

Appendix 1

Description of three exponential growth models allowing unequal intervals in the time series, with computer program in R

Here we first describe the assumptions and statistical methods entailed in fitting the EGOE and EGN models to data, and then describe the EGSS model, including a novel extension of its statistical inferences to accommodate unequal time intervals in the trajectory of abundance observations. As in the main text, we use the statistical notation convention of lower case to denote data (a particular outcome of a stochastic model) as well as constants, and upper case to denote the stochastic process (random variable) that generates data. So, we write $n(0)$, $n(t_1)$, ..., $n(t_q)$ for a recorded time series of population abundances (the data) at times 0 ($= t_0$), t_1 , ..., t_q , and $N(t)$ for a random population abundance at time t with some associated probability distribution.

EGOE model

We first consider the case where the population abundances in the time series ($n(0)$, $n(t_1)$, ..., $n(t_q)$) contain observation or sampling error but negligible process variance in the form of environmental or individual variation. The lognormal distribution is an often reasonable model of observation error because it reflects the heterogeneity of ecological sampling conditions (Dennis et al. 2006). Let $x(t) = \ln n(t)$, and let $Y(t_i)$ be a value of $x(t)$ estimated or observed with error at time t_i . Under these assumptions, we can write the EGOE model as

$$X(t) = x(0) + (\ln \lambda)t \quad (A1)$$

$$Y(t_i) = x(t_i) + F_i \quad (A2)$$

where F_i has a normal distribution with mean 0 and variance τ^2 ($F_i \sim \text{normal}(0, \tau^2)$). Generally, pure observation or sampling error is independent across sampling occasions, so F_0, F_1, \dots, F_q can be assumed to be uncorrelated. Presence of autocorrelated errors would tend to suggest that the observations are influenced by stochastic process noise in the population in addition to observation error and would warrant use of a different model. An equivalent way of writing the model for $Y(t_i)$ is

$$Y(t_i) \sim \text{normal}(\beta\mu t_i + \tau^2), \quad (A3)$$

where $\beta = x(0)$ and $\mu = \ln \lambda$, with $Y(0), Y(t_1), \dots, Y(t_q)$ independent. The parameter μ is the trend parameter and can be interpreted as the expected difference in observations separated by one time unit:

$$E[Y(t) - Y(t-1)] = \mu \quad (A4)$$

The variance of such a difference would be $2\tau^2$.

This statistical model for log-scale linear trend and observation-error-only is that of ordinary linear regression. For convenience we denote the time series of estimated log-abundances data by $y(0) = y_0, y(t_1) = y_1, \dots, y(t_q) = y_q$. The likelihood function for the unknown parameters β, μ , and τ^2 is a product of normal probability density functions:

$$L(\beta, \mu, \tau^2) = \prod_{i=0}^q (\tau^2 2\pi)^{-1/2} \exp \left\{ -\frac{[y_i - (\beta + \mu t_i)]^2}{2\tau^2} \right\} \quad (A5)$$

This likelihood function is identical to the likelihood for a linear regression using the y_i 's as the response variable values and the t_i 's as the predictor variable values. The maximum likelihood (ML) estimates for β and μ are the values that jointly maximize the likelihood and are the familiar least squares estimates; that is, trend (μ) is estimated by the slope of the linear regression. The bias-corrected ML estimate, $\hat{\tau}^2$, of the variance parameter is the sum of squared model residuals divided by $q-1$ (mean squared error in most regression packages), the total number of observations being $q+1$.

The confidence interval is calculated as $\hat{\mu} \pm t_{\alpha/2, q-1} \text{SE}(\hat{\mu})$, where $t_{\alpha/2, q-1}$ is the $100[1 - (\alpha/2)]$ th percentile of a student's t -distribution with $q-1$ degrees of freedom and $\text{SE}(\hat{\mu})$ is:

$$\text{SE}(\hat{\mu}) = \sqrt{\frac{\hat{\tau}^2}{\sum_{i=0}^q (t_i - \bar{t})^2}} \quad (A6)$$

If the data production process justifies use of the EGOE model, that is, if stochastic variability in the underlying population can be assumed negligible, then the statistical analysis has some conveniences. All the inferences from standard linear regression, such as confidence intervals and hypothesis tests, remain valid. The point estimate and confidence interval boundaries for λ are found by exponentiating those for μ . In addition, if population abundance is appropriately indexed, that is, if the observation process has a mean proportional to population abundance, or log-abundance is estimated with an additive sampling bias (if for instance a consistent proportion of animals elude sampling), then the bias or index constant will be absorbed into the intercept parameter β . The trend estimate will remain the same.

EGPN model

The EGN model is the “diffusion process” model of Dennis et al. (1991). The model assumes that during any small time interval dt , the population's log-scale growth rate experiences a random perturbation due to environmental variability:

$$d \ln N(t) = (\ln \lambda) dt + dB(t) \quad (A7)$$

or

$$dX(t) = (\ln \lambda)dt + dB(t) \quad (A8)$$

Here $dB(T) \sim \text{normal}(0, \sigma^2 dt)$. The random quantity $dB(t)$ is an increment of a Brownian motion stochastic process over a small time interval, with the correlation between $dB(t_i)$ and $dB(t_j)$ equal to 0 if $t_i \neq t_j$, and σ^2 is a positive constant. Written this way as a 'stochastic differential equation,' Eq. A8 can be regarded as just a recipe for simulating a trajectory of population abundance for specified values of λ and σ^2 (for instance, Allen 2003, Higham 2001): (1) Over a tiny time interval dt , generate an increment $dX(t)$ of log-abundance from a normal distribution with a mean of $(\ln \lambda)dt$ and a variance of $\sigma^2 dt$. (2) Update log-population abundance as $X(t) + dX(t)$ and update time as $t + dt$. (3) Return to step (1) and repeat the process for a new time interval, generating a new normal random number uncorrelated with the previous one.

