C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006,
ISBN 026218253X. www.GaussianProcess.org/gpml

Gaussian Processes for Machine Learning

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006,
ISBN 026218253X. www.GaussianProcess.org/gpml

Adaptive Computation and Machine Learning
Thomas Dietterich, Editor
Christopher Bishop, David Heckerman, Michael Jordan, and Michael Kearns, Associate Editors

Bioinformatics: The Machine Learning Approach,
Pierre Baldi and Søren Brunak

Reinforcement Learning: An Introduction,
Richard S. Sutton and Andrew G. Barto

Graphical Models for Machine Learning and Digital Communication,
Brendan J. Frey

Learning in Graphical Models,
Michael I. Jordan

Causation, Prediction, and Search, second edition,
Peter Spirtes, Clark Glymour, and Richard Scheines

Principles of Data Mining,
David Hand, Heikki Mannila, and Padhraic Smyth

Bioinformatics: The Machine Learning Approach, second edition,
Pierre Baldi and Søren Brunak

Learning Kernel Classifiers: Theory and Algorithms,
Ralf Herbrich

Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond,
Bernhard Schölkopf and Alexander J. Smola

Introduction to Machine Learning,
Ethem Alpaydin

Gaussian Processes for Machine Learning,
Carl Edward Rasmussen and Christopher K. I. Williams

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006,
ISBN 026218253X. www.GaussianProcess.org/gpml

Gaussian Processes for Machine Learning

Carl Edward Rasmussen
Christopher K. I. Williams

The MIT Press
Cambridge, Massachusetts
London, England

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. © 2006 Massachusetts Institute of Technology. www.GaussianProcess.org/gpml

Typeset by the authors using LaTeX 2_ε.
This book was printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data

Rasmussen, Carl Edward.
Gaussian processes for machine learning / Carl Edward Rasmussen, Christopher K. I. Williams.
p. cm. —(Adaptive computation and machine learning)
Includes bibliographical references and indexes.
ISBN 0-262-18253-X
1. Gaussian processes—Data processing. 2. Machine learning—Mathematical models.
I. Williams, Christopher K. I. II. Title. III. Series.

QA274.4.R37 2006
519.2'3—dc22

2005053433

10 9 8 7 6 5 4 3 2

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

> The actual science of logic is conversant at present only with things either certain, impossible, or entirely doubtful, none of which (fortunately) we have to reason on. Therefore the true logic for this world is the calculus of Probabilities, which takes account of the magnitude of the probability which is, or ought to be, in a reasonable man's mind.
>
> — James Clerk Maxwell [1850]

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

Contents

∗Sections marked by an asterisk contain advanced material that may be omitted on a first reading.

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

Series Foreword

The goal of building systems that can adapt to their environments and learn from their experience has attracted researchers from many fields, including computer science, engineering, mathematics, physics, neuroscience, and cognitive science. Out of this research has come a wide variety of learning techniques that have the potential to transform many scientific and industrial fields. Recently, several research communities have converged on a common set of issues surrounding supervised, unsupervised, and reinforcement learning problems. The MIT Press series on Adaptive Computation and Machine Learning seeks to unify the many diverse strands of machine learning research and to foster high quality research and innovative applications.

One of the most active directions in machine learning has been the development of practical Bayesian methods for challenging learning problems. *Gaussian Processes for Machine Learning* presents one of the most important Bayesian machine learning approaches based on a particularly effective method for placing a prior distribution over the space of functions. Carl Edward Rasmussen and Chris Williams are two of the pioneers in this area, and their book describes the mathematical foundations and practical application of Gaussian processes in regression and classification tasks. They also show how Gaussian processes can be interpreted as a Bayesian version of the well-known support vector machine methods. Students and researchers who study this book will be able to apply Gaussian process methods in creative ways to solve a wide range of problems in science and engineering.

Thomas Dietterich

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

Preface

kernel machines

Over the last decade there has been an explosion of work in the "kernel machines" area of machine learning. Probably the best known example of this is work on support vector machines, but during this period there has also been much activity concerning the application of Gaussian process models to machine learning tasks. The goal of this book is to provide a systematic and unified treatment of this area. Gaussian processes provide a principled, practical, probabilistic approach to learning in kernel machines. This gives advantages with respect to the interpretation of model predictions and provides a well-founded framework for learning and model selection. Theoretical and practical developments of over the last decade have made Gaussian processes a serious competitor for real supervised learning applications.

