CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Data-Driven Science and Engineering

Data-driven discovery is revolutionizing the modeling, prediction, and control of complex systems. This textbook brings together machine learning, engineering mathematics, and mathematical physics to integrate modeling and control of dynamical systems with modern methods in data science. It highlights many of the recent advances in scientific computing that enable data-driven methods to be applied to a diverse range of complex systems such as turbulence, the brain, climate, epidemiology, finance, robotics, and autonomy.

Aimed at advanced undergraduate and beginning graduate students in the engineering and physical sciences, the text presents a range of topics and methods from introductory to state of the art.

Steven L. Brunton is Associate Professor of Mechanical Engineering at the University of Washington. He is also Adjunct Associate Professor of Applied Mathematics and a Data-Science Fellow at the eScience Institute. His research applies data science and machine learning for dynamical systems and control to fluid dynamics, biolocomotion, optics, energy systems, and manufacturing. He is an author of two textbooks, received the Army and Air Force Young Investigator Program awards, and was awarded the University of Washington College of Engineering teaching and Junior Faculty awards.

J. Nathan Kutz is the Robert Bolles and Yasuko Endo Professor of Applied Mathematics at the University of Washington, and served as department chair until 2015. He is also Adjunct Professor of Electrical Engineering and Physics and a Senior Data-Science Fellow at the eScience institute. His research interests are in complex systems and data analysis where machine learning can be integrated with dynamical systems and control for a diverse set of applications. He is an author of two textbooks and has received the Applied Mathematics Boeing Award of Excellence in Teaching and an NSF CAREER award.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Data-Driven Science and Engineering

Machine Learning, Dynamical Systems, and Control

STEVEN L. BRUNTON
University of Washington

J. NATHAN KUTZ
University of Washington

CAMBRIDGE
UNIVERSITY PRESS

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India

79 Anson Road, #06–04/06, Singapore 079906

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning, and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781108422093
DOI: 10.1017/9781108380690

© Steven L. Brunton and J. Nathan Kutz 2019

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2019

Printed and bound in Great Britain by Clays Ltd, Elcograf S.p.A.

A catalogue record for this publication is available from the British Library.

Library of Congress Cataloging-in-Publication Data
Names: Brunton, Steven L. (Steven Lee), 1984– author. | Kutz, Jose Nathan, author.
Title: Data-driven science and engineering : machine learning, dynamical systems, and control / Steven L. Brunton,
University of Washington, J. Nathan Kutz, University of Washington.
Description: Cambridge : Cambridge University Press, 2019. |
Includes bibliographical references and index.
Identifiers: LCCN 2018029888 | ISBN 9781108422093 (hardback : alk. paper)
Subjects: LCSH: Engineering–Data processing. | Science–Data processing. | Mathematical analysis.
Classification: LCC TA330 .B78 2019 | DDC 620.00285/631–dc23
LC record available at https://lccn.loc.gov/2018029888

ISBN 978-1-108-42209-3 Hardback

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party Internet Web sites referred to in this publication and does not guarantee that any content on such Web sites is, or will remain, accurate or appropriate.

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Contents

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

© in this web service Cambridge University Press www.cambridge.org

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

© in this web service Cambridge University Press www.cambridge.org

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Preface

This book is about the growing intersection of data-driven methods, applied optimization, and the classical fields of engineering mathematics and mathematical physics. We have been developing this material over a number of years, primarily to educate our advanced undergrad and beginning graduate students from engineering and physical science departments. Typically, such students have backgrounds in linear algebra, differential equations, and scientific computing, with engineers often having some exposure to control theory and/or partial differential equations. However, most undergraduate curricula in engineering and science fields have little or no exposure to data methods and/or optimization. Likewise, computer scientists and statisticians have little exposure to dynamical systems and control. Our goal is to provide a broad entry point to applied data science for both of these groups of students. We have chosen the methods discussed in this book for their (1) relevance, (2) simplicity, and (3) generality, and we have attempted to present a range of topics, from basic introductory material up to research-level techniques.

Data-driven discovery is currently revolutionizing how we model, predict, and control complex systems. The most pressing scientific and engineering problems of the modern era are not amenable to empirical models or derivations based on first-principles. Increasingly, researchers are turning to data-driven approaches for a diverse range of complex systems, such as turbulence, the brain, climate, epidemiology, finance, robotics, and autonomy. These systems are typically nonlinear, dynamic, multi-scale in space and time, high-dimensional, with dominant underlying patterns that should be characterized and modeled for the eventual goal of sensing, prediction, estimation, and control. With modern mathematical methods, enabled by unprecedented availability of data and computational resources, we are now able to tackle previously unattainable challenge problems. A small handful of these new techniques include robust image reconstruction from sparse and noisy random pixel measurements, turbulence control with machine learning, optimal sensor and actuator placement, discovering interpretable nonlinear dynamical systems purely from data, and reduced order models to accelerate the study and optimization of systems with complex multi-scale physics.

