

## Information Science and Statistics

Series Editors:

M. Jordan

J. Kleinberg

B. Schölkopf

#### **Information Science and Statistics**

Akaike and Kitagawa: The Practice of Time Series Analysis.

Bishop: Pattern Recognition and Machine Learning.

Cowell, Dawid, Lauritzen, and Spiegelhalter: Probabilistic Networks and

Expert Systems.

Doucet, de Freitas, and Gordon: Sequential Monte Carlo Methods in Practice.

Fine: Feedforward Neural Network Methodology.

Hawkins and Olwell: Cumulative Sum Charts and Charting for Quality Improvement.

Jensen: Bayesian Networks and Decision Graphs.

Marchette: Computer Intrusion Detection and Network Monitoring:

A Statistical Viewpoint.

Rubinstein and Kroese: The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte Carlo Simulation, and Machine Learning.

Studený: Probabilistic Conditional Independence Structures.

Vapnik: The Nature of Statistical Learning Theory, Second Edition.

Wallace: Statistical and Inductive Inference by Minimum Massage Length.

# Pattern Recognition and Machine Learning



Christopher M. Bishop F.R.Eng. Assistant Director Microsoft Research Ltd Cambridge CB3 0FB, U.K. cmbishop@microsoft.com http://research.microsoft.com/~cmbishop

Series Editors
Michael Jordan
Department of Computer
Science and Department
of Statistics
University of California,
Berkeley
Berkeley, CA 94720
USA

Professor Jon Kleinberg Department of Computer Science Cornell University Ithaca, NY 14853 USA Bernhard Schölkopf Max Planck Institute for Biological Cybernetics Spemannstrasse 38 72076 Tübingen Germany

Library of Congress Control Number: 2006922522

ISBN-10: 0-387-31073-8 ISBN-13: 978-0387-31073-2

Printed on acid-free paper.

#### © 2006 Springer Science+Business Media, LLC

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed in Singapore. (KYO)

987654321

springer.com

# This book is dedicated to my family: Jenna, Mark, and Hugh



Total eclipse of the sun, Antalya, Turkey, 29 March 2006.

#### **Preface**

Pattern recognition has its origins in engineering, whereas machine learning grew out of computer science. However, these activities can be viewed as two facets of the same field, and together they have undergone substantial development over the past ten years. In particular, Bayesian methods have grown from a specialist niche to become mainstream, while graphical models have emerged as a general framework for describing and applying probabilistic models. Also, the practical applicability of Bayesian methods has been greatly enhanced through the development of a range of approximate inference algorithms such as variational Bayes and expectation propagation. Similarly, new models based on kernels have had significant impact on both algorithms and applications.

This new textbook reflects these recent developments while providing a comprehensive introduction to the fields of pattern recognition and machine learning. It is aimed at advanced undergraduates or first year PhD students, as well as researchers and practitioners, and assumes no previous knowledge of pattern recognition or machine learning concepts. Knowledge of multivariate calculus and basic linear algebra is required, and some familiarity with probabilities would be helpful though not essential as the book includes a self-contained introduction to basic probability theory.

Because this book has broad scope, it is impossible to provide a complete list of references, and in particular no attempt has been made to provide accurate historical attribution of ideas. Instead, the aim has been to give references that offer greater detail than is possible here and that hopefully provide entry points into what, in some cases, is a very extensive literature. For this reason, the references are often to more recent textbooks and review articles rather than to original sources.

The book is supported by a great deal of additional material, including lecture slides as well as the complete set of figures used in the book, and the reader is encouraged to visit the book web site for the latest information:

 $http://research.microsoft.com/{\sim}cmbishop/PRML$ 

#### **Exercises**

The exercises that appear at the end of every chapter form an important component of the book. Each exercise has been carefully chosen to reinforce concepts explained in the text or to develop and generalize them in significant ways, and each is graded according to difficulty ranging from  $(\star)$ , which denotes a simple exercise taking a few minutes to complete, through to  $(\star \star \star)$ , which denotes a significantly more complex exercise.

It has been difficult to know to what extent these solutions should be made widely available. Those engaged in self study will find worked solutions very beneficial, whereas many course tutors request that solutions be available only via the publisher so that the exercises may be used in class. In order to try to meet these conflicting requirements, those exercises that help amplify key points in the text, or that fill in important details, have solutions that are available as a PDF file from the book web site. Such exercises are denoted by <a href="https://www.solutions.org/www.solutions">www.solutions</a> for the remaining exercises are available to course tutors by contacting the publisher (contact details are given on the book web site). Readers are strongly encouraged to work through the exercises unaided, and to turn to the solutions only as required.