Gaussian process

Roughly speaking a stochastic *process* is a generalization of a probability distribution (which describes a finite-dimensional random variable) to *functions*. By focussing on processes which are *Gaussian*, it turns out that the computations required for inference and learning become relatively easy. Thus, the supervised learning problems in machine learning which can be thought of as learning a function from examples can be cast directly into the Gaussian process framework.

Gaussian processes in machine learning

Our interest in Gaussian process (GP) models in the context of machine learning was aroused in 1994, while we were both graduate students in Geoff Hinton's Neural Networks lab at the University of Toronto. This was a time when the field of neural networks was becoming mature and the many connections to statistical physics, probabilistic models and statistics became well known, and the first kernel-based learning algorithms were becoming popular. In retrospect it is clear that the time was ripe for the application of Gaussian processes to machine learning problems.

neural networks

Many researchers were realizing that neural networks were not so easy to apply in practice, due to the many decisions which needed to be made: what architecture, what activation functions, what learning rate, etc., and the lack of a principled framework to answer these questions. The probabilistic framework was pursued using approximations by MacKay [1992b] and using Markov chain Monte Carlo (MCMC) methods by Neal [1996]. Neal was also a graduate student in the same lab, and in his thesis he sought to demonstrate that using the Bayesian formalism, one does not necessarily have problems with "overfitting" when the models get large, and one should pursue the limit of large models. While his own work was focused on sophisticated Markov chain methods for inference in large finite networks, he did point out that some of his networks became Gaussian processes in the limit of infinite size, and "there may be simpler ways to do inference in this case."

large neural networks ≡ Gaussian processes

adaptive basis functions

It is perhaps interesting to mention a slightly wider historical perspective. The main reason why neural networks became popular was that they allowed the use of *adaptive* basis functions, as opposed to the well known linear models. The adaptive basis functions, or hidden units, could "learn" hidden features

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

useful for the modelling problem at hand. However, this adaptivity came at the cost of a lot of practical problems. Later, with the advancement of the "kernel era", it was realized that the limitation of fixed basis functions is not a big restriction if only one has enough of them, i.e. typically infinitely many, and one is careful to control problems of overfitting by using priors or regularization. The resulting models are much easier to handle than the adaptive basis function models, but have similar expressive power.

many fixed basis functions

Thus, one could claim that (as far a machine learning is concerned) the adaptive basis functions were merely a decade-long digression, and we are now back to where we came from. This view is perhaps reasonable if we think of models for solving practical learning problems, although MacKay [2003, ch. 45], for example, raises concerns by asking "did we throw out the baby with the bath water?", as the kernel view does not give us any hidden representations, telling us what the useful features are for solving a particular problem. As we will argue in the book, one answer may be to learn more sophisticated covariance functions, and the "hidden" properties of the problem are to be found here. An important area of future developments for GP models is the use of more expressive covariance functions.

useful representations

supervised learning in statistics

Supervised learning problems have been studied for more than a century in statistics, and a large body of well-established theory has been developed. More recently, with the advance of affordable, fast computation, the machine learning community has addressed increasingly large and complex problems.

statistics and machine learning

data and models

algorithms and predictions

Much of the basic theory and many algorithms are shared between the statistics and machine learning community. The primary differences are perhaps the types of the problems attacked, and the goal of learning. At the risk of oversimplification, one could say that in statistics a prime focus is often in understanding the *data* and relationships in terms of *models* giving approximate summaries such as linear relations or independencies. In contrast, the goals in machine learning are primarily to make predictions as accurately as possible and to understand the behaviour of learning *algorithms*. These differing objectives have led to different developments in the two fields: for example, neural network algorithms have been used extensively as black-box function approximators in machine learning, but to many statisticians they are less than satisfactory, because of the difficulties in interpreting such models.

bridging the gap

Gaussian process models in some sense bring together work in the two communities. As we will see, Gaussian processes are mathematically equivalent to many well known models, including Bayesian linear models, spline models, large neural networks (under suitable conditions), and are closely related to others, such as support vector machines. Under the Gaussian process viewpoint, the models may be easier to handle and interpret than their conventional counterparts, such as e.g. neural networks. In the statistics community Gaussian processes have also been discussed many times, although it would probably be excessive to claim that their use is widespread except for certain specific applications such as spatial models in meteorology and geology, and the analysis of computer experiments. A rich theory also exists for Gaussian process models

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

in the time series analysis literature; some pointers to this literature are given in Appendix B.

