1 Introducing Deep Learning: Why you should learn it

In this chapter:

- Why you should learn deep learning
- Why you should read this book
- What you need to get started

Do not worry about your difficulties in mathematics.

I can assure you mine are still greater.

—**Albert Einstein**

1.1 Welcome to Grokking Deep Learning

You're about to learn some of the most valuable skills of the century!

I'm very excited that you're here! You should be too! Deep Learning represents an exciting intersection of Machine Learning and Artificial Intelligence, and a very significant disruption to society and industry. The methods discussed in this book are changing the world all around you. From optimizing the engine of your car to deciding which content you view on social media, it's everywhere, it's powerful, and fortunately, it's fun!

1.2 Why you should learn Deep Learning

From the beginning of time, humans have been building better and better tools to understand and control the environment around us. Deep Learning is today's chapter in this story of innovation. Perhaps what makes this chapter so compelling is that this field is more of a *mental* innovation than a *mechanical* one. Much like its sister fields in Machine Learning, Deep Learning seeks to *automate intel igence* bit by bit, and in the past few years it has achieved enourmous success and progress in this endeavor, exceeding previous records in Computer Vision, Speech Recognition, Machine Translation, and many other tasks. This is particularly extraordinary given that Deep Learning seems to use *largely the same brain-inspired algorithm* (Neural Networks) for achieving these accomplishments across a vast number of fields. Even though Deep Learning is still an actively developing field with many challenges, recent developments have lead to tremendous excitement that perhaps we have in fact discovered more than just a great tool, but a window into our own minds as wel .

MEAP

Deep Learning has the potential for significant automation of skilled labor.

There is a substantial amount of hype around the potential impacts of Deep Learning if the current trend of progress is extrapolated at varying speeds. While many of these predictions are over-zealous, there is one that I think merits your consideration: job displacement. I think that this claim stands out from the rest for no other reason than if Deep Learning's innovations stopped *today*, there would already be an incredible impact on skilled labor around the globe. Call center operators, taxi drivers, and low-level business analysts are compelling examples where Deep Learning can provide a low-cost alternative. Fortunately, the economy doesn't turn on a dime, but in many ways we are already past the point of concern with the current power of the technology. It is my hope that you (and people you know) will be enabled by this book to transition from perhaps one of the industries facing disruption into an industry ripe with growth and prosperity:

Personal y, I got into Deep Learning because it's fascinating. It's an amazing intersection between man and machine. Unpacking exactly what it means to think, to reason, and to create is enlightening, engaging, and for me it's quite inspiring. Consider having a dataset filled with every painting ever painted, and then using that to teach a machine how to paint like Monet. Insanely, it's possible, and it's mind-bogglingly cool to see how it works.

1.3 Will this be hard to learn?

How hard will you have to work before there is a "fun" payoff?

This is my favorite question. My definition of a "fun" payoff is the experience of witnessing something that I built *learning.* There's just something amazing about seeing a creation of your hands do something like that. If you also feel this way, then the answer is simple. A few pages into Chapter 3, you will create your first neural network. The only work involved between now and then is reading the pages between here and there.

After Chapter 3, you may be interested to know that the ***next*** fun payoff occurs after you have memorized a small snippet of code and proceeded to read to the midway of

Chapter 4. Each chapter will continue to work this way. Memorize a small code segment from the previous chapter, read the next chapter, and then experience the payoff of a new learning neural network.

1.4 Why you should read this book

It has a uniquely low barrier to entry.

mathematics (i.e. college degree in a mathy field). Don't get me wrong, there are real y good reasons for teaching it using math. Math is, after al , a language. It is certainly more **efficient** to teach Deep Learning using this language, but

I don't think it's absolutely necessary to assume advanced knowledge of math in order to become a skilled, knowledgable practitioner who has a firm understanding of the "how" behind Deep Learning. So, why should you learn Deep Learning using this book? I'm going to assume you have a High School level background in math (and that it's rusty), and *explain everything else you need to know as we go along.* Remember multiplication? Remember x-y graphs (the square with lines on it)? Awesome! You'll be fin e.

1.5 Why you should read this book (cont.)

To help you understand what's inside a framework (Torch, TensorFlow, etc.).

