

Preface

By and large, this book grew out of research conducted in my group as well as classes and lectures given at the University of California, Irvine (UCI) and elsewhere over the years. It can be used as a textbook for an undergraduate or graduate course in machine learning, or as an introduction to the topic for scientists from other fields. Basic prerequisites for understanding the material include college level algebra, calculus, and probability. Familiarity with information theory, statistics, coding theory, and computational complexity at an elementary level are also helpful. I have striven to focus primarily on fundamental principles and provide a treatment that is both self-contained and rigorous, sometimes referring to the literature for well known technical results, or to the exercises, which are an integral part of the book. The exercises vary in difficulty substantially. Should you become frustrated at trying to solve one of them, remind yourself that it is only when you are struggling with a problem that your brain is really learning something.

In order to solve some of the problems in the book, or more broadly to think about scientific and other questions, I recommend my students to systematically try at least four different approaches. The first of course is to simplify. When a question seems too difficult at first, look for special or simpler cases. When trying to understand a theorem, look at the case of “small n ”, or fix the values of certain parameters, or switch to the linear case, or try to interpolate. The second is the opposite way of thinking: generalize or extrapolate. How can a result be applied to more general cases? Can the conditions under which a theorem is true be relaxed? The third way of thinking is “to take the limit”, to look at what happens at the boundaries of a certain domain, under extreme conditions, to let n go to zero, or to infinity. And finally, the fourth way, is to always invert, looking at things somehow from an opposite perspective. Thus, for example, when thinking about an autoencoder, one may want to first simplify it by studying how to solve the top layer given the lower layer, which is usually an easier problem; and then invert this approach by studying how the lower layer can be solved given the top layer, which is usually a harder problem. Of course these four principles are not the panacea to every situation and, for instance, identifying the right form of “inversion” in a given

situation may not be obvious. However, the discipline of trying to apply these four principles in a systematic manner can be helpful and, incidentally, remains a major challenge for Artificial Intelligence (AI).

In writing this book, one my goals has been to provide a rigorous treatment from first principles, as much as possible, in a still rapidly evolving field. This is one of the meanings of “in science” in the title. In this regard, the flow of the book is dictated primarily by complexity issues, going from shallow networks in their different forms, to deep feedforward networks, to recurrent and recursive networks. Two-layer networks, of which autoencoders are the prototypical example, provide the hinge between shallow and deep learning. For each kind of networks, it is useful to consider special “hardware” cases, such as networks of linear units. Contrary to widespread belief, the linear case is often interesting and far from trivial. But this is not the only case where using a particular hardware model is helpful. Another example is the use of unrestricted Boolean units, another model that may seem trivial at first sight, but which leads to useful insights for both autoencoders and deep architectures. Yet another important example is provided by networks of linear or polynomial threshold gates.

A second characteristic of this book is its connection to biology. Neural networks, deep learning, and the entire field of AI are deeply rooted in biology, in trying to understand how the brain works and the space of possible strategies to replicate and surpass its capabilities. This is evident in Turing’s foundational work on Turing machines, guided by the fundamental intuition of a brain capable of having only a finite number of states [633] and in the vocabulary of computer science, which is full of words clearly rooted in biology such as AI, machine learning, memory, computer vision, computer virus, genetic algorithms, and so forth. It is regrettable to see young students and practitioners of machine learning misled to believe that artificial neural networks have little to do with biology, or that machine learning is the set of techniques used to maximize engineering or business goals, such as advertisement revenues for search engines. In addition, not only computers and neural networks are inspired by biology, but they are of course also being successfully used to analyze biological data, for instance high-throughput omic data, and through one of these surprising self-recursions only mankind seem to have produced, the results of these bioinformatics and systems biology analyses are progressively informing our understanding of the brain, helping to reveal for instance key gene expression and protein mechanisms involved in synaptic formation and biological memory.

A third characteristic of this book is precisely in the applications. The second meaning of “in science” in the title is “for science”. I have focused on applications of deep learning to the natural sciences—primarily physics, chemistry, and biology for the past three decades or so. These applications are expanding rapidly today, but were almost nonexistent in the 1980s. Plenty of textbooks and other material can

be found dealing with applications of neural networks to problems in engineering and other related areas.

A fourth characteristic is the emphasis placed on storage, specifically on the neural-style of information storage, in fundamental contrast to the Turing-style of information storage, ironically introduced by Turing precisely while thinking about the brain. This theme goes together with the importance of recognizing the virtualization process hidden behind most of today's neural network applications. In most applications of neural networks today, there are no neurons and no synapses, only their digital mirage. This comes at a price that can only be understood by thinking about "learning in the machine", as opposed to machine learning. In a physical neural system, learning rules must be local both in space and time. Among other things, this locality principle helps clarify the relationship between Hebbian learning and backpropagation and explains why Hebbian learning applied to feed-forward convolutional architectures has never worked. It also naturally leads to random backpropagation and recirculation algorithms, important topics that are poorly known because not particularly useful for current applications. For readers primarily interested in applications, or for courses with tight time limitations, I recommend using the abbreviated sequence of chapters: 2, 3, 6, and 10, covering most of the practical aspects.