intended audience

The book is primarily intended for graduate students and researchers in machine learning at departments of Computer Science, Statistics and Applied Mathematics. As prerequisites we require a good basic grounding in calculus, linear algebra and probability theory as would be obtained by graduates in numerate disciplines such as electrical engineering, physics and computer science. For preparation in calculus and linear algebra any good university-level textbook on mathematics for physics or engineering such as Arfken [1985] would be fine. For probability theory some familiarity with multivariate distributions (especially the Gaussian) and conditional probability is required. Some background mathematical material is also provided in Appendix A.

focus

The main focus of the book is to present clearly and concisely an overview of the main ideas of Gaussian processes in a machine learning context. We have also covered a wide range of connections to existing models in the literature, and cover approximate inference for faster practical algorithms. We have presented detailed algorithms for many methods to aid the practitioner. Software implementations are available from the website for the book, see Appendix C. We have also included a small set of exercises in each chapter; we hope these will help in gaining a deeper understanding of the material.

scope

In order limit the size of the volume, we have had to omit some topics, such as, for example, Markov chain Monte Carlo methods for inference. One of the most difficult things to decide when writing a book is what sections not to write. Within sections, we have often chosen to describe one algorithm in particular in depth, and mention related work only in passing. Although this causes the omission of some material, we feel it is the best approach for a monograph, and hope that the reader will gain a general understanding so as to be able to push further into the growing literature of GP models.

book organization

The book has a natural split into two parts, with the chapters up to and including chapter 5 covering core material, and the remaining sections covering the connections to other methods, fast approximations, and more specialized properties. Some sections are marked by an asterisk. These sections may be omitted on a first reading, and are not pre-requisites for later (un-starred) material.

∗

acknowledgements

We wish to express our considerable gratitude to the many people with whom we have interacted during the writing of this book. In particular Moray Allan, David Barber, Peter Bartlett, Miguel Carreira-Perpiñán, Marcus Gallagher, Manfred Opper, Anton Schwaighofer, Matthias Seeger, Hanna Wallach, Joe Whittaker, and Andrew Zisserman all read parts of the book and provided valuable feedback. Dilan Görür, Malte Kuss, Iain Murray, Joaquin Quiñonero-Candela, Leif Rasmussen and Sam Roweis were especially heroic and provided comments on the whole manuscript. We thank Chris Bishop, Miguel Carreira-Perpiñán, Nando de Freitas, Zoubin Ghahramani, Peter Grünwald, Mike Jordan, John Kent, Radford Neal, Joaquin Quiñonero-Candela, Ryan Rifkin, Stefan Schaal, Anton Schwaighofer, Matthias Seeger, Peter Sollich, Ingo Steinwart,

Amos Storkey, Volker Tresp, Sethu Vijayakumar, Grace Wahba, Joe Whittaker and Tong Zhang for valuable discussions on specific issues. We also thank Bob Prior and the staff at MIT Press for their support during the writing of the book. We thank the Gatsby Computational Neuroscience Unit (UCL) and Neil Lawrence at the Department of Computer Science, University of Sheffield for hosting our visits and kindly providing space for us to work, and the Department of Computer Science at the University of Toronto for computer support. Thanks to John and Fiona for their hospitality on numerous occasions. Some of the diagrams in this book have been inspired by similar diagrams appearing in published work, as follows: Figure 3.5, Schölkopf and Smola [2002]; Figure 5.2, MacKay [1992b]. CER gratefully acknowledges financial support from the German Research Foundation (DFG). CKIW thanks the School of Informatics, University of Edinburgh for granting him sabbatical leave for the period October 2003-March 2004.

Finally, we reserve our deepest appreciation for our wives Agnes and Barbara, and children Ezra, Kate, Miro and Ruth for their patience and understanding while the book was being written.

errata

Despite our best efforts it is inevitable that some errors will make it through to the printed version of the book. Errata will be made available via the book's website at

`http://www.GaussianProcess.org/gpml`

We have found the joint writing of this book an excellent experience. Although hard at times, we are confident that the end result is much better than either one of us could have written alone.

looking ahead

Now, ten years after their first introduction into the machine learning community, Gaussian processes are receiving growing attention. Although GPs have been known for a long time in the statistics and geostatistics fields, and their use can perhaps be traced back as far as the end of the 19th century, their application to real problems is still in its early phases. This contrasts somewhat the application of the non-probabilistic analogue of the GP, the support vector machine, which was taken up more quickly by practitioners. Perhaps this has to do with the probabilistic mind-set needed to understand GPs, which is not so generally appreciated. Perhaps it is due to the need for computational short-cuts to implement inference for large datasets. Or it could be due to the lack of a self-contained introduction to this exciting field—with this volume, we hope to contribute to the momentum gained by Gaussian processes in machine learning.