The process $X(t)$ is the same as Brownian motion with drift (Dennis et al. 1991); the original scale process $N(t) = \exp(X(t))$ is often termed geometric Brownian motion, especially in investment analysis (Ross 2002). The model is a time series model that induces dependence among the population abundances $N(0)$, $N(t_1)$, ..., $N(t_q)$. However, the process $N(t)$ (along with $X(t)$) is a Markov process, that is, the statistical properties of the distribution of $N(t + s)$ given the value of $N(t)$ do not depend on observations previous to $N(t)$. Furthermore, it can be shown that the conditional distribution of $X(t + s)$ given that $X(t)$ is fixed at the value $x(t)$ is a normal distribution:

$$X(t + s) | \{X(t) = x(t)\} \sim \text{normal}(x(t) + \mu, \sigma^2 s) \quad (A9)$$

where $\mu = \ln \lambda$ and $s \geq 0$. In discrete time, with equal time intervals between observations, the process $X(t)$ is a form of nonstationary autoregressive model that has served as the null hypothesis in statistical tests of density dependence (Dennis and Taper 1994). Our trend parameter defined by $\mu = \ln \lambda$ in the EGP model is interpreted as the expected change of a population's log-abundance in one time unit. For the EGSS model, the geometric mean of $N(t)$ given by

$$\exp\{E[\log N(t)]\} = \exp\{E[X(t)]\} = \exp(\mu t) \quad (A10)$$

characterizes 'typical' sample paths of $N(t)$ better than does the mean population size

$$E[N(t)] = \exp\left[\left(\mu + \sigma^2/2\right)t\right] \quad (A11)$$

The geometric mean happens to be the same as the median for the highly skewed lognormal distribution of population size. The quantity e^{μ} gives a better portrait of the growth rate of the bulk of the sample paths than does $e^{\mu + \sigma^2/2}$ (Tuljapurkar 1989, Dennis et al. 1991). The environmental 'process noise' is additive on the logarithmic scale and produces proportional variability at all population abundances (unlike 'demographic' process noise which produces essentially deterministic behavior at large population abundances). Unlike observation error, process noise causes stochastic fluctuations regardless of whether or not the population is observed.

The observations x_0, x_1, \dots, x_q recorded at times $0, t_1, \dots, t_q$ are now assumed to be log-transformed population census values (i.e. no observation error). The likelihood function used for this model is the joint probability density of x_1, \dots, x_q , given the starting population of x_0 . Due to the above-mentioned Markov property, the likelihood function can be conveniently written as a product of conditional normal densities:

$$L(\mu, \sigma^2) = \prod_{j=1}^q \left(\sigma^2 s_j 2\pi \right)^{-1/2} \exp \left\{ - \frac{[x_j - (x_{j-1} + \mu s_j)]^2}{2\sigma^2 s_j} \right\} \quad (A12)$$

Here $s_j = t_j - t_{j-1}$. This is the likelihood function for the log-transformed abundances; the likelihood displayed in Dennis et al. (1991) is a product of lognormal densities for the untransformed abundances. Either likelihood yields the same parameter estimates. Here the initial population abundance is not an unknown parameter, because there is no observation error in the model. As described by Dennis et al. (1991), the ML estimates for μ and σ^2 can be calculated with formulas (Eq. 24–26 in Dennis et al. 1991) or by simple linear regression with the intercept fixed at zero using the transformed values $(x_j - x_{j-1})/\sqrt{s_j}$ as the response variable,

and the transformed time intervals $\sqrt{s_j}$ as the predictor variable. The resulting slope estimate is the ML estimate of μ , and the mean squared error is the bias-corrected ML estimate of σ^2 . The time intervals (s_1, s_2, \dots) do not need to be equal to obtain the ML estimates of μ and σ^2 .

Note that this maximum likelihood estimator of trend reduces (see equation 24 in Dennis et al. (1991) to a form using only the first ($n(t_0)$) and last ($n(t_q)$) abundance estimates in the time series, and the total duration of the survey (i.e. the time interval between the first and last observations):

$$\hat{\mu} = \frac{1}{t_q - t_0} \ln \left(\frac{n(t_q)}{n(t_0)} \right) \quad (A13)$$

All counts are used, however, to estimate the variance.

The confidence interval is calculated as $\hat{\mu} \pm t_{\alpha/2, q-1} SE(\hat{\mu})$, where $t_{\alpha/2, q-1}$ is the $100[1 - (\alpha/2)]$ th percentile of a student's t -distribution with $q-1$ degrees of freedom and $SE(\hat{\mu})$ is:

$$SE(\hat{\mu}) = \sqrt{\frac{\hat{\sigma}^2}{t_q - t_0}} \quad (A14)$$

EGSS model, extended to unequal time intervals between observations

An exponential growth model with both process noise and observation error can be constructed by combining the EGOE and the EGP model, creating a state space model with an unobserved stochastic population component, $X(t)$, and an observed or estimated component, $Y(t)$:

$$DX(t) = (\ln \lambda)dt + dB(t) \quad (A15)$$

$$Y(t_i) = X(t_i) + F_i \quad (A16)$$

Here $dB(t) \sim \text{normal}(0, \sigma^2 dt)$ and $F_i \sim \text{normal}(0, \tau^2)$, with no auto- or cross-correlations. In other words, the unobserved population is governed by the EGP model, but the data are created with observation error as in the EGOE model. The EGSS model was proposed by Holmes (2001), and parameter estimation was studied by Lindley (2003), Staples et al. (2004) and Dennis et al. (2006).

These authors defined the model in discrete time, with equal intervals between samples. Equation A15 and A16 incorporate the continuous time version of $X(t)$ and constitute an extension of the EGSS model to unevenly spaced sampling times.

In simulations, the ML parameter estimates for the EGSS model tend to have a persistent small-sample bias (Staples et al. 2004). An improvement to ML estimation can be made, akin to restricted maximum likelihood (REML) estimation in mixed effects models. Staples et al. (2004) showed that REML estimates based on second differences of the observations were superior to ML estimates for the EGSS model; their ML and REML estimates, however, were based on equally spaced time intervals between observations. The ML and REML estimates are extended to unequally spaced sampling times as follows. Staudenmayer and Buonaccorsi (2006) provide a more theoretical treatment.

The model has four unknown parameters: β ($= x_0$), μ ($= \ln \lambda$), σ^2 and τ^2 . The observations $Y(0), Y(t_1), \dots, Y(t_q)$ are dependent but not Markovian, and they have a joint multivariate normal distribution in which the mean of $Y(t)$ is $\beta + \mu t$, the variance of $Y(t)$ is $\sigma^2 t + \tau^2$, and the covariance of $Y(t)$ and $Y(t + s)$ is $\sigma^2 t$, for any times $t, s \geq 0$. For ML estimation, the multivariate normal log-likelihood function given by

$$\ln L(\beta, \mu, \sigma^2, \tau^2) = -\frac{(q+1)}{2} \ln(2\pi) - \frac{1}{2} \ln(|V|) - \frac{1}{2} (y - m)' V^{-1} (y - m) \quad (A17)$$

