Bayesian Inference and Data Assimilation

Prof. Dr.-Ing. Sebastian Reich

Universität Potsdam

12 April 2021

Content:

- Prolog: How to Produce Forecasts
- Quantification of Uncertainty
  - Introduction to Probability
  - Computational Statistics
  - Stochastic Processes
  - Bayesian Inference
- Bayesian Data Assimilation
  - Basic Data Assimilation
  - Modern Data Assimilation: The Ensemble Kalman Filter
  - Parameter Estimation and Model Selection

References:

Probabilistic Forecasting and Bayesian Data Assimilation, Sebastian Reich & Colin Cotter, Cambridge University Press, 2015

Data Assimilation – A Mathematical Introduction, Kody Law, Andrew Stuart & Konstantinos Zygalakis, Springer-Verlag, 2015

Prolog: How to Produce Forecasts

Forecasting future events is one of the major challenges for the human intellect.

- Financial markets
- Weather prediction
- Elections
- Robotics
- AlphaGo

Pierre Simon Laplace (Essai philosophique sur les probabilités, 1814):

We may regard the present state of the universe as the effect of its past and the cause of its future. An intellect which at a certain moment would know all forces that set nature in motion, and all positions of all its items of which nature is composed, if this intellect were also vast enough to submit these data to analysis, it would embrace in a single formula the movements of the greatest bodies of the universe and those of the tiniest atoms; for such an intellect nothing would be uncertain and the future just like the past would be present before its eyes.

1.1 Physical processes and observations

The ideal mathematical scenario:

Definition (Laplace's demon)

A state space $\mathbb{R}^{N_z}$, an initial state $z^0 \in \mathbb{R}^{N_z}$, an iteration

$$z^{n+1} = \Psi(z^n, t_n), \qquad t_{n+1} = t_n + \Delta t,$$

for $n \geq 0$ with step-size $\delta t > 0$,

We will also use the continuous-time interpolation

$$z_{\rm ref}(t) = z^n + (t - t_n)\frac{z^{n+1} - z^n}{\delta t} \quad \text{for} \quad t \in [t_n, t_{n_1}]$$

as our reference solution ("truth")

Example (Euler discretized ODE/ Lorenz-63)

Consider a time-dependent ODE

$$\frac{\mathrm{d}z}{\mathrm{d}t} = f(z) + g(t)$$

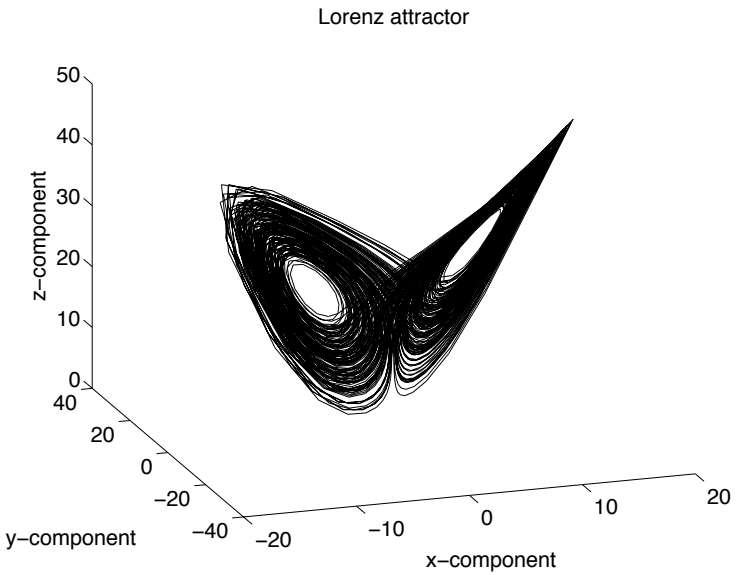
and its Euler discretization

$$z^{n+1} = z^n + \delta t \left(f(z^n) + g(t_n) \right) \quad =: \Psi(z^n, t_n)$$

The Lorenz-63 system is given by $z = (\mathrm{x}, \mathrm{y}, \mathrm{z})^{\mathrm{T}} \in \mathbb{R}^3$,

$$f(z) := \begin{pmatrix} \sigma(\mathrm{y} - \mathrm{x}) \\ \mathrm{x}(\rho - \mathrm{z}) - \mathrm{y} \\ \mathrm{xy} - \beta \mathrm{z} \end{pmatrix},$$

and parameter values $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$.



In reality we neither have access to the map Ψ, the initial state z^0, nor the implied reference trajectory $z_{\rm ref}(t)$.

Definition (Observations)

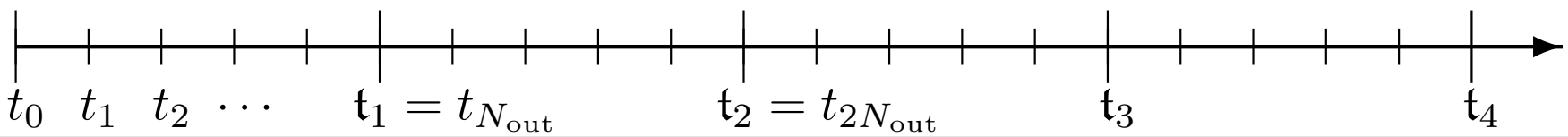
We assume that we can observe $z_{\rm ref}(t)$ via partial and noisy observations

$$y_{\rm obs}(\mathrm{t}_k) = h(z_{\rm ref}(\mathrm{t}_k)) + \sum_{i=1}^{I} \eta_i(\mathrm{t}_k)$$

at discrete time instances

$$\mathrm{t}_k = k\,\Delta t_{\rm out}, \qquad k \geq 1,$$

and $\Delta t_{\rm out} = \delta t N_{\rm out}$ for given integer $N_{\rm out} \geq 1$. Herer $h : \mathbb{R}^{N_z} \to \mathbb{R}^{N_y}$ is called the forward operator and the η_i's represent measurement errors.



Example (Partially observed Lorenz-63)

The forward operator is given by

$$\mathrm{x}_{\mathrm{ref}}(t) = h(z_{\mathrm{ref}}(t))$$

i.e., the first component of the state vector. The observed values are obtained from

$$\mathrm{x}_{\mathrm{obs}}(\mathrm{t}_k) = \mathrm{x}_{\mathrm{ref}}(\mathrm{t}_k) + \frac{1}{20}\sum_{i=1}^{20} \eta_i(\mathrm{t}_k).$$

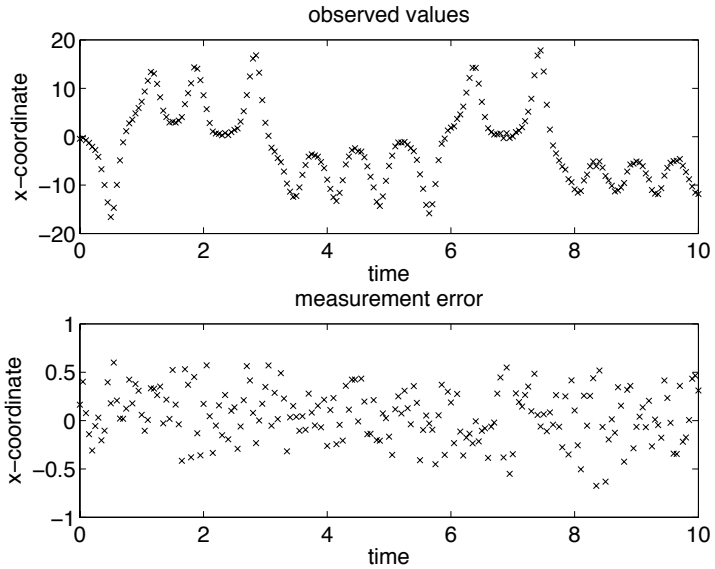


Figure: Observed values for the x-component and their measurement errors over the time interval $[0, 10]$ with observations taken every $\Delta t_{\mathrm{out}} = 0.05$ time units.