Although this book focuses on concepts and principles, in a taught course the students should ideally have the opportunity to experiment with some of the key algorithms using appropriate data sets. A companion volume (Bishop and Nabney, 2008) will deal with practical aspects of pattern recognition and machine learning, and will be accompanied by Matlab software implementing most of the algorithms discussed in this book.

#### **Acknowledgements**

First of all I would like to express my sincere thanks to Markus Svensén who has provided immense help with preparation of figures and with the typesetting of the book in LaTeX. His assistance has been invaluable.

I am very grateful to Microsoft Research for providing a highly stimulating research environment and for giving me the freedom to write this book (the views and opinions expressed in this book, however, are my own and are therefore not necessarily the same as those of Microsoft or its affiliates).

Springer has provided excellent support throughout the final stages of preparation of this book, and I would like to thank my commissioning editor John Kimmel for his support and professionalism, as well as Joseph Piliero for his help in designing the cover and the text format and MaryAnn Brickner for her numerous contributions during the production phase. The inspiration for the cover design came from a discussion with Antonio Criminisi.

I also wish to thank Oxford University Press for permission to reproduce excerpts from an earlier textbook, *Neural Networks for Pattern Recognition* (Bishop, 1995a). The images of the Mark 1 perceptron and of Frank Rosenblatt are reproduced with the permission of Arvin Calspan Advanced Technology Center. I would also like to thank Asela Gunawardana for plotting the spectrogram in Figure 13.1, and Bernhard Schölkopf for permission to use his kernel PCA code to plot Figure 12.17.

Many people have helped by proofreading draft material and providing comments and suggestions, including Shivani Agarwal, Cédric Archambeau, Arik Azran, Andrew Blake, Hakan Cevikalp, Michael Fourman, Brendan Frey, Zoubin Ghahramani, Thore Graepel, Katherine Heller, Ralf Herbrich, Geoffrey Hinton, Adam Johansen, Matthew Johnson, Michael Jordan, Eva Kalyvianaki, Anitha Kannan, Julia Lasserre, David Liu, Tom Minka, Ian Nabney, Tonatiuh Pena, Yuan Qi, Sam Roweis, Balaji Sanjiya, Toby Sharp, Ana Costa e Silva, David Spiegelhalter, Jay Stokes, Tara Symeonides, Martin Szummer, Marshall Tappen, Ilkay Ulusoy, Chris Williams, John Winn, and Andrew Zisserman.

Finally, I would like to thank my wife Jenna who has been hugely supportive throughout the several years it has taken to write this book.

Chris Bishop Cambridge February 2006

#### **Mathematical notation**

I have tried to keep the mathematical content of the book to the minimum necessary to achieve a proper understanding of the field. However, this minimum level is nonzero, and it should be emphasized that a good grasp of calculus, linear algebra, and probability theory is essential for a clear understanding of modern pattern recognition and machine learning techniques. Nevertheless, the emphasis in this book is on conveying the underlying concepts rather than on mathematical rigour.

I have tried to use a consistent notation throughout the book, although at times this means departing from some of the conventions used in the corresponding research literature. Vectors are denoted by lower case bold Roman letters such as  $\mathbf{x}$ , and all vectors are assumed to be column vectors. A superscript T denotes the transpose of a matrix or vector, so that  $\mathbf{x}^T$  will be a row vector. Uppercase bold roman letters, such as  $\mathbf{M}$ , denote matrices. The notation  $(w_1,\ldots,w_M)$  denotes a row vector with M elements, while the corresponding column vector is written as  $\mathbf{w} = (w_1,\ldots,w_M)^T$ .

The notation [a,b] is used to denote the *closed* interval from a to b, that is the interval including the values a and b themselves, while (a,b) denotes the corresponding *open* interval, that is the interval excluding a and b. Similarly, [a,b) denotes an interval that includes a but excludes b. For the most part, however, there will be little need to dwell on such refinements as whether the end points of an interval are included or not.

The  $M \times M$  identity matrix (also known as the unit matrix) is denoted  $\mathbf{I}_M$ , which will be abbreviated to  $\mathbf{I}$  where there is no ambiguity about it dimensionality. It has elements  $I_{ij}$  that equal 1 if i = j and 0 if  $i \neq j$ .

