Probability and Computing

Randomization and probabilistic techniques play an important role in modern computer science, with applications ranging from combinatorial optimization and machine learning to communication networks and secure protocols.

This textbook provides an indispensable teaching tool to accompany a one- or two-semester course for advanced undergraduate or beginning graduate students in computer science and applied mathematics. It offers a comprehensive introduction to the role of randomization and probabilistic techniques in modern computer science, in particular to techniques and paradigms used in the development and probabilistic analysis of algorithms and for data analyses. It assumes only an elementary background in discrete mathematics and gives a rigorous yet accessible treatment of the material, with numerous examples and applications.

The first half of the book covers core material, including random sampling, expectations, Markov's inequality, Chebyshev's inequality, Chernoff bounds, balls-and-bins models, the probabilistic method, and Markov chains. In the second half, the authors delve into more advanced topics such as continuous probability, applications of limited independence, entropy, Markov chain Monte Carlo methods, coupling, martingales, and balanced allocations.

This greatly expanded new edition includes several newly added chapters and sections, covering topics including normal distributions, sample complexity, VC dimension, Rademacher complexity, power laws and related distributions, cuckoo hashing, and applications of the Lovász Local Lemma. New material relevant to machine learning and big data analysis enables students to learn up-to-date techniques and applications. Among the many new exercises and examples are programming-related exercises that provide students with practical experience and training related to the theoretical concepts covered in the text.

Michael Mitzenmacher is a Professor of Computer Science in the School of Engineering and Applied Sciences at Harvard University, where he was also the Area Dean for Computer Science from 2010 to 2013. Michael has authored or co-authored over 200 conference and journal publications on a variety of topics, including algorithms for the Internet, efficient hash-based data structures, erasure and error-correcting codes, power laws, and compression. His work on low-density parity-check codes shared the 2002 IEEE Information Theory Society Best Paper Award and won the 2009 ACM SIGCOMM Test of Time Award. He is an ACM Fellow, and was elected as the Chair of the ACM Special Interest Group on Algorithms and Computation Theory in 2015.

Eli Upfal is a Professor of Computer Science at Brown University, where he was also the department chair from 2002 to 2007. Prior to joining Brown in 1998, he was a researcher and project manager at the IBM Almaden Research Center, and a Professor of Applied Mathematics and Computer Science at the Weizmann Institute of Science. His main research interests are randomized algorithms, probabilistic analysis of algorithms, and computational statistics, with applications ranging from combinatorial and stochastic optimization, computational biology, and computational finance. He is a Fellow of both the IEEE and the ACM.

Probability and Computing

Randomization and Probabilistic Techniques in Algorithms and Data Analysis

Second Edition

Michael Mitzenmacher Eli Upfal

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

Shaftesbury Road, Cambridge CB2 8EA, 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

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment,
a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of
education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781107154889

10.1017/9781316651124

First published 2017
7th printing 2022

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

Library of Congress Cataloging-in-Publication data
Names: Mitzenmacher, Michael, 1969– author. | Upfal, Eli, 1954– author.
Title: Probability and computing / Michael Mitzenmacher Eli Upfal.
Description: Second edition. | Cambridge, United Kingdom ;
New York, NY, USA : Cambridge University Press, [2017] |
Includes bibliographical references and index.
Identifiers: LCCN 2016041654 | ISBN 9781107154889
Subjects: LCSH: Algorithms. | Probabilities. | Stochastic analysis.
Classification: LCC QA274.M574 2017 | DDC 518/.1 – dc23
LC record available at https://lccn.loc.gov/2016041654

ISBN 978-1-107-15488-9 Hardback

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

Cambridge University Press & Assessment 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.

To

Stephanie, Michaela, Jacqueline, and Chloe
M.M.

Liane, Tamara, and Ilan
E.U.