Driving modern data science is the availability of vast and increasing quantities of data, enabled by remarkable innovations in low-cost sensors, orders-of-magnitudes increases in computational power, and virtually unlimited data storage and transfer capabilities. Such vast quantities of data are affording engineers and scientists across all disciplines new opportunities for data-driven discovery, which has been referred to as the fourth paradigm of scientific discovery [245]. This fourth paradigm is the natural culmination of the first three paradigms: empirical experimentation, analytical derivation, and computational investigation. The integration of these techniques provides a transformative framework for

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

data-driven discovery efforts. This process of scientific discovery is not new, and indeed mimics the efforts of leading figures of the scientific revolution: Johannes Kepler (1571–1630) and Sir Isaac Newton (1642–1727). Each played a critical role in developing the theoretical underpinnings of celestial mechanics, based on a combination of empirical data-driven and analytical approaches. Data science is not replacing mathematical physics and engineering, but is instead augmenting it for the twenty-first century, resulting in more of a renaissance than a revolution.

Data science itself is not new, having been proposed more than 50 years ago by John Tukey who envisioned the existence of a scientific effort focused on learning from data, or *data analysis* [152]. Since that time, data science has been largely dominated by two distinct cultural outlooks on data [78]. The *machine learning* community, which is predominantly comprised of computer scientists, is typically centered on prediction quality and scalable, fast algorithms. Although not necessarily in contrast, the *statistical learning* community, often centered in statistics departments, focuses on the inference of interpretable models. Both methodologies have achieved significant success and have provided the mathematical and computational foundations for data-science methods. For engineers and scientists, the goal is to leverage these broad techniques to infer and compute models (typically nonlinear) from observations that correctly identify the underlying dynamics *and* generalize qualitatively and quantitatively to unmeasured parts of phase, parameter, or application space. Our goal in this book is to leverage the power of both statistical and machine learning to solve engineering problems.

Themes of This Book

There are a number of key themes that have emerged throughout this book. First, many complex systems exhibit *dominant low-dimensional patterns* in the data, despite the rapidly increasing resolution of measurements and computations. This underlying structure enables efficient sensing, and compact representations for modeling and control. Pattern extraction is related to the second theme of finding *coordinate transforms* that simplify the system. Indeed, the rich history of mathematical physics is centered around coordinate transformations (e.g., spectral decompositions, the Fourier transform, generalized functions, etc.), although these techniques have largely been limited to simple idealized geometries and linear dynamics. The ability to derive *data-driven* transformations opens up opportunities to generalize these techniques to new research problems with more complex geometries and boundary conditions. We also take the perspective of *dynamical systems and control* throughout the book, applying data-driven techniques to model and control systems that evolve in time. Perhaps the most pervasive theme is that of *data-driven applied optimization*, as nearly every topic discussed is related to optimization (e.g., finding *optimal* low-dimensional patterns, *optimal* sensor placement, machine learning *optimization*, *optimal* control, etc.). Even more fundamentally, most data is organized into arrays for analysis, where the extensive development of numerical linear algebra tools from the early 1960s onward provides many of the foundational mathematical underpinnings for matrix decompositions and solution strategies used throughout this text.

Acknowledgments

We are indebted to many wonderful students, collaborators, and colleagues for valuable feedback, suggestions, and support. We are especially grateful to Joshua Proctor, who was

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

instrumental in the origination of this book and who helped guide much of the framing and organization. We have also benefited from extensive interactions and conversations with Bing Brunton, Igor Mezić, Bernd Noack, and Sam Taira. This work would also not be possible without our many great colleagues and collaborators, with whom we have worked and whose research is featured throughout this book.

Throughout the writing of this book and teaching of related courses, we have received great feedback and comments from our excellent students and postdocs: Travis Askham, Michael Au-Yeung, Zhe Bai, Ido Bright, Kathleen Champion, Emily Clark, Charles Delahunt, Daniel Dylewski, Ben Erichson, Charlie Fiesler, Xing Fu, Chen Gong, Taren Gorman, Jacob Grosek, Seth Hirsh, Mikala Johnson, Eurika Kaiser, Mason Kamb, James Kunert, Bethany Lusch, Pedro Maia, Krithika Manohar, Niall Mangan, Ariana Mendible, Thomas Mohren, Megan Morrison, Markus Quade, Sam Rudy, Susanna Sargsyan, Isabel Scherl, Eli Shlizerman, George Stepaniants, Ben Strom, Chang Sun, Roy Taylor, Meghana Velagar, Jake Weholt, and Matt Williams. Our students are our inspiration for this book, and they make it fun and exciting to come to work every day.

We would also like to thank our publisher Lauren Cowles at Cambridge University Press for being a reliable supporter throughout this process.

