

Machine Learning Refined

Providing a unique approach to machine learning, this text contains fresh and intuitive, yet rigorous, descriptions of all fundamental concepts necessary to conduct research, build products, tinker, and play. By prioritizing geometric intuition, algorithmic thinking, and practical real-world applications in disciplines including computer vision, natural language processing, economics, neuroscience, recommender systems, physics, and biology, this text provides readers with both a lucid understanding of foundational material as well as the practical tools needed to solve real-world problems. With in-depth Python and MATLAB/OCTAVE-based computational exercises and a complete treatment of cutting edge numerical optimization techniques, this is an essential resource for students and an ideal reference for researchers and practitioners working in machine learning, computer science, electrical engineering, signal processing, and numerical optimization.

Key features:

- A presentation built on lucid geometric intuition
- A unique treatment of state-of-the-art numerical optimization techniques
- A fused introduction to logistic regression and support vector machines
- Inclusion of feature design and learning as major topics
- An unparalleled presentation of advanced topics through the lens of function approximation
- A refined description of deep neural networks and kernel methods

Jeremy Watt received his PhD in Computer Science and Electrical Engineering from Northwestern University. His research interests lie in machine learning and computer vision, as well as numerical optimization.

Reza Borhani received his PhD in Computer Science and Electrical Engineering from Northwestern University. His research interests lie in the design and analysis of algorithms for problems in machine learning and computer vision.

Aggelos K. Katsaggelos is a professor and holder of the Joseph Cummings chair in the Department of Electrical Engineering and Computer Science at Northwestern University, where he also heads the Image and Video Processing Laboratory.

Machine Learning Refined

Foundations, Algorithms, and Applications

JEREMY WATT, REZA BORHANI, AND
AGGELOS K. KATSAGGELOS

Northwestern University



CAMBRIDGE
UNIVERSITY PRESS

Cambridge University Press
978-1-107-12352-6 — Machine Learning Refined
Jeremy Watt, Reza Borhani, Aggelos Katsaggelos
Frontmatter
[More Information](#)

CAMBRIDGE UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

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/9781107123526

© Cambridge University Press 2016

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 2016

Printed in the United Kingdom by Clays, St Ives plc

A catalog record for this publication is available from the British Library

Library of Congress Cataloging in Publication data

Names: Watt, Jeremy, author. | Borhani, Reza. | Katsaggelos, Aggelos Konstantinos, 1956-

Title: Machine learning refined : foundations, algorithms, and applications / Jeremy Watt, Reza Borhani, Aggelos Katsaggelos.

Description: New York : Cambridge University Press, 2016.

Identifiers: LCCN 2015041122 | ISBN 9781107123526 (hardback)

Subjects: LCSH: Machine learning.

Classification: LCC Q325.5 .W38 2016 | DDC 006.3/1—dc23

LC record available at <http://lcn.loc.gov/2015041122>

ISBN 978-1-107-12352-6 Hardback

Additional resources for this publication at www.cambridge.org/watt

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

Contents

	<i>Preface</i>	<i>page xi</i>
1	Introduction	1
	1.1 Teaching a computer to distinguish cats from dogs	1
	1.1.1 The pipeline of a typical machine learning problem	5
	1.2 Predictive learning problems	6
	1.2.1 Regression	6
	1.2.2 Classification	9
	1.3 Feature design	12
	1.4 Numerical optimization	15
	1.5 Summary	16
	Part I Fundamental tools and concepts	19
2	Fundamentals of numerical optimization	21
	2.1 Calculus-defined optimality	21
	2.1.1 Taylor series approximations	21
	2.1.2 The first order condition for optimality	22
	2.1.3 The convenience of convexity	24
	2.2 Numerical methods for optimization	26
	2.2.1 The big picture	27
	2.2.2 Stopping condition	27
	2.2.3 Gradient descent	29
	2.2.4 Newton’s method	33
	2.3 Summary	38
	2.4 Exercises	38
3	Regression	45
	3.1 The basics of linear regression	45
	3.1.1 Notation and modeling	45
	3.1.2 The Least Squares cost function for linear regression	47
	3.1.3 Minimization of the Least Squares cost function	48