is used. Here the data values y_0, y_1, \dots, y_q recorded at times $0, t_1, \dots, t_q$ are the elements of the column vector y , the means $\beta, \beta + \mu t_1, \beta + \mu t_2, \dots$ are the elements in the column vector m , the variances $\tau^2, \sigma^2 t_1 + \tau^2, \sigma^2 t_2 + \tau^2, \dots$ are the elements on main diagonal in the matrix V , and the (i, j) th and (j, i) th covariance elements in V are both $\sigma^2 t_i$, where $t_i < t_j$:

$$V = \begin{bmatrix} \tau^2 & 0 & 0 & 0 & \dots \\ 0 & \sigma^2 t_1 + \tau^2 & \sigma^2 t_1 & \sigma^2 t_1 & \sigma^2 t_1 \\ 0 & \sigma^2 t_1 & \sigma^2 t_2 + \tau^2 & \sigma^2 t_2 & \sigma^2 t_2 \\ 0 & \sigma^2 t_1 & \sigma^2 t_2 & \sigma^2 t_3 + \tau^2 & \sigma^2 t_3 \\ \vdots & \sigma^2 t_1 & \sigma^2 t_2 & \sigma^2 t_3 & \ddots \end{bmatrix} \quad (A18)$$

The ML parameter estimates are calculated from the log-likelihood function with numerical maximization (for instance, with the 'optim' function in R; R Core Development Team 2006).

The REML estimates are found by rescaling the differences of the observations. The scaled first differences of the observations, defined by

$$W_i = \frac{1}{s_i} [Y(t_i) - Y(t_{i-1})], i = 1, 2, \dots, q \quad (A19)$$

with $s_i = t_i - t_{i-1}$, have a multivariate normal distribution with a constant mean vector equal to $j\mu$ (j is a column vector of ones), that is, $E(W_i) = \mu$. The variance-covariance matrix V_1 of the W_i 's is found as the matrix product

$$V_1 = D_1 V D_1'$$

where

$$D_1 = \begin{bmatrix} -\frac{1}{s_1} & \frac{1}{s_1} & 0 & 0 \\ 0 & -\frac{1}{s_2} & \frac{1}{s_2} & 0 \\ 0 & 0 & \ddots & \ddots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (A20)$$

is a $q \times (q+1)$ matrix with all elements zero except for those on the two long diagonals, and with V being the variance-covariance matrix of the $Y(t_i)$'s. By carrying out the matrix multiplications we find the resulting variances and covariances of the W_i 's to be: $V(W_i) = (\sigma^2/s_i) + (2\tau^2/s_i^2)$, $\text{Cov}(W_i, W_{i+1}) = -\tau^2/(s_i s_{i+1})$, $\text{Cov}(W_i, W_{i+k}) = 0$ ($k \geq 2$). Note that one parameter, the initial condition β , has been eliminated from the distribution of the W_i 's. When the intervals s_i are equal, the model for the W_i 's is equivalent to a normal linear mixed model in which the variance-covariance matrix has constant main diagonal, equivalent and constant subdiagonals, and zeros elsewhere. As Staples et al. (2004) noted, existing software packages for linear mixed models often allow such a 'banded Toeplitz' structure to be specified, and so REML estimates of EGSS parameters can be readily obtained (programs provided by Staples et al. 2004). When the intervals s_i are unequal, however, the variances and covariances of the W_i 's are unequal as well. It is not clear how to adapt current software packages for linear mixed models to accommodate the varied structure. Instead, REML estimates for the EGSS model with varying time intervals are obtained by numerical maximization.

The REML estimates are constructed from the second differences of the observations (first differences of the W_i 's):

$$U_i = W_{i+1} - W_i, i = 1, 2, \dots, q-1 \quad (A21)$$

It can be shown that U_1, U_2, \dots, U_{q-1} have a multivariate normal distribution in which the mean of each U_i is 0, and the variance-covariance matrix is found as

$$V_2 = D_2 V_1 D_2' = D_2 D_1 V D_1' D_2' \quad (A22)$$

where

$$D_2 = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & \ddots & \ddots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (A23)$$

is an ordinary $(q-1) \times q$ differencing matrix. Let

$$u_i = w_{i+1} - w_i = \frac{1}{s_{i+1}} (y_{i+1} - y_i) - \frac{1}{s_i} (y_i - y_{i-1}), i = 1, 2, \dots, q-1 \quad (A24)$$

be the twice-differenced data. The log-likelihood for the REML estimates is then given by

$$\ln L(\sigma^2, \tau^2) = -\frac{(q-1)}{2} \ln(2\pi) - \frac{1}{2} \ln(|V_2|) - \frac{1}{2} u' V_2^{-1} u \quad (A25)$$

in which u is a column vector containing the values u_1, u_2, \dots, u_{q-1} , and V_2 is the variance-covariance matrix given above. Now

two parameters, β and μ , have been eliminated by the differencing; however, the REML log-likelihood must still be maximized numerically for the REML estimates $\hat{\sigma}^2$ and $\hat{\tau}^2$. The estimate of the trend parameter μ then is calculated as

$$\hat{\mu} = \frac{j'V_1^{-1}w}{j'V_1^{-1}j} \quad (A26)$$

where w is the column vector containing w_1, w_2, \dots, w_n , with the elements in V_1 (the variance-covariance matrix for the \tilde{W}_i 's) evaluated using the REML estimates $\hat{\sigma}^2$ and $\hat{\tau}^2$. The variance in the trend estimate is:

$$\text{Var}(\hat{\mu}) = \frac{1}{j'V_1^{-1}j} \quad (A27)$$

and this variance can be estimated using the REML estimates $\hat{\sigma}^2$ and $\hat{\tau}^2$. Finally, the estimate of the initial condition β becomes

$$\hat{\beta} = \frac{j'V^{-1}(y - t\hat{\mu})}{j'V^{-1}j} \quad (A28)$$

Here $t = [t_0, t_1, \dots, t_n]'$, with the elements in V (the variance-covariance matrix for the $Y(t_i)$'s) evaluated using the REML estimates $\hat{\sigma}^2$ and $\hat{\tau}^2$.

For the numerical maximizations required for ML or REML estimates, initial parameter values are required for starting the iterations. Ideally, for a given data set the ML or REML estimates should be verified by trying many initial parameter values, because the EGSS model can produce likelihoods with multiple local maxima. However some strategy for automating the calculation of initial values is important for processing many data sets or for repetitive fitting techniques such as bootstrapping. One possibility is to fit both the EGOE and EGP models and then cut in half the resulting estimates of τ^2 and σ^2 . Another possibility is to use the one-step covariance of the vector of first differences \tilde{W}_i to estimate an initial value for τ^2 (see Eq. 33 in Dennis et al. 2006) and then use the variance of \tilde{W}_i to estimate an initial value for σ^2 (Eq. 32 in Dennis et al. 2006). Also, a standard numerical trick to prevent negative parameter values in iterations is to use transformed parameters, i.e. $\sigma^2 = \exp(\theta)$ and $\tau^2 = \exp(\delta)$, and maximize log-likelihoods for θ and δ as real-valued parameters.