Online Material

We have designed this book to make extensive use of online supplementary material, including codes, data, videos, homeworks, and suggested course syllabi. All of this material can be found at the following website:

databookuw.com

In addition to course resources, all of the code and data used in the book are available. The codes online are more extensive than those presented in the book, including code used to generate publication quality figures. Data visualization was ranked as the top used data-science method in the Kaggle 2017 *The State of Data Science and Machine Learning* study, and so we highly encourage readers to download the online codes and make full use of these plotting commands.

We have also recorded and posted video lectures on YouTube for most of the topics in this book. We include supplementary videos for students to fill in gaps in their background on scientific computing and foundational applied mathematics. We have designed this text both to be a reference as well as the material for several courses at various levels of student preparation. Most chapters are also modular, and may be converted into stand-alone *boot camps*, containing roughly 10 hours of materials each.

How to Use This Book

Our intended audience includes beginning graduate students, or advanced undergraduates, in engineering and science. As such, the machine learning methods are introduced at a beginning level, whereas we assume students know how to model physical systems with differential equations and simulate them with solvers such as **ode45**. The diversity of topics covered thus range from introductory to state-of-the-art research methods. Our aim is to provide an integrated viewpoint and mathematical toolset for solving engineering and science problems. Alternatively, the book can also be useful for computer science and

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

statistics students who often have limited knowledge of dynamical systems and control. Various courses can be designed from this material, and several example syllabi may be found on the book website; this includes homework, data sets, and code.

First and foremost, we want this book to be fun, inspiring, eye-opening, and empowering for young scientists and engineers. We have attempted to make everything as simple as possible, while still providing the depth and breadth required to be useful in research. Many of the chapter topics in this text could be entire books in their own right, and many of them are. However, we also wanted to be as comprehensive as may be reasonably expected for a field that is so big and moving so fast. We hope that you enjoy this book, master these methods, and change the world with applied data science!

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Common Optimization Techniques, Equations, Symbols, and Acronyms

Most Common Optimization Strategies

Least-Squares (discussed in Chapters 1 and 4) minimizes the sum of the squares of the residuals between a given fitting model and data. Linear least-squares, where the residuals are linear in the unknowns, has a closed form solution which can be computed by taking the derivative of the residual with respect to each unknown and setting it to zero. It is commonly used in the engineering and applied sciences for fitting polynomial functions. Nonlinear least-squares typically requires iterative refinement based upon approximating the nonlinear least-squares with a linear least-squares at each iteration.

Gradient Descent (discussed in Chapters 4 and 6) is the industry leading, convex optimization method for high-dimensional systems. It minimizes residuals by computing the gradient of a given fitting function. The iterative procedure updates the solution by *moving downhill* in the residual space. The Newton–Raphson method is a one-dimensional version of gradient descent. Since it is often applied in high-dimensional settings, it is prone to find only local minima. Critical innovations for big data applications include stochastic gradient descent and the backpropagation algorithm which makes the optimization amenable to computing the gradient itself.

Alternating Descent Method (ADM) (discussed in Chapter 4) avoids computations of the gradient by optimizing in one unknown at a time. Thus all unknowns are held constant while a line search (non-convex optimization) can be performed in a single variable. This variable is then updated and held constant while another of the unknowns is updated. The iterative procedure continues through all unknowns and the iteration procedure is repeated until a desired level of accuracy is achieved.

Augmented Lagrange Method (ALM) (discussed in Chapters 3 and 8) is a class of algorithms for solving constrained optimization problems. They are similar to penalty methods in that they replace a constrained optimization problem by a series of unconstrained problems and add a penalty term to the objective which helps enforce the desired constraint. ALM adds another term designed to mimic a Lagrange multiplier. The augmented Lagrangian is not the same as the method of Lagrange multipliers.

Linear Program and Simplex Method are the workhorse algorithms for convex optimization. A linear program has an objective function which is linear in the unknown and the constraints consist of linear inequalities and equalities. By computing its feasible region, which is a convex polytope, the linear programming algorithm finds a point in the polyhedron where this function has the smallest (or largest) value if such a point exists. The simplex method is a specific iterative technique for linear programs which aims to take a given basic feasible solution to another basic feasible solution for which the objective function is smaller, thus producing an iterative procedure for optimizing.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Most Common Equations and Symbols

Linear Algebra

Linear System of Equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{0.1}$$

The matrix $\mathbf{A} \in \mathbb{R}^{p\times n}$ and vector $\mathbf{b} \in \mathbb{R}^p$ are generally known, and the vector $\mathbf{x} \in \mathbb{R}^n$ is unknown.

Eigenvalue Equation

$$\mathbf{A}\mathbf{T} = \mathbf{T}\boldsymbol{\Lambda}. \tag{0.2}$$

The columns $\boldsymbol{\xi}_k$ of the matrix $\mathbf{T}$ are the eigenvectors of $\mathbf{A} \in \mathbb{C}^{n\times n}$ corresponding to the eigenvalue λ_k: $\mathbf{A}\boldsymbol{\xi}_k = \lambda_k\boldsymbol{\xi}_k$. The matrix $\boldsymbol{\Lambda}$ is a diagonal matrix containing these eigenvalues, in the simple case with n distinct eigenvalues.