	3.1.4	The efficacy of a learned model	50
	3.1.5	Predicting the value of new input data	50
3.2		Knowledge-driven feature design for regression	51
	3.2.1	General conclusions	54
3.3		Nonlinear regression and ℓ_2 regularization	56
	3.3.1	Logistic regression	56
	3.3.2	Non-convex cost functions and ℓ_2 regularization	59
3.4		Summary	61
3.5		Exercises	62
4		Classification	73
4.1		The perceptron cost functions	73
	4.1.1	The basic perceptron model	73
	4.1.2	The softmax cost function	75
	4.1.3	The margin perceptron	78
	4.1.4	Differentiable approximations to the margin perceptron	80
	4.1.5	The accuracy of a learned classifier	82
	4.1.6	Predicting the value of new input data	83
	4.1.7	Which cost function produces the best results?	84
	4.1.8	The connection between the perceptron and counting costs	85
4.2		The logistic regression perspective on the softmax cost	86
	4.2.1	Step functions and classification	87
	4.2.2	Convex logistic regression	89
4.3		The support vector machine perspective on the margin perceptron	91
	4.3.1	A quest for the hyperplane with maximum margin	91
	4.3.2	The hard-margin SVM problem	93
	4.3.3	The soft-margin SVM problem	93
	4.3.4	Support vector machines and logistic regression	95
4.4		Multiclass classification	95
	4.4.1	One-versus-all multiclass classification	96
	4.4.2	Multiclass softmax classification	99
	4.4.3	The accuracy of a learned multiclass classifier	103
	4.4.4	Which multiclass classification scheme works best?	104
4.5		Knowledge-driven feature design for classification	104
	4.5.1	General conclusions	106
4.6		Histogram features for real data types	107
	4.6.1	Histogram features for text data	109
	4.6.2	Histogram features for image data	112
	4.6.3	Histogram features for audio data	115
4.7		Summary	117
4.8		Exercises	118

Part II	Tools for fully data-driven machine learning	129
5	Automatic feature design for regression	131
5.1	Automatic feature design for the ideal regression scenario	131
5.1.1	Vector approximation	132
5.1.2	From vectors to continuous functions	133
5.1.3	Continuous function approximation	134
5.1.4	Common bases for continuous function approximation	135
5.1.5	Recovering weights	140
5.1.6	Graphical representation of a neural network	140
5.2	Automatic feature design for the real regression scenario	141
5.2.1	Approximation of discretized continuous functions	142
5.2.2	The real regression scenario	142
5.3	Cross-validation for regression	146
5.3.1	Diagnosing the problem of overfitting/underfitting	149
5.3.2	Hold out cross-validation	149
5.3.3	Hold out calculations	151
5.3.4	k -fold cross-validation	152
5.4	Which basis works best?	155
5.4.1	Understanding of the phenomenon underlying the data	156
5.4.2	Practical considerations	156
5.4.3	When the choice of basis is arbitrary	156
5.5	Summary	158
5.6	Exercises	158
5.7	Notes on continuous function approximation	165
6	Automatic feature design for classification	166
6.1	Automatic feature design for the ideal classification scenario	166
6.1.1	Approximation of piecewise continuous functions	166
6.1.2	The formal definition of an indicator function	168
6.1.3	Indicator function approximation	170
6.1.4	Recovering weights	170
6.2	Automatic feature design for the real classification scenario	171
6.2.1	Approximation of discretized indicator functions	171
6.2.2	The real classification scenario	172
6.2.3	Classifier accuracy and boundary definition	178
6.3	Multiclass classification	179
6.3.1	One-versus-all multiclass classification	179
6.3.2	Multiclass softmax classification	180
6.4	Cross-validation for classification	180
6.4.1	Hold out cross-validation	182
6.4.2	Hold out calculations	182