Finally, the field of neural networks has been polluted by fads and a significant amount of cronyism and collusion over the past few decades, that a fragmented, multigenerational, and often unaware community could do little to stop. These are nothing new in human affairs, but they have distorted or slowed down the development of the field. Readers should read more widely, check what has been published – where and when – and decide for themselves which results are supported by mathematical proofs or sound simulations, and which are not. In the end, towering over human affairs, only matter the beauty of deep learning and the underlying mysteries it is intimately connected to: from whether silicon can be conscious to the fundamental nature of the universe.

Acknowledgment

The number of people I am indebted to keeps growing every year, and I can only mention a few of them.

- As a graduate student at Caltech (1983-1986) and visiting lecturer at UCSD (1986-1988), I was fortunate to be able to participate and contribute to the early beginnings of neural networks in the 1980s. Being at those two universities, which were the hotbeds of neural networks research at the time, resulted to a large extent from a series of chance encounters with several individuals, of

whom I can only mention two: Brian Ekin and Gill Williamson. Brian, who I met by chance in Paris, told me to apply to Caltech, a name I had never heard before. And while bartending for an alumni reunion in the basement of the Caltech Athaneum, I met Gill Williamson who was a Professor at UCSD and, while still sober, offered me my first academic job.

- From those early times, I wish also to acknowledge two kind mentors—Edward Posner and Walter Heiligenberg—both of whom died prematurely in tragic transportation accidents; as well as some of my early collaborators including: Amir Atiya, Eric Baum, Yves Chauvin, Kurt Hornik, Ron Meir, Fernando Pineda, Yosi Rinott, and Santosh Venkatesh.
- From more recent times, I wish to acknowledge past and present faculty colleagues at UCI in AI and machine learning, including: Rina Dechter, Charles Fowlkes, Roy Fox, Richard Granger, Alexander Ihler, Dennis Kibler, Richard Lathrop, Stephan Mandt, Eric Mjølness, Michael Pazzani, Deva Ramanan, Sameer Singh, Padhraic Smyth, Erik Sudderth, Max Welling, and Xiaohui Xie.
- Successful applications of deep learning to the sciences vitally require interdisciplinary collaborations. I am deeply grateful to all my collaborators from the natural sciences. Among the current ones at UCI, I can only mention: Roman Vershynin (Mathematics) and Babak Shahbaba (Statistics), Daniel Whiteson and Jianming Bian (Physics), Michael Pritchard (Earth Sciences), David Van Vranken (Chemistry), and Paolo Sassone-Corsi and Marcelo Wood (Biology). As I was about to send the final draft to Cambridge University Press, Paolo unexpectedly died and I wish to honor his memory here. He was an outstanding scientist, friend, and collaborator.
- Together with my external collaborators, I am also deeply grateful to current and past students and postdoctoral fellows in my research group, who have contributed in some many ways to this book over the years, including: Forest Agostinelli, Alessio Andronico, Chloe Azencott, Kevin Bache, Pierre-François Baisnée, Ryan Benz, Vincenzo Bonnici, Martin Brandon, Andrew Brethorst, Jocelyne Bruand, Francesco Ceccarelli, Nicholas Ceglia, Ivan Chang, Jonathan Chen, Siwei Chen, Jianlin Cheng, Davide Chicco, Julian Collado, Kenneth Daily, Ekaterina Deyneka, Pietro Di Lena, Yimeng Dou, David Fooshee, Clovis Galliez, Steven Hampson, Lars Hertel, Qian-Nan Hu, Raja Jurdak, Matt Kayala, John B. Lanier, Christine Lee, Lingge Li, Erik Linstead, Junze Liu, Yadong Lu, Alessandro Lusci, Christophe Magnan, Antonio Maratea, Stephen McAleer, Ken Nagata, Francesco Napolitano, Ramzi Nasr, Jordan Ott, Vishal Patel, Gianluca Pollastri, Liva Ralaivola, Arlo Randall, Paul Rigor, Alex Sadovsky, Peter Sadowski, Muntaha Samad, Hiroto

Saigo, Siyu Shao, Alexander Shmakov, Suman Sundaresh, S. Joshua Swamidass, Mike Sweredoski, Amin Tavakoli, Gregor Urban, Alessandro Vullo, Eric Wang, Lin Wu, Yu Liu, and Michael Zeller.

- I am equally grateful to Janet Ko, who has assisted me and other faculty, for so many years with her loyalty and outstanding administrative skills, always shielding scientific research from bureaucracy.
- As far as the book itself is directly concerned, two chapters and the appendix reuse material from three previously published articles [70, 557, 243] and I thank the publishers—Springer and Annual Reviews—for their permissions. I wish also to thank Annie Vogel-Cierna for providing me two of the microscopy images in the last chapter.
- This book was finished during the COVID-19 pandemic. It has been a real pleasure to work with the staff of Cambridge University Press, and in particular with David Tranah. I thank them for their outstanding support and professionalism, and their deep understanding of academic publishing.
- Last but not least, I am deeply grateful to my closest friends and to my family.