Change of Coordinates

$$\mathbf{x} = \boldsymbol{\Psi}\mathbf{a}. \tag{0.3}$$

The vector $\mathbf{x} \in \mathbb{R}^n$ may be written as $\mathbf{a} \in \mathbb{R}^n$ in the coordinate system given by the columns of $\boldsymbol{\Psi} \in \mathbb{R}^{n\times n}$.

Measurement Equation

$$\mathbf{y} = \mathbf{C}\mathbf{x}. \tag{0.4}$$

The vector $\mathbf{y} \in \mathbb{R}^p$ is a measurement of the state $\mathbf{x} \in \mathbb{R}^n$ by the measurement matrix $\mathbf{C} \in \mathbb{R}^{p\times n}$.

Singular Value Decomposition

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^* \approx \tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^*. \tag{0.5}$$

The matrix $\mathbf{X} \in \mathbb{C}^{n\times m}$ may be decomposed into the product of three matrices $\mathbf{U} \in \mathbb{C}^{n\times n}$, $\boldsymbol{\Sigma} \in \mathbb{C}^{n\times m}$, and $\mathbf{V} \in \mathbb{C}^{m\times m}$. The matrices $\mathbf{U}$ and $\mathbf{V}$ are *unitary*, so that $\mathbf{U}\mathbf{U}^* = \mathbf{U}^*\mathbf{U} = \mathbf{I}_{n\times n}$ and $\mathbf{V}\mathbf{V}^* = \mathbf{V}^*\mathbf{V} = \mathbf{I}_{m\times m}$, where * denotes complex conjugate transpose. The columns of $\mathbf{U}$ (resp. $\mathbf{V}$) are orthogonal, called left (resp. right) *singular vectors*. The matrix $\boldsymbol{\Sigma}$ contains decreasing, nonnegative diagonal entries called *singular values*.

Often, $\mathbf{X}$ is approximated with a low-rank matrix $\tilde{\mathbf{X}} = \tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^*$, where $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ contain the first $r \ll n$ columns of $\mathbf{U}$ and $\mathbf{V}$, respectively, and $\tilde{\boldsymbol{\Sigma}}$ contains the first $r \times r$ block of $\boldsymbol{\Sigma}$. The matrix $\tilde{\mathbf{U}}$ is often denoted $\boldsymbol{\Psi}$ in the context of spatial modes, reduced order models, and sensor placement.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Regression and Optimization

Overdetermined and Underdetermined Optimization for Linear Systems

$$\underset{\mathbf{x}}{\operatorname{argmin}} \left(\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2 + \lambda g(\mathbf{x})\right) \qquad \text{or} \tag{0.6a}$$

$$\underset{\mathbf{x}}{\operatorname{argmin}}\, g(\mathbf{x}) \;\; \text{subject to} \;\; \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2 \leq \epsilon\,, \tag{0.6b}$$

Here $g(\mathbf{x})$ is a regression penalty (with penalty parameter λ for overdetermined systems). For over- and underdetermined linear systems of equations, which result in either no solutions or an infinite number of solutions of $\mathbf{A}\mathbf{x} = \mathbf{b}$, a choice of constraint or penalty, which is also known as *regularization*, must be made in order to produce a solution.

Overdetermined and Underdetermined Optimization for Nonlinear Systems

$$\underset{\mathbf{x}}{\operatorname{argmin}} \left(f(\mathbf{A},\mathbf{x},\mathbf{b}) + \lambda g(\mathbf{x})\right) \qquad \text{or} \tag{0.7a}$$

$$\underset{\mathbf{x}}{\operatorname{argmin}}\, g(\mathbf{x}) \;\; \text{subject to} \;\; f(\mathbf{A},\mathbf{x},\mathbf{b}) \leq \epsilon \tag{0.7b}$$

This generalizes the linear system to a nonlinear system $f(\cdot)$ with regularization $g(\cdot)$. These over- and underdetermined systems are often solved using gradient descent algorithms.

Compositional Optimization for Neural Networks

$$\underset{\mathbf{A}_j}{\operatorname{argmin}} \big(f_M(\mathbf{A}_M, \cdots f_2(\mathbf{A}_2, (f_1(\mathbf{A}_1,\mathbf{x}))\cdots) + \lambda g(\mathbf{A}_j)\big) \tag{0.8}$$

Each $\mathbf{A}_k$ denotes the weights connecting the neural network from the kth to $(k+1)$th layer. It is typically a massively underdetermined system which is regularized by $g(\mathbf{A}_j)$. Composition and regularization are critical for generating expressive representations of the data as well as preventing overfitting.