An asymptotic $100(1 - \alpha)\%$ confidence interval is calculated as $\hat{\mu} \pm z_{\alpha/2} \text{SE}(\hat{\mu})$ where $z_{\alpha/2}$ is the $100[1 - (\alpha/2)]$ th percentile of the standard normal distribution, and $\text{SE}(\hat{\mu})$ is:

$$\text{SE}(\hat{\mu}) = \sqrt{\text{Var}(\hat{\mu})} \quad (A29)$$

If a statistical test for zero trend is desired, we suggest using $\text{SE}(\hat{\mu})$ and a standard normal percentile in an equivalence testing framework (Dixon and Pechmann 2005). The null hypothesis is that a substantial trend is present (i.e. μ is outside of a fixed, specified interval containing zero), and the alternative hypothesis is that trend is negligible for practical purposes (μ is inside the specified interval containing zero). Dixon and Pechmann (2005) give further details about equivalence testing and illustrate the concept with tests for the trend parameter in the EGOE model.

Although finite samples in many models can lead to ML (or REML) estimates with skewed sampling distributions accompanied by finite-sample bias, leading to poor CI coverage, we found that the sampling distribution of the REML estimates of μ were symmetrical and normal. This suggests that μ is being estimated well and that the asymptotic theory of ML/REML estimation is providing useful approximations for CI construction.

References

- Allen, L. J. S. 2003. An introduction to stochastic processes with applications to biology. – Pearson Education, Upper Saddle River.
- Dennis, B. and Taper, M. L. 1994. Density dependence in time series observations of natural populations: estimation and testing. – *Ecol. Monogr.* 64: 205–224.
- Dennis, B. et al. 1991. Estimation of growth and extinction parameters for endangered species. – *Ecol. Monogr.* 61: 115–143.
- Dennis, B. et al. 2006. Estimating density dependence, process noise, and observation error. – *Ecol. Monogr.* 76: 323–341.
- Dixon, P. M. and Pechmann, J. H. K. 2005. A statistical test to show negligible trend. – *Ecology* 86: 1751–1756.
- Higham, D. J. 2001. An algorithmic introduction to numerical simulation of stochastic differential equations. – *Siam Rev.* 43: 525–546.
- Holmes, E. E. 2001. Estimating risks in declining populations with poor data. – *Proc. Natl Acad. Sci. USA* 98: 5072–5077.
- Lindley, S. T. 2003. Estimation of population growth and extinction parameters from noisy data. – *Ecol. Appl.* 13: 806–813.
- Ross, S. M. 2002. An elementary introduction to mathematical finance: options and other topics (2nd ed.). – Cambridge Univ. Press.
- Staples, D. F. et al. 2004. Estimating population trend and process variation for PVA in the presence of sampling error. – *Ecology* 85: 923–929.
- Staudenmayer, J. and Buonaccorsi, J. P. 2006. Measurement error in a random walk model with applications to population dynamics. – *Biometrics* 62: 1178–1189.
- Tuljapurkar, S. 1989. An uncertain life – demography in random environments. – *Theor. Popul. Biol.* 35: 227–294.

Computer program, in the R language, for calculating ML and REML estimates of parameters in the EGSS model

```
# Exponential Growth State Space model:
# Version Nov.28.2009
#
# R program for calculating maximum likelihood (ML) and restricted maximum
# likelihood (REML) estimates of unknown parameters for the Exponential
# Growth State Space (EGSS) model of stochastic population growth.
# The model is
#
#  $dX(t) = \mu \cdot dt + dB(t)$ 
#           with  $dB(t) \sim \text{normal}(0, \text{ssq} \cdot dt),$ 
#  $Y(t) = X(t) + F(t)$ 
#           with  $F(t) \sim \text{normal}(0, \text{tsq}).$ 
#
# Here  $X(t)$  is log-population abundance,  $Y(t)$  is observed or estimated value
# of  $X(t)$ ,  $x_0$ ,  $\mu$ ,  $\text{ssq}$ ,  $\text{tsq}$  are parameters. The parameter  $\text{ssq}$  is the
# variance of the process noise, and  $\text{tsq}$  is the variance of the observation
# error.
#
# The model takes population abundance  $N(t) = \exp(X(t))$  to be governed by
# a stochastic, density independent model, with the observed abundances
#  $O(t) = N(t) \cdot \exp(F(t))$  arising from lognormal sampling error.
#
# User provides time series of observed population abundances  $o(0)$ ,  $o(1)$ ,
# ...,  $o(q)$ , which are log-transformed by the program into  $y(0)$ ,  $y(1)$ , ...,
#  $y(q)$ , assumed to be a time series realization of  $Y(t)$ . Likelihood
# function of  $y(0)$ ,  $y(1)$ , ...,  $y(q)$  is that of a multivariate normal
# distribution. The observation times  $t_0$ ,  $t_1$ ,  $t_2$ , ...,  $t_q$  can have
# unequal intervals.
#
# Program computes initial parameter values for iterations. The program
# should be re-run for several sets of initial values, as the likelihood
# function for the model frequently has multiple local maxima, see program
# section 4.
#
# Alternative programs, for observation times with equal intervals,
# are available as an online appendix to Staples et al. (2004).
# See also Staudenmayer and Buonaccorsi (2006) for a more theoretical
# development.
#
# Program citations:
#   Dennis et al. 2006. Ecological Monographs.
#   Humbert et al. 2009. Oikos.
#   Staples et al. 2004. Ecology.
#   Staudenmayer and Buonaccorsi. 2006. Biometrics.
#
```

```

#-----
#           1. USER INPUT SECTION
#-----
#
# The best way to conduct these analyses is by preparing data in Excel, then
# saving as a text file.
#
# In Excel, in cell A1, type (Exactly) the word «Observed.t», without the
# quotes. Put the observed abundance in column A, starting with row 2.
# In cell B1, type (Exactly) the word «Time.t» without the quotes. Put the
# time step identifiers that correspond to the observed population size in
# column B, starting with row 2.
# Your first time step can be 0, or 1, or anything else (e.g. a year).
#
# ** NOTE: If you have years with no data, that is fine.
# Just omit the year and the associated abundance. However, if you have a
# year where you sampled but got zero abundance, you cannot use these
# approaches. **
#
# Once the data sheet is prepared in Excel, save it as on the C:\ drive
# as "c:\my_data.txt". It must be a tab delimited text file.
# Excel may try to name it my_data.txt.txt or my_data.txt.xls but don't
# let it! You can confirm that it is, indeed, a text file by double clicking
# on it and confirming that it opens with Notepad, not Excel.
#
# ** NOTE: You can change the drive or the name of the input file by
# changing the second next line. **

rm(list=ls(all=TRUE))          # Clears all objects from memory
my_data <- data.frame(read.table(«C:/my_data.txt», header=TRUE, sep=»\t»))
Observed.t <- my_data$Observed.t
Time.t <- my_data$Time.t
print.table(cbind(Observed.t,Time.t))

# OUTPUT: 2 Files will go the same place as the input file:
#     GRAPHICS                my_graph.png
#     INTERPRET OUTPUT        my_results.txt

# REST OF THIS SECTION CAN BE IGNORED IF USING EXCEL INPUT
# Example data below are American Redstart counts from North American
# Breeding Bird Survey, record # 02014 3328 08636, 1966-95 (Table 1 in
# Dennis et al. 2006).
# To run this example, comment out the 5 lines immediately
# above this paragraph, using the pound sign, and uncomment the 4 lines
# immediately below this paragraph, but do not change anything outside
# of the user input section. When you uncomment the lines below, remove
# only the pound signs at the left side of this document. Do not remove
# the pound signs after the semicolons.
# Observed.t=c(18,10,9,14,17,14,5,10,9,5,11,11,4,5,4,8,2,3,9,2,4,7,4,1,2,
# 4,11,11,9,6);      # No zeros! (With zeros, you must use another model)
# Time.t=c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,