A functional is denoted f[y] where y(x) is some function. The concept of a functional is discussed in Appendix D.

The notation g(x) = O(f(x)) denotes that |f(x)/g(x)| is bounded as  $x \to \infty$ . For instance if  $g(x) = 3x^2 + 2$ , then  $g(x) = O(x^2)$ .

The expectation of a function f(x,y) with respect to a random variable x is denoted by  $\mathbb{E}_x[f(x,y)]$ . In situations where there is no ambiguity as to which variable is being averaged over, this will be simplified by omitting the suffix, for instance

#### xii MATHEMATICAL NOTATION

 $\mathbb{E}[x]$ . If the distribution of x is conditioned on another variable z, then the corresponding conditional expectation will be written  $\mathbb{E}_x[f(x)|z]$ . Similarly, the variance is denoted var[f(x)], and for vector variables the covariance is written  $\text{cov}[\mathbf{x}, \mathbf{y}]$ . We shall also use  $\text{cov}[\mathbf{x}]$  as a shorthand notation for  $\text{cov}[\mathbf{x}, \mathbf{x}]$ . The concepts of expectations and covariances are introduced in Section 1.2.2.

If we have N values  $\mathbf{x}_1, \dots, \mathbf{x}_N$  of a D-dimensional vector  $\mathbf{x} = (x_1, \dots, x_D)^T$ , we can combine the observations into a data matrix  $\mathbf{X}$  in which the  $n^{\text{th}}$  row of  $\mathbf{X}$  corresponds to the row vector  $\mathbf{x}_n^T$ . Thus the n, i element of  $\mathbf{X}$  corresponds to the  $i^{\text{th}}$  element of the  $n^{\text{th}}$  observation  $\mathbf{x}_n$ . For the case of one-dimensional variables we shall denote such a matrix by  $\mathbf{X}$ , which is a column vector whose  $n^{\text{th}}$  element is  $x_n$ . Note that  $\mathbf{X}$  (which has dimensionality N) uses a different typeface to distinguish it from  $\mathbf{x}$  (which has dimensionality D).

### **Contents**

Pr	eface			vii
M	athen	natical r	notation	xi
Co	onten	ts		xiii
1	Inti	oductio	on	1
	1.1	Exam	ple: Polynomial Curve Fitting	4
	1.2	Probal	bility Theory	12
		1.2.1	Probability densities	
		1.2.2	Expectations and covariances	19
		1.2.3	Bayesian probabilities	21
		1.2.4	The Gaussian distribution	24
		1.2.5	Curve fitting re-visited	28
		1.2.6	Bayesian curve fitting	30
	1.3	Model	l Selection	32
	1.4	The C	Curse of Dimensionality	33
	1.5	Decisi	ion Theory	38
		1.5.1	Minimizing the misclassification rate	39
		1.5.2	Minimizing the expected loss	41
		1.5.3	The reject option	42
		1.5.4	Inference and decision	42
		1.5.5	Loss functions for regression	46
	1.6	Inforn	nation Theory	48
		1.6.1	Relative entropy and mutual information	55
	Exe	cises .		58

#### xiv CONTENTS

2	Pro	bability	Distributions 67
	2.1	Binar	y Variables
		2.1.1	The beta distribution
	2.2	Multi	nomial Variables
		2.2.1	The Dirichlet distribution
	2.3	The C	Gaussian Distribution
		2.3.1	Conditional Gaussian distributions
		2.3.2	Marginal Gaussian distributions
		2.3.3	Bayes' theorem for Gaussian variables
		2.3.4	Maximum likelihood for the Gaussian
		2.3.5	Sequential estimation
		2.3.6	Bayesian inference for the Gaussian
		2.3.7	Student's t-distribution
		2.3.8	Periodic variables
		2.3.9	Mixtures of Gaussians
	2.4		Exponential Family
		2.4.1	Maximum likelihood and sufficient statistics
		2.4.2	Conjugate priors
		2.4.3	Noninformative priors
	2.5	Nonp	arametric Methods
		2.5.1	Kernel density estimators
		2.5.2	Nearest-neighbour methods
	Exe		
3			dels for Regression 137
	3.1	Linea	r Basis Function Models
		3.1.1	Maximum likelihood and least squares 140
		3.1.2	Geometry of least squares
		3.1.3	Sequential learning
		3.1.4	Regularized least squares
		3.1.5	Multiple outputs
	3.2	The B	Sias-Variance Decomposition
	3.3	Bayes	sian Linear Regression
		3.3.1	Parameter distribution
		3.3.2	Predictive distribution
		3.3.3	Equivalent kernel
	3.4	Bayes	sian Model Comparison
	3.5		vidence Approximation
		3.5.1	Evaluation of the evidence function 166
		3.5.2	Maximizing the evidence function
		3.5.3	Effective number of parameters
	3.6	Limit	ations of Fixed Basis Functions
	Exer	cises	173

