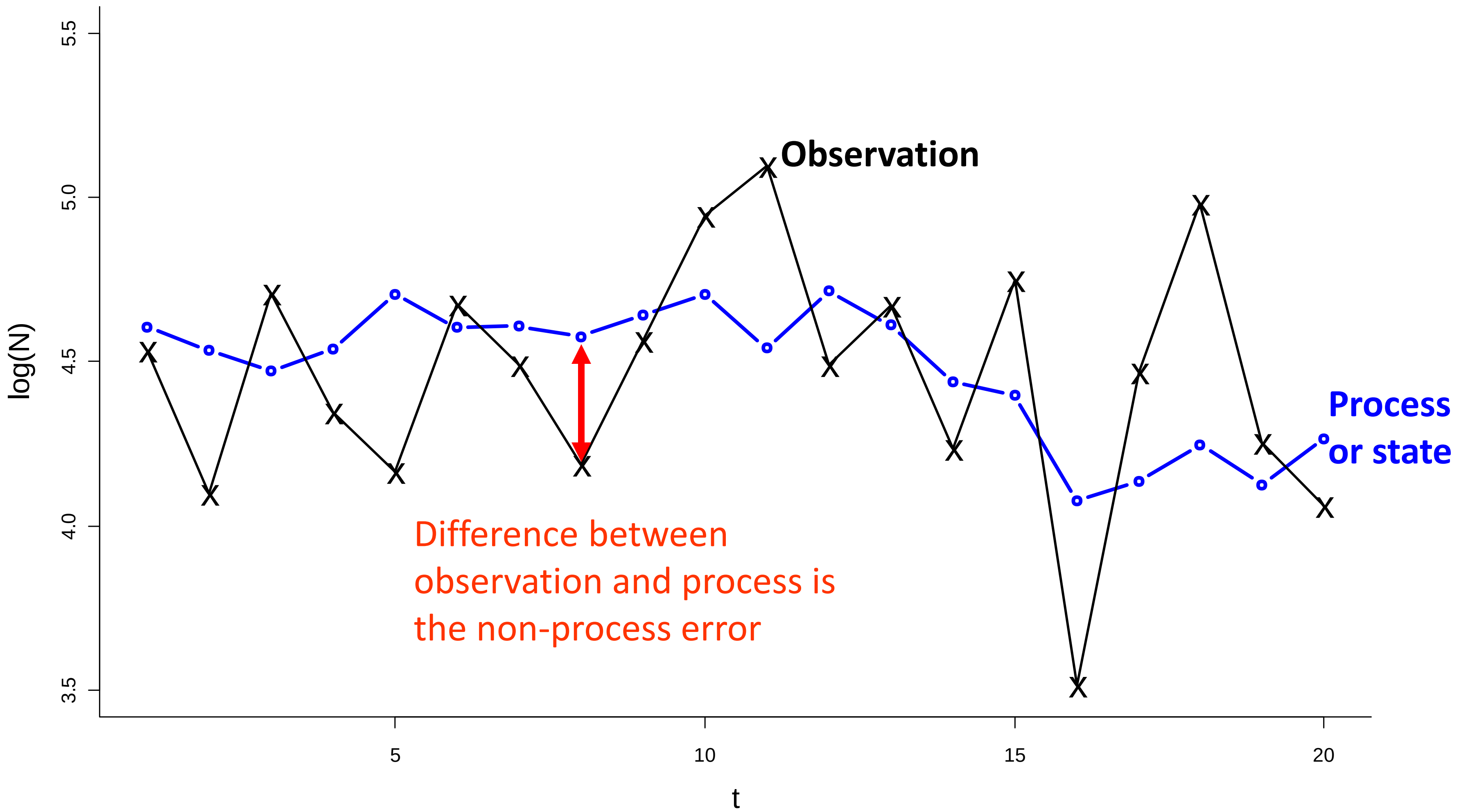
Versus a state-space model

Autoregressive state-space models fit a RANDOM WALK through the data
variability = “observation error” + “process error”



Two types of variability

#1 observation or "non-process" variability



Two types of variability

#1 observation or “non-process” variability

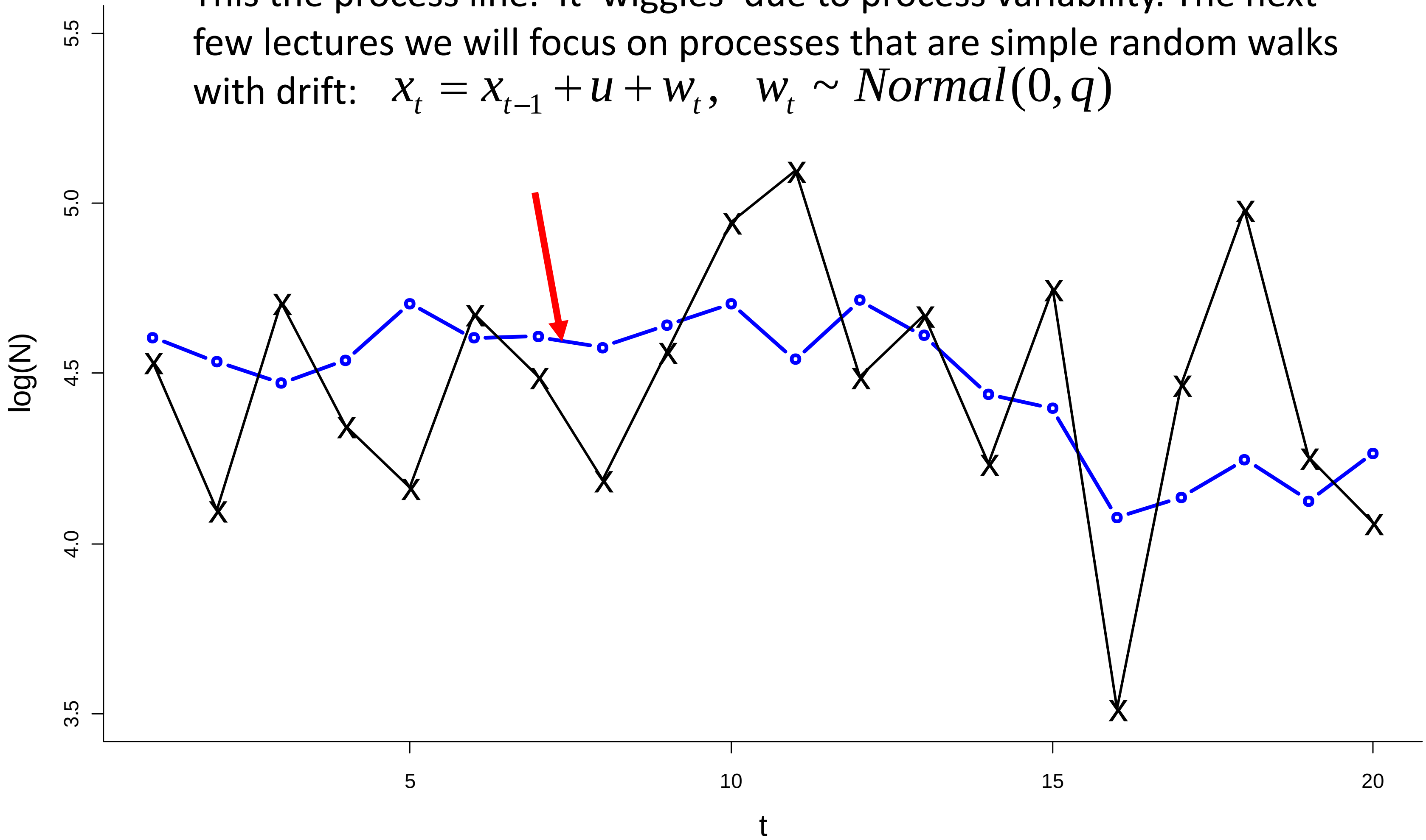
The non-process (observation) variance is often unknowable in fisheries and ecological data

- *Sightability varies due to factors that may not be fully understood or measureable*
  - Environmental factors (tides, temperature, etc.)
  - Population factors (age structure, sex ratio, etc.)
  - Species interactions (prey distribution, prey density, predator distribution or density, etc.)