Cambridge University Press & Assessment
978-1-107-15488-9 — Probability and Computing
2nd Edition
Michael Mitzenmacher , Eli Upfal
Frontmatter
More Information

Contents

Cambridge University Press & Assessment
978-1-107-15488-9 — Probability and Computing
2nd Edition
Michael Mitzenmacher , Eli Upfal
Frontmatter
More Information

Cambridge University Press & Assessment
978-1-107-15488-9 — Probability and Computing
2nd Edition
Michael Mitzenmacher , Eli Upfal
Frontmatter
More Information

Cambridge University Press & Assessment
978-1-107-15488-9 — Probability and Computing
2nd Edition
Michael Mitzenmacher , Eli Upfal
Frontmatter
More Information

Note: Asterisks indicate advanced material for this chapter.

Preface to the Second Edition

In the ten years since the publication of the first edition of this book, probabilistic methods have become even more central to computer science, rising with the growing importance of massive data analysis, machine learning, and data mining. Many of the successful applications of these areas rely on algorithms and heuristics that build on sophisticated probabilistic and statistical insights. Judicious use of these tools requires a thorough understanding of the underlying mathematical concepts. Most of the new material in this second edition focuses on these concepts.

The ability in recent years to create, collect, and store massive data sets, such as the World Wide Web, social networks, and genome data, lead to new challenges in modeling and analyzing such structures. A good foundation for models and analysis comes from understanding some standard distributions. Our new chapter on the normal distribution (also known as the Gaussian distribution) covers the most common statistical distribution, as usual with an emphasis on how it is used in settings in computer science, such as for tail bounds. However, an interesting phenomenon is that in many modern data sets, including social networks and the World Wide Web, we do not see normal distributions, but instead we see distributions with very different properties, most notably unusually heavy tails. For example, some pages in the World Wide Web have an unusually large number of pages that link to them, orders of magnitude larger than the average. The new chapter on power laws and related distributions covers specific distributions that are important for modeling and understanding these kinds of modern data sets.

Machine learning is one of the great successes of computer science in recent years, providing efficient tools for modeling, understanding, and making predictions based on large data sets. A question that is often overlooked in practical applications of machine learning is the accuracy of the predictions, and in particular the relation between accuracy and the sample size. A rigorous introduction to approaches to these important questions is presented in a new chapter on sample complexity, VC dimension, and Rademacher averages.

We have also used the new edition to enhance some of our previous material. For example, we present some of the recent advances on algorithmic variations of the powerful Lovász local lemma, and we have a new section covering the wonderfully named and increasingly useful hashing approach known as cuckoo hashing. Finally, in addition to all of this new material, the new edition includes updates and corrections, and many new exercises.

We thank the many readers who sent us corrections over the years – unfortunately, too many to list here!

Preface to the First Edition

Why Randomness?

Why should computer scientists study and use randomness? Computers appear to behave far too unpredictably as it is! Adding randomness would seemingly be a disadvantage, adding further complications to the already challenging task of efficiently utilizing computers.

Science has learned in the last century to accept randomness as an essential component in modeling and analyzing nature. In physics, for example, Newton's laws led people to believe that the universe was a deterministic place; given a big enough calculator and the appropriate initial conditions, one could determine the location of planets years from now. The development of quantum theory suggests a rather different view; the universe still behaves according to laws, but the backbone of these laws is probabilistic. "God does not play dice with the universe" was Einstein's anecdotal objection to modern quantum mechanics. Nevertheless, the prevailing theory today for subparticle physics is based on random behavior and statistical laws, and randomness plays a significant role in almost every other field of science ranging from genetics and evolution in biology to modeling price fluctuations in a free-market economy.

Computer science is no exception. From the highly theoretical notion of probabilistic theorem proving to the very practical design of PC Ethernet cards, randomness and probabilistic methods play a key role in modern computer science. The last two decades have witnessed a tremendous growth in the use of probability theory in computing. Increasingly more advanced and sophisticated probabilistic techniques have been developed for use within broader and more challenging computer science applications. In this book, we study the fundamental ways in which randomness comes to bear on computer science: randomized algorithms and the probabilistic analysis of algorithms.