```

```

# 25,26,27,28,29); # Initial time can be nonzero.

#-----
#          2. PROGRAM INITIALIZATION SECTION
#-----
library(MASS);          # Loads miscellaneous functions (ginv, etc.)
T.t=Time.t-Time.t[1];   # Time starts at zero.
Y.t=log(Observed.t);    # Log-transform the observations.
q=length(Y.t)-1;        # Number of time series transitions, q.
qp1=q+1;                # q+1 gets used a lot, too.
S.t=T.t[2:qp1]-T.t[1:q]; # Time intervals.
m=rep(1,qp1);           # Will contain Kalman means for Kalman calculations.
v=rep(1,qp1);           # Will contain variances for Kalman calculations.
sink(file = «C:/my_results.txt», append = FALSE, type = «output», split = T)
                        # Tells the program where to write the output
# ** CHANGE LINE ABOVE IF YOU WANT RESULTS TO GO SOMEWHERE OTHER THAN C:/ DRIVE

#-----
#          3. SECTION FOR DEFINING ML & REML LOG-LIKELIHOODS
#-----

# ML objective function «negloglike.ml» is negative of log-likelihood;
# the Nelder-Mead optimization routine in R, «optim», is a minimization
# routine. The ML objective function uses equations 24-26 from Dennis et
# al. (2006). The three function arguments are: theta, vector of
# parameters (transformed to the real line), yt, vector of time series
# observations, and tt, vector of observation times.
negloglike.ml=function(theta,yt,tt)
{
  muu=theta[1];
  sigmasq=exp(theta[2]);      # Constrains ssq > 0.
  tausq=exp(theta[3]);        # Constrains tsq > 0.
  xzero=theta[4];
  q=length(yt)-1;
  qp1=q+1;
  yt=matrix(yt,nrow=qp1,ncol=1);
  vx=matrix(0,qp1,qp1);
  for (ti in 1:q)
  {
    vx[(ti+1):qp1,(ti+1):qp1]=matrix(1,1,(qp1-ti))*tt[ti+1];
  }
  Sigma.mat=sigmasq*vx;
  Itausq=matrix(rep(0,(qp1*qp1)),nrow=q+1,ncol=q+1);
  diag(Itausq)=rep(tausq,q+1);
  V=Sigma.mat+Itausq;
  mu=matrix((xzero+muu*tt),nrow=qp1,ncol=1);
  ofn=((qp1/2)*log(2*pi)+(0.5*log(det(V)))+(0.5*(t(yt-mu)%*%ginv(V)%*%(yt-mu))));
  return(ofn);
}