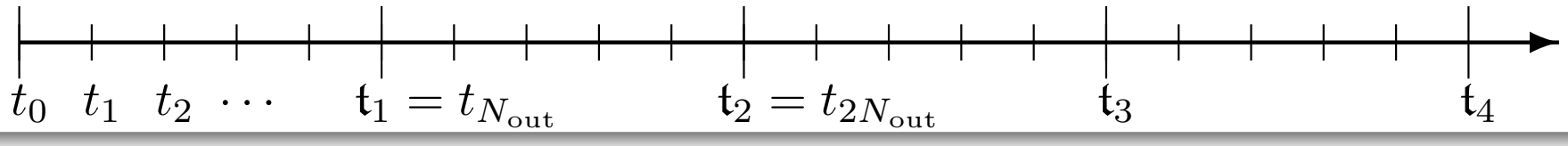
1.2 Data driven forecasting

Definition (Data driven forecasting)

We now assume that $N_{\rm obs}$ scalar observations $y_{\rm obs}(\mathrm{t}_k) \in \mathbb{R}$ at $\mathrm{t}_k = k\,\Delta t_{\rm out}$, $k = 1, 2, \ldots, N_{\rm obs}$, have been made at time intervals of $\Delta t_{\rm out}$. To define what we understand by a *forecast* or a *prediction*, we select a point in time t_{k_*} that we denote the *present*. Relative to t_{k_*}, we can define the *past* $t < \mathrm{t}_{k_*}$ and the *future* $t > \mathrm{t}_{k_*}$. A possible forecasting (or prediction) problem would be to produce an *estimate* for

$$y_{\rm ref}(t) := h(z_{\rm ref}(t))$$

with $t > \mathrm{t}_{k_*}$ and only observations from the past and present available.



Definition (Linear extrapolation)

We can fit a polynomial

$$q(t) = b_0 + b_1 t + b_2 t^2 + \cdots b_p t^p$$

of order p through available observations $y_{\rm obs}(\mathrm{t}_k)$, $k \leq k^*$. For example, linear interpolation ($p = 1$) leads to

$$\begin{aligned} q(t) &= y_{\rm obs}(\mathrm{t}_{k_*}) + (t - \mathrm{t}_{k_*}) \frac{y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*-1})}{\mathrm{t}_{k_*} - \mathrm{t}_{k_*-1}} \\ &= y_{\rm obs}(\mathrm{t}_{k_*}) + (t - \mathrm{t}_{k_*}) \frac{y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*} - \Delta t_{\rm out})}{\Delta t_{\rm out}}. \end{aligned}$$

Upon setting $t = \mathrm{t}_{k_*+1}$ we obtain the extrapolation formula

$$y_{\rm predict}(\mathrm{t}_{k_*+1}) := q(\mathrm{t}_{k_*+1}) = 2y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*-1}).$$

Example (Lorenz-63)

We consider linear extrapolation for the observed x-component of the Lorenz-63 model:

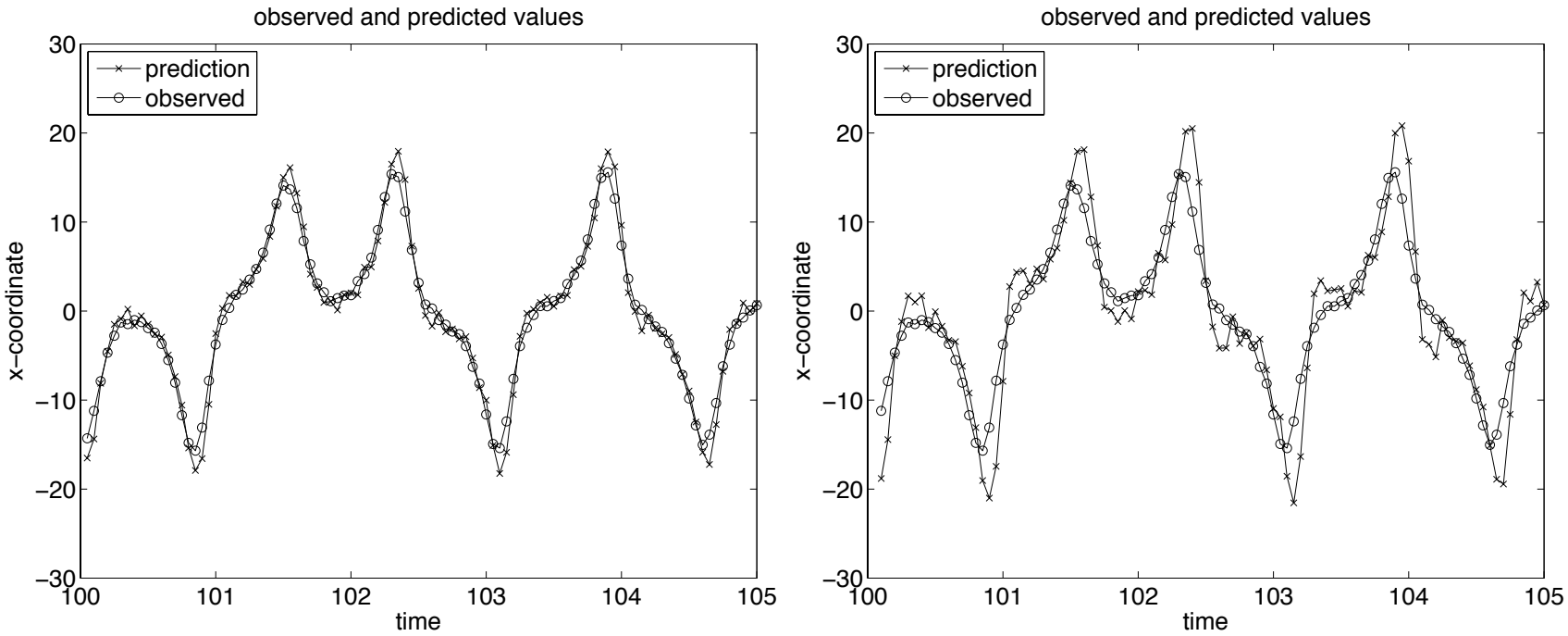


Figure: Observed values for the x-component and its predicted values using linear extrapolation. The figure on the left shows the results from linear extrapolation over a single observation interval $\Delta t_{\mathrm{out}} = 0.05$ while the figure on the right shows results when doubling the prediction interval to 0.1 time units.