Randomized algorithms: Randomized algorithms are algorithms that make random choices during their execution. In practice, a randomized program would use values generated by a random number generator to decide the next step at several branches of its execution. For example, the protocol implemented in an Ethernet card uses random numbers to decide when it next tries to access the shared Ethernet communication medium. The randomness is useful for breaking symmetry, preventing different cards from repeatedly accessing the medium at the same time. Other commonly used applications of randomized algorithms include Monte Carlo simulations and primality testing in cryptography. In these and many other important applications, randomized algorithms are significantly more efficient than the best known deterministic solutions. Furthermore, in most cases the randomized algorithms are also simpler and easier to program.

These gains come at a price; the answer may have some probability of being incorrect, or the efficiency is guaranteed only with some probability. Although it may seem unusual to design an algorithm that may be incorrect, if the probability of error is sufficiently small then the improvement in speed or memory requirements may well be worthwhile.

Probabilistic analysis of algorithms: Complexity theory tries to classify computation problems according to their computational complexity, in particular distinguishing between easy and hard problems. For example, complexity theory shows that the Traveling Salesman problem is NP-hard. It is therefore very unlikely that we will ever know an algorithm that can solve any instance of the Traveling Salesman problem in time that is subexponential in the number of cities. An embarrassing phenomenon for the classical worst-case complexity theory is that the problems it classifies as hard to compute are often easy to solve in practice. Probabilistic analysis gives a theoretical explanation for this phenomenon. Although these problems may be hard to solve on some set of pathological inputs, on most inputs (in particular, those that occur in real-life applications) the problem is actually easy to solve. More precisely, if we think of the input as being randomly selected according to some probability distribution on the collection of all possible inputs, we are very likely to obtain a problem instance that is easy to solve, and instances that are hard to solve appear with relatively small probability. Probabilistic analysis of algorithms is the method of studying how algorithms perform when the input is taken from a well-defined probabilistic space. As we will see, even NP-hard problems might have algorithms that are extremely efficient on almost all inputs.

The Book

This textbook is designed to accompany one- or two-semester courses for advanced undergraduate or beginning graduate students in computer science and applied mathematics. The study of randomized and probabilistic techniques in most leading universities has moved from being the subject of an advanced graduate seminar meant for theoreticians to being a regular course geared generally to advanced undergraduate and beginning graduate students. There are a number of excellent advanced, research-oriented books on this subject, but there is a clear need for an introductory textbook. We hope that our book satisfies this need.

The textbook has developed from courses on probabilistic methods in computer science taught at Brown (CS 155) and Harvard (CS 223) in recent years. The emphasis in these courses and in this textbook is on the probabilistic techniques and paradigms, not on particular applications. Each chapter of the book is devoted to one such method or technique. Techniques are clarified though examples based on analyzing randomized algorithms or developing probabilistic analysis of algorithms on random inputs. Many of these examples are derived from problems in networking, reflecting a prominent trend in the networking field (and the taste of the authors).

The book contains fourteen chapters. We may view the book as being divided into two parts, where the first part (Chapters 1–7) comprises what we believe is core material. The book assumes only a basic familiarity with probability theory, equivalent to what is covered in a standard course on discrete mathematics for computer scientists. Chapters 1–3 review this elementary probability theory while introducing some interesting applications. Topics covered include random sampling, expectation, Markov's inequality, variance, and Chebyshev's inequality. If the class has sufficient background in probability, then these chapters can be taught quickly. We do not suggest skipping them, however, because they introduce the concepts of randomized algorithms and probabilistic analysis of algorithms and also contain several examples that are used throughout the text.

Chapters 4–7 cover more advanced topics, including Chernoff bounds, balls-and-bins models, the probabilistic method, and Markov chains. The material in these chapters is more challenging than in the initial chapters. Sections that are particularly challenging (and hence that the instructor may want to consider skipping) are marked with an asterisk. The core material in the first seven chapters may constitute the bulk of a quarter- or semester-long course, depending on the pace.