Dynamical Systems and Reduced Order Models

Nonlinear Ordinary Differential Equation (Dynamical System)

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), t; \boldsymbol{\beta}). \tag{0.9}$$

The vector $\mathbf{x}(t) \in \mathbb{R}^n$ is the state of the system evolving in time t, $\boldsymbol{\beta}$ are parameters, and $\mathbf{f}$ is the vector field. Generally, $\mathbf{f}$ is Lipschitz continuous to guarantee existence and uniqueness of solutions.

Linear Input–Output System

$$\frac{d}{dt}\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{0.10a}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}. \tag{0.10b}$$

The state of the system is $\mathbf{x} \in \mathbb{R}^n$, the inputs (actuators) are $\mathbf{u} \in \mathbb{R}^q$, and the outputs (sensors) are $\mathbf{y} \in \mathbb{R}^p$. The matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ define the dynamics, the effect of actuation, the sensing strategy, and the effect of actuation feed-through, respectively.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Nonlinear Map (Discrete-Time Dynamical System)

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k). \tag{0.11}$$

The state of the system at the kth iteration is $\mathbf{x}_k \in \mathbb{R}^n$, and $\mathbf{F}$ is a possibly nonlinear mapping. Often, this map defines an iteration forward in time, so that $\mathbf{x}_k = \mathbf{x}(k\Delta t)$; in this case the flow map is denoted $\mathbf{F}_{\Delta t}$.

Koopman Operator Equation (Discrete-Time)

$$\mathcal{K}_t g = g \circ \mathbf{F}_t \quad \Longrightarrow \quad \mathcal{K}_t \varphi = \lambda \varphi. \tag{0.12}$$

The linear Koopman operator $\mathcal{K}_t$ advances measurement functions of the state $g(\mathbf{x})$ with the flow $\mathbf{F}_t$. Eigenvalues and eigenvectors of $\mathcal{K}_t$ are λ and $\varphi(\mathbf{x})$, respectively. The operator $\mathcal{K}_t$ operates on a Hilbert space of measurements.

Nonlinear Partial Differential Equation

$$\mathbf{u}_t = \mathbf{N}(\mathbf{u}, \mathbf{u}_x, \mathbf{u}_{xx}, \cdots, x, t; \boldsymbol{\beta}). \tag{0.13}$$

The state of the PDE is $\mathbf{u}$, the nonlinear evolution operator is $\mathbf{N}$, subscripts denote partial differentiation, and x and t are the spatial and temporal variables, respectively. The PDE is parameterized by values in $\boldsymbol{\beta}$. The state $\mathbf{u}$ of the PDE may be a continuous function $u(x,t)$, or it may be discretized at several spatial locations, $\mathbf{u}(t) = \begin{bmatrix} u(x_1,t) & u(x_2,t) & \cdots & u(x_n,t) \end{bmatrix}^T \in \mathbb{R}^n$.

Galerkin Expansion

The continuous Galerkin expansion is:

$$u(x,t) \approx \sum_{k=1}^{r} a_k(t)\psi_k(x). \tag{0.14}$$

The functions $a_k(t)$ are temporal coefficients that capture the time dynamics, and $\psi_k(x)$ are spatial modes. For a high-dimensional discretized state, the Galerkin expansion becomes: $\mathbf{u}(t) \approx \sum_{k=1}^{r} a_k(t)\boldsymbol{\psi}_k$. The spatial modes $\boldsymbol{\psi}_k \in \mathbb{R}^n$ may be the columns of $\boldsymbol{\Psi} = \tilde{\mathbf{U}}$.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Complete Symbols

Dimensions

K	Number of nonzero entries in a K-sparse vector $\mathbf{s}$
m	Number of data snapshots (i.e., columns of $\mathbf{X}$)
n	Dimension of the state, $\mathbf{x} \in \mathbb{R}^n$
p	Dimension of the measurement or output variable, $\mathbf{y} \in \mathbb{R}^p$
q	Dimension of the input variable, $\mathbf{u} \in \mathbb{R}^q$
r	Rank of truncated SVD, or other low-rank approximation

Scalars

s	Frequency in Laplace domain
t	Time
δ	learning rate in gradient descent
Δt	Time step
x	Spatial variable
Δx	Spatial step
σ	Singular value
λ	Eigenvalue
λ	Sparsity parameter for sparse optimization (Section 7.3)
λ	Lagrange multiplier (Sections. 3.7, 8.4, and 11.4)
τ	Threshold