```

```
# REML objective function „negloglike.reml“ is negative of log-likelihood
# for second differences of the log-scale observations. The REML objective
# function uses equations A18-A25 from Humbert et al. (2009). The three
# function arguments are: theta, vector of parameters (transformed to the
# real line), yt, vector of time series observations (log scale), and
# tt, vector of observation times. Function performs the differencing.
```

```
negloglike.reml=function(theta,yt,tt)
{
  sigsq=exp(theta[1]);      # Constrains ssq > 0.
  tausq=exp(theta[2]);      # Constrains tsq > 0.
  q=length(yt)-1;
  qp1=q+1;
  vx=matrix(0,qp1,qp1);
  for (ti in 1:q)
  {
    vx[(ti+1):qp1,(ti+1):qp1]=matrix(1,1,(qp1-ti))*tt[ti+1];
  }
  Sigma.mat=sigsq*vx;
  Itausq=matrix(rep(0,(qp1*qp1)),nrow=q+1,ncol=q+1);
  diag(Itausq)=rep(tausq,q+1);
  V=Sigma.mat+Itausq;
  ss=tt[2:qp1]-tt[1:q];
  D1mat=cbind(-diag(1/ss),matrix(0,q,1))+cbind(matrix(0,q,1),diag(1/ss));
  D2mat=cbind(-diag(1,q-1),matrix(0,q-1,1))+
    cbind(matrix(0,q-1,1),diag(1,q-1));
  Phi.mat=D2mat%*%D1mat%*%V%*%t(D1mat)%*%t(D2mat);
  wt=(yt[2:qp1]-yt[1:q])/ss;
  ut=wt[2:q]-wt[1:q-1];
  ofn=(q/2)*log(2*pi)+(0.5*log(det(Phi.mat)))+(
    0.5*(ut%*%ginv(Phi.mat)%*%ut));
  return(ofn);
}
```

```
#-----
#           4. SECTION FOR CALCULATING EGOE AND EGPN ESTIMATES
#           (FOR USE AS INITIAL VALUES)
#-----
```

```
# The EGOE estimates
Ybar=mean(Y.t);
Tbar=mean(T.t);
mu.egoe=sum((T.t-Tbar)*(Y.t-Ybar))/sum((T.t-Tbar)*(T.t-Tbar));
x0.egoe=Ybar-mu.egoe*Tbar;
ssq.egoe=0;
Yhat.egoe=x0.egoe+mu.egoe*T.t;
tsq.egoe=sum((Y.t-Yhat.egoe)*(Y.t-Yhat.egoe))/(q-1);
```

```
# The EGPN estimates
Ttr=sqrt(S.t);
Ytr=(Y.t[2:qp1]-Y.t[1:q])/Ttr;
mu.egpn=sum(Ttr*Ytr)/sum(Ttr*Ttr);
Ytrhat=mu.egpn*Ttr;
```

```

ssq.egpn=sum((Ytr-Ytrhat)*(Ytr-Ytrhat))/(q-1);
tsq.egpn=0;
x0.egpn=Y.t[1];

# Initial values for EGSS are averages of EGOE and EGPn values
mu0=(mu.egoe+mu.egpn)/2;      # For ML only
ssq0=ssq.egpn/2;              # For ML and REML
tsq0=tsq.egoe/2;              # For ML and REML
x00=x0.egoe;                  # For ML only

# To set different initial values for iterations, enter manually a value
# after the equal sign of the concern parameter instead of the
# automatically generated value. Then run again the line and the program
# section 5 below.
# Initial values near the EGOE and EGPn models are good for exploring
# possible alternative local maxima. The values which produce the largest
# log-likelihood should be used. To see the log-likelihood for the REML
# estimates type:
# EGSSreml$value[1];
# See Dennis et al. 2006 for more details.

#-----
#           5. SECTION FOR CALCULATING ML & REML PARAMETER ESTIMATES
#-----

# The ML estimates.
EGSSml=optim(par=c(mu0,log(ssq0),log(tsq0),x00),
  negloglike.ml,NULL,method="Nelder-Mead",yt=Y.t,tt=T.t);
params.ml=c(EGSSml$par[1],exp(EGSSml$par[2]),exp(EGSSml$par[3]),
  EGSSml$par[4]);
lnlike.ml=-EGSSml$value[1];
AIC.egss=-2*lnlike.ml+2*length(params.ml);

mu.ml=params.ml[1];           # These are the ML estimates.
ssq.ml=params.ml[2];          #           --
tsq.ml=params.ml[3];          #           --
x0.ml=params.ml[4];           #           --

# The REML estimates.
EGSSreml=optim(par=c(log(ssq0),log(tsq0)),
  negloglike.reml,NULL,method="Nelder-Mead",yt=Y.t,tt=T.t);
params.reml=c(exp(EGSSreml$par[1]),exp(EGSSreml$par[2]))

ssq.reml=params.reml[1];      # These are the REML estimates.
tsq.reml=params.reml[2];      #           --

vx=matrix(0,qp1,qp1);
for (ti in 1:q)
{
  vx[(ti+1):qp1,(ti+1):qp1]=matrix(1,1,(qp1-ti))*T.t[ti+1];
}

```

```

Sigma.mat=ssq.reml*vx;
Itausq=matrix(rep(0, (qp1*qp1)),nrow=q+1,ncol=q+1);
diag(Itausq)=rep(tsq.reml,q+1);
V=Sigma.mat+Itausq;
Dlmat=cbind(-diag(1/S.t),matrix(0,q,1))+cbind(matrix(0,q,1),diag(1/S.t));
Vlmat=Dlmat%*%V%*%t(Dlmat);
W.t=(Y.t[2:qp1]-Y.t[1:q])/S.t;
j1=matrix(1,q,1);
Vlinv=ginv(Vlmat);
mu.reml=(t(j1)%*%Vlinv%*%W.t)/(t(j1)%*%Vlinv%*%j1);
j=matrix(1,qp1,1);
Vinv=ginv(V);
x0.reml=(t(j)%*%Vinv%*%(Y.t-mu.reml*T.t))/(t(j)%*%Vinv%*%j);
Var_mu.reml=1/(t(j1)%*%Vlinv%*%j1);          # Variance of mu
mu_hi.reml=mu.reml+1.96*sqrt(Var_mu.reml);    # 95% CI for mu
mu_lo.reml=mu.reml-1.96*sqrt(Var_mu.reml);    # --

# Calculate estimated population sizes for EGSS model
# with Kalman filter, for plotting.
#
# Choose ML or REML estimates here for calculating model values
# for plotting (by commenting out the unwanted, default is REML).
# mu=mu.ml; ssq=ssq.ml; tsq=tsq.ml; x0=x0.ml;
mu=mu.reml; ssq=ssq.reml; tsq=tsq.reml; x0=x0.reml;

m[1]=x0;          # Initial mean of Y(t).
v[1]=tsq;         # Initial variance of Y(t).

for (ti in 1:q)   # Loop to generate estimated population abundances
{
    # using Kalman filter (see equations 6 & 7,
    # Dennis et al. (2006)).
    m[ti+1]=mu+(m[ti]+((v[ti]-tsq)/v[ti])*(Y.t[ti]-m[ti]));
    v[ti+1]=tsq*((v[ti]-tsq)/v[ti])+ssq+tsq;
}

# The following statement calculates  $\exp\{E[X(t) \mid Y(t), Y(t-1), \dots, Y(0)]\}$ ;
# see equation 54 in Dennis et al. (2006).
Predict.t=exp(m+((v-tsq)/v)*(Y.t-m));

# Plot the data & model-fitted values
plot(Observed.t ~ Time.t,xlab=»time»,ylab=»population abundance»,
     type=»o»,lty=»solid»,pch=1,cex=1);
     # Population data are circles.
points(Predict.t ~ Time.t,type=»l»,lty=»dashed»,lwd=1);
     # Estimated abundances are dashed line.
legend(«top», c(«Observed.t»,»Predict.t»),lty=c(1,2),pch=c(«o»,»»),bty=»n»)
     # Graph legend

# Print the parameter estimates
parms.egoe=c(mu.egoe,ssq.egoe,tsq.egoe,x0.egoe); # Collect for printing
parms.egpn=c(mu.egpn,ssq.egpn,tsq.egpn,x0.egpn); # --