6.4.3	<i>k</i> -fold cross-validation	184
6.4.4	<i>k</i> -fold cross-validation for one-versus-all multiclass classification	187
6.5	Which basis works best?	187
6.6	Summary	188
6.7	Exercises	189
7	Kernels, backpropagation, and regularized cross-validation	195
7.1	Fixed feature kernels	195
7.1.1	The fundamental theorem of linear algebra	196
7.1.2	Kernelizing cost functions	197
7.1.3	The value of kernelization	197
7.1.4	Examples of kernels	199
7.1.5	Kernels as similarity matrices	201
7.2	The backpropagation algorithm	202
7.2.1	Computing the gradient of a two layer network cost function	203
7.2.2	Three layer neural network gradient calculations	205
7.2.3	Gradient descent with momentum	206
7.3	Cross-validation via ℓ_2 regularization	208
7.3.1	ℓ_2 regularization and cross-validation	209
7.3.2	Regularized <i>k</i> -fold cross-validation for regression	210
7.3.3	Regularized cross-validation for classification	211
7.4	Summary	212
7.5	Further kernel calculations	212
7.5.1	Kernelizing various cost functions	212
7.5.2	Fourier kernel calculations – scalar input	214
7.5.3	Fourier kernel calculations – vector input	215
	Part III Methods for large scale machine learning	217
8	Advanced gradient schemes	219
8.1	Fixed step length rules for gradient descent	219
8.1.1	Gradient descent and simple quadratic surrogates	219
8.1.2	Functions with bounded curvature and optimally conservative step length rules	221
8.1.3	How to use the conservative fixed step length rule	224
8.2	Adaptive step length rules for gradient descent	225
8.2.1	Adaptive step length rule via backtracking line search	226
8.2.2	How to use the adaptive step length rule	227
8.3	Stochastic gradient descent	229
8.3.1	Decomposing the gradient	229
8.3.2	The stochastic gradient descent iteration	230
8.3.3	The value of stochastic gradient descent	232

8.3.4	Step length rules for stochastic gradient descent	233
8.3.5	How to use the stochastic gradient method in practice	234
8.4	Convergence proofs for gradient descent schemes	235
8.4.1	Convergence of gradient descent with Lipschitz constant fixed step length	236
8.4.2	Convergence of gradient descent with backtracking line search	236
8.4.3	Convergence of the stochastic gradient method	238
8.4.4	Convergence rate of gradient descent for convex functions with fixed step length	239
8.5	Calculation of computable Lipschitz constants	241
8.6	Summary	243
8.7	Exercises	243
9	Dimension reduction techniques	245
9.1	Techniques for data dimension reduction	245
9.1.1	Random subsampling	245
9.1.2	K -means clustering	246
9.1.3	Optimization of the K -means problem	249
9.2	Principal component analysis	250
9.2.1	Optimization of the PCA problem	256
9.3	Recommender systems	256
9.3.1	Matrix completion setup	257
9.3.2	Optimization of the matrix completion model	258
9.4	Summary	259
9.5	Exercises	260
	Part IV Appendices	263
A	Basic vector and matrix operations	265
A.1	Vector operations	265
A.2	Matrix operations	266
B	Basics of vector calculus	268
B.1	Basic definitions	268
B.2	Commonly used rules for computing derivatives	269
B.3	Examples of gradient and Hessian calculations	269
C	Fundamental matrix factorizations and the pseudo-inverse	274
C.1	Fundamental matrix factorizations	274
C.1.1	The singular value decomposition	274
C.1.2	Eigenvalue decomposition	276
C.1.3	The pseudo-inverse	277

x	Contents	
D	Convex geometry	278
D.1	Definitions of convexity	278
D.1.1	Zeroth order definition of a convex function	278
D.1.2	First order definition of a convex function	279
	<i>References</i>	280
	<i>Index</i>	285

Preface

In the last decade the user base of machine learning has grown dramatically. From a relatively small circle in computer science, engineering, and mathematics departments the users of machine learning now include students and researchers from every corner of the academic universe, as well as members of industry, data scientists, entrepreneurs, and machine learning enthusiasts. The book before you is the result of a complete tearing down of the standard curriculum of machine learning into its most basic components, and a curated reassembly of those pieces (painstakingly polished and organized) that we feel will most benefit this broadening audience of learners. It contains fresh and intuitive yet rigorous descriptions of the most fundamental concepts necessary to conduct research, build products, tinker, and play.

Intended audience and book pedagogy

This book was written for readers interested in understanding the core concepts of machine learning from first principles to practical implementation. To make full use of the text one only needs a basic understanding of linear algebra and calculus (i.e., vector and matrix operations as well as the ability to compute the gradient and Hessian of a multivariate function), plus some prior exposure to fundamental concepts of computer programming (i.e., conditional and looping structures). It was written for first time learners of the subject, as well as for more knowledgeable readers who yearn for a more intuitive and serviceable treatment than what is currently available today.