Carl Edward Rasmussen and Chris Williams
Tübingen and Edinburgh, summer 2005

Second printing: We thank Baback Moghaddam, Mikhail Parakhin, Leif Rasmussen, Benjamin Sobotta, Kevin S. Van Horn and Aki Vehtari for reporting errors in the first printing which have now been corrected.

C. E. Rasmussen & C. K. I. Williams, Gaussian Processes for Machine Learning, the MIT Press, 2006, ISBN 026218253X. www.GaussianProcess.org/gpml

Symbols and Notation

Matrices are capitalized and vectors are in bold type. We do not generally distinguish between probabilities and probability densities. A subscript asterisk, such as in X_*, indicates reference to a *test set* quantity. A superscript asterisk denotes complex conjugate.

Symbol	Meaning
$\backslash$	left matrix divide: $A\backslash\mathbf{b}$ is the vector $\mathbf{x}$ which solves $A\mathbf{x}=\mathbf{b}$
$\triangleq$	an equality which acts as a definition
$\overset{c}{=}$	equality up to an additive constant
$\lvert K\rvert$	determinant of K matrix
$\lvert\mathbf{y}\rvert$	Euclidean length of vector $\mathbf{y}$, i.e. $\left(\sum_i y_i^2\right)^{1/2}$
$\langle f, g\rangle_{\mathcal{H}}$	RKHS inner product
$\lVert f\rVert_{\mathcal{H}}$	RKHS norm
$\mathbf{y}^\top$	the transpose of vector $\mathbf{y}$
$\propto$	proportional to; e.g. $p(x\lvert y)\propto f(x,y)$ means that $p(x\lvert y)$ is equal to $f(x,y)$ times a factor which is independent of x
$\sim$	distributed according to; example: $x\sim\mathcal{N}(\mu,\sigma^2)$
∇ or $\nabla_{\mathbf{f}}$	partial derivatives (w.r.t. $\mathbf{f}$)
$\nabla\nabla$	the (Hessian) matrix of second derivatives
$\mathbf{0}$ or $\mathbf{0}_n$	vector of all 0's (of length n)
$\mathbf{1}$ or $\mathbf{1}_n$	vector of all 1's (of length n)
C	number of classes in a classification problem
cholesky(A)	Cholesky decomposition: L is a lower triangular matrix such that $LL^\top=A$
cov$(\mathbf{f}_*)$	Gaussian process posterior covariance
D	dimension of input space $\mathcal{X}$
$\mathcal{D}$	data set: $\mathcal{D}=\{(\mathbf{x}_i,y_i)\lvert i=1,\ldots,n\}$
diag$(\mathbf{w})$	(vector argument) a diagonal matrix containing the elements of vector $\mathbf{w}$
diag(W)	(matrix argument) a vector containing the diagonal elements of matrix W
δ_{pq}	Kronecker delta, $\delta_{pq}=1$ iff $p=q$ and 0 otherwise
$\mathbb{E}$ or $\mathbb{E}_{q(x)}[z(x)]$	expectation; expectation of $z(x)$ when $x\sim q(x)$
$f(\mathbf{x})$ or $\mathbf{f}$	Gaussian process (or vector of) latent function values, $\mathbf{f}=(f(\mathbf{x}_1),\ldots,f(\mathbf{x}_n))^\top$
$\mathbf{f}_*$	Gaussian process (posterior) prediction (random variable)
$\bar{\mathbf{f}}_*$	Gaussian process posterior mean
$\mathcal{GP}$	Gaussian process: $f\sim\mathcal{GP}\big(m(\mathbf{x}),k(\mathbf{x},\mathbf{x}')\big)$, the function f is distributed as a Gaussian process with mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x},\mathbf{x}')$
$h(\mathbf{x})$ or $\mathbf{h}(\mathbf{x})$	*either* fixed basis function (or set of basis functions) *or* weight function
H or $H(X)$	set of basis functions evaluated at all training points
I or I_n	the identity matrix (of size n)
$J_\nu(z)$	Bessel function of the first kind
$k(\mathbf{x},\mathbf{x}')$	covariance (or kernel) function evaluated at $\mathbf{x}$ and $\mathbf{x}'$
K or $K(X,X)$	$n\times n$ covariance (or Gram) matrix
K_*	$n\times n_*$ matrix $K(X,X_*)$, the covariance between training and test cases
$\mathbf{k}(\mathbf{x}_*)$ or $\mathbf{k}_*$	vector, short for $K(X,\mathbf{x}_*)$, when there is only a single test case
K_f or K	covariance matrix for the (noise free) $\mathbf{f}$ values