				CONTENTS	XV
4	Lin	ear Mo	odels for Classification		179
•	4.1		iminant Functions		
		4.1.1	Two classes		
		4.1.2	Multiple classes		
		4.1.3	Least squares for classification		
		4.1.4	Fisher's linear discriminant		
		4.1.5	Relation to least squares		
		4.1.6	Fisher's discriminant for multiple classe		
		4.1.7	The perceptron algorithm		
	4.2		abilistic Generative Models		
	1.2	4.2.1	Continuous inputs		
		4.2.2	Maximum likelihood solution		
		4.2.3	Discrete features		. 202
		4.2.4	Exponential family		
	4.3		abilistic Discriminative Models		
	т.Э	4.3.1	Fixed basis functions		
		4.3.2	Logistic regression		
		4.3.3	Iterative reweighted least squares		
		4.3.4	Multiclass logistic regression		
		4.3.5	Probit regression		. 210
		4.3.6	Canonical link functions		
	4.4		Laplace Approximation		
	7.7	4.4.1	Model comparison and BIC		
	4.5		sian Logistic Regression		
	т.Э	4.5.1	Laplace approximation		
		4.5.2	Predictive distribution		. 218
	Exe	cises			
5		ıral Ne			225
	5.1		forward Network Functions		
		5.1.1	Weight-space symmetries		
	5.2		ork Training		. 232
		5.2.1	Parameter optimization		
		5.2.2	Local quadratic approximation		
		5.2.3	Use of gradient information		
		5.2.4	Gradient descent optimization		
	5.3		Backpropagation		
		5.3.1	Evaluation of error-function derivatives		
		5.3.2	A simple example		
		5.3.3	Efficiency of backpropagation		. 246
		5.3.4	The Jacobian matrix		
	5.4		Hessian Matrix		
		5.4.1	Diagonal approximation		
		5.4.2	Outer product approximation		
		5.4.3	Inverse Hessian		. 252

#### xvi CONTENTS

		5.4.4	Finite differences
		5.4.5	Exact evaluation of the Hessian
		5.4.6	Fast multiplication by the Hessian
	5.5	Regul	arization in Neural Networks
		5.5.1	Consistent Gaussian priors
		5.5.2	Early stopping
		5.5.3	Invariances
		5.5.4	Tangent propagation
		5.5.5	Training with transformed data
		5.5.6	Convolutional networks
		5.5.7	Soft weight sharing
	5.6		re Density Networks
	5.7	Bayes	ian Neural Networks
	5.7	5.7.1	Posterior parameter distribution
		5.7.2	Hyperparameter optimization
		5.7.3	Bayesian neural networks for classification
	Exer	cises .	284
	Later		20
6	Ker	nel Mei	thods 291
	6.1	Dual l	Representations
	6.2	Const	ructing Kernels
	6.3	Radia	l Basis Function Networks
		6.3.1	Nadaraya-Watson model
	6.4	Gauss	ian Processes
		6.4.1	Linear regression revisited
		6.4.2	Gaussian processes for regression
		6.4.3	Learning the hyperparameters
		6.4.4	Automatic relevance determination
		6.4.5	Gaussian processes for classification
		6.4.6	Laplace approximation
		6.4.7	Connection to neural networks
	Exer	cises .	320
_			
7	-		rnel Machines 325
	7.1		num Margin Classifiers
		7.1.1	Overlapping class distributions
		7.1.2	Relation to logistic regression
			Multiclass SVMs
		7.1.4	SVMs for regression
		7.1.5	Computational learning theory
	7.2		ance Vector Machines
		7.2.1	RVM for regression
		7.2.2	Analysis of sparsity
		7.2.3	RVM for classification
	Evar	cicac	357

