

1

Introduction

1.1 State of a Dynamic System

In many branches of science and engineering, deriving a probabilistic model for sequential data plays a key role. System theory provides guidelines for studying the underlying dynamics of sequential data (time series). In describing a dynamic system, the notion of *state* is a key concept [1]:

Definition 1.1 *State of a dynamic system* is the smallest collection of variables that must be specified at a time instant k_0 in order to be able to predict the behavior of the system for any time instant $k \geq k_0$. To be more precise, the state is the minimal record of the past history, which is required to predict the future behavior.

According to the *principle of causality*, any dynamic system may be described from the state perspective. Deploying a state-transition model allows for determining the future state of a system, $\mathbf{x}_k \in \mathbb{R}^{n_x}$, at any time instant $k \geq k_0$, given its initial state, \mathbf{x}_{k_0} , at time instant k_0 as well as the inputs to the system, $\mathbf{u}_k \in \mathbb{R}^{n_u}$, for $k \geq k_0$. The output of the system, $\mathbf{y}_k \in \mathbb{R}^{n_y}$, is a function of the state, which can be computed using a measurement model. In this regard, state-space models are powerful tools for analysis and control of dynamic systems.

1.2 State Estimation

Observability is a key concept in system theory, which refers to the ability to reconstruct the *hidden* or *latent* state variables that cannot be directly measured, from

the measured variables in the minimum possible length of time [1]. In building state-space models, two key questions deserve special attention [2]:

- (i) Is it possible to identify the governing dynamics from data?
- (ii) Is it possible to perform inference from observables to the latent state variables?

At time instant k , the inference problem to be solved is to find the estimate of $\mathbf{x}_{k+\alpha}$ in the presence of noise, which is denoted by $\hat{\mathbf{x}}_{k+\alpha}$. Depending of the value of α , estimation algorithms are categorized into three groups [3]:

- (i) *Prediction*: $\alpha > 0$,
- (ii) *Filtering*: $\alpha = 0$,
- (iii) *Smoothing*: $\alpha < 0$.

Regarding the mentioned two challenging questions, in order to improve performance, sophisticated representations can be deployed for the system under study. However, the corresponding inference algorithms may become computationally demanding. Hence, for designing efficient data-driven inference algorithms, the following points must be taken into account [2]:

- (i) The underlying assumptions for building a state-space model must allow for reliable system identification and plausible long-term prediction of the system behavior.
- (ii) The inference mechanism must be able to capture rich dependencies.
- (iii) The algorithm must be able to inherit the merit of *learning machines* to be trainable on raw data such as sensory inputs in a control system.
- (iv) The algorithm must be scalable to big data regarding the optimization of model parameters based on the *stochastic gradient descent* method.

Regarding the important role of computation in inference problems, Section 1.3 provides a brief account of the foundations of computing.

1.3 Construals of Computing

According to [4], a comprehensive *theory of computing* must meet three criteria:

- (i) *Empirical criterion*: Doing justice to practice by keeping the analysis grounded in real-world examples.
- (ii) *Conceptual criterion*: Being understandable in terms of what it says, where it comes from, and what it costs.
- (iii) *Cognitive criterion*: Providing an intelligible foundation for the computational theory of mind that underlies both artificial intelligence and cognitive science.

Following this line of thinking, it was proposed in [4] to distinguish the following construals of computation:

1. *Formal symbol manipulation* is rooted in formal logic and metamathematics. The idea is to build machines that are capable of manipulating symbolic or meaningful expressions regardless of their interpretation or semantic content.
2. *Effective computability* deals with the question of what can be done, and how hard it is to do it mechanically.
3. *Execution of an algorithm or rule following* focuses on what is involved in following a set of rules or instructions, and what behavior would be produced.
4. *Calculation of a function* considers the behavior of producing the value of a mathematical function as output, when a set of arguments is given as input.
5. *Digital state machine* is based on the idea of a finite-state automaton.
6. *Information processing* focuses on what is involved in storing, manipulating, displaying, and trafficking of information.
7. *Physical symbol systems* is based on the idea that the way computers interact with symbols depends on their mutual physical embodiment. In this regard, computers may be assumed to be made of symbols.
8. *Dynamics* must be taken into account in terms of the roles that nonlinear elements, attractors, criticality, and emergence play in computing.
9. *Interactive agents* are capable of interacting and communicating with other agents and even people.
10. *Self-organizing or complex adaptive systems* are capable of adjusting their organization or structure in response to changes in their environment in order to survive and improve their performance.
11. *Physical implementation* emphasizes on the occurrence of computational practice in real-world systems.