- *Sampling variability*--due to how you actually count animals--is just one component of observation variance

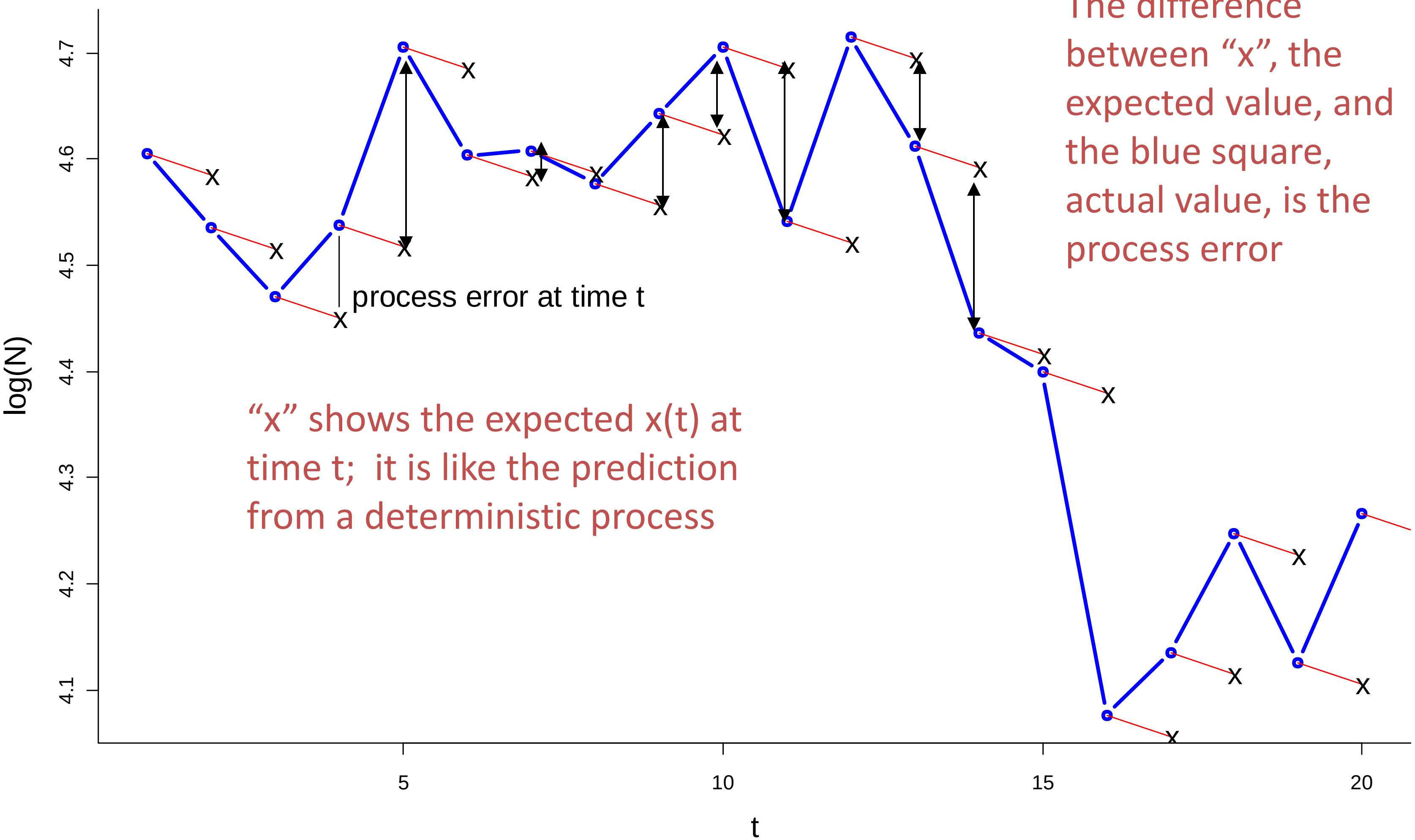
#2 Process variability

This the process line. It 'wiggles' due to process variability. The next few lectures we will focus on processes that are simple random walks with drift: $x_t = x_{t-1} + u + w_t, \quad w_t \sim Normal(0, q)$



Process error is the difference between the expected x(t) and the actual value

Let's say that in $x(t)=x(t-1)-0.02+e(t)$*

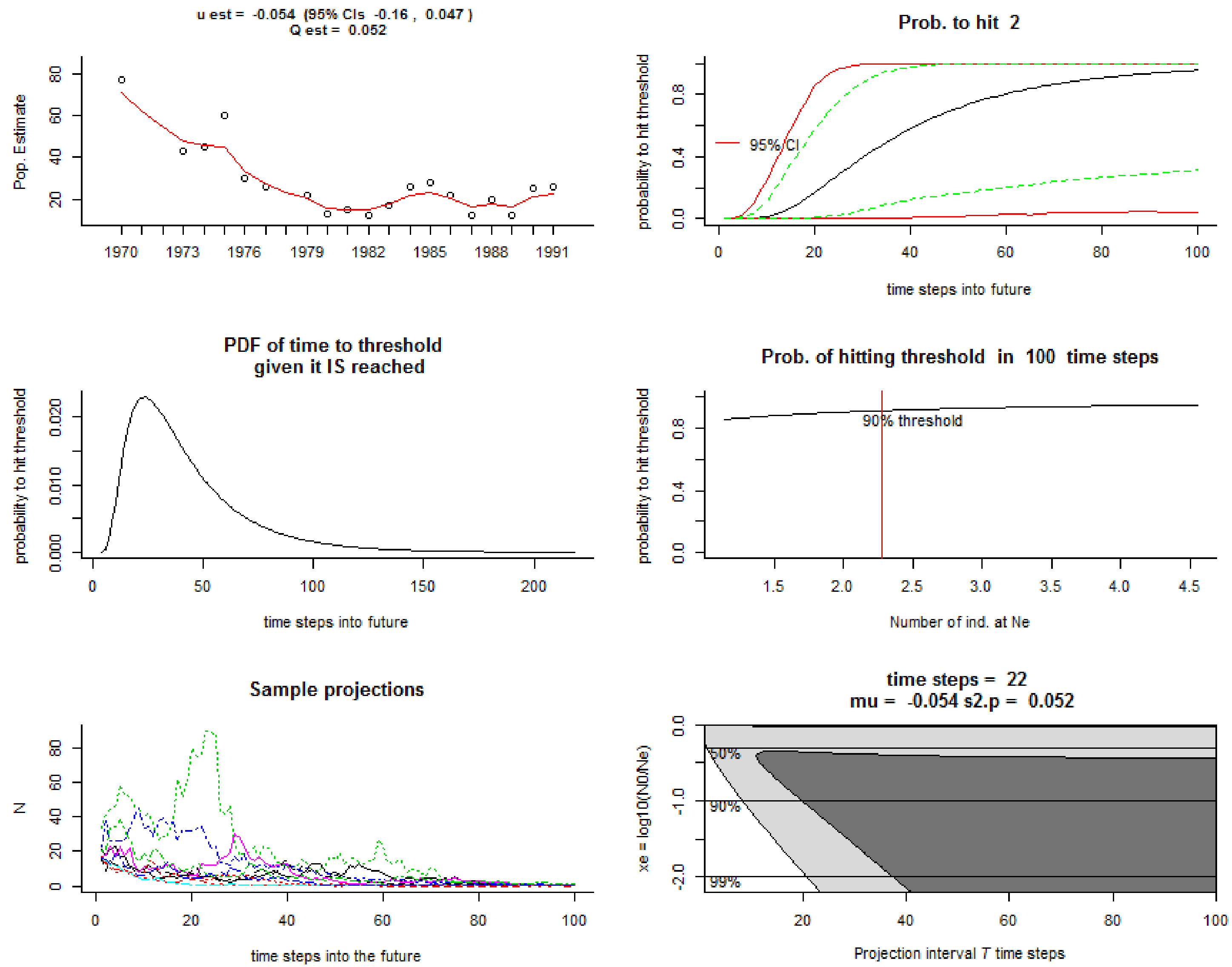


"x" shows the expected x(t) at time t; it is like the prediction from a deterministic process

