Extracted from:

Programming Machine Learning

From Coding to Deep Learning

Programming Machine Learning

From Coding to
Deep Learning

Paolo Perrotta

edited by Katharine Dvorak

Programming Machine Learning

From Coding to Deep Learning

Paolo Perrotta

Pragmatic Bookshelf

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

For our complete catalog of hands-on, practical, and Pragmatic content for software developers, please visit *https://pragprog.com*.

The team that produced this book includes:

Publisher: Andy Hunt
VP of Operations: Janet Furlow
Executive Editor: Dave Rankin
Development Editor: Katharine Dvorak
Copy Editor: Jasmine Kwityn
Indexing: Potomac Indexing, LLC
Layout: Gilson Graphics

For sales, volume licensing, and support, please contact *support@pragprog.com*.

For international rights, please contact *rights@pragprog.com*.

To my wife Irene,

making my every day.

How the Heck Is That Possible?

Machine learning can seem like magic. How can a computer recognize the objects in an image? How can a car drive itself?

Those feats are baffling—not just to the layman, but to many software developers like you and me. Even after writing code for many years, I had no idea how machine learning could possibly work. While I tinkered with the latest web framework, someone out there was writing amazing software that looked like science fiction—and I couldn't even comprehend it.

I wanted in on the action. I wanted to be able to build those things myself.

I knew how to write software, so I assumed that I would grok machine learning quickly. I mean, how hard could it be? I put on a confident smile and started studying. Then I kept smiling confidently as I slammed my muzzle into a long sequence of brick walls.

To us developers, machine learning feels… *foreign*. The field is teeming with math jargon, researchy conventions, and frankly, bad code. Instead of tutorials, people point you at lectures and research papers. For many of us, machine learning is as intimidating as it's intriguing.

This is the book I missed when I got started with machine learning: an introduction for developers, written in our own language. After reading it, you'll be comfortable with the fundamentals, and able to write machine learning programs. No, you probably won't be able to build your own self-driving car just yet—but at least you will know how the heck that's possible.
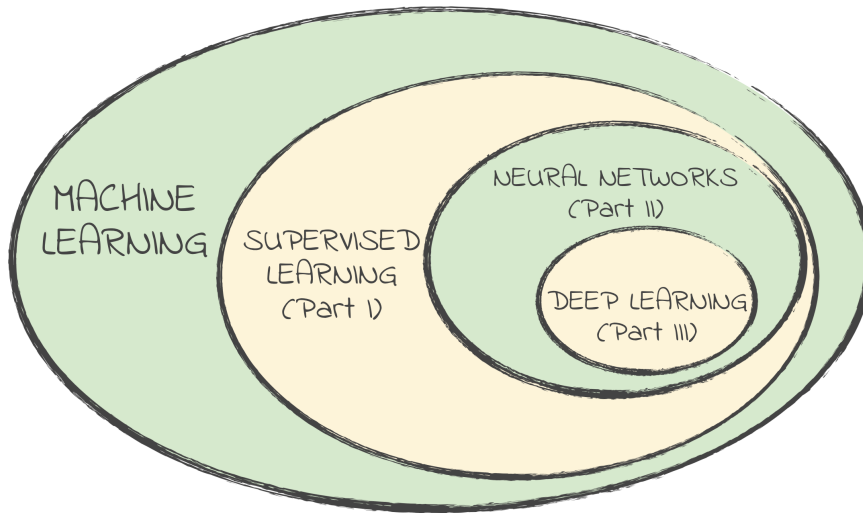
Come in.

About This Book

This is a book for developers who want to learn machine learning from scratch.

Machine learning is a broad field, and no book can cover it all. We'll focus on the three facets of machine learning that are most important today: *supervised*

learning, *neural networks*, and *deep learning*. We'll look into those terms as we go through the book, but here's a picture and a few quick definitions to get you started:

Supervised learning is a specific type of machine learning. Machine learning comes in a few different flavors, but supervised learning is the most popular one these days. Part I of this book, *From Zero to Image Recognition*, is a hands-on supervised learning tutorial. Within a couple of chapters, we'll write a minimal learning program. Then we'll evolve this program step by step, progressively turning it into a machine learning system called a *perceptron*. Our perceptron will be a bona fide computer vision program, powerful enough to recognize handwritten characters. We'll craft it all by ourselves, without using machine learning libraries. You'll understand each single line of its code.

