Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

Charu C. Aggarwal

Neural Networks and Deep Learning

A Textbook

Springer

Perceptive Exhibit 2041
IPR2025-01574

Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

Charu C. Aggarwal
IBM T. J. Watson Research Center
International Business Machines
Yorktown Heights, NY, USA

ISBN 978-3-319-94462-3 ISBN 978-3-319-94463-0 (eBook)
https://doi.org/10.1007/978-3-319-94463-0

Library of Congress Control Number: 2018947636

© Springer International Publishing AG, part of Springer Nature 2018
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Perceptive Exhibit 2041
IPR2025-01574

Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

Preface

"Any A.I. smart enough to pass a Turing test is smart enough to know to fail it."—Ian McDonald

Neural networks were developed to simulate the human nervous system for machine learning tasks by treating the computational units in a learning model in a manner similar to human neurons. The grand vision of neural networks is to create artificial intelligence by building machines whose architecture simulates the computations in the human nervous system. This is obviously not a simple task because the computational power of the fastest computer today is a minuscule fraction of the computational power of a human brain. Neural networks were developed soon after the advent of computers in the fifties and sixties. Rosenblatt's perceptron algorithm was seen as a fundamental cornerstone of neural networks, which caused an initial excitement about the prospects of artificial intelligence. However, after the initial euphoria, there was a period of disappointment in which the data hungry and computationally intensive nature of neural networks was seen as an impediment to their usability. Eventually, at the turn of the century, greater data availability and increasing computational power lead to increased successes of neural networks, and this area was reborn under the new label of "deep learning." Although we are still far from the day that artificial intelligence (AI) is close to human performance, there are specific domains like image recognition, self-driving cars, and game playing, where AI has matched or exceeded human performance. It is also hard to predict what AI might be able to do in the future. For example, few computer vision experts would have thought two decades ago that any automated system could ever perform an intuitive task like categorizing an image more accurately than a human.

Neural networks are *theoretically* capable of learning any mathematical function with sufficient training data, and some variants like recurrent neural networks are known to be *Turing complete*. Turing completeness refers to the fact that a neural network can simulate any learning algorithm, *given sufficient training data*. The sticking point is that the amount of data required to learn even simple tasks is often extraordinarily large, which causes a corresponding increase in training time (if we assume that enough training data is available in the first place). For example, the training time for image recognition, which is a simple task for a human, can be on the order of weeks even on high-performance systems. Furthermore, there are practical issues associated with the stability of neural network training, which are being resolved even today. Nevertheless, given that the speed of computers is

Perceptive Exhibit 2041
IPR2025-01574

Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

expected to increase rapidly over time, and fundamentally more powerful paradigms like quantum computing are on the horizon, the computational issue might not eventually turn out to be quite as critical as imagined.

Although the biological analogy of neural networks is an exciting one and evokes comparisons with science fiction, the mathematical understanding of neural networks is a more mundane one. The neural network abstraction can be viewed as a modular approach of enabling learning algorithms that are based on continuous optimization on a computational graph of dependencies between the input and output. To be fair, this is not very different from traditional work in control theory; indeed, some of the methods used for optimization in control theory are strikingly similar to (and historically preceded) the most fundamental algorithms in neural networks. However, the large amounts of data available in recent years together with increased computational power have enabled experimentation with deeper architectures of these computational graphs than was previously possible. The resulting success has changed the broader perception of the potential of deep learning.

The chapters of the book are organized as follows:

1. *The basics of neural networks:* Chapter 1 discusses the basics of neural network design. Many traditional machine learning models can be understood as special cases of neural learning. Understanding the relationship between traditional machine learning and neural networks is the first step to understanding the latter. The simulation of various machine learning models with neural networks is provided in Chapter 2. This will give the analyst a feel of how neural networks push the envelope of traditional machine learning algorithms.

2. *Fundamentals of neural networks:* Although Chapters 1 and 2 provide an overview of the training methods for neural networks, a more detailed understanding of the training challenges is provided in Chapters 3 and 4. Chapters 5 and 6 present radial-basis function (RBF) networks and restricted Boltzmann machines.