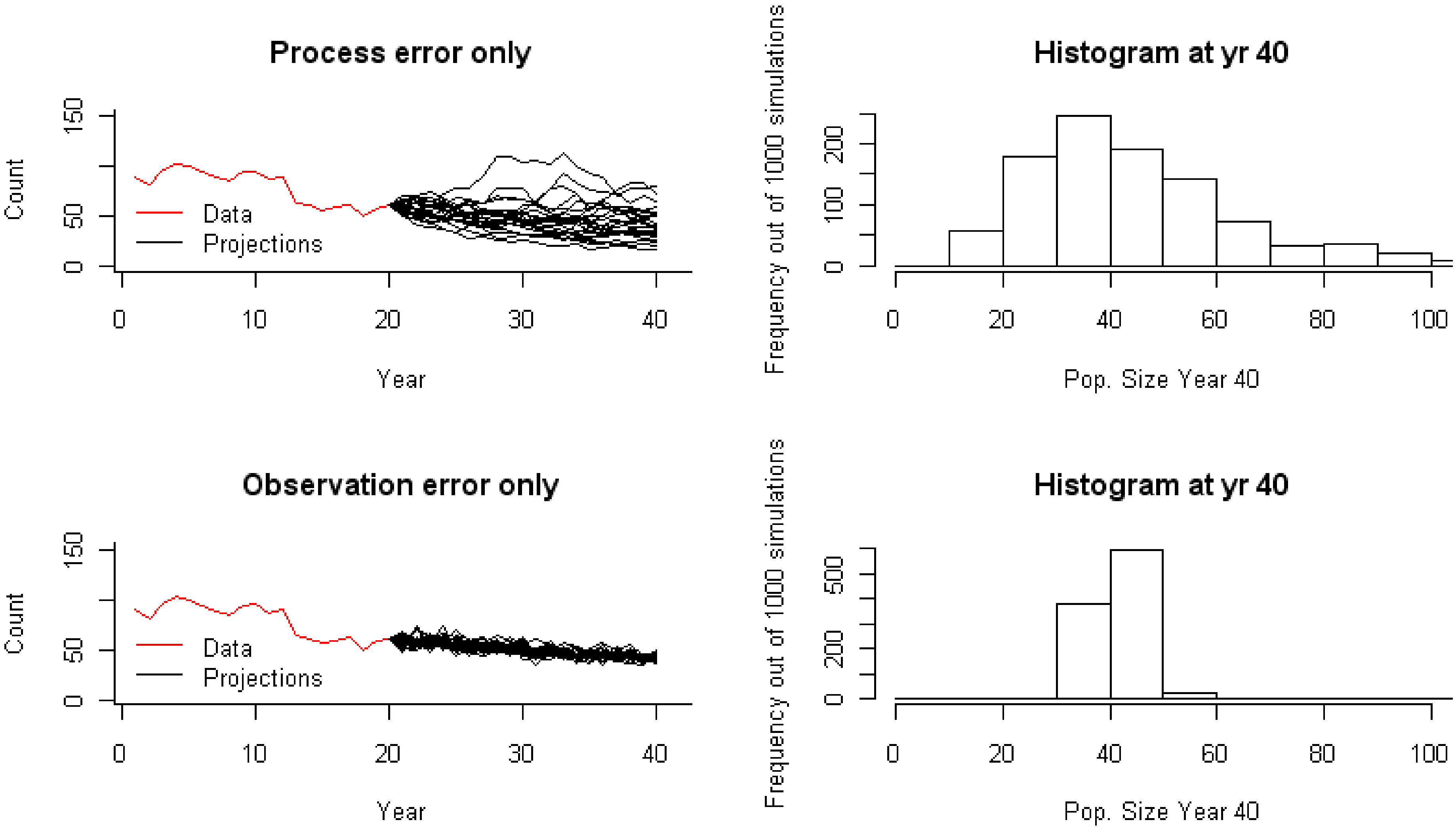
The difference between "x", the expected value, and the blue square, actual value, is the process error

*If this were a population model, that means a the mean rate of decline is ca 2% per year

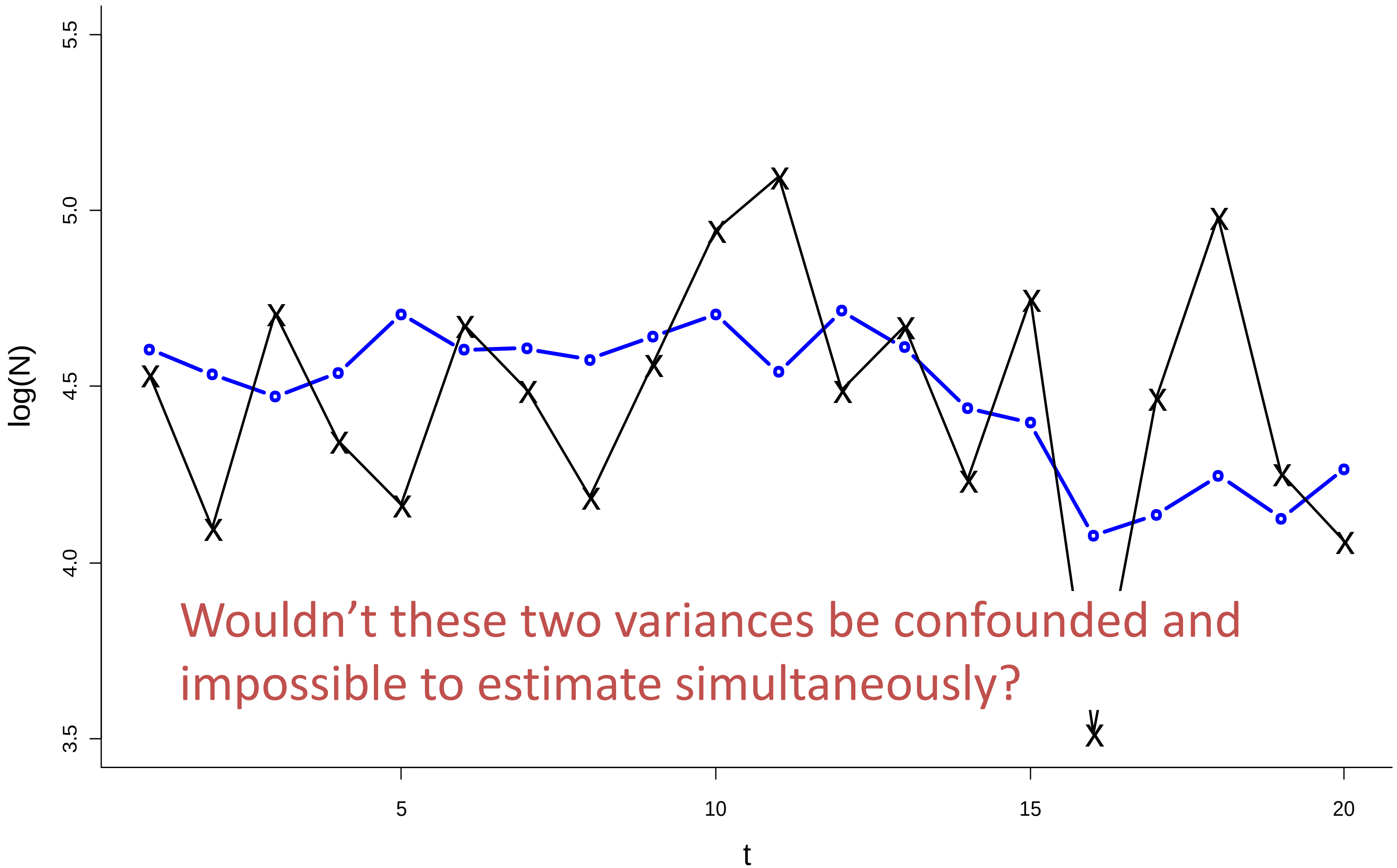
One use of univariate state-space models is “count-based” population viability analysis (chap 6 HWS2014)