Vectors

$\mathbf{a}$	Vector of mode amplitudes of $\mathbf{x}$ in basis $\boldsymbol{\Psi}$, $\mathbf{a} \in \mathbb{R}^r$
$\mathbf{b}$	Vector of measurements in linear system $\mathbf{Ax} = \mathbf{b}$
$\mathbf{b}$	Vector of DMD mode amplitudes (Section 7.2)
$\mathbf{Q}$	Vector containing potential function for PDE-FIND
$\mathbf{r}$	Residual error vector
$\mathbf{s}$	Sparse vector, $\mathbf{s} \in \mathbb{R}^n$
$\mathbf{u}$	Control variable (Chapters 8, 9, and 10)
$\mathbf{u}$	PDE state vector (Chapters 11 and 12)
$\mathbf{w}$	Exogenous inputs
$\mathbf{w}_d$	Disturbances to system
$\mathbf{w}_n$	Measurement noise
$\mathbf{w}_r$	Reference to track
$\mathbf{x}$	State of a system, $\mathbf{x} \in \mathbb{R}^n$
$\mathbf{x}_k$	Snapshot of data at time t_k
$\mathbf{x}_j$	Data sample $j \in Z := \{1, 2, \cdots, m\}$ (Chapters 5 and 6)
$\tilde{\mathbf{x}}$	Reduced state, $\tilde{\mathbf{x}} \in \mathbb{R}^r$, so that $\mathbf{x} \approx \tilde{\mathbf{U}}\tilde{\mathbf{x}}$
$\hat{\mathbf{x}}$	Estimated state of a system
$\mathbf{y}$	Vector of measurements, $\mathbf{y} \in \mathbb{R}^p$
$\mathbf{y}_j$	Data label $j \in Z := \{1, 2, \cdots, m\}$ (Chapters 5 and 6)
$\hat{\mathbf{y}}$	Estimated output measurement
$\mathbf{z}$	Transformed state, $\mathbf{x} = \mathbf{Tz}$ (Chapters 8 and 9)
$\boldsymbol{\epsilon}$	Error vector

© in this web service Cambridge University Press www.cambridge.org

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Vectors, continued

$\boldsymbol{\beta}$	Bifurcation parameters
$\boldsymbol{\xi}$	Eigenvector of Koopman operator (Sections 7.4 and 7.5)
$\boldsymbol{\xi}$	Sparse vector of coefficients (Section 7.3)
$\boldsymbol{\phi}$	DMD mode
$\boldsymbol{\psi}$	POD mode
$\boldsymbol{\Upsilon}$	Vector of PDE measurements for PDE-FIND

Matrices

$\mathbf{A}$	Matrix for system of equations or dynamics
$\tilde{\mathbf{A}}$	Reduced dynamics on r-dimensional POD subspace
$\mathbf{A}_{\mathbf{X}}$	Matrix representation of linear dynamics on the state $\mathbf{x}$
$\mathbf{A}_{\mathbf{Y}}$	Matrix representation of linear dynamics on the observables $\mathbf{y}$
$(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{B})$	Matrices for continuous-time state-space system
$(\mathbf{A}_d, \mathbf{B}_d, \mathbf{C}_d, \mathbf{B}_d)$	Matrices for discrete-time state-space system
$(\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{B}})$	Matrices for state-space system in new coordinates $\mathbf{z} = \mathbf{T}^{-1}\mathbf{x}$
$(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}})$	Matrices for reduced state-space system with rank r
$\mathbf{B}$	Actuation input matrix
$\mathbf{C}$	Linear measurement matrix from state to measurements
$\mathcal{C}$	Controllability matrix
$\mathcal{F}$	Discrete Fourier transform
$\mathbf{G}$	Matrix representation of linear dynamics on the states and inputs $[\mathbf{x}^T \mathbf{u}^T]^T$
$\mathbf{H}$	Hankel matrix
$\mathbf{H}'$	Time-shifted Hankel matrix
$\mathbf{I}$	Identity matrix
$\mathbf{K}$	Matrix form of Koopman operator (Chapter 7)
$\mathbf{K}$	Closed-loop control gain (Chapter 8)
$\mathbf{K}_f$	Kalman filter estimator gain
$\mathbf{K}_r$	LQR control gain
$\mathbf{L}$	Low-rank portion of matrix $\mathbf{X}$ (Chapter 3)
$\mathcal{O}$	Observability matrix
$\mathbf{P}$	Unitary matrix that acts on columns of $\mathbf{X}$
$\mathbf{Q}$	Weight matrix for state penalty in LQR (Sec. 8.4)
$\mathbf{Q}$	Orthogonal matrix from QR factorization
$\mathbf{R}$	Weight matrix for actuation penalty in LQR (Sec. 8.4)
$\mathbf{R}$	Upper triangular matrix from QR factorization
$\mathbf{S}$	Sparse portion of matrix $\mathbf{X}$ (Chapter 3)
$\mathbf{T}$	Matrix of eigenvectors (Chapter 8)
$\mathbf{T}$	Change of coordinates (Chapters 8 and 9)
$\mathbf{U}$	Left singular vectors of $\mathbf{X}$, $\mathbf{U} \in \mathbb{R}^{n\times n}$
$\hat{\mathbf{U}}$	Left singular vectors of economy SVD of $\mathbf{X}$, $\mathbf{U} \in \mathbb{R}^{n\times m}$
$\tilde{\mathbf{U}}$	Left singular vectors (POD modes) of truncated SVD of $\mathbf{X}$, $\mathbf{U} \in \mathbb{R}^{n\times r}$
$\mathbf{V}$	Right singular vectors of $\mathbf{X}$, $\mathbf{V} \in \mathbb{R}^{m\times m}$
$\tilde{\mathbf{V}}$	Right singular vectors of truncated SVD of $\mathbf{X}$, $\mathbf{V} \in \mathbb{R}^{m\times r}$