There are two major groups of Deep Learning educational material (books, courses, etc.). One group is focused around how to use popular frameworks and code libraries such as Torch, Tensorflow, Keras, and others. The other group is focused around teaching Deep Learning itself, otherwise known as the*science under the hood* of these major frameworks. Ultimately, learning about ***both*** is important. It's like if you want to be a NASCAR driver. You need to learn both about the particular model of car you're driving (the framework), and about driving itself (the science/skill). However, just learning about a framework is like learning about the pros and cons of a Generation-6 Chevrolet SS before you know what a stick shift is. This book is about teaching you what *Deep Learning* is so that you can then be prepared to learn a framework.

Whenever I encounter a math formula in the wild, I take a two-step approach. The first is to translate its methods into an intuitive *analogy* to the real world. I almost never just take a formula at face value. I break it into *parts*, each with a story of its own. That will be the approach of this book as wel . Anytime we encounter a math concept, I'll offer an alternative *analogy* for what the formula is actual y doing.

"Everything should be made as simple as possible, but no simpler"

– Albert Einstein

Everything after the introduction chapters is "project" based.

MEAP

If there is one thing I hate when learning something new, it is when I have to question whether or not what I'm learning is useful/relevant. If someone is teaching me everything there is to know about a hammer without actual y taking my hand and helping me drive in a nail, then they're not real y teaching me how to use a hammer. I know that there are going to be dots that weren't connected, and if I was thrown out into the real world with a hammer, a box of nails, and a bunch of 2x4s, I'm going to have to do some guesswork.

This book is about giving you the wood, nails, and a hammer *before* telling you about what they do. Each lesson is about picking up the tools and building stuff with them, explaining how stuff works along the way. In this way, you don't leave with a list of facts about the various Deep Learning tools we'll work with, you leave with the ability to use them to solve problems. Furthermore, you will understand the most important part: when and why each tool is appropriate for each problem you want to solve. It is with this knowledge that you will be empowered to pursue a career in research and/or industry.

My absolute favorite place to work is a Jupyter Notebook. One of the most important parts of learning Deep Learning (for me), is the abiliy to stop a network while it's training and tear apart absolutely every piece to see what it looks like. This is something that Jupyter

Notebook is incredibly useful for. As for Numpy, perhaps the most compelling case for why this book leaves nothing out is that we'll only be using a single matrix library. In this way, you will understand **how** everything works, not just how to call a framework. This book teaches Deep Learning from absolute scratch, soup to nuts. Instal ation instructions for these two tools can be found at (http://jupyter.org/) for Jupyter and (http://numpy.org) for NumPy. I will be building these examples in Python 2.7, but will test them for Python 3 as wel . For easy instal ation, I also recommend the Anaconda framework (https://docs.continuum.io/anaconda/install) .

MEAP

Pass High School Mathematics

There are some mathematical assumptions that are simply out of depth for this book, but the goal of this book is to teach Deep Learning only assuming you understand basic algebra.

Find a personal problem you are interested in

This might seem like an optional "need" to get started. I guess it could be, but seriously, I highly, highly recommend finding one. Everyone I know who has become successful at this stuff had some sort of problem they were trying to solve. Learning Deep Learning was just a "dependency" to solving some other interesting task. For me, it was using Twitter to predict the stock market. It's just something that I thought was real y fascinating. It's what drove me to sit down and read the next chapter and build the next prototype. And as it turns out, this field is **so new**, and is changing **so fast**, that if you spend the next

Learning, having a problem you're fascinated with that involves using one dataset to predict another is the key catalyst! Go find one!

1.7 You'll probably need some Python knowledge

Python is my teaching library of choice, but I'll provide a few others online.

Python is an amazingly intuitive language. I think it just might be the most widely adopted and intuitively readable language yet constructed. Furthermore, the Python community has a passion for simplicity that can't be beat. For these reasons, I want to stick with Python for all of the examples (Python 2.7 is what I'm working in). On this book's Github, I'll provide all of the examples in a variety of other languages as wel , but for the in-page explanations, we're going to use Python.

MEAP

1.8 How much coding experience should you have?

Scan through the Python Codecademy course (https://www.codecademy.com/learn/ python). If you can read through the table of contents and feel comfortable with the terms mentioned, you're all set! If not, then just take the course and come back when you're done!

It's designed to be a beginner course and it's very well crafted.

1.9 Conclusion and Primer for Chapter 2

If you've got your Jupyter Notebook in-hand and feel comfortable with

Up next...

2 Fundamental Concepts: How do machines learn?

- What are Deep Learning, Machine Learning, and Artificial Intelligence?
- What is a Parametric Model?
- What is a Non-Parametric Model?
- What is Supervised Learning?
- What is Unsupervised Learning?
- How can machines learn?

MEAP