How you model your data has a large impact on your forecasts



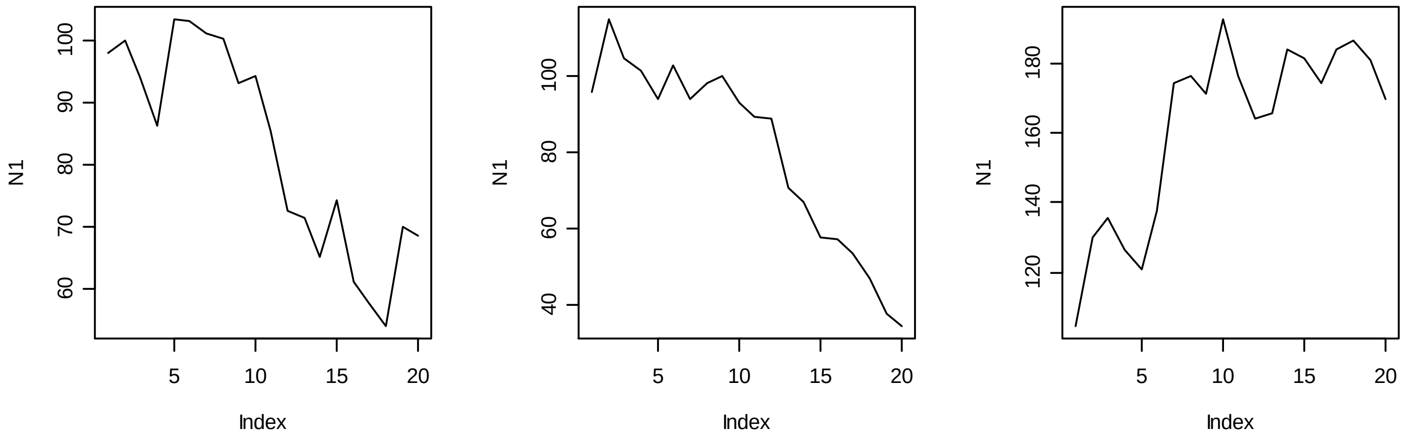
How can we separate process and non-process variance?



How can we separate process and observation variance? They have different temporal patterns.

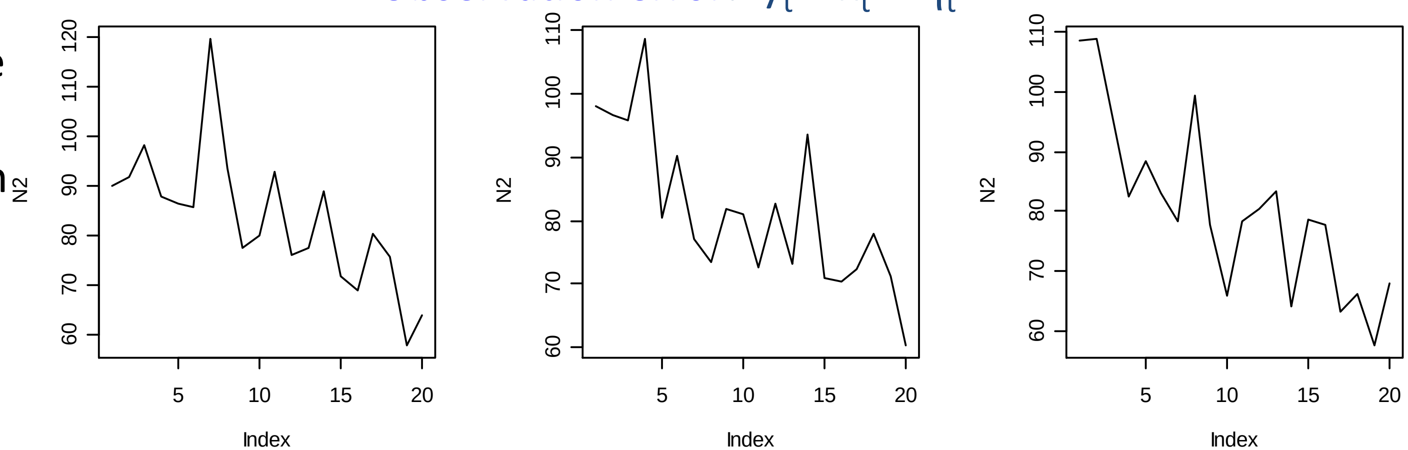
Process error: $x_t = x_{t-1} + u + e_t$

multiple sims of x(t) with same u and q



Observation error: $y_t = x_t + \eta_t$

multiple sims of y(t) with same x(t)



An AR-1 state-space model combines a model for the hidden AR-1 process with a model for the observation process

…and allows us to separate the variances

Process model

$$x_t = x_{t-1} + u + w_t$$
$$w_t \sim Normal(0, q)$$

AR lag-1
random walk with drift
normally distributed
process errors

Observation model

$$y_t = x_t + v_t$$
$$v_t \sim Normal(0, r)$$

observation errors
normally distributed
process errors

Kalman Filter: Estimate the x in a state-space model

A mathematical algorithm that solves for the 'optimal' (least error or maximum-likelihood) x_t given all the data (y) from time 1 to t

Predict: Given an x_0, predict x_1 from your model

Update: Given y_1, update your x_1 estimate

Predict: Given an x_1, predict x_2 from your model

Update: Given y_2, update your x_2 estimate

Predict: Given an x_2, predict x_3 from your model

Update: Given y_3, update your x_3 estimate

Let’s simulate and try fitting some models

- Open up R and follow after me
- univariate_example_1.R
- univariate_example_2.R
- univariate_example_3.R

How to write a straight-line as AR-1

- ##Preliminaries: how to write ##x=intercept+slope*t as a AR-1
- x(0)=intercept
- x(1)=x(0)+slope #this is x at t=1
- x(2)=x[1]+slope
- so..
- x(t)=x(t-1)+slope+w(t), w(t)~N(0,0)

MARSS R Package

- Fits MARSS models (multivariate AR-1 state-space)
- General, fits any MARSS model with Gaussian errors

- But
- Maximum likelihood
- Slow. Students working with large data sets have gotten huge speed improvements by coding their models in TMB

MARSS R Package

- Fits MARSS models (multivariate AR-1 state-space)
- MARSS model syntax

X(t) = **B** X(t-1) + **U** + w(t), w(t) ~ N(0, **Q**)

Y(t) = **Z** X(t) + **A** + v(t), v(t) ~ N(0,**R**)

- **`fit2=MARSS(y,model=mod.list)`**

- **`y`** is data; **`model`** tells MARSS what the parameters are
- The parameters are MATRICES
- You write matrices just like they appear in your model on paper
- You pass **`model`** to MARSS as a list

X(t) = **B** X(t-1) + **U** + w(t), w(t) ~ N(0, **Q**)

Y(t) = **Z** X(t) + **A** + v(t), v(t) ~ N(0,**R**)

Let's say we want to fit this model:

```
mod.list=list(
  U=matrix("u"),
  x0=matrix(0),
  B=matrix(1),
  Q=matrix(0.1),
  Z=matrix(1),
  A=matrix(0),
  R=matrix("r"),
  tinitx=0)
```

$$x_t = x_{t-1} + u + w_t, w_t \sim N(0, \sigma^2 = 0.1)$$

$$y_t = x_t + v_t, v_t \sim N(0, r)$$

$$x_0 = 0$$

How do you know when to use a process error or observation error model?