The second part of the book (Chapters 8–17) covers additional advanced material that can be used either to fill out the basic course as necessary or for a more advanced second course. These chapters are largely self-contained, so the instructor can choose the topics best suited to the class. The chapters on continuous probability and entropy are perhaps the most appropriate for incorporating into the basic course. Our introduction to continuous probability (Chapter 8) focuses on uniform and exponential distributions, including examples from queueing theory. Our examination of entropy (Chapter 10) shows how randomness can be measured and how entropy arises naturally in the context of randomness extraction, compression, and coding.

Chapters 11 and 12 cover the Monte Carlo method and coupling, respectively; these chapters are closely related and are best taught together. Chapter 13, on martingales, covers important issues on dealing with dependent random variables, a theme that continues in a different vein in Chapter 15 is the development of pairwise independence and derandomization. Finally, the chapter on balanced allocations (Chapter 17) covers a topic close to the authors' hearts and ties in nicely with Chapter 5 concerning analysis of balls-and-bins problems.

The order of the subjects, especially in the first part of the book, corresponds to their relative importance in the algorithmic literature. Thus, for example, the study of Chernoff bounds precedes more fundamental probability concepts such as Markov chains. However, instructors may choose to teach the chapters in a different order. A course with more emphasis on general stochastic processes, for example, may teach Markov chains (Chapter 7) immediately after Chapters 1–3, following with the chapter on balls, bins, and random graphs (Chapter 5, omitting the Hamiltonian cycle example). Chapter 6 on the probabilistic method could then be skipped, following instead with continuous probability and the Poisson process (Chapter 8). The material from Chapter 4 on Chernoff bounds, however, is needed for most of the remaining material.

Most of the exercises in the book are theoretical, but we have included some programming exercises – including two more extensive exploratory assignments that require some programming. We have found that occasional programming exercises are often helpful in reinforcing the book's ideas and in adding some variety to the course.

We have decided to restrict the material in this book to methods and techniques based on rigorous mathematical analysis; with few exceptions, all claims in this book are followed by full proofs. Obviously, many extremely useful probabilistic methods do not fall within this strict category. For example, in the important area of Monte Carlo methods, most practical solutions are heuristics that have been demonstrated to be effective and efficient by experimental evaluation rather than by rigorous mathematical analysis. We have taken the view that, in order to best apply and understand the strengths and weaknesses of heuristic methods, a firm grasp of underlying probability theory and rigorous techniques – as we present in this book – is necessary. We hope that students will appreciate this point of view by the end of the course.

Acknowledgments

Our first thanks go to the many probabilists and computer scientists who developed the beautiful material covered in this book. We chose not to overload the textbook with numerous references to the original papers. Instead, we provide a reference list that includes a number of excellent books giving background material as well as more advanced discussion of the topics covered here.

The book owes a great deal to the comments and feedback of students and teaching assistants who took the courses CS 155 at Brown and CS 223 at Harvard. In particular we wish to thank Aris Anagnostopoulos, Eden Hochbaum, Rob Hunter, and Adam Kirsch, all of whom read and commented on early drafts of the book.

Special thanks to Dick Karp, who used a draft of the book in teaching CS 174 at Berkeley during fall 2003. His early comments and corrections were most valuable in improving the manuscript. Peter Bartlett taught CS 174 at Berkeley in spring 2004, also providing many corrections and useful comments.

We thank our colleagues who carefully read parts of the manuscript, pointed out many errors, and suggested important improvements in content and presentation: Artur Czumaj, Alan Frieze, Claire Kenyon, Joe Marks, Salil Vadhan, Eric Vigoda, and the anonymous reviewers who read the manuscript for the publisher.

We also thank Rajeev Motwani and Prabhakar Raghavan for allowing us to use some of the exercises in their excellent book *Randomized Algorithms*.

We are grateful to Lauren Cowles of Cambridge University Press for her editorial help and advice in preparing and organizing the manuscript.

Writing of this book was supported in part by NSF ITR Grant no. CCR-0121154.