How can we assess the quality of a forecast? A simple measure is given by the following quantitity:

Definition (root mean square error)

For a set of predictions and observations at times $\{\mathrm{t}_1, \mathrm{t}_2, \cdots, \mathrm{t}_N\}$ the time averaged root mean square error (RMSE) is given by

$$\text{time averaged RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} |y_{\rm obs}(\mathrm{t}_k) - y_{\rm predict}(\mathrm{t}_k)|^2}.$$

The previous example resulted in a time averaged RMSE of 1.2951 and 3.3654, respectively. Can we do better with higher-order extrapolation (i.e. $p > 1$):

$$y_{\rm predict}(\mathrm{t}_{k_*+1}) = \sum_{l=0}^{p} a_l \, y_{\rm obs}(\mathrm{t}_{k_*-l}).$$

Example (Lorenz-63)

We set $p = 4$ and obtain extrapolation coefficients
$a_0 = 5, a_1 = -10, a_2 = 10, a_3 = -5, a_4 = 1$ and
$a_0 = 15, a_1 = -40, a_2 = 45, a_3 = -24, a_4 = 5, respectively$

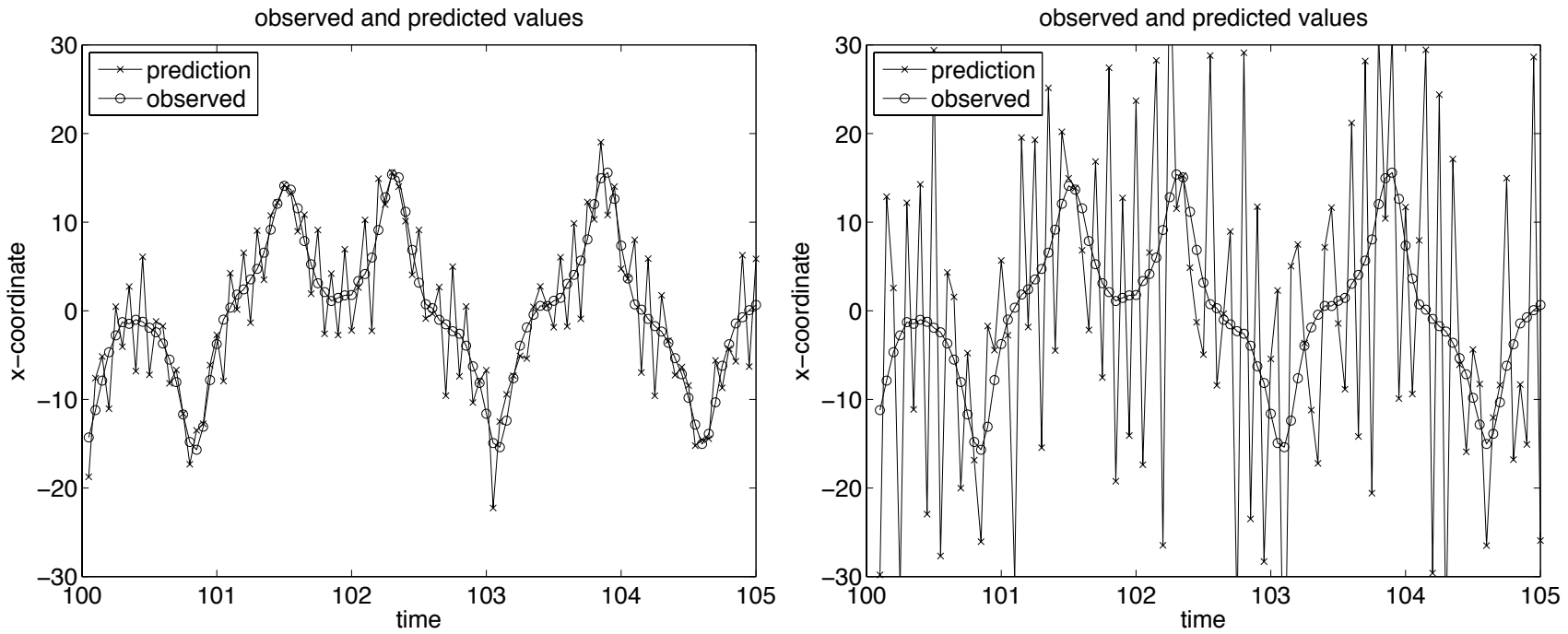


Figure: Observed and predicted values using fourth-order extrapolation. Left: results from extrapolation over a single observation interval $\Delta t_{\text{out}} = 0.05$; Right: results for doubling the prediction interval to 0.1 time units. The time averaged RMSE for the predictions on the left is 4.2707!

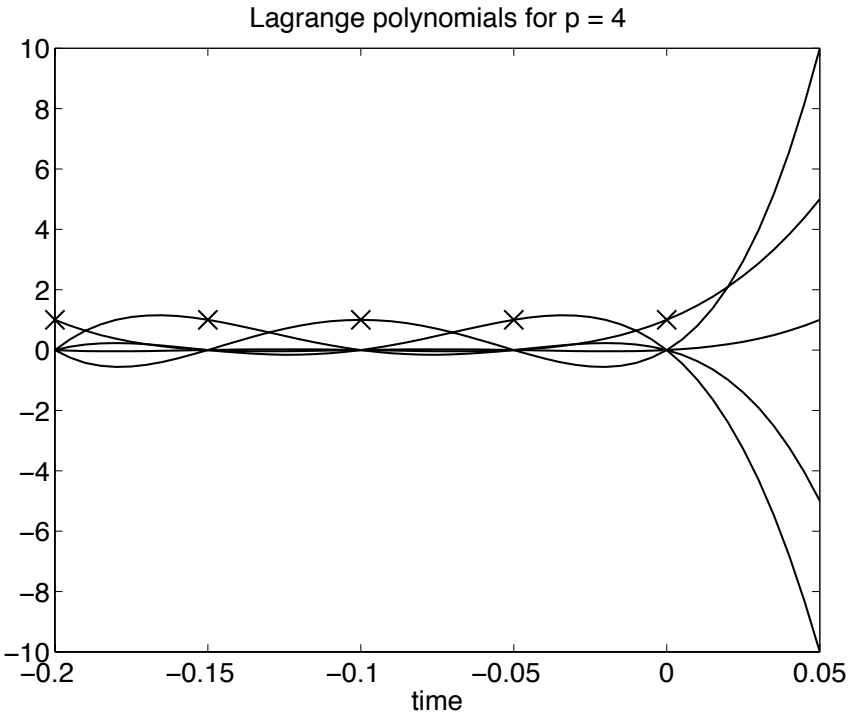
The extrapolation coefficients a_l are given by the value of the associated Lagrange polynomials

$$l_j(t) = \frac{\prod_{i \neq j}(t - \mathrm{t}_i)}{\prod_{i \neq j}(\mathrm{t}_j - \mathrm{t}_i)}$$

via

$$a_l = l_{k_* - l}(\mathrm{t}_{k_*+1}), \qquad l = 0, 1, \ldots, p.$$

Lagrange polynomials $l_j(t)$ of order four corresponding to observations at $\mathrm{t}_i = 0, -0.05, -0.1, -0.15, -.20$. The coefficients a_l are equal to the values of the Lagrangian polynomials at $\mathrm{t} = 0.05$. Crosses mark the points where each polynomial takes the value one. Note that the other polynomials are zero at those interpolation points, and note the steep increase in magnitude outside the interpolation interval $t \in [-0.2, 0]$.