Symbol	Meaning
K_y	covariance matrix for the (noisy) $\mathbf{y}$ values; for independent homoscedastic noise, $K_y = K_f + \sigma_n^2 I$
$K_\nu(z)$	modified Bessel function
$\mathcal{L}(a, b)$	loss function, the loss of predicting b, when a is true; note argument order
$\log(z)$	natural logarithm (base e)
$\log_2(z)$	logarithm to the base 2
ℓ or ℓ_d	characteristic length-scale (for input dimension d)
$\lambda(z)$	logistic function, $\lambda(z) = 1/\big(1 + \exp(-z)\big)$
$m(\mathbf{x})$	the mean function of a Gaussian process
μ	a measure (see section A.7)
$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ or $\mathcal{N}(\mathbf{x}\|\boldsymbol{\mu}, \Sigma)$	(the variable $\mathbf{x}$ has a) Gaussian (Normal) distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ
$\mathcal{N}(\mathbf{x})$	short for unit Gaussian $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, I)$
n and n_*	number of training (and test) cases
N	dimension of feature space
N_H	number of hidden units in a neural network
$\mathbb{N}$	the natural numbers, the positive integers
$\mathcal{O}(\cdot)$	big Oh; for functions f and g on $\mathbb{N}$, we write $f(n) = \mathcal{O}(g(n))$ if the ratio $f(n)/g(n)$ remains bounded as $n \to \infty$
O	*either* matrix of all zeros *or* differential operator
$y\|x$ and $p(y\|x)$	conditional random variable y given x and its probability (density)
$\mathbb{P}_N$	the regular n-polygon
$\boldsymbol{\phi}(\mathbf{x}_i)$ or $\Phi(X)$	feature map of input $\mathbf{x}_i$ (or input set X)
$\Phi(z)$	cumulative unit Gaussian: $\Phi(z) = (2\pi)^{-1/2} \int_{-\infty}^{z} \exp(-t^2/2) dt$
$\pi(\mathbf{x})$	the sigmoid of the latent value: $\pi(\mathbf{x}) = \sigma(f(\mathbf{x}))$ (stochastic if $f(\mathbf{x})$ is stochastic)
$\hat{\pi}(\mathbf{x}_*)$	MAP prediction: π evaluated at $\bar{f}(\mathbf{x}_*)$.
$\bar{\pi}(\mathbf{x}_*)$	mean prediction: expected value of $\pi(\mathbf{x}_*)$. Note, in general that $\hat{\pi}(\mathbf{x}_*) \neq \bar{\pi}(\mathbf{x}_*)$
$\mathbb{R}$	the real numbers
$R_{\mathcal{L}}(f)$ or $R_{\mathcal{L}}(c)$	the risk or expected loss for f, or classifier c (averaged w.r.t. inputs and outputs)
$\tilde{R}_{\mathcal{L}}(l\|\mathbf{x}_*)$	expected loss for predicting l, averaged w.r.t. the model's pred. distr. at $\mathbf{x}_*$
$\mathcal{R}_c$	decision region for class c
$S(\mathbf{s})$	power spectrum
$\sigma(z)$	any sigmoid function, e.g. logistic $\lambda(z)$, cumulative Gaussian $\Phi(z)$, etc.
σ_f^2	variance of the (noise free) signal
σ_n^2	noise variance
$\boldsymbol{\theta}$	vector of hyperparameters (parameters of the covariance function)
$\mathrm{tr}(A)$	trace of (square) matrix A
$\mathbb{T}_l$	the circle with circumference l
$\mathbb{V}$ or $\mathbb{V}_{q(x)}[z(x)]$	variance; variance of $z(x)$ when $x \sim q(x)$
$\mathcal{X}$	input space and also the index set for the stochastic process
X	$D \times n$ matrix of the training inputs $\{\mathbf{x}_i\}_{i=1}^n$: the design matrix
X_*	matrix of test inputs
$\mathbf{x}_i$	the ith training input
x_{di}	the dth coordinate of the ith training input $\mathbf{x}_i$
$\mathbb{Z}$	the integers $\ldots, -2, -1, 0, 1, 2, \ldots$