

# Machine Learning from Scratch-ish

How I learned to stop worrying about back-propagation



Matthew Dunn [Follow](#)

Nov 1 · 12 min read ★



Actor Peter Sellers in “**Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb**”

## The Problem

**“I am really interested in Machine Learning, but I don’t have a background in Computer Science or Math.”**

You might have come across this sentiment before. There is a good chance, if you are reading this article with any purpose and intent, that you have come across this sentiment quite often. And, in fact, I am guessing that there is a better than average chance it is possible that you have *personally* experienced this sentiment.

I certainly have. That is the genesis and the foundation of this article. For quite some time, I was genuinely frustrated to the point of distraught, as I truly and honestly believed that someone with my educational background and experience was likely incapable of understanding or working with Machine Learning techniques.

## Can an “average” person do Machine Learning?

The short answer is *yes*. We can argue over what “average” means in this context, but my own personal experience (which is admittedly quite limited) indicates to me that a background in those disciplines that are *assumed* to be required (*e.g.* computer science, calculus and other forms of higher-level math, statistics, programming, etc.), while certainly helpful, is either **(i)** not required at all or, **(ii)** if actually required, typically at a significantly more surface level than is initially perceived.

## Becoming a Machine Learning Beginner

### An overview

I can't tell you how to become a Machine Learning expert for no other reason than I myself am just beginning down this path. But, I do feel comfortable in saying that the path that I have started on to “Machine Learning Beginner” is accessible and should be repeatable across a broad spectrum of individuals with a wide variety of backgrounds.

### Who am I?

As I mentioned above, I have been interested in Machine Learning, Artificial Intelligence, Natural Language Processing and other areas of cutting-edge computing, but honestly and truly felt that my lack of a “technical” education background was holding me back.

To be clear, by society's bizarre standard of equating a diploma with actual intellectual accomplishment, I am considered very educated. I have an undergraduate degree in Russian Language and Literature from Dartmouth College, a master's degree in the same from Harvard University, and a law degree from Suffolk University (in Boston, Massachusetts).

Until 2016, I was practicing as a corporate attorney at a very large, very prominent, international law firm, where I focused very narrowly on private equity transactions. In the middle of 2016, I made a completely age-appropriate decision (at age 45) to abandon a very lucrative career as a partner-track attorney to become a computer programmer, a discipline in which I had no prior formal training and no real substantive knowledge.

For the sake of transparency, it wasn't *quite* that bad. I had always been interested in computers and had, prior to attending law school, worked for a few software companies, but any programming experience I had was **(i)** extraordinarily minimal, **(ii)** completely self-taught, and **(iii)** acquired in 2001–2002, which means that the knowledge was, as a practical matter, essentially useless.

After about a year of self-teaching, I convinced a company to take a chance on me. I used that job to learn C#, improved my JavaScript, continued to learn about web application architecture, etc. Fast forward to today, I work at the Center for Clinical Data Science, but not as a Data Scientist.

So, this is where I stand as I begin my journey into Data Science in the second half of 2019:

- **I don't have any formal education in Computer Science and I have not taken a class in Mathematics since my junior year of high school, in 1989.**
- **I have been working professionally as a programmer since 2016, but I have fairly limited experience with Python.**
- **I work in close proximity to Data Scientists, but I don't work *with* any of them and am certainly not privy to any learning from them that couldn't otherwise readily be gleaned from Google.**

If you believe you are, or might be, similar to me, I sincerely urge you to continue reading.

# How to fail

Tutorials and help articles are great at trying to tell you what you should *affirmatively* be doing, but they traditionally spend significantly less time explaining what *not to do*. I believe that setting yourself up for success includes making sure to avoid potential traps and failures, and this journey has more than its fair share of those for the unwary.

You will almost certainly fail in your journey to become a Machine Learning Beginner if:

## You lack patience and resolve

I can tell you as a programmer, if not as a Machine Learning practitioner, that learning to program is an exercise in patience. Computers are, in a very important (and yet ironic) sense, extraordinarily dumb machines — they do only what you tell them and literally nothing more.

If you do not have the patience to sit in front of a computer and engage in the iterative process of coding, then debugging, and then researching, this is probably not going to be enjoyable for you.

## You lack curiosity

Regardless of the path you eventually choose to take to acquire your Machine Learning knowledge, it is almost certain that you will be given the advice to “type in all the code” or “work along with the exercises” or “go out and build the thing that we just built, but slightly different”. These are all just variants of the same idea, which is that there is value in just “tinkering” — experimenting, reading the documentation when something fails, looking at other peoples’ source code, etc. If you are not fundamentally driven and sustained by *your own* curiosity and a desire to “build”, you will almost certainly not keep up with those who are.

## You overestimate the difficulty of the learning curve

This is, in essence, a re-statement of the initial concern, meaning that this is simply a concern as to whether someone with a non-traditional background can succeed in the world of Machine Learning. If this article accomplishes anything, I hope that it will at least dispel the myth that Machine Learning is completely unapproachable, or

otherwise only for the lucky few that happen to have dual PhDs in Computer Science and Applied Mathematics/Statistics.

## You underestimate the difficulty of the learning curve

I have posited that this is all *do-able*, not that it is *easy*. If it was easy, suffice it to say that everyone would be doing it, and if everyone was doing it, educational background could not, by force of logic, be a relevant factor, which means the operative question that this article attempts to address would be moot.

The point simply is that there will be hard bumps in the road and how you deal with those will likely be more of a determining factor in the outcome of your success. Are you going to start scouring forums, mailings lists, Twitter, Medium, etc. and really pound the pavement until you get a satisfactory answer? Or are you going to be unsettled in the face of unfamiliar material? See **“You lack patience and resolve”** above.

## You are intending to learn all of Machine Learning/AI next week, next month, this year...

The amount of material that can/could be learned in the process of just becoming a Machine Learning Beginner is simply overwhelming in its volume. To further complicate matters, the body of scholarship and relevant work product is increasing at such a rate that we tend to measure relevance in terms of single months and years as opposed to decades.

There is no “end” to the development of technology now or in the foreseeable future and to try to identify some point at which you will know “everything about machine learning” is to worship at the alter of a false god. Take your time, go over the relevant material a second and third time, write more code, build more things, document your learning process, and enjoy and