Definition (Statistical Learning)

Instead of using polynomial interpolation, we shall seek the coefficients a_l that optimise the prediction errors for a chosen subset of the observations, which we call the training set. These "extrapolation"coefficients are then fixed, and can be used to predict future observations.
We shall assess the performance of our extrapolation coefficients on the remaining observations points, which we shall call the test set.
For simplicity, let us assume that the training set consists of the first $N_T < N_{\rm obs}$ observations $\{y_{\rm obs}(\mathrm{t}_1), \ldots, y_{\rm obs}(\mathrm{t}_{N_T})\}$, and use the remaining data points as the test set.

Definition (Method of least squares)

Given a chosen set of coefficients $a_l \in \mathbb{R}$, $l = 0, \ldots, p$, we can obtain a prediction of $y_{\rm obs}(\mathrm{t}_{j+p+1})$ for $0 < j \leq N_T - p - 1$ by using

$$y_{\rm predict}(\mathrm{t}_{k_*+1}) = \sum_{l=0}^{p} a_l \, y_{\rm obs}(\mathrm{t}_{k_*-l}).$$

The quality of the predictions is measured by the residuals

$$\begin{aligned} r_j &= y_{\rm obs}(\mathrm{t}_{j+p+1}) - y_{\rm predict}(\mathrm{t}_{j+p+1}) \\ &= y_{\rm obs}(\mathrm{t}_{j+p+1}) - \sum_{l=0}^{p} a_l y_{\rm obs}(\mathrm{t}_{j+p-l}) \end{aligned} \tag{1}$$

for $j = 1, 2, \ldots, J$ with $J = N_T - p - 1$.
We now seek the coefficients a_l such that the resulting time averaged RMSE is minimised over the training set. This is equivalent to minimising the functional

$$L(\{a_l\}) = \frac{1}{2} \sum_{j=1}^{J} r_j^2.$$

Example (Lorenz-63)

We find that implementing the method of least squares for $p = 4$ and a training set with $N_T = N_{\text{obs}}/2 = 2000$ leads to a time averaged RMSE of 0.9718 with $a_0 = 2.0503, a_1 = -1.2248, a_2 = -0.2165, a_3 = 0.4952, a_4 = -0.1397$. The RMSE increases to 2.3039 when doubling the prediction interval.

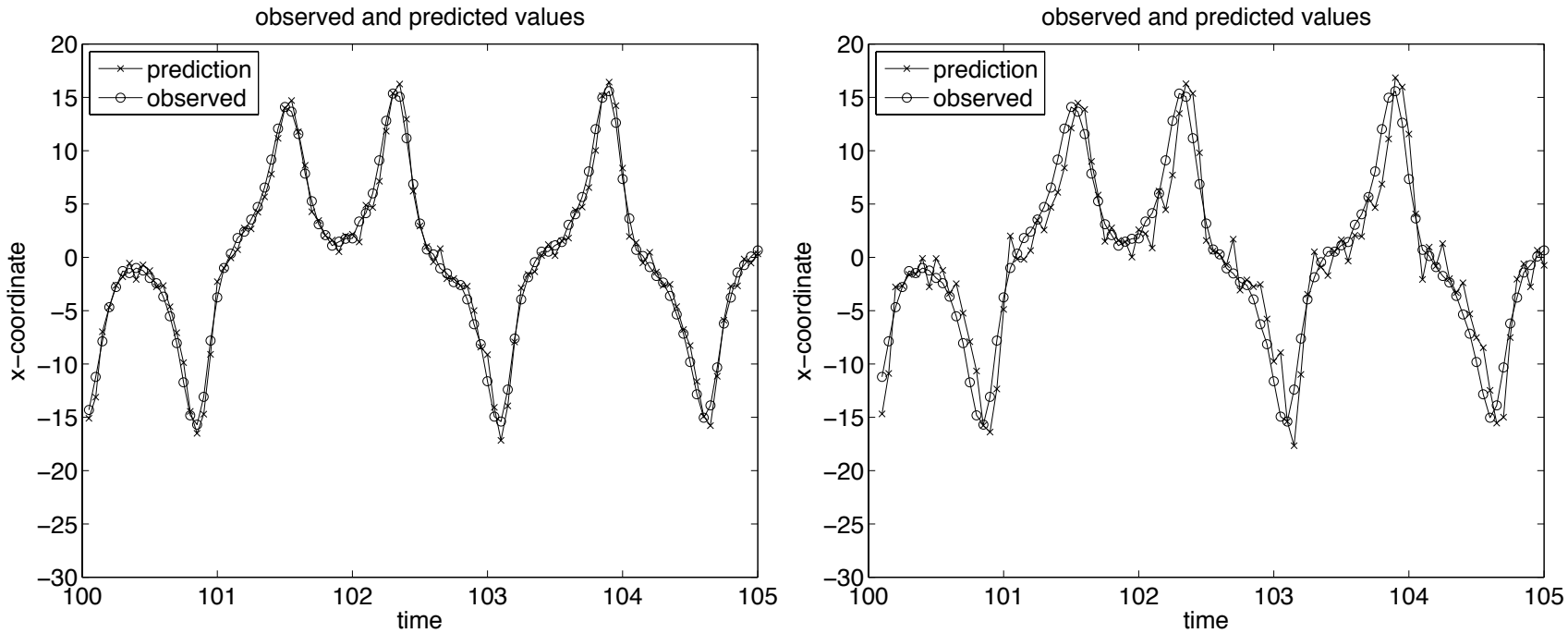


Figure: Observed values of the x-component and corresponding predicted values using the method of least squares with $p = 4$. Left: predictions over a single observation interval $\Delta t_{\text{out}} = 0.05$, Right: results when doubling the prediction interval to 0.1 time units.