- If your time-series data contain both types, use a model with both types.
- To estimate both variances, you need a) 20+ time steps **OR** b) multi-site data.
- If you don't have enough data, you need to use assumptions about one of the variances. Meaning a) fix the value or b) incorporate a prior.
- Diagnostics: Observation error induces autocorrelation in the noise of an autoregressive process. Fit a process-error only model ($R=0$) and check for autocorrelation of residuals

Other types of “non-process” error

- Fluctuations that don’t have “feedback” (variance doesn’t explode)

- Lots of biological processes also create noise that looks like that
  - age-structure cycles
  - density-dependence
    - cyclic variability in fecundity
    - predator-prey interactions

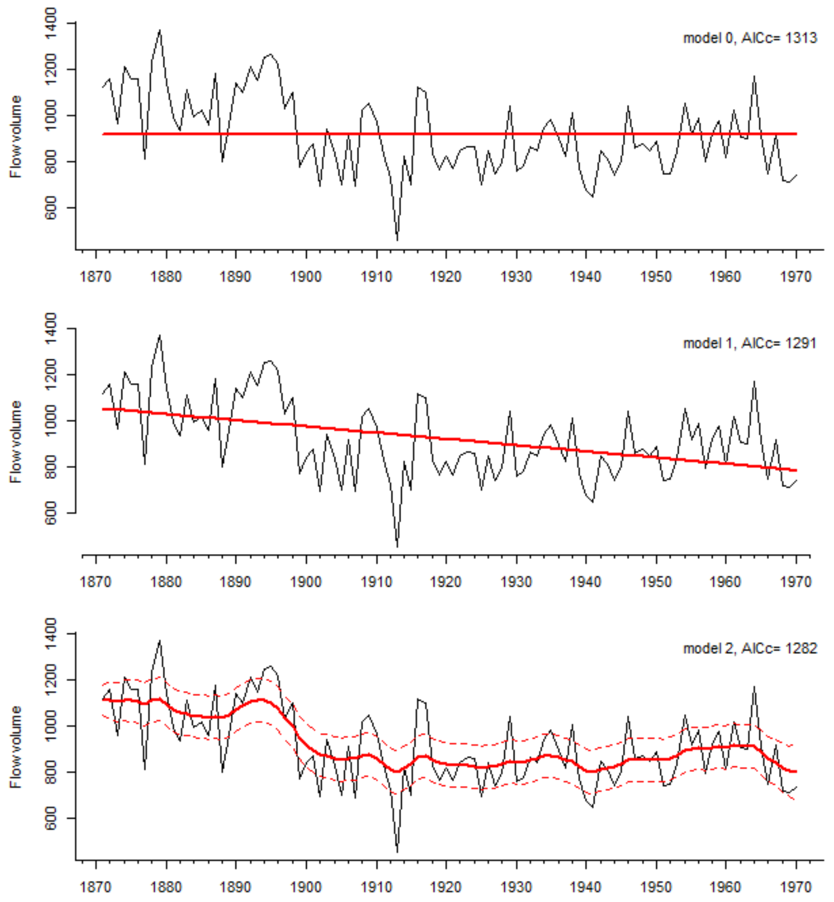
- If your model cannot accommodate that cycling,
  - it tends to get ‘soaked’ up in the ‘non-process’ error component

- If your model can accommodate that cycling,
  - estimation of ‘observation error’ variance can be confounded, unless you have long, long datasets or replicates