3. *Advanced topics in neural networks:* A lot of the recent success of deep learning is a result of the specialized architectures for various domains, such as recurrent neural networks and convolutional neural networks. Chapters 7 and 8 discuss recurrent and convolutional neural networks. Several advanced topics like deep reinforcement learning, neural Turing mechanisms, and generative adversarial networks are discussed in Chapters 9 and 10.

We have taken care to include some of the "forgotten" architectures like RBF networks and Kohonen self-organizing maps because of their potential in many applications. The book is written for graduate students, researchers, and practitioners. Numerous exercises are available along with a solution manual to aid in classroom teaching. Where possible, an application-centric view is highlighted in order to give the reader a feel for the technology.

Throughout this book, a vector or a multidimensional data point is annotated with a bar, such as $\overline{X}$ or $\overline{y}$. A vector or multidimensional point may be denoted by either small letters or capital letters, as long as it has a bar. Vector dot products are denoted by centered dots, such as $\overline{X} \cdot \overline{Y}$. A matrix is denoted in capital letters without a bar, such as R. Throughout the book, the $n \times d$ matrix corresponding to the entire training data set is denoted by D, with n documents and d dimensions. The individual data points in D are therefore d-dimensional row vectors. On the other hand, vectors with one component for each data

Perceptive Exhibit 2041
IPR2025-01574

Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

point are usually n-dimensional column vectors. An example is the n-dimensional column vector $\overline{y}$ of class variables of n data points. An observed value y_i is distinguished from a predicted value $\hat{y}_i$ by a circumflex at the top of the variable.

Yorktown Heights, NY, USA Charu C. Aggarwal

Perceptive Exhibit 2041
IPR2025-01574

Computers connected to subscribing institutions download at:
https://rd.springer.com/book/10.1007/978-3-319-94463-0

Author Biography

Charu C. Aggarwal is a Distinguished Research Staff Member (DRSM) at the IBM T. J. Watson Research Center in Yorktown Heights, New York. He completed his undergraduate degree in Computer Science from the Indian Institute of Technology at Kanpur in 1993 and his Ph.D. from the Massachusetts Institute of Technology in 1996.

He has worked extensively in the field of data mining. He has published more than 350 papers in refereed conferences and journals and authored over 80 patents. He is the author or editor of 18 books, including textbooks on data mining, recommender systems, and outlier analysis. Because of the commercial value of his patents, he has thrice been designated a Master Inventor at IBM. He is a recipient of an IBM Corporate Award (2003) for his work on bioterrorist threat detection in data streams, a recipient of the IBM Outstanding Innovation Award (2008) for his scientific contributions to privacy technology, and a recipient of two IBM Outstanding Technical Achievement Awards (2009, 2015) for his work on data streams/high-dimensional data. He received the EDBT 2014 Test of Time Award for his work on condensation-based privacy-preserving data mining. He is also a recipient of the IEEE ICDM Research Contributions Award (2015), which is one of the two highest awards for influential research contributions in the field of data mining.

He has served as the general co-chair of the IEEE Big Data Conference (2014) and as the program co-chair of the ACM CIKM Conference (2015), the IEEE ICDM Conference (2015), and the ACM KDD Conference (2016). He served as an associate editor of the IEEE Transactions on Knowledge and Data Engineering from 2004 to 2008. He is an associate editor of the IEEE Transactions on Big Data, an action editor of the Data Mining and Knowledge Discovery Journal, and an associate editor of the Knowledge and Information Systems Journal. He serves as the editor-in-chief of the ACM Transactions on Knowledge Discovery from Data as well as the ACM SIGKDD Explorations. He serves on the advisory board of the Lecture Notes on Social Networks, a publication by Springer. He has served as the vice-president of the SIAM Activity Group on Data Mining and is a member of the SIAM industry committee. He is a fellow of the SIAM, ACM, and the IEEE, for "contributions to knowledge discovery and data mining algorithms."

Perceptive Exhibit 2041
IPR2025-01574