Remark (Method of least squares)

The desired vector of coefficients $x = \{a_0, a_1, \ldots, a_p)^{\mathrm{T}} \in \mathbb{R}^{p+1}$ *can be found as the solution of the following minimisation problem*

$$x_* = \arg\min \|r\|^2 = \arg\min \|Ax - b\|^2.$$

Hence

$$\nabla_x L(x_*) = 2A^{\mathrm{T}} A x_* - A^{\mathrm{T}} b = 0, \qquad L(x) = \|Ax - b\|^2$$

for $A \in \mathbb{R}^{J \times (p+1)}$, $b \in \mathbb{R}^J$, $J = N_{\mathrm{T}} - p - 1$, *appropriately defined. See book for more details.*

1.3 Model driven forecasting and data assimilation

Definition (Model driven forecasting)

We now introduce method for making forecasts, which is based on mechanistic or top-down models of the physical process that are derived from *first principles*, a process well established in the context of classical mechanics, for example.

In practice such first principles might be provided by conservation of mass and/or energy or by Newton's laws of motion, or other analogues in *e.g.* biology, sociology or economics. Given an estimate of the system state $z(t_0)$ at time t_0, a model allows us to obtain estimates of the system state $z(t)$ for $t > t_0$. In almost all cases the model is imperfect, and model errors lead to increased errors in the state estimate over time, unless it is corrected by introducing more data at later times.

Example (Lorenz-63)

We assume that our "mechanistic" model is given by

$$z^{n+1} = z^n + \delta t f(z^n), \quad t_{n+1} = t_n + \delta t.$$

The resulting model error (i.e. prediction error over a single time-step) is in this case represented by

$$e^{n+1} = -\delta t\, g(t_n), \qquad t_n = n\,\delta t.$$

The precise form of the model error is in practice unknown.

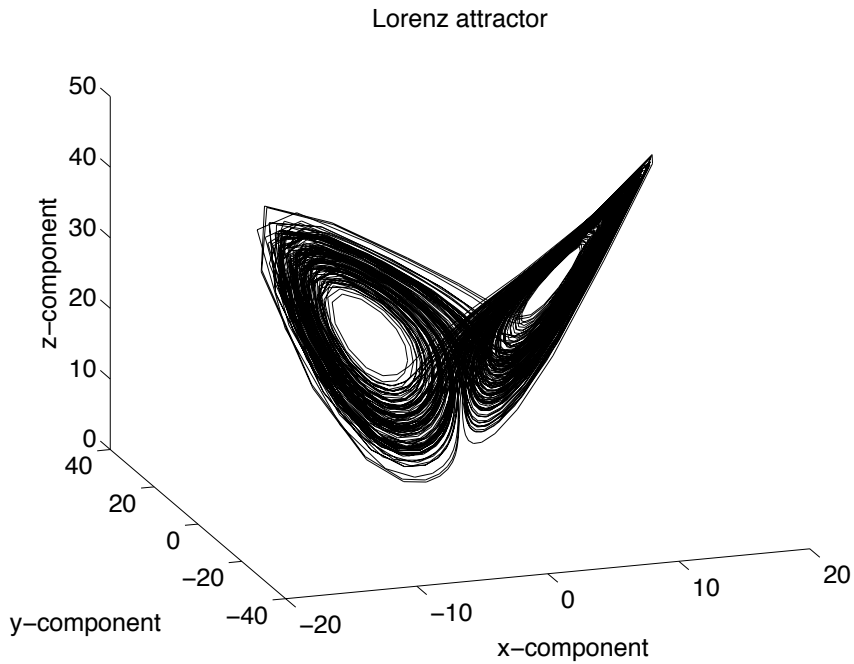


Figure: Long trajectory from our mechanistic model. The shape of the attractor is nearly identical to what has previously been displayed for the "exact" model.

Example (Lorenz-63)

What is the accumulated impact of the model error on our predictions? We start the "exact" and "mechanistic" model from an identical initial condition z^0.

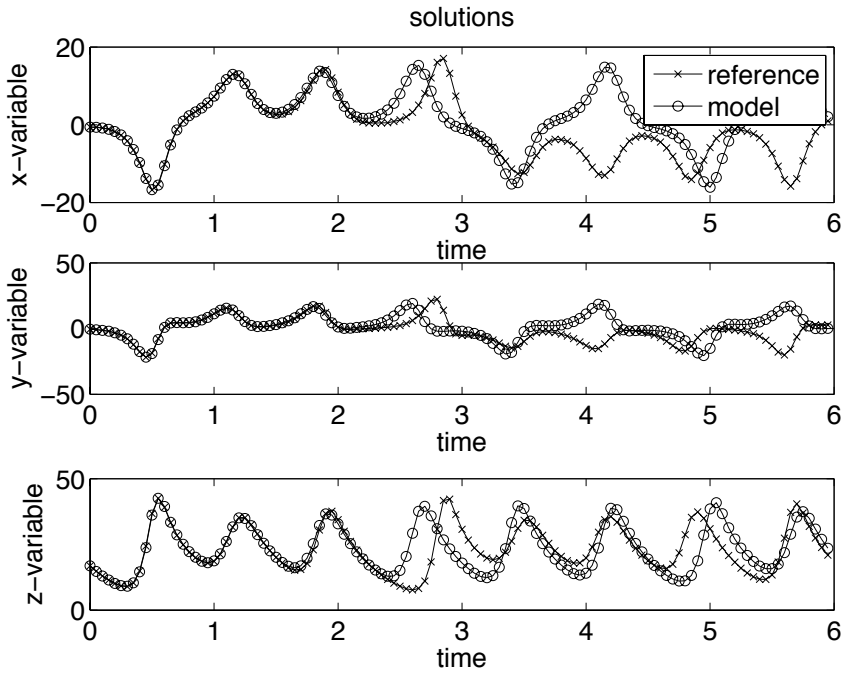


Figure: We compare the behaviour of our mechanistic model to the reference trajectory. The differences between the model and nature are caused by the non-autonomous driving terms $g^n = g(t_n)$ and their accumulative effect. These differences become significant at about $t = 2$ as can be seen from the above panel, which displays the differences in all three solution components as a function of time.

Remark (Data Assimilation)

We conclude that (i) we need methods for estimating appropriate initial conditions for our mechanistic model from the available observations, and that (ii) we should strive to improve our mechanistic models by making the unaccounted contributions from $g(t)$ as small as possible.

Both tasks can be addressed by clever combinations of mechanistic models with observational data. Associated computational techniques are often referred to as data assimilation in the geosciences and filtering/smoothing in the engineering community.

Throughout this book we will primarily use the term data assimilation, which, broadly speaking, covers the task of combining mechanistic models with partial and noisy observations in order to produce more skillful forecasts.

Definition (Nonlinear method of least squares)

The differences between simulated and true observations is measured in a residual

$$r_k = y_{\rm model}(\mathrm{t}_k) - y_{\rm obs}(\mathrm{t}_k) = h(z_{\rm model}(\mathrm{t}_k)) - y_{\rm obs}(\mathrm{t}_k), \qquad k = 1, \ldots, N_a.$$

The residual implicitly depends on the model initial condition z^0, since this changes the entire model trajectory and therefore the simulated observations $y_{\rm model}(\mathrm{t}_k)$.