Basic diagnostics #1 Plot your residuals

Basic diagnostics

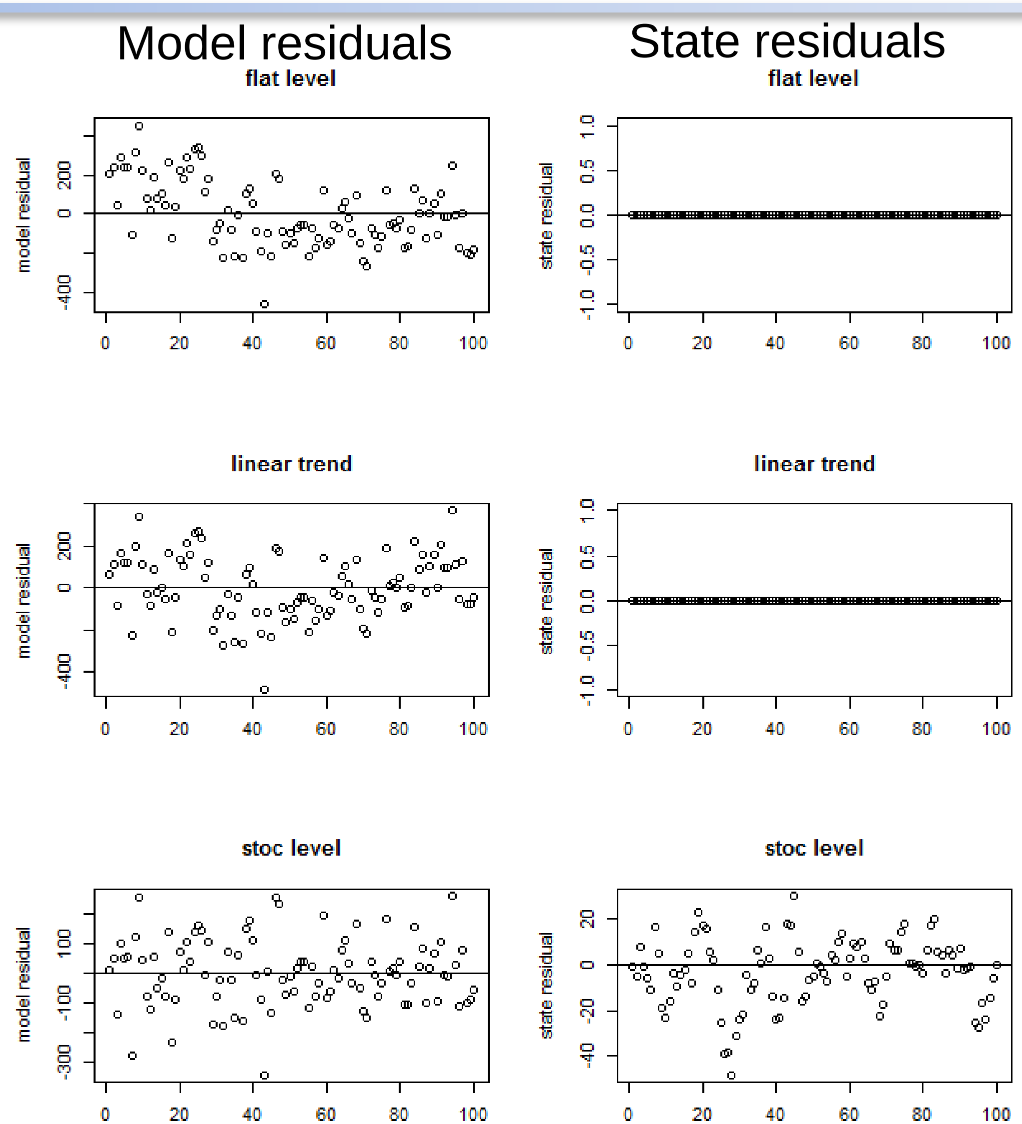
Nile River models from the lab handout



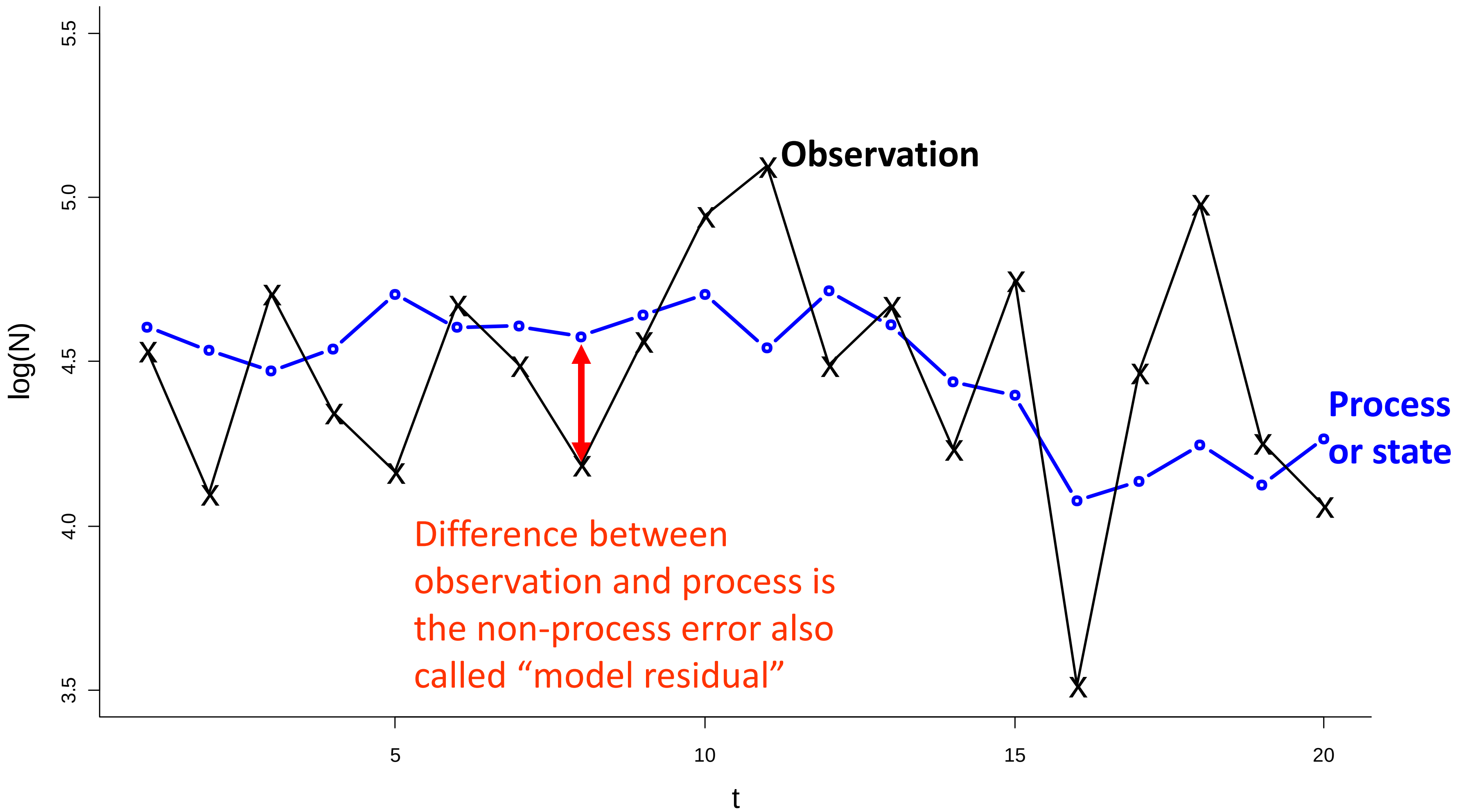
Basic diagnostics: plot the residuals

There should be no temporal trends!

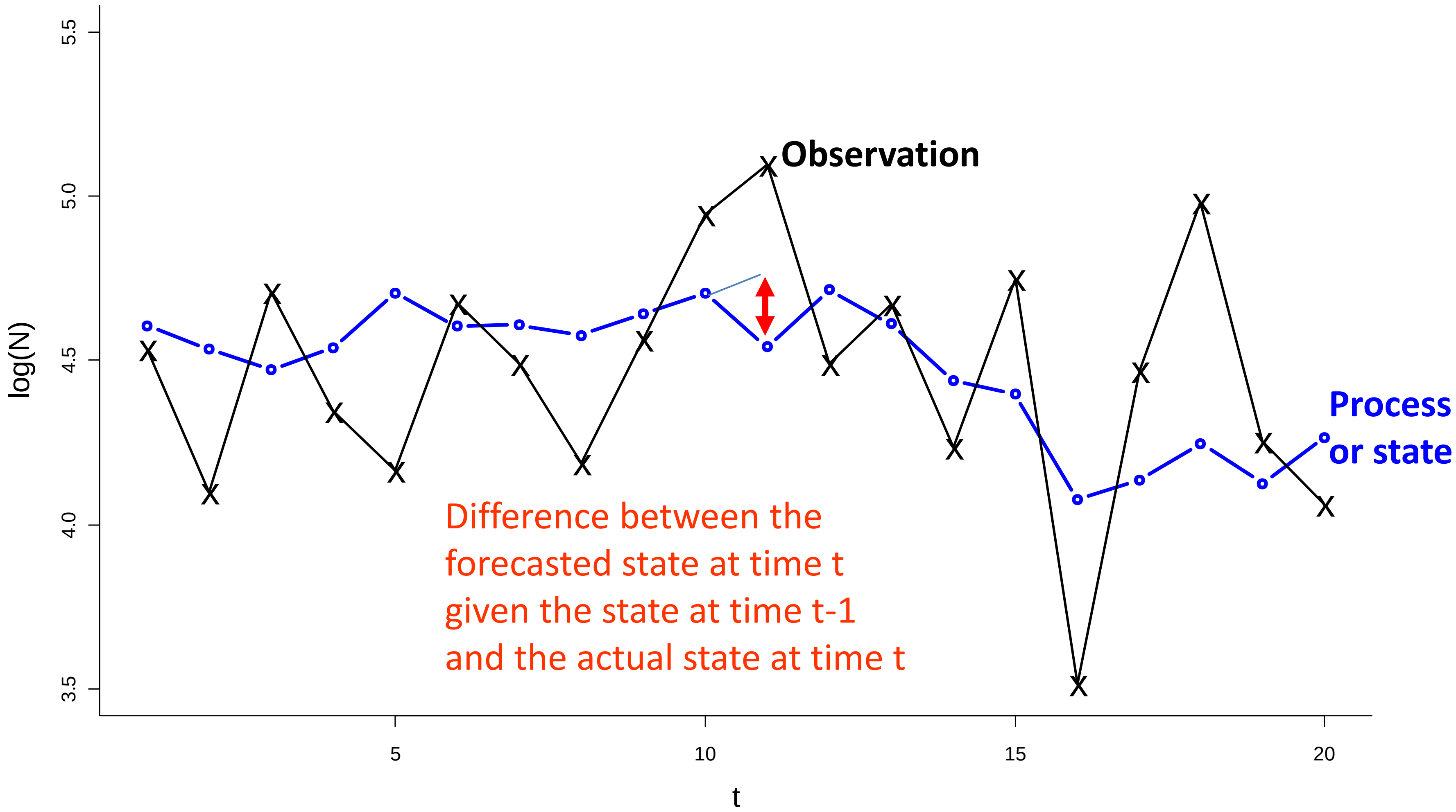
They should be centered about 0.