```

```

parms.reml=c(mu.reml,ssq.reml,tsq.reml,x0.reml); # --
parms.ml=c(mu.ml,ssq.ml,tsq.ml,x0.ml); # --
names=c("mu","ssq","tsq","x0"); # --
types=c("EGOE","EGPN","EGSS-ML","EGSS-REML"); # --

# Print stuff
matrix(cbind(parms.egoe,parms.egpn,parms.ml,parms.reml),
       nrow=4,ncol=4,byrow=TRUE,dimnames=list(types,names));

# Print CI, default is for EGSS-REML
matrix(cbind(mu_lo.reml,mu_hi.reml),nrow=1,ncol=2,byrow=TRUE,
       dimnames=list(„95% CI for MU“,c(„LO“,„HI“)));

# Print log-likelihood and AIC for EGSS ML
matrix(cbind(lnlike.ml,AIC.egss),nrow=1,ncol=2,byrow=TRUE,
       dimnames=list(„EGSS ML RESULTS“,c(„LN-LIKELIHOOD“,„AIC“)));

# Plot the data & model-fitted values to a png file
png(file = „C:/my_graph.png“) # Open a png file for plotting
# ** CHANGE LINE ABOVE IF YOU WANT RESULTS TO GO SOMEWHERE OTHER THAN C:/ DRIVE

plot(Observed.t ~ Time.t,xlab="time",ylab="population abundance",
     type="o",lty="solid",pch=1,cex=1);
# Population data are circles.
points(Predict.t ~ Time.t,type="l",lty="dashed",lwd=1);
# Estimated abundances are dashed line.
legend(„top“, c(„Observed.t“,„Predict.t“),lty=c(1,2),pch=c(„o“,„“),bty="n")
# Graph legend
graphics.off() # Close graphics file
sink() # Remove output diversion to results file so output will
# be sent back to the screen

```

Appendix 2

Results from some other parameter combinations

Showing that inferences from the main text Fig. 1 and 2 are robust and generalizable.

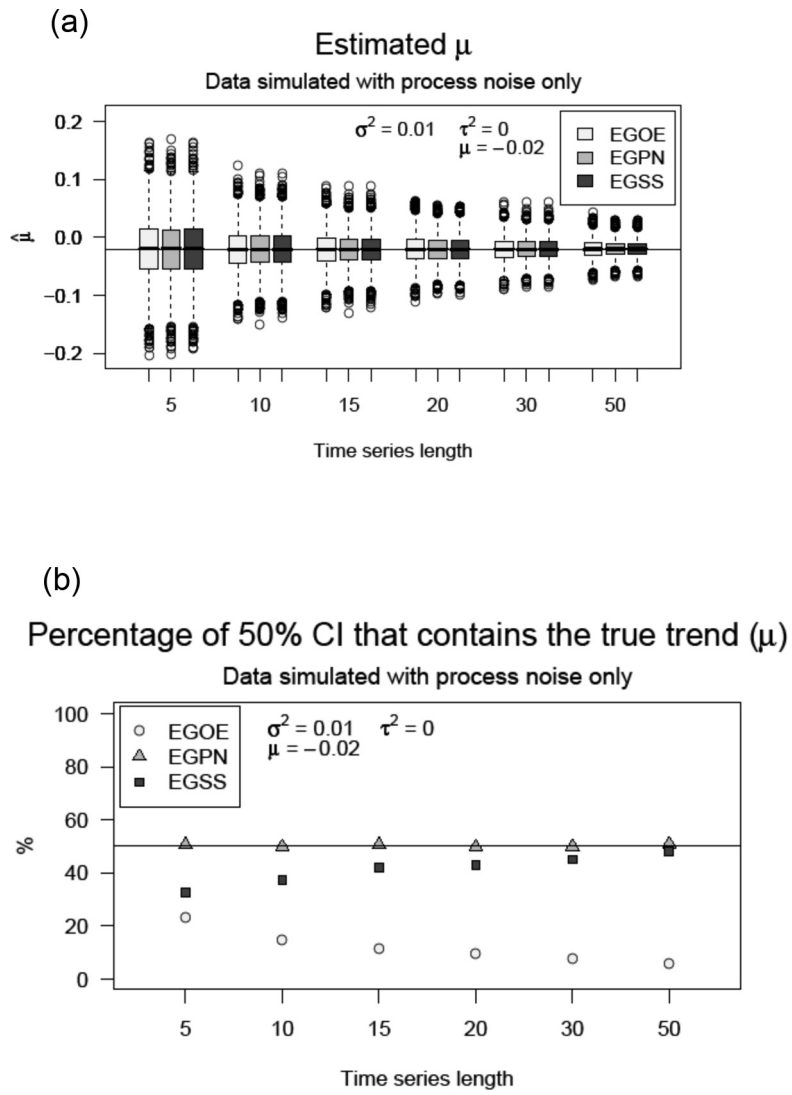


Figure A1a–b. Data with process noise only: $\mu = -0.02$, $\sigma^2 = 0.01$ and $\tau^2 = 0$.

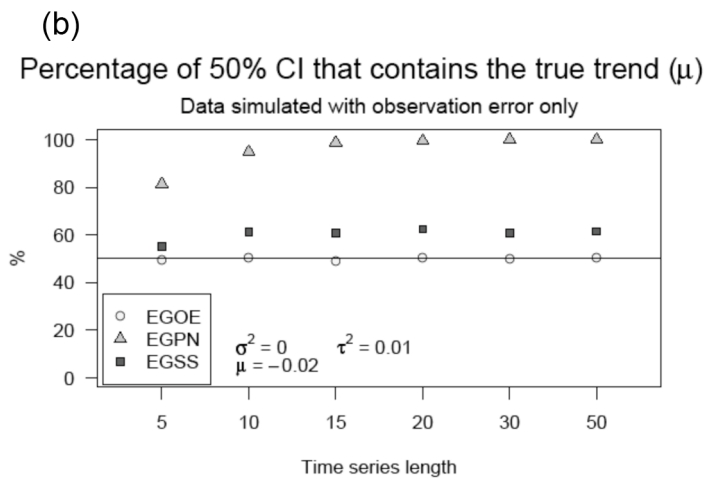
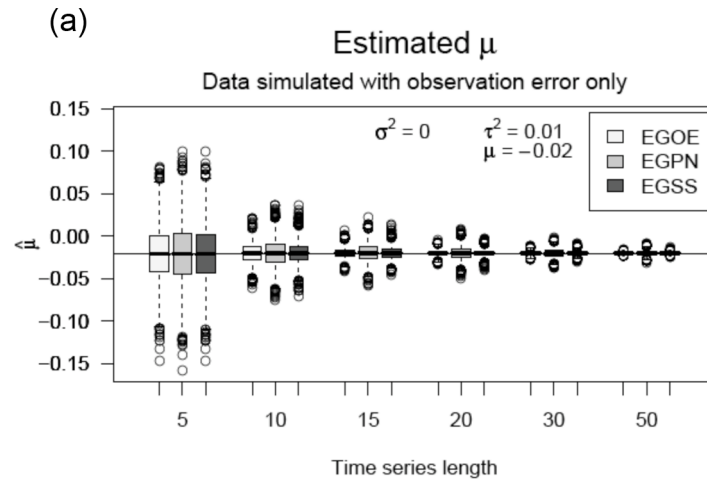
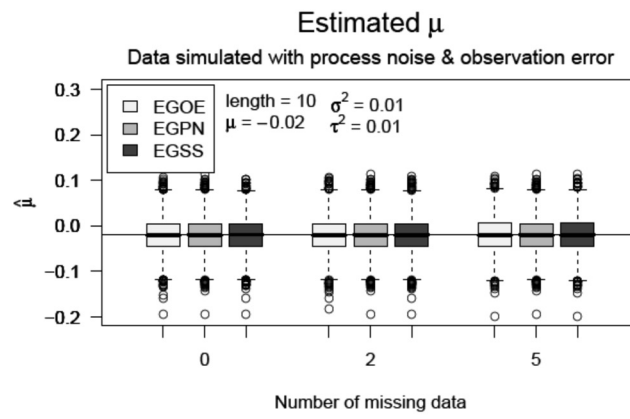