Adopting the method of least squares, we seek the initial condition z^0 that minimises the residual sum

$$L(z^0) = \frac{1}{2} \sum_{k=1}^{N_a} \|r_k\|^2. \tag{2}$$

We denote a minimiser by z^0_*.

The main complication arises from the nonlinear dependence of $z_{\rm model}(\mathrm{t}_k)$ on z^0 (which may results in nonexistence or nonuniqueness of minimisers).

Remark (gradient of nonlinear method of least squares)

We introduce the map ψ as a shorthand for the $N_{\rm out}$-fold application of the mechanistic model, i.e. if we define

$$z_{\rm model}(\mathrm{t}_{k+1}) = \psi(z_{\rm model}(\mathrm{t}_k)), \quad k \geq 0.$$

Then

$$z_{\rm model}(\mathrm{t}_k) = \psi^k(z^0) := \underbrace{\psi(\psi(\cdots\psi(z^0)))}_{k \text{ fold application of } \psi}$$

and

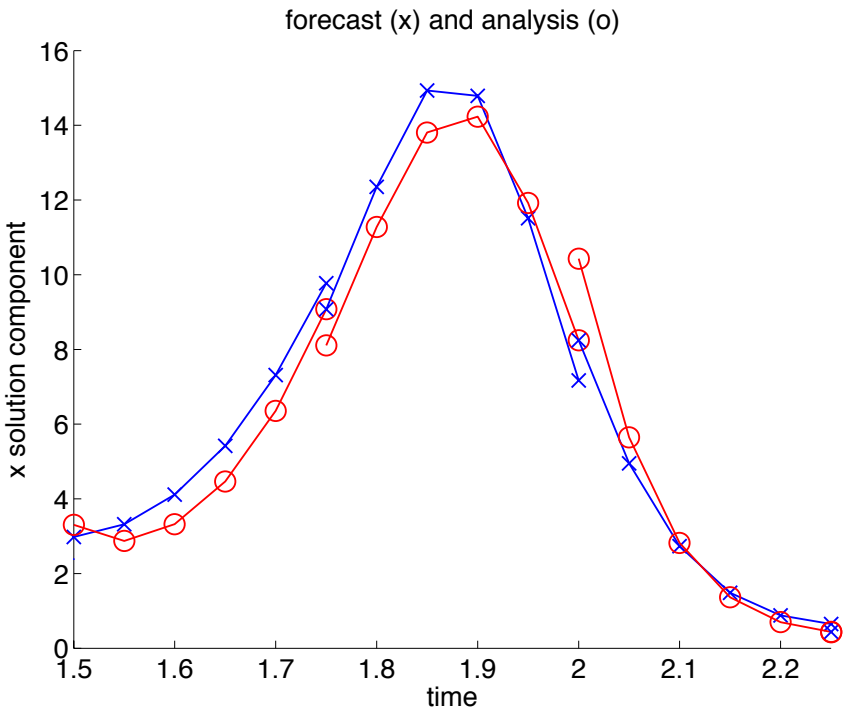
$$\nabla_{z^0} L(z^0) = \sum_{k=1}^{N_a} (D\psi^k(z^0))^T H^T r_k, \quad r_k = H\psi^k(z^0) - y_{\rm obs}(\mathrm{t}_k),$$

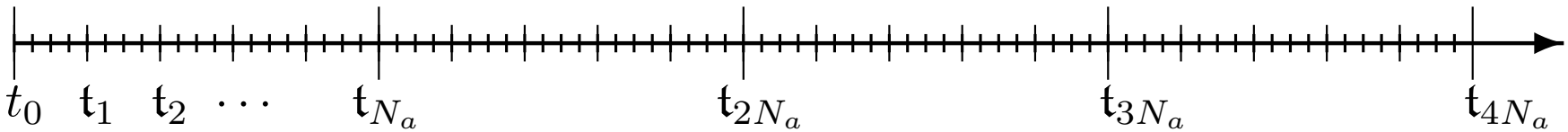
for linear forward operator $h(z) = Hz$. The minimiser z_*^0 must satisfy

$$\nabla_{z^0} L(z_*^0) = 0.$$

Definition (Windowed data assimilation)

In contrast to the forecast $z_{\rm model}(t)$, $t \geq 0$, which does not make use of the observations $y_{\rm obs}(\mathrm{t}_k)$, $k = 1, \ldots, N_{\rm a}$, the minimiser z_*^0 of L provides an improved (retrospective) approximation $z^a_{\rm model}(t)$, called the analysis.
Once time t is increased beyond $t = \mathrm{t}_{N_a}$ the analysis $z^a_{\rm model}(t)$ turns into the next forecast.
This forecast-analysis cycle can be repeated. Once observations at t_k, $k = N_a + 1, \ldots, 2N_a$, have become available, the next assimilation cycle covering the interval $[\mathrm{t}_{N_a}, \mathrm{t}_{2N_a}]$ delivers the initial condition for the forecast over the interval $[\mathrm{t}_{2N_a}, \mathrm{t}_{3N_a}]$.





Example (Lorenz-63)