© in this web service Cambridge University Press www.cambridge.org

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Matrices, continued

$\mathbf{\Sigma}$	Matrix of singular values of $\mathbf{X}$, $\mathbf{\Sigma} \in \mathbb{R}^{n\times m}$
$\hat{\mathbf{\Sigma}}$	Matrix of singular values of economy SVD of $\mathbf{X}$, $\mathbf{\Sigma} \in \mathbb{R}^{m\times m}$
$\tilde{\mathbf{\Sigma}}$	Matrix of singular values of truncated SVD of $\mathbf{X}$, $\mathbf{\Sigma} \in \mathbb{R}^{r\times r}$
$\mathbf{W}$	Eigenvectors of $\tilde{\mathbf{A}}$
$\mathbf{W}_c$	Controllability Gramian
$\mathbf{W}_o$	Observability Gramian
$\mathbf{X}$	Data matrix, $\mathbf{X} \in \mathbb{R}^{n\times m}$
$\mathbf{X}'$	Time-shifted data matrix, $\mathbf{X}' \in \mathbb{R}^{n\times m}$
$\mathbf{Y}$	Projection of $\mathbf{X}$ matrix onto orthogonal basis in randomized SVD (Sec. 1.8)
$\mathbf{Y}$	Data matrix of observables, $\mathbf{Y} = \mathbf{g}(\mathbf{X})$, $\mathbf{Y} \in \mathbb{R}^{p\times m}$ (Chapter 7)
$\mathbf{Y}'$	Shifted data matrix of observables, $\mathbf{Y}' = \mathbf{g}(\mathbf{X}')$, $\mathbf{Y}' \in \mathbb{R}^{p\times m}$ (Chapter 7)
$\mathbf{Z}$	Sketch matrix for randomized SVD, $\mathbf{Z} \in \mathbb{R}^{n\times r}$ (Sec. 1.8)
$\mathbf{\Theta}$	Measurement matrix times sparsifying basis, $\mathbf{\Theta} = \mathbf{C}\mathbf{\Psi}$ (Chapter 3)
$\mathbf{\Theta}$	Matrix of candidate functions for SINDy (Sec. 7.3)
$\mathbf{\Gamma}$	Matrix of derivatives of candidate functions for SINDy (Sec. 7.3)
$\mathbf{\Xi}$	Matrix of coefficients of candidate functions for SINDy (Sec. 7.3)
$\mathbf{\Xi}$	Matrix of nonlinear snapshots for DEIM (Sec. 12.5)
$\mathbf{\Lambda}$	Diagonal matrix of eigenvalues
$\mathbf{\Upsilon}$	Input snapshot matrix, $\mathbf{\Upsilon} \in \mathbb{R}^{q\times m}$
$\mathbf{\Phi}$	Matrix of DMD modes, $\mathbf{\Phi} \triangleq \mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{W}$
$\mathbf{\Psi}$	Orthonormal basis (e.g., Fourier or POD modes)

Tensors

$(\mathcal{A}, \mathcal{B}, \mathcal{M})$	N-way array tensors of size $I_1 \times I_2 \times \cdots \times I_N$

Norms

$\|\cdot\|_0$	ℓ_0 pseudo-norm of a vector $\mathbf{x}$ the number of nonzero elements in $\mathbf{x}$
$\|\cdot\|_1$	ℓ_1 norm of a vector $\mathbf{x}$ given by $\|\mathbf{x}\|_1 = \sum_{i=1}^n \lvert x_i \rvert$
$\|\cdot\|_2$	ℓ_2 norm of a vector $\mathbf{x}$ given by $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n (x_i^2)}$
$\|\cdot\|_2$	2-norm of a matrix $\mathbf{X}$ given by $\|\mathbf{X}\|_2 = \max_{\mathbf{x}} \frac{\|\mathbf{X}\mathbf{x}\|_2}{\|\mathbf{x}\|_2}$
$\|\cdot\|_F$	Frobenius norm of a matrix $\mathbf{X}$ given by $\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^m \lvert X_{ij}\rvert^2}$
$\|\cdot\|_*$	Nuclear norm of a matrix $\mathbf{X}$ given by $\|\mathbf{X}\|_* = \text{trace}\left(\sqrt{\mathbf{X}^*\mathbf{X}}\right) = \sum_{i=1}^m \sigma_i$ (for $m \leq n$)
$\langle\cdot,\cdot\rangle$	Inner product. For functions, $\langle f(x), g(x)\rangle = \int_{-\infty}^{\infty} f(x)g^*(x)dx$.
$\langle\cdot,\cdot\rangle$	Inner product. For vectors, $\langle \mathbf{u}, \mathbf{v}\rangle = \mathbf{u}^*\mathbf{v}$.

