The Book Review Column [1]
by **Nicholas Tran** (ntran@scu.edu)
Department of Mathematics & Computer Science, Santa Clara University

1 Notable New Releases

Games on Graphs: From Logic and Automata to Algorithms (Cambridge University Press, 2026), by Nathanaël Fijalkow (Ed.) (University of Warsaw), is a collection of articles on infinite duration games on graphs, with applications in program verification and synthesis.

Extremal Graph and Hypergraph Theory (Cambridge University Press, 2026), by Dhruv Mubayi (University of Illinois, Chicago) and Jacques Verstraete (University of California, San Diego), contains both introductory and advanced material in extremal graph theory, hypergraph theory and Ramsey theory.

Quantum Computing: Foundations and Practice (Oxford University Press, 2026), by Steven Herbert (University of Cambridge), presents the most important quantum algorithms and communication protocols, weighs the question of quantum advantage, and addresses the subject of quantum error correction.

2 This Column

Shiva Houshmand appreciates the textbook *Understanding Cryptography: From Established Symmetric and Asymmetric Ciphers to Post-Quantum Cryptography* (Springer, 2024), by Christof Paar, Jan Pelzl, and Tim Güneysu, for its emphasis on practical understanding and correct application of cryptographic techniques. The book is well-suited for undergraduate and beginning graduate courses in applied cryptography and also serves as an excellent reference for practitioners.

Why Machines Learn: The Elegant Math Behind Modern AI (Penguin Random House, 2025), by Anil Ananthaswamy, is a must-read for anyone seeking an accessible introduction to the mathematical foundations of machine learning. I particularly recommend it to first-year computer science students.

3 How to Contribute

Declutter your reading list by writing a review of the book you have been meaning to read for *SIGACT News*. Either choose from the books listed below or propose your own. In either case, the publisher will send you a free copy of the book. Guidelines and a LaTeX template can be found at `https://algoplexity.com/~ntran`.

BOOKS THAT NEED REVIEWERS FOR THE SIGACT NEWS COLUMN

Algorithms & Complexity

1. Brody, J. (2025). *The Joy of Quantum Computing: A Concise Introduction*. Princeton University Press.

2. Dalzell, A., & McArdle, S., & Berta, M., & Bienias, P., & Chen, C.-F., & Gilyén, A., & Hann, C., & Kastoryano, M., & Khabiboulline, E., & Kubica, A., & Salton, G., & Wang, S., & Brandão, F. (2025). *Quantum Algorithms: A Survey of Applications and End-to-end Complexities*. Cambridge University Press.

3. Erciyes, K. (2025). *Guide to Distributed Algorithms: Design, Analysis and Implementation Using Python*. Springer.

4. Morazán, M. T. (2025). *Programming-based Formal Languages and Automata Theory: Design, Implement, Validate, and Prove*. Springer.

5. Herbert, S. (2026). *Quantum Computing: Foundations and Practice*. Oxford University Press.

Computability & Logic

1. Badia, G., Crossley, J. N., & Stillwell, J. C. (2026). *What is Mathematical Logic?, 2nd ed.* Oxford University Press.

Programming

1. Lichtman, E. (2025). *The Computer Always Wins: How Algorithms Beat Us at Our Own Games*. The MIT Press.

2. Laaksonen, A. (2026). *Guide to Using Generative AI in Programming*. Springer.

Miscellaneous Computer Science & Mathematics

1. Wilson, R. (2025). *Sum Stories: Equations and Their Origins*. Oxford University Press.

2. O'Rourke, J. (2025). *The Mathematics of Origami*. Cambridge University Press.

3. Meister, M., Lee, K. H., & Portugues, R. (2025). *Mathematical Biology*. The MIT Press.

Data Science

1. Alpaydın, E. (2025). *Fundamentals of Probability and Statistics for Machine Learning*. The MIT Press.

2. Tromp, J. (2025). *A Geometrical Introduction to Tensor Calculus*. Princeton University Press.

Discrete Mathematics and Computing

1. Devadoss, S., & O'Rourke, J. (2025). *Discrete and Computational Geometry, 2nd ed.* Princeton University Press.

Cryptography and Security

1. Garfinkel, S. (2025). *Differential Privacy.* The MIT Press.
2. Beyne, T., & Rijmen, V. (2025). *Linear Cryptanalysis.* Cambridge University Press.

Combinatorics and Graph Theory

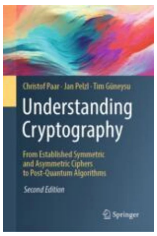
1. Mubayi, D., & Verstraete, J. (2026). *Extremal Graph and Hypergraph Theory.* Cambridge University Press.
2. Fijalkow, N. (Ed.) (2026). *Games on Graphs: From Logic and Automata to Algorithms.* Cambridge University Press.