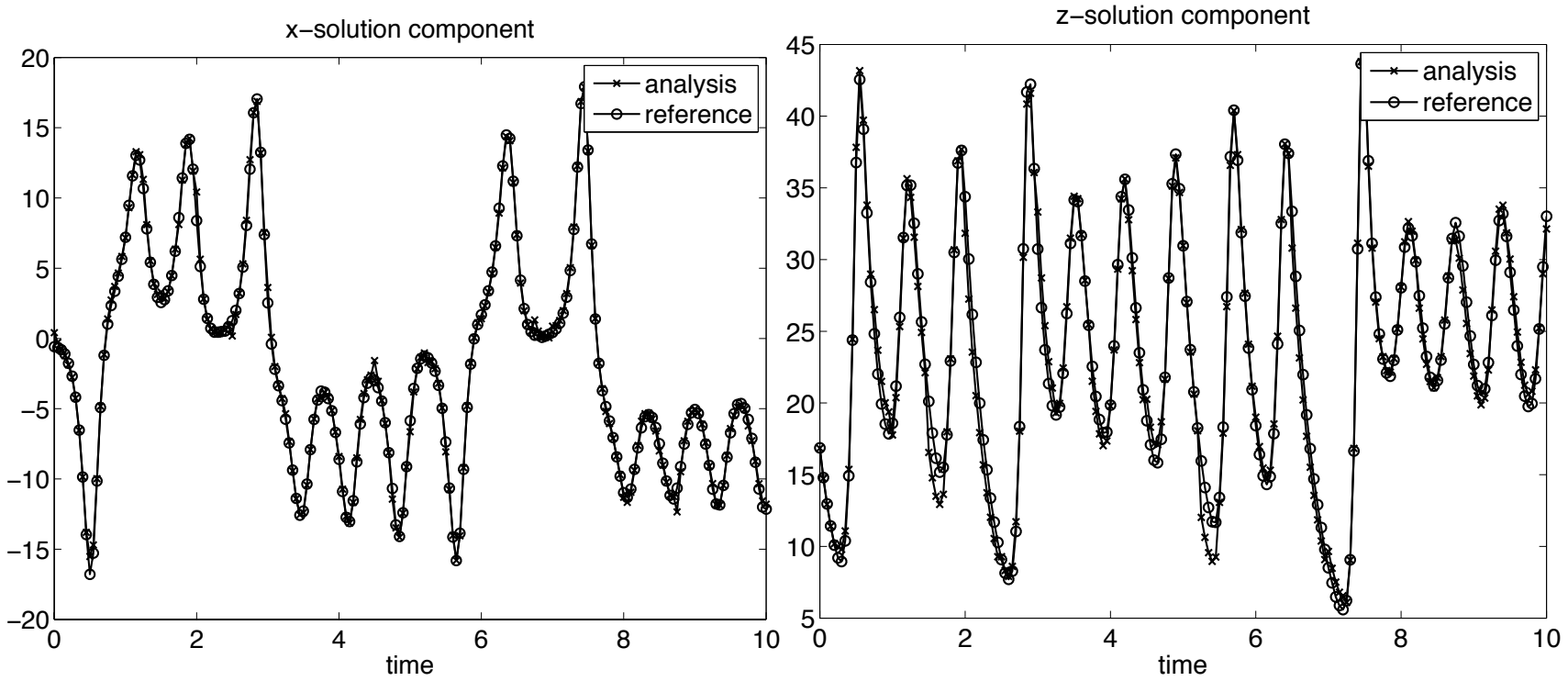


Figure: We display the results from 40 data assimilation cycles each over a window of length $5\,\Delta t_{\rm out} = 0.25$. Only the x-variable is observed in intervals of $\Delta t_{\rm out} = 0.05$. The synchronising effect of the nonlinear least square approach can be clearly seen both in the x variable and the unobserved z variable, while the model output without adjustments from the data assimilation cycles loses track of the underlying physical process at about $t = 2$.

Example (Lorenz-63)

We use the last analysis from the previous slide to start a forecast at $t = 10$:

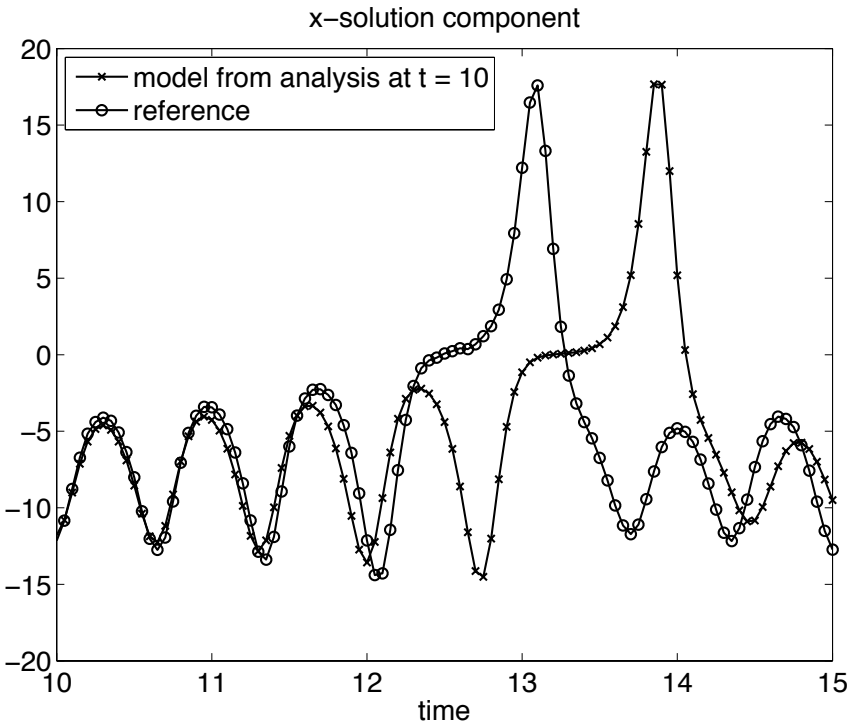


Figure: Forecast started from the analysis at time $t = 10$ and reference solution from the surrogate physical process. It can be seen that the forecast stays close to the reference solution for about 2 time units after which its starts diverging due to errors in the analysis and model errors.

Example (Lorenz-63)