Operators, Functions, and Maps

$\mathcal{F}$	Fourier transform
$\mathbf{F}$	Discrete-time dynamical system map
$\mathbf{F}_t$	Discrete-time flow map of dynamical system through time t
$\mathbf{f}$	Continuous-time dynamical system
$\mathcal{G}$	Gabor transform

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Operators, Functions, and Maps, continued

$\mathbf{G}$	Transfer function from inputs to outputs (Chapter 8)
g	Scalar measurement function on $\mathbf{x}$
$\mathbf{g}$	Vector-valued measurement functions on $\mathbf{x}$
J	Cost function for control
ℓ	Loss function for support vector machines (Chapter 5)
$\mathcal{K}$	Koopman operator (continuous time)
$\mathcal{K}_t$	Koopman operator associated with time t flow map
$\mathcal{L}$	Laplace transform
$\mathbf{L}$	Loop transfer function (Chapter 8)
$\mathbf{L}$	Linear partial differential equation (Chapters 11 and 12)
$\mathbf{N}$	Nonlinear partial differential equation
$\mathcal{O}$	Order of magnitude
$\mathbf{S}$	Sensitivity function (Chapter 8)
$\mathbf{T}$	Complementary sensitivity function (Chapter 8)
$\mathcal{W}$	Wavelet transform
μ	Incoherence between measurement matrix $\mathbf{C}$ and basis $\boldsymbol{\Psi}$
κ	Condition number
φ	Koopman eigenfunction
∇	Gradient operator
$*$	Convolution operator

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Most Common Acronyms

CNN	Convolutional neural network
DL	Deep learning
DMD	Dynamic mode decomposition
FFT	Fast Fourier transform
ODE	Ordinary differential equation
PCA	Principal components analysis
PDE	Partial differential equation
POD	Proper orthogonal decomposition
ROM	Reduced order model
SVD	Singular value decomposition

Other Acronyms

ADM	Alternating directions method
AIC	Akaike information criterion
ALM	Augmented Lagrange multiplier
ANN	Artificial neural network
ARMA	Autoregressive moving average
ARMAX	Autoregressive moving average with exogenous input
BIC	Bayesian information criterion
BPOD	Balanced proper orthogonal decomposition
DMDc	Dynamic mode decomposition with control
CCA	Canonical correlation analysis
CFD	Computational fluid dynamics
CoSaMP	Compressive sampling matching pursuit
CWT	Continuous wavelet transform
DEIM	Discrete empirical interpolation method
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
DMDc	Dynamic mode decomposition with control
DNS	Direct numerical simulation
DWT	Discrete wavelet transform
ECOG	Electrocorticography
eDMD	Extended DMD
EIM	Empirical interpolation method
EM	Expectation maximization
EOF	Empirical orthogonal functions
ERA	Eigensystem realization algorithm
ESC	Extremum-seeking control
GMM	Gaussian mixture model
HAVOK	Hankel alternative view of Koopman
JL	Johnson–Lindenstrauss
KL	Kullback–Leibler
ICA	Independent component analysis

© in this web service Cambridge University Press www.cambridge.org

Cambridge University Press
978-1-108-42209-3 — Data-Driven Science and Engineering
Steven L. Brunton , J. Nathan Kutz
Frontmatter
More Information

Other Acronyms, continued

KLT	Karhunen–Loève transform
LAD	Least absolute deviations
LASSO	Least absolute shrinkage and selection operator
LDA	Linear discriminant analysis
LQE	Linear quadratic estimator
LQG	Linear quadratic Gaussian controller
LQR	Linear quadratic regulator
LTI	Linear time invariant system
MIMO	Multiple input, multiple output
MLC	Machine learning control
MPE	Missing point estimation
mrDMD	Multi-resolution dynamic mode decomposition
NARMAX	Nonlinear autoregressive model with exogenous inputs
NLS	Nonlinear Schrödinger equation
OKID	Observer Kalman filter identification
PBH	Popov–Belevitch–Hautus test
PCP	Principal component pursuit
PDE-FIND	Partial differential equation functional identification of nonlinear dynamics
PDF	Probability distribution function
PID	Proportional-integral-derivative control
PIV	Particle image velocimetry
RIP	Restricted isometry property
rSVD	Randomized SVD
RKHS	Reproducing kernel Hilbert space
RNN	Recurrent neural network
RPCA	Robust principal components analysis
SGD	Stochastic gradient descent
SINDy	Sparse identification of nonlinear dynamics
SISO	Single input, single output
SRC	Sparse representation for classification
SSA	Singular spectrum analysis
STFT	Short time Fourier transform
STLS	Sequential thresholded least-squares
SVM	Support vector machine
TICA	Time-lagged independent component analysis
VAC	Variational approach of conformation dynamics

© in this web service Cambridge University Press www.cambridge.org