non-process error or model residual

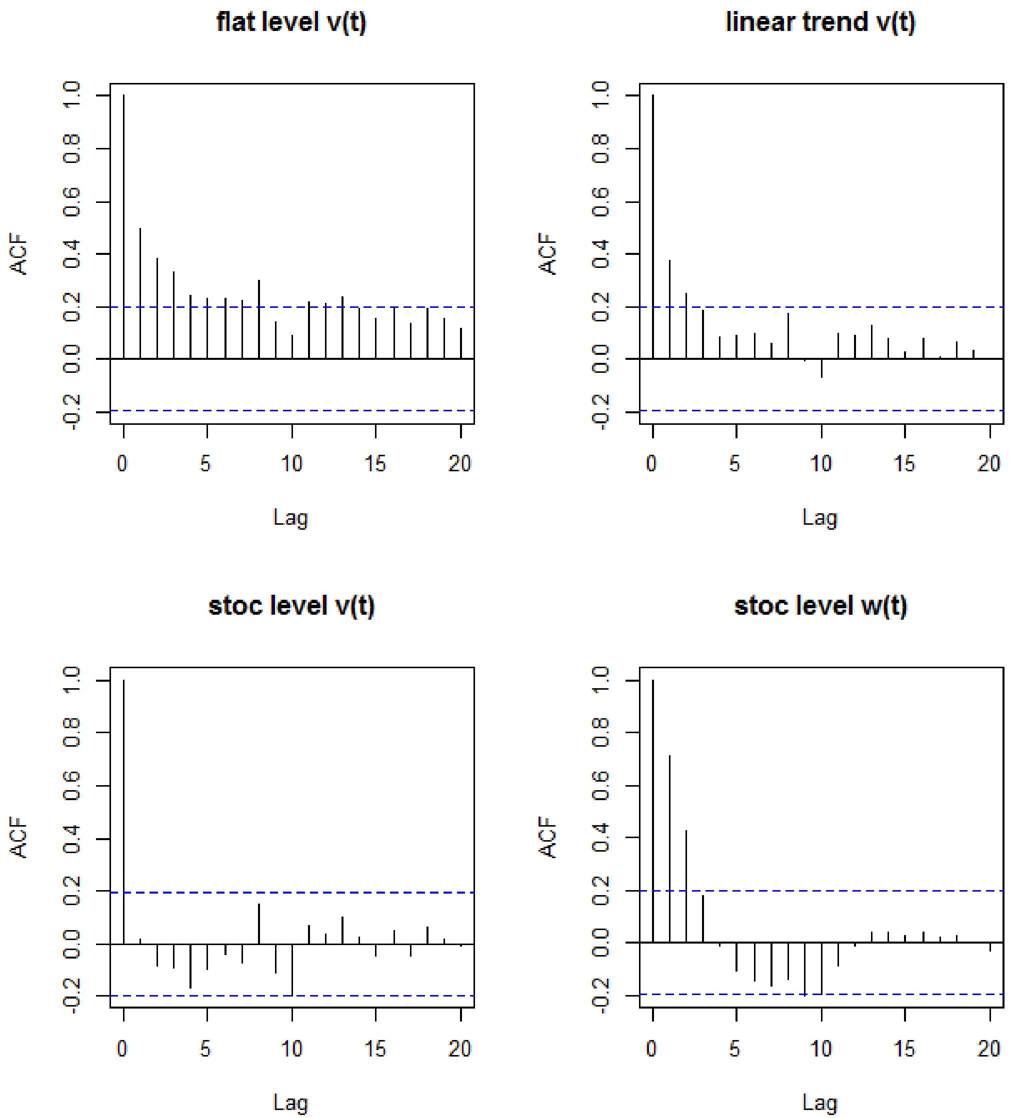


process error or state residual



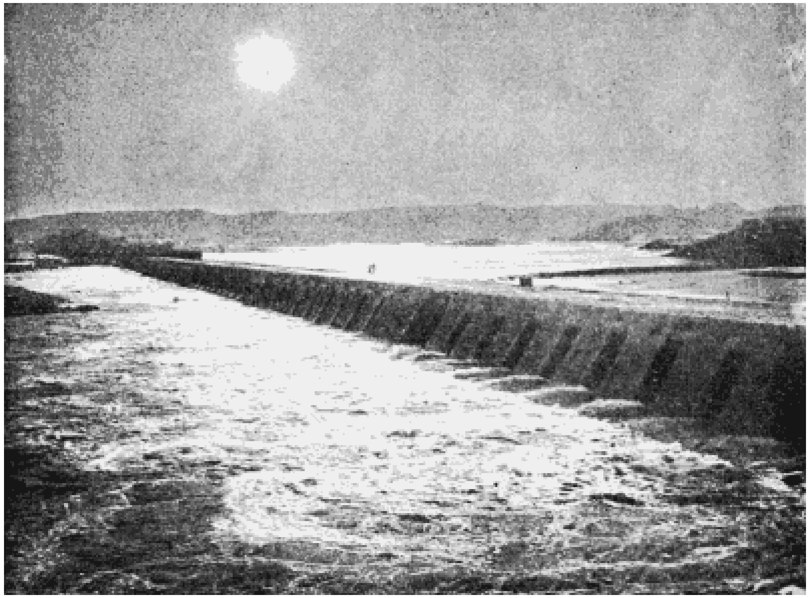
Basic diagnostics: check acf of residuals