There are many ways to implement a supervised learning system. The most popular of those is the *neural network*—a brilliant algorithm that was loosely inspired by the connections of neurons in our own brains. Part II of this book is dedicated to neural networks. We'll grow the program from Part I into a full-fledged neural network. We'll have to overcome a few challenges along the way, but the payoff will be worth it: the final neural network will be way more powerful than the fledgling program we'll start off with. Once again, we'll write the code ourselves, line by line. Its inner workings will be open for you to play with.

Neural networks got a big boost in recent years, when researchers came up with breakthrough techniques to design and use them. This souped-up technology

is vastly more powerful than the simple neural networks of old—so much so that it got its own name: *deep learning.* That's also the title of Part III of this book. In it, we'll rewrite our neural network using a modern machine learning library. The resulting code will be our starting point to understand what deep learning is about. Finally, as we wrap up the book, we'll take a look at a few advanced deep learning techniques, paving the way for your future explorations.

In truth, things aren't quite as clean-cut as our picture implies. For example, neural networks can be used in other fields of machine learning, not just in supervised learning. However, the diagram just shown is a good starting point to get a sense of the topics in this book, and how they fit together.

Before We Begin

This book cannot turn you into a machine learning pro overnight, but it can give you an intuitive, practical understanding of how machine learning works. I want to open the hood of this discipline, show you the gears, and demystify the magic. Once you grasp the fundamental principles of machine learning, you'll find it much easier to dig deeper, incorporate these techniques in your daily job, and maybe even embark on a career as a machine learning engineer.

You don't need to be a senior developer to read this book. However, you should be comfortable writing short programs. If you know Python, then you just lucked out: that's the language that I will use throughout, so you'll feel right at home. Even if you don't know Python yet, no worries. It's a friendly language, and the code in this book will never get too complicated. Read Appendix 1, Just Enough Python, on page ? to get up to speed, and be ready to Google for more information if you get stuck.

Machine learning involves a lot of mathematics. I won't dumb down the math, but I'll make it as intuitive as I can. To go through this book, you need some high school level concepts: I'm going to assume that you can read a Cartesian chart, that you know what the "axes" and their "origin" are, and that you can make sense of a function plot. Other than that, you don't need much math knowledge. In fact, you might be able to read through even if you consider yourself terrible at math… But be ready to be proven wrong about that!

On the other hand, if you have a solid background in linear algebra and calculus, then you might find some of the math obvious. In that case, feel free to breeze over the explanations you don't need.

Math Deep Dives

We all love intuitive mathematics, but sometimes you might strive for a more formal explanation. If you ever feel lost while parsing a formula, or if you like math and want to dig deeper, then look for "Math Deep Dive" boxes like this one. They'll point you at relevant math screencasts on the excellent Khan Academy.[1] No matter what your current level of math is, this site has got you covered.

Just to be clear, these additional lessons are optional. You don't need them to read this book—only if you wish to really wrap your mind around the mathematics of machine learning.

Machine learning has a rich and specific vocabulary. You're likely to stumble upon new words, or new meanings for old words. Take it easy and don't feel like you have to remember everything. I will remind you of many of those words' meaning the next time we encounter them. Whenever a term gives you that obnoxious "I cannot quite remember what this means" feeling, you can look it up in Appendix 2, The Words of Machine Learning, on page ?.

One word about the datasets that I'll use in the examples: many of them are collections of images. Rest assured that machine learning can do much more besides image recognition: it can analyze text, generate music, or even hold conversations. However, image recognition makes for very intuitive examples, so it will be our go-to application throughout the book.

Finally, there are a couple of online resources that you should know about. One is this book's official page[2] on the Pragmatic Bookshelf. From there, you can download the examples' source code and report errata—typos, bugs, and factual mistakes.

This book also has a companion website called ProgML[3] that contains a few additional explanations that I couldn't fit in these pages. Every now and then, you'll find references to ProgML in the page margins, pointing you at those optional explanations. Go read them if you're eager for more details.

References to ProgML look like this.

That's enough public service announcements. Let's dive into Part I.

---

1. www.khanacademy.org
2. https://pragprog.com/book/pplearn/programming-machine-learning
3. www.progml.com