				CONTENTS	xvii
8	Gra	phical l	Models		359
	8.1		ian Networks		
	-	8.1.1	Example: Polynomial regression		
		8.1.2	Generative models		. 365
		8.1.3	Discrete variables		. 366
		8.1.4			. 370
	8.2		tional Independence		. 372
		8.2.1	Three example graphs		. 373
		8.2.2	D-separation		. 378
	8.3		ov Random Fields		. 383
		8.3.1	Conditional independence properties .		. 383
		8.3.2	Factorization properties		. 384
		8.3.3	Illustration: Image de-noising		. 387
		8.3.4	Relation to directed graphs		. 390
	8.4	Infere	nce in Graphical Models		. 393
		8.4.1	Inference on a chain		. 394
		8.4.2	Trees		. 398
		8.4.3	Factor graphs		. 399
		8.4.4	The sum-product algorithm		. 402
		8.4.5	The max-sum algorithm		. 411
		8.4.6	Exact inference in general graphs		. 416
		8.4.7	Loopy belief propagation		. 417
		8.4.8	Learning the graph structure		. 418
	Exer	cises .			. 418
9	Mix	ture M	odels and EM		423
	9.1		ans Clustering		
		9.1.1	Image segmentation and compression		
	9.2	Mixtu	res of Gaussians		. 430
		9.2.1	Maximum likelihood		. 432
		9.2.2	EM for Gaussian mixtures		. 435
	9.3	An Al	ternative View of EM		. 439
		9.3.1			. 441
		9.3.2			. 443
		9.3.3	Mixtures of Bernoulli distributions		. 444
		9.3.4	EM for Bayesian linear regression		
	9.4		M Algorithm in General		
	Exer				
10	Ann	roxima	ate Inference		461
	10.1		ional Inference		
	10.1	10.1.1	Factorized distributions		
		10.1.2			
		10.1.3			
			Model comparison		
	10.2	Illustr	ation: Variational Mixture of Gaussians		. 474

#### xviii CONTENTS

			./5
		10.2.2 Variational lower bound	81
			82
			83
			85
	10.3		86
	10.0	$\boldsymbol{\mathcal{C}}$	86
			.88
			.89
	10.4		.90
	10.7		.91
	10.5		.93
	10.5		.98
	10.0		.98
			90 00
	10.7	V 1 1	02
	10.7		05
		1 1	11
	_		13
	Exerc	cises	17
11	Sam	pling Methods 5	23
			26
			26
			28
		11.1.3 Adaptive rejection sampling	30
			32
			34
			36
	11 2		37
	11.2		39
			41
	11 2	Gibbs Sampling	42
	11.3	1 6	46
	11.5	1 6	48
	11.3		
			48
	11.6		52
		E	54
	Exerc	cises	56
12	Con	tinuous Latent Variables 5	59
	12.1	Principal Component Analysis	61
			61
		12.1.2 Minimum-error formulation	63
			65
		12.1.4 PCA for high-dimensional data	

12.2 Probabilistic PCA	
12.2.1 Maximum likelihood PCA	
12.2.2 EM algorithm for PCA	
12.2.3 Bayesian PCA	
12.2.4 Factor analysis	
12.3 Kernel PCA	
12.4 Nonlinear Latent Variable Models	
12.4.1 Independent component analysis	
12.4.2 Autoassociative neural networks	
12.4.3 Modelling nonlinear manifolds	
Exercises	
13 Sequential Data	
13.1 Markov Models	
13.2 Hidden Markov Models	
13.2.1 Maximum likelihood for the HMM	
13.2.2 The forward-backward algorithm	
13.2.3 The sum-product algorithm for the HMM	
13.2.4 Scaling factors	
13.2.5 The Viterbi algorithm	
13.2.6 Extensions of the hidden Markov model	
13.3 Linear Dynamical Systems	
13.3.1 Inference in LDS	
13.3.2 Learning in LDS	
13.3.3 Extensions of LDS	
13.3.4 Particle filters	
Exercises	
14 Combining Models	
14.1 Bayesian Model Averaging	
14.2 Committees	
14.3 Boosting	
14.3.1 Minimizing exponential error	
14.3.2 Error functions for boosting	
14.4 Tree-based Models	
14.5 Conditional Mixture Models	
14.5.1 Mixtures of linear regression models	
14.5.2 Mixtures of logistic models	
14.5.3 Mixtures of experts	
Exercises	
Appendix A Data Sets	
A P . D . D . L . L . 124 . D . 4 . 1 . 4	
Appendix B Probability Distributions	

#### **XX** CONTENTS

Appendix D	Calculus of Variations	703
Appendix E	Lagrange Multipliers	707
References		711
Index		729