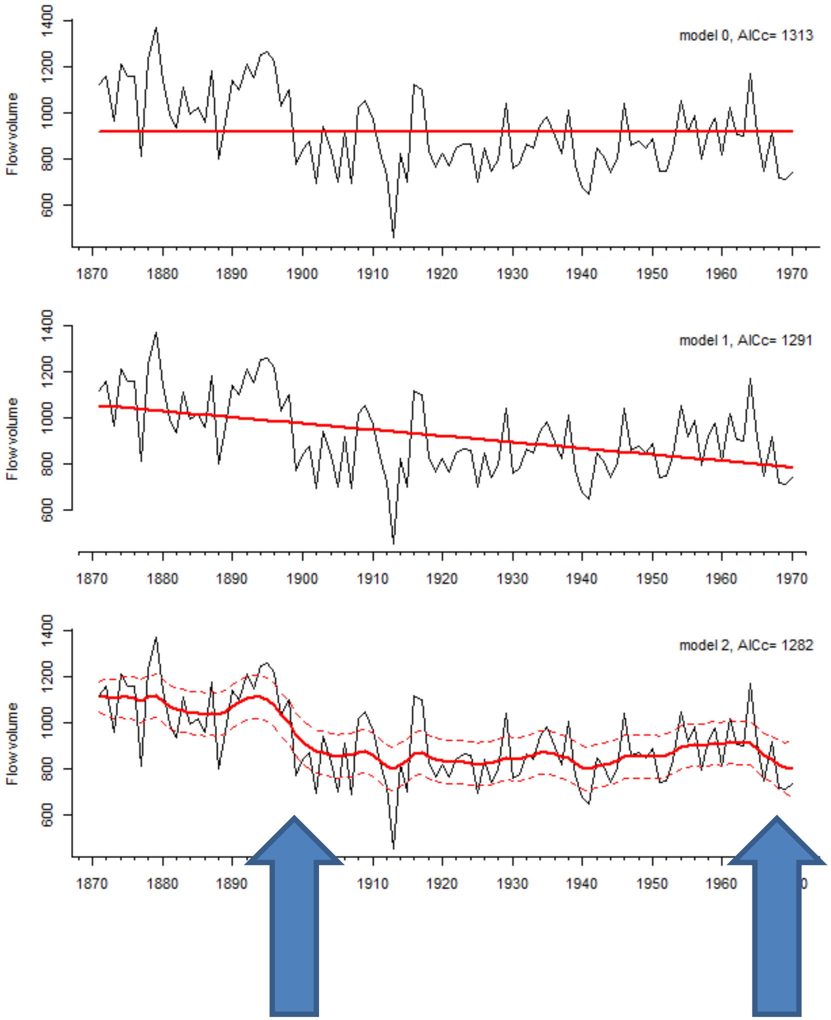
v(t) are model residuals



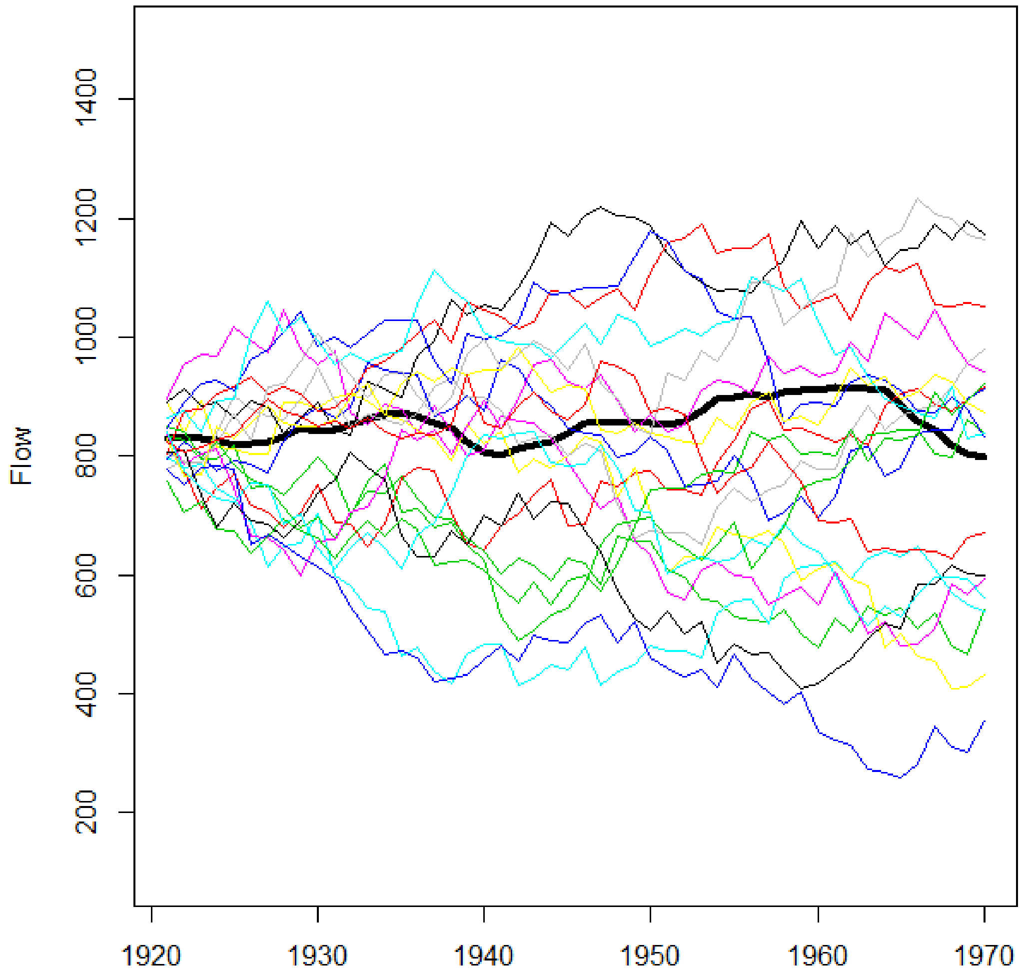
w(t) are state residuals

Even our ‘best’ model is missing something…

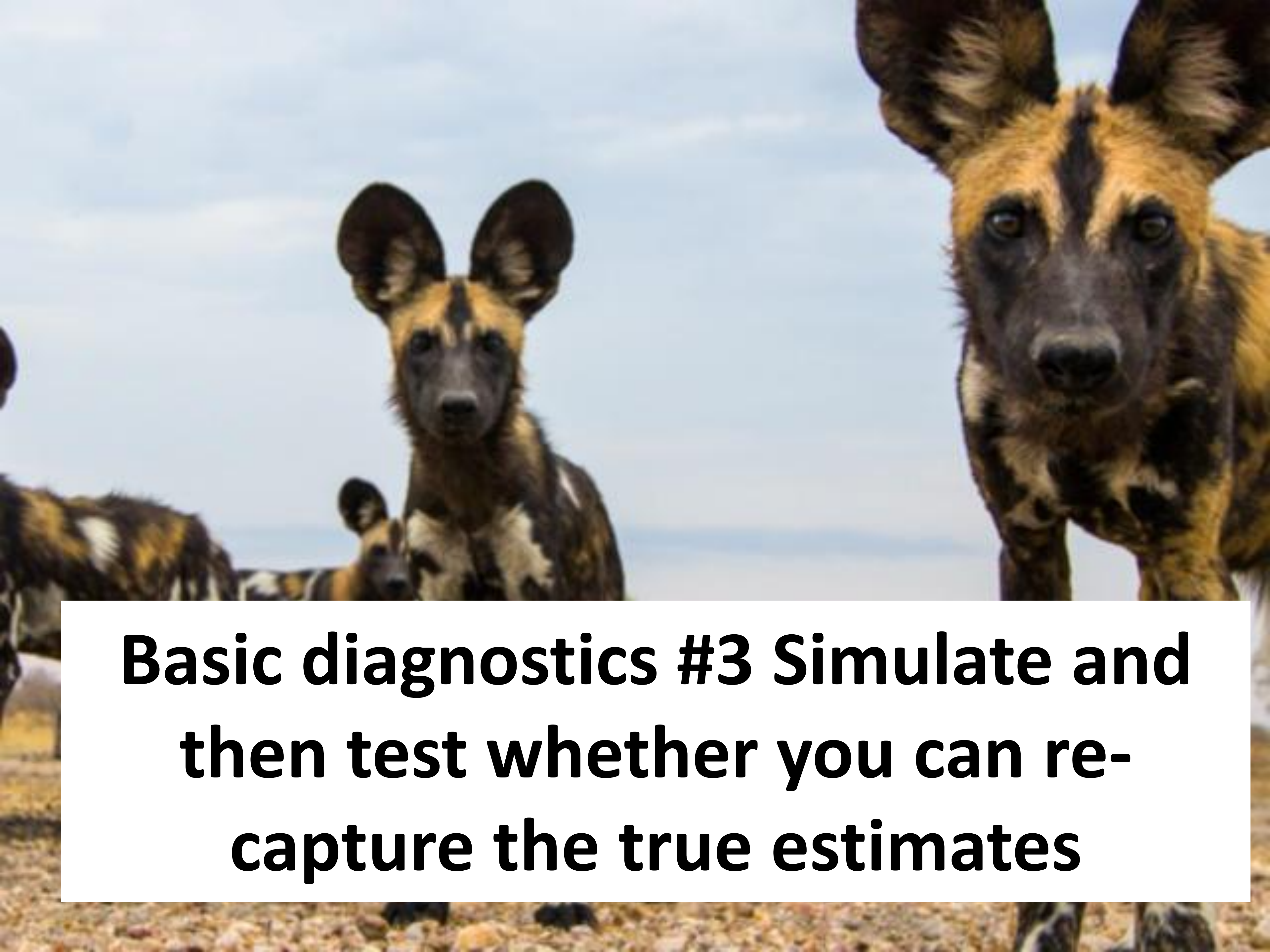




Basic diagnostics #2: Simulate from your estimated model and compare to the data.



Black line is the estimated state from model 2



Basic diagnostics #3 Simulate and then test whether you can re-capture the true estimates

Thursday lecture: multivariate state-space

$$\begin{bmatrix} y_{1,t} \\ y_{2,t} \\ y_{3,t} \\ y_{4,t} \\ y_{5,t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_{JF,t} \\ x_{N,t} \\ x_{S,t} \end{bmatrix} + \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} + \begin{bmatrix} \eta_{1,t} \\ \eta_{2,t} \\ \eta_{3,t} \\ \eta_{4,t} \\ \eta_{5,t} \end{bmatrix}$$

Thursday lab: fitting univariate and multivariate state-space models