To this end, throughout the text, in describing the fundamentals of each concept, we defer the use of probabilistic, statistical, and neurological views of the material in favor of a fresh and consistent geometric perspective. We believe that this not only permits a more intuitive understanding of many core concepts, but helps establish revealing connections between ideas often regarded as fundamentally distinct (e.g., the logistic regression and support vector machine classifiers, kernels, and feed-forward neural networks). We also place significant emphasis on the design and implementation of algorithms, and include many coding exercises for the reader to practice at the end of each chapter. This is because we strongly believe that the bulk of learning this subject takes place when learners “get their hands dirty” and code things up for themselves. In short, with this text we have aimed to create a learning experience for the reader where intuitive leaps precede intellectual ones and are tempered by their application.

What this book is about

The core concepts of our treatment of machine learning can be broadly summarized in four categories. *Predictive learning*, the first of these categories, comprises two kinds of tasks where we aim to either predict a continuous valued phenomenon (like the future location of a celestial body), or distinguish between distinct kinds of things (like different faces in an image). The second core concept, *feature design*, refers to a broad set of engineering and mathematical tools which are crucial to the successful performance of predictive learning models in practice. Throughout the text we will see that features are generated along a spectrum based on the level of our own understanding of a dataset. The third major concept, *function approximation*, is employed when we know too little about a dataset to produce proper features ourselves (and therefore must learn them strictly from the data itself). The final category, *numerical optimization*, powers the first three and is the engine that makes machine learning run in practice.

Overview of the book

This book is separated into three parts, with the latter parts building thematically on each preceding stage.

Part I: Fundamental tools and concepts

Here we detail the fundamentals of predictive modeling, numerical optimization, and feature design. After a general introduction in Chapter 1, Chapter 2 introduces the rudiments of numerical optimization, those critical tools used to properly tune predictive learning models. We then introduce predictive modeling in Chapters 3 and 4, where the regression and classification tasks are introduced, respectively. Along the way we also describe a number of examples where we have some level of knowledge about the underlying process generating the data we receive, which can be leveraged for the design of features.

Part 2: Tools for fully data-driven machine learning

In the absence of useful knowledge about our data we must broaden our perspective in order to design, or learn, features for regression and classification tasks. In Chapters 5 and 6 we review the classical tools of *function approximation*, and see how they are applied to deal with general regression and classification problems. We then end in Chapter 7 by describing several advanced topics related to the material in the preceding two chapters.

Part 3: Methods for large scale machine learning

In the final stage of the book we describe common procedures for scaling regression and classification algorithms to large datasets. We begin in Chapter 8 by introducing

a number of advanced numerical optimization techniques. A continuation of the introduction in Chapter 2, these methods greatly enhance the power of predictive learning by means of more effective optimization algorithms. We then detail in Chapter 9 general techniques for properly lowering the dimension of input data, allowing us to deflate large datasets down to more manageable sizes.

Readers: how to use this book

As mentioned earlier, the only technical prerequisites for the effective use of this book are a basic understanding of linear algebra and vector calculus, as advanced concepts are introduced as necessary throughout the text, as well as some prior computer programming experience. Readers can find a brief tutorial on the Python and MATLAB/OCTAVE programming environments used for completing coding exercises, which introduces proper syntax for both languages as well as necessary libraries to download (for Python) as well as useful built-in functions (for MATLAB/OCTAVE), on the book website.

For self-study one may read all the chapters in order, as each builds on its direct predecessor. However, a solid understanding of the first six chapters is sufficient preparation for perusing individual topics of interest in the final three chapters of the text.

Instructors: how to use this book

The contents of this book have been used for a number of courses at Northwestern University, ranging from an introductory course for senior level undergraduate and beginning graduate students, to a specialized course on advanced numerical optimization for an audience largely consisting of PhD students. Therefore, with its treatment of foundations, applications, and algorithms this book is largely self-contained and can be used for a variety of machine learning courses. For example, it may be used as the basis for:

A single quarter or semester long senior undergraduate/beginning graduate level introduction to standard machine learning topics. This includes coverage of basic techniques from numerical optimization, regression/classification techniques and applications, elements of feature design and learning, and feed-forward neural networks. Chapters 1–6 provide the basis for such a course, with Chapters 7 and 9 (on kernel methods and dimension reduction/unsupervised learning techniques) being optimal add-ons.

A single quarter or semester long senior level undergraduate/graduate course on large scale optimization for machine learning. Chapters 2 and 6–8 provide the basis for a course on introductory and advanced optimization techniques for solving the applications and models introduced in the first two-thirds of the book.