1.4 Statistical Modeling

Statistical modeling aims at extracting information about the underlying data mechanism that allows for making predictions. Then, such predictions can be used to make decisions. There are two cultures in deploying statistical models for data analysis [5]:

- *Data modeling culture* is based on the idea that a given stochastic model generates the data.
- *Algorithmic modeling culture* uses algorithmic models to deal with an unknown data mechanism.

An algorithmic approach has the advantage of being able to handle large complex datasets. Moreover, it can avoid irrelevant theories or questionable conclusions.

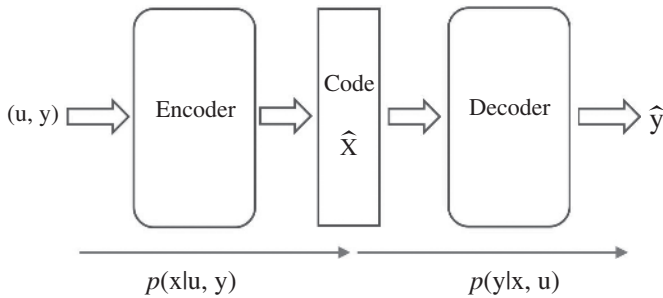


Figure 1.1 The encoder of an asymmetric autoencoder plays the role of a nonlinear filter.

Taking an algorithmic approach, in *machine learning*, statistical models can be classified as [6]:

- (i) *Generative models* predict visible effects from hidden causes, $p(\mathbf{y}|\mathbf{x}, \mathbf{u})$.
- (ii) *Discriminative models* infer hidden causes from visible effects, $p(\mathbf{x}|\mathbf{u}, \mathbf{y})$.

While the former is associated with the measurement process in a state-space model, the latter is associated with the state estimation or filtering problem. Deploying machine learning, a wide range of filtering algorithms can be developed that are able to learn the corresponding state-space models. For instance, an asymmetric autoencoder can be designed by combining a generative model and a discriminative model as shown in Figure 1.1 [7]. Deep neural networks can be used to implement both the encoder and the decoder. Then, the resulting autoencoder can be trained in an unsupervised manner. After training, the encoder can be used as a filter, which estimates the latent state variables.

1.5 Vision for the Book

This book provides an algorithmic perspective on the nonlinear state/parameter estimation problem for discrete-time systems, where measurements are available at discrete sampling times and estimators are implemented using digital processors. In Chapter 2, guidelines are provided for discretizing continuous-time linear and nonlinear state-space models. The rest of the book is organized as follows:

- Chapter 2 presents the notion of observability for deterministic and stochastic systems.

Chapters 3–7 cover classic estimation algorithms:

- Chapter 3 is dedicated to observers as state estimators for deterministic systems.

- Chapter 4 presents the general formulation of the optimal Bayesian filtering for stochastic systems.
- Chapter 5 covers the Kalman filter as the optimal Bayesian filter in the sense of minimizing the mean-square estimation error for linear systems with Gaussian noise. Moreover, Kalman filter variants are presented that extend its applicability to nonlinear or non-Gaussian cases.
- Chapter 6 covers the particle filter, which handles severe nonlinearity and non-Gaussianity by approximating the corresponding distributions using a set of particles (random samples).
- Chapter 7 covers the smooth variable-structure filter, which provides robustness against bounded uncertainties and noise. In addition to the innovation vector, this filter benefits from a secondary set of performance indicators.

Chapters 8–11 cover learning-based estimation algorithms:

- Chapter 8 covers the basics of deep learning.
- Chapter 9 covers deep-learning-based filtering algorithms using supervised and unsupervised learning.
- Chapter 10 presents the expectation maximization algorithm and its variants, which are used for joint state and parameter estimation.
- Chapter 11 presents the reinforcement learning-based filter, which is built on viewing variational inference and reinforcement learning as instances of a generic expectation maximization problem.

The last chapter is dedicated to nonparametric Bayesian models:

- Chapter 12 covers measure-theoretic probability concepts as well as the notions of exchangeability, posterior computability, and algorithmic sufficiency. Furthermore, it provides guidelines for constructing nonparametric Bayesian models from finite parametric Bayesian models.

In each chapter, selected applications of the presented filtering algorithms are reviewed, which cover a wide range of problems. Moreover, the last section of each chapter usually refers to a few topics for further study.