Review of [1]

Understanding Cryptography

From Established Symmetric and Asymmetric Ciphers to Post-Quantum Algorithms

Second Edition



Christof Paar, **Jan Pelzl**, and **Tim Güneysu**

Springer, 2024
543 pages, $79.99 Hardcover, $64.99 Ebook

Review by
Shiva Houshmand
Department of Mathematics and Computer Science
Santa Clara University

1 Overview

Today, almost every part of modern life relies on digital systems, which makes cybersecurity more important than ever. Technologies such as social networking platforms, medical devices, and emerging systems like autonomous vehicles all depend on strong security mechanisms to function safely. Cryptography is a foundational pillar of modern computer security and a primary tool for building secure systems. This book offers clear explanations of the cryptographic mechanisms most commonly used in real-world systems, with a strong emphasis on practical relevance.

Designing, implementing, and using cryptographic mechanisms correctly is subtle and critical, and even small misunderstandings can lead to serious security failures. Aimed at both students and practitioners, the book helps readers develop a solid conceptual understanding of cryptographic schemes, protocols, and implementation issues, including how they withstand known attacks and how to choose appropriate key lengths. Mathematical foundations are introduced when needed, without going deeply into formal proofs or provable-security frameworks. Overall, the book gives readers a clear picture of how cryptographic systems are built, why they are considered secure, and under what assumptions that security might no longer hold.

2 Summary of Contents

The organization of the book follows a structure that is typical of introductory cryptography texts, beginning with an overview of the field of cryptography.

Chapter 1 introduces symmetric encryption using historical ciphers to motivate fundamental concepts such as message secrecy and key space. It also briefly discusses basic cryptanalysis techniques and introduces modular arithmetic, which provides the mathematical background needed for later chapters.

Chapter 2 is about stream ciphers (a family of symmetric encryption). Since randomness plays a major role in stream ciphers, the chapter presents pseudorandom number generators and explains

how keystreams are derived from cryptographically secure pseudorandom number generators. It concludes with a discussion of modern stream ciphers currently in practical use.

The next three chapters focus on block ciphers (another family of symmetric encryption scheme). **Chapter 3** covers DES, one of the earliest modern encryption algorithms, which is now considered obsolete. The chapter explains DES's internal structure, including Feistel networks and S-boxes, and introduces core block cipher design principles such as diffusion and confusion. These concepts remain central to the design of modern block ciphers.

Chapter 4 is dedicated to AES, the most widely used block cipher today. It walks through the design process behind AES and explains its internal structure in detail.

Chapter 5 then explores how block ciphers can be used as building blocks for other cryptographic mechanisms, including stream ciphers, pseudorandom number generators, hash functions, and message authentication codes, discussed in the context of their modes of operation.

Chapter 6 introduces asymmetric-key cryptography (aka public-key cryptography) along with the number theory concepts that are essential for understanding public-key algorithms.

Chapter 7 focuses on RSA, one of the earliest public-key encryption schemes that remains widely used in practice. The chapter explains the core RSA algorithm and then discusses its practical weaknesses, emphasizing the need for proper padding schemes in real-world implementations.

Chapter 8 continues the discussion of public-key cryptography by presenting algorithms based on the discrete logarithm problem, including Diffie–Hellman and ElGamal.

Chapter 9 introduces elliptic curve cryptography, a third family of public-key algorithms that has become increasingly popular in practice due to its performance advantages and shorter key lengths.

Chapter 10 introduces digital signatures, a fundamental tool based on public-key cryptography that provides integrity and authentication. The chapter covers RSA- and ElGamal-based signature schemes, along with its widely used extensions, including the Digital Signature Algorithm (DSA) and the Elliptic Curve Digital Signature Algorithm (ECDSA).

Chapter 11 focuses on hash functions, a core cryptographic primitive used across a wide range of protocols and systems. Hash functions produce a short, fixed-length, and unique representation of a message and play a central role in applications such as digital signatures, message authentication codes, cryptocurrencies, password security, and post-quantum cryptographic constructions.

Chapter 12 introduces post-quantum cryptography by explaining the threat that quantum computers pose to widely deployed public-key schemes. It presents the basic ideas behind post-quantum approaches—such as lattice-based, hash-based, and code-based cryptography—at a high level. This chapter reinforces a central theme of the book: cryptographic security depends on underlying assumptions, and those assumptions must be revisited as technology evolves.