Figure A2a–b. Data with observation error only: $\mu = -0.02$, $\sigma^2 = 0$ and $\tau^2 = 0.01$.

(a)



(b)

Percentage of 50% CI that contains the true trend (μ)

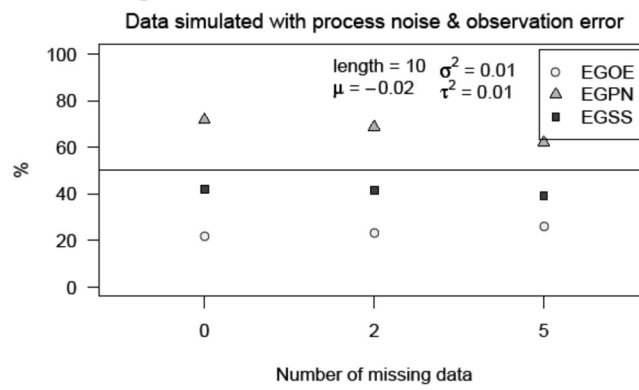
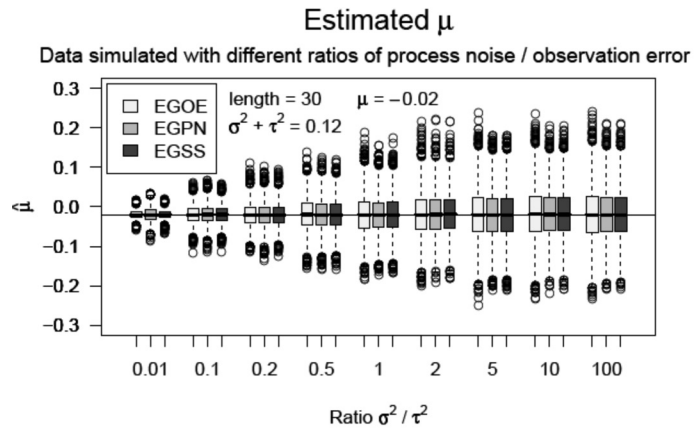


Figure 3a–b. Shorter time series: time series length = 10, $\mu = -0.02$, $\sigma^2 = 0.01$ and $\tau^2 = 0.01$.

(a)



(b)

Percentage of 50% CI that contains the true trend (μ)

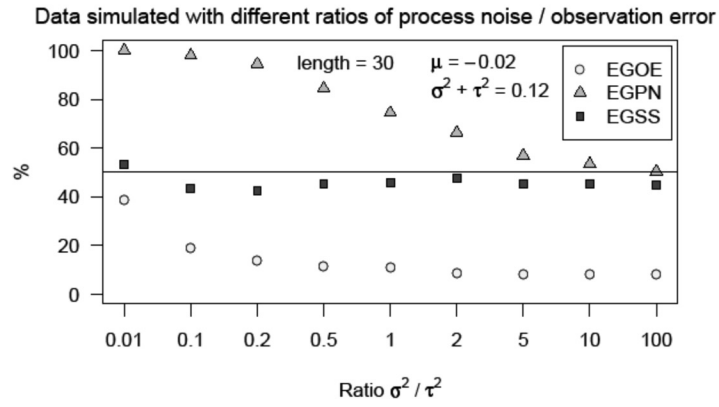
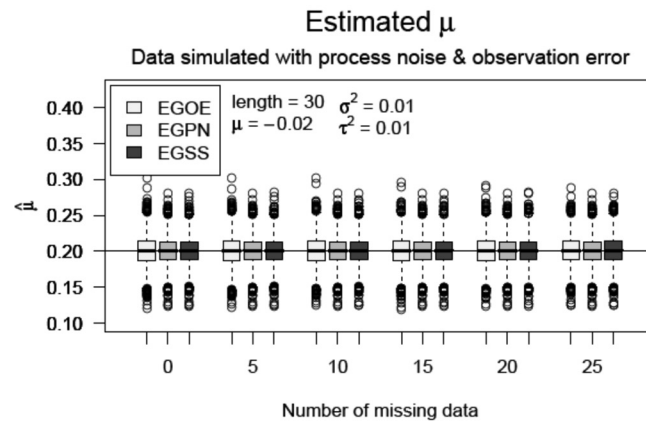


Figure 4a–b. Higher variance: time series length = 30, $\mu = 0$, $\sigma^2 + \tau^2 = 0.12$. Ratio of process (σ^2) to sampling (τ^2) variance ranging from 0.01 (trivial process variance) to 100 (trivial sampling variance).

(a)



(b)

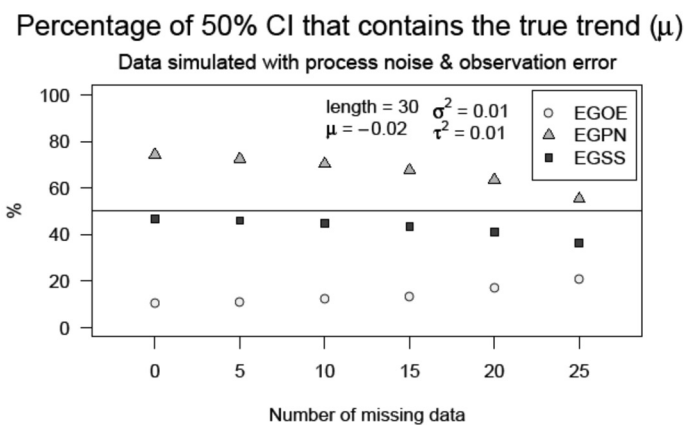


Figure 5a–b. Large positive trend: time series length = 30, $\mu = 0.20$, $\sigma^2 = 0.01$ and $\tau^2 = 0.01$.