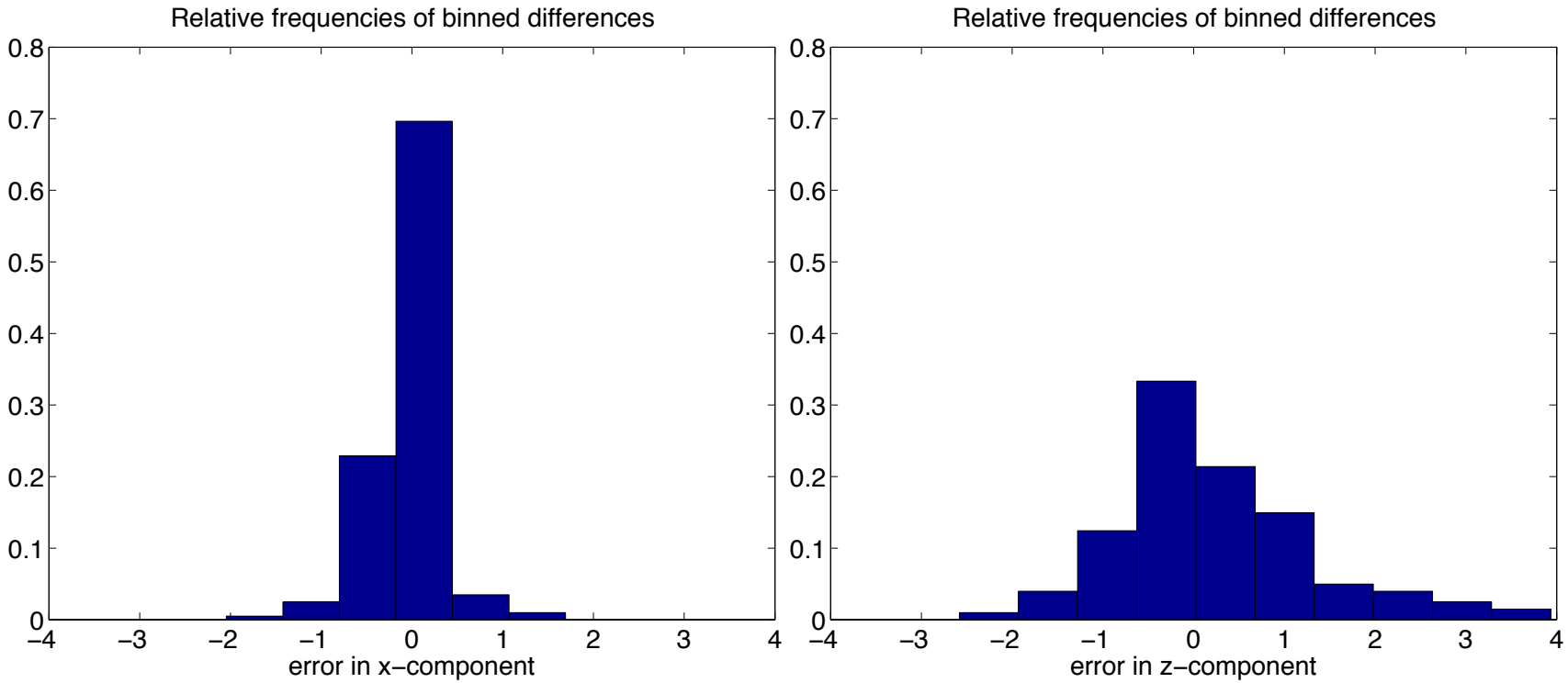


Figure: Relative frequencies of binned differences between the reference solution and the analysis, in both x and z. It is tempting to view these relative frequencies as arising from finite samples of an underlying random variable with unknown specifications. We could then discuss the probability of an analysis falling within a certain range of the (generally unavailable explicitly) true system state. It is a task of statistics/uncertainty quantification to characterise such probabilities.

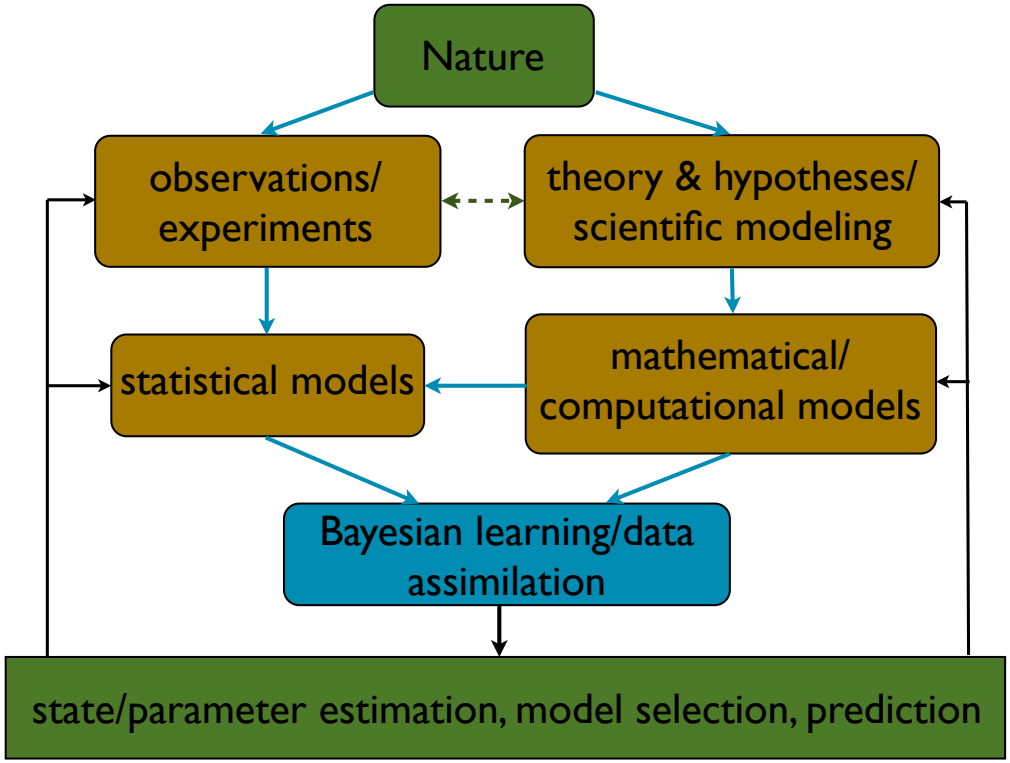
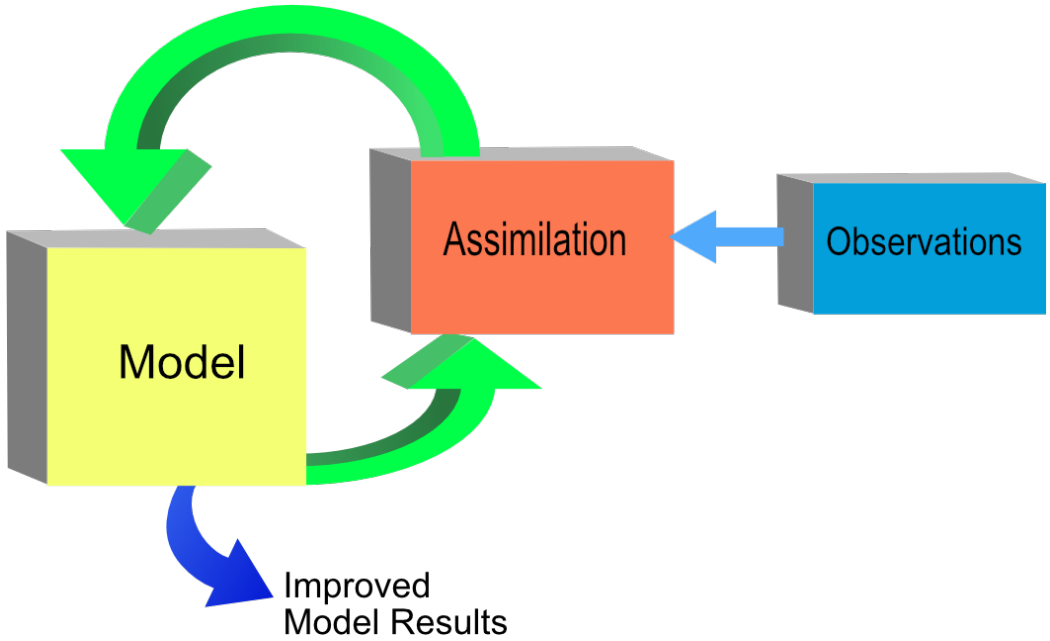


Figure: A schematic presentation of the complex process of building and using mathematical models. An essential building block is data assimilation which is a mathematical technique for combining mechanistic models with statistical models based on data. Here a statistical model refers to a set of probability distributions describing observed data, while a mechanistic model consists of evolution equations which give rise to deterministic and/or stochastic processes. Mechanistic models depend on parameters (including the state of the model), which we treat as random variables.



Remark (Data assimilation)

The seamless integration of large data sets into sophisticated computational models provides one of the central research challenges for the mathematical sciences in the 21st century.

When the computational model is based on evolutionary equations and the data set is time-ordered, the process of combining models and data is called data assimilation. The assimilation of data into computational models serves a wide spectrum of purposes ranging from model calibration and model comparison all the way to the validation of novel model design principles.