Chapter 13 focuses on message authentication codes (MACs), which provide message integrity and authentication using symmetric-key techniques. While they serve a purpose similar to digital signatures, MACs are widely used in practice due to their efficiency and role in many security protocols.

The final chapter, **Chapter 14**, addresses practical deployment and key management issues, covering the mechanisms used to securely generate, store, and distribute the secret keys that are essential to most of the cryptographic algorithms discussed in the book.

3 Opinion

This book would work very well as a textbook for undergraduate and graduate courses in applied cryptography, and it is also a good fit for practitioners with a solid technical background. It assumes some familiarity with programming and basic mathematics, but it does not expect the reader to have prior experience with cryptography. One thing I particularly appreciate is that the book intentionally avoids heavy mathematical formalism. This is a thoughtful choice that keeps the focus on practical understanding and correct application of cryptographic techniques without getting lost in formal proofs.

A major strength of the book is how each chapter is structured. Each chapter explains the core ideas behind each cryptographic algorithm, while also discussing known attacks, relevant NIST recommendations, and practical implementation issues in software and hardware when appropriate. Having all of this information brought together in a single chapter is especially valuable for readers who want to understand how these schemes are actually used in practice.

Each chapter also points readers to additional resources for those who want to dive deeper into the mathematical details or specific attacks, and the chapter concludes with a concise summary of the key takeaways. Another aspect I found valuable is the explicit discussion of cryptography in the presence of quantum computing. The authors explain how known quantum algorithms affect existing cryptographic schemes and why this forces us to rethink long-standing security assumptions. In this edition, the new chapter on post-quantum cryptography gives readers a clear picture of which types of algorithms are expected to remain secure and which are not once powerful quantum computers become available, along with an overview of the main post-quantum approaches.

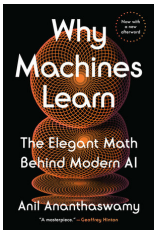
Naturally, some chapters, especially Chapter 9 on elliptic curve cryptography and Chapter 12 on post-quantum cryptography, are more mathematically demanding than earlier ones. Even so, the presentation remains approachable, with an emphasis on intuition and high-level structure.

Review of [1]

Why Machines Learn

The Elegant Math Behind Modern AI

Anil Ananthaswamy



Penguin Random House, 2025
496 pages, $22 Paperback

Review by

Nicholas Tran
Department of Mathematics and Computer Science
Santa Clara University

1 Summary

This highly readable book traces the evolution of machine learning over the past seventy years by examining its key ideas, people, and mathematics. The reader is treated to a *Quanta Magazine*–style college course on the subject that assumes no background in advanced mathematics.

The title of the book comes from a story in the *New York Times* reporting on Frank Rosenblatt's invention of the perceptron in 1958, which ushered in the age of machines that self-adjust, or learn, as they process data. When asked to explain why the perceptron learned, the inventor demurred, saying that he could only do so in highly technical terms. Author Anil Ananthaswamy ably closes this gap by explaining the fundamental mathematical concepts from linear algebra, calculus, probability and statistics, and optimization behind the important advances in the field.

The first two chapters explain the mathematics behind the perceptron, starting with the neurode, a computational model of the biological neuron by Warren McCulloch and Walter Pitts, who showed that networks of these building blocks can simulate Turing machines. A perceptron is a special type of these networks that consists of a single layer of input neurodes whose discrete outputs, each with its own adjustable weight, are summed together to produce a yes/no answer based on the sign of the sum. The network makes multiple passes over a preclassified data set, adjusting its weights if it makes a mistake on a data point, until it learns the correct classification perfectly. A network of this type is characterized by a hyperplane whose normal vector is given by the weights. Rosenblatt devised a method for updating the weights (normal vector) that is guaranteed to learn the classification after a finite number of steps for any linearly separable data set, i.e., one for which there exists a hyperplane separating the "yes" data points from the "no" data points. The concepts of vectors, matrices, dot products, and normals are gently introduced and illustrated with two-dimensional examples, culminating in Rosenblatt's algorithm and a proof of its convergence.

Soon after the introduction of the perceptron, Bernard Widrow and Ted Hoff introduced a different type of single-layer neural network with continuous output that uses the now-ubiquitous method of gradient descent to improve its weights. Their network, called ADALINE, learns to classify a linearly separable data set using a stochastic estimate of the gradient of the mean squared

error between the predicted and actual output values. Chapter 3 illustrates the method of gradient descent using quadratic curves and surfaces and presents Widrow and Hoff's ADALINE training algorithm.

The excitement that drove research on neural networks at the beginning of the 1960s eventually gave way to the realization of serious limitations in their power. In 1969, Marvin Minsky and Seymour Papert published a mathematical study of perceptron-like networks which, among other things, showed that they cannot learn the XOR function, which is not linearly separable. Researchers began exploring alternative approaches to learning data sets whose classification requires boundaries more complex than hyperplanes.

In the probabilistic setting, there exists a provably best classifier that has the lowest error rate among all classifiers on average. This theoretical Bayes optimal classifier is often impractical or even impossible to compute. Surprisingly, Thomas Cover and Peter Hart proved in 1967 that it can be asymptotically approximated within a factor of 2 by a very simple algorithm called the k-nearest-neighbor (kNN) algorithm: a new data point is assigned to the most popular group among its k nearest neighbors, where k is a predefined value. Chapter 4 gives a crash course in Bayesian statistics and shows how to apply it to the famous Monty Hall problem and some real-world classification problems. Chapter 5 explains the kNN algorithm and points out its Achilles' heel: as the number of dimensions of the data set increases, most points are far away from any given point, and hence the underlying assumption of the kNN algorithm that distance correlates with similarity is no longer valid.

One approach to mitigating this "curse of dimensionality" is to consolidate dimensions that highly correlate with one another, transforming the original data set into one having fewer dimensions. Chapter 6 explains how these so-called principal components can be obtained by computing the eigenvectors and eigenvalues of the covariance matrix of the original data set.

Another strategy for classifying non-linearly separable data involves mapping data points into higher-dimensional spaces where linear separation becomes possible. In these spaces, the optimal separating hyperplane is computed using Lagrange multipliers, as shown by Vladimir Vapnik. However, calculations in high dimensions can be expensive. Chapter 7 covers the mathematics behind support vector machines (SVMs) and the associated kernel trick, developed by Bernhard Boser, Isabelle Guyon, and Vapnik. The central insights are that the optimal hyperplane depends solely on the dot products of a subset of the data points called support vectors, and that there exist kernel functions that yield the same dot product between two original data points as the dot product between those points when lifted to higher dimensions.

Neural network research experienced a renaissance in the 1980s. John Hopfield showed in 1982 how to store information in a neural network whose neurons are all connected to one another and whose weights between two neurons are the same in each direction. Regardless of the starting state, a Hopfield network dynamically adjusts itself until reaching an equilibrium, which is one of the stored memories. Chapter 8 explains Hopfield networks and provides a proof that when perturbed, they will eventually reach a stable state. George Cybenko proved in 1989 that all continuous functions can be approximated with a neural network having an additional (hidden) layer between the inputs and output. Here the output of a neurode is no longer a binary value but the value of the sigmoid function, $\sigma(x) = 1/(1 + e^{-x})$, which is continuous and ranges between 0 and 1. Chapter 9 gives a proof sketch of this theorem. In 1986, David Rumelhart, Geoffrey Hinton, and Ronald Williams published the backpropagation algorithm (independently discovered several times in other settings)that allows multilayer neural networks using gradient descent to update

their weights. This is done by computing the gradient backward using the chain rule. An example of the propagation calculation appears at the end of Chapter 10.

In 1989, Yann LeCun developed LeNet, a multilayer convolutional neural network that learns images in a spatially invariant manner similar to the way information is processed hierarchically in the visual cortex. Chapter 11 explains the evolution of LeNet from its predecessor, the neocognitron, to its successor, AlexNet.

The last chapter discusses phenomena in neural networks that cannot yet be explained by current theories, followed by an epilogue and an afterword explaining the latest revolution in AI: large language models and their underlying engine, transformers.

2 Opinion

This book will appeal to a wide audience, from general readers interested in AI to students and practitioners of machine learning. The author's engaging style and lucid explanations make the substantial mathematical concepts accessible to those with only a high school mathematics background. The book is filled with historical anecdotes and profiles of key figures in the field, adding depth and context to the technical content. The inclusion of proofs and algorithmic details provides a solid foundation for readers who wish to delve deeper into the subject.

A few minor blemishes may detract from the reading experience for some. The book adopts a somewhat idiosyncratic mathematical notation such as *x1*, w_1*32*, and *yhat* instead of x_1, $w_1^{3,2}$ and $\hat{y}$. Some explanations of technical terms like "stochastic" in the context of gradient descent are somewhat imprecise. Finally, the book glosses over the deep philosophical differences between the symbolic AI approach and the connectionist approach embodied by neural networks, giving the impression that the dispute was mainly about funding.

Overall, *Why Machines Learn* is a gem of an introduction to the still-unfolding story of the AI revolution. It is a must-read for anyone interested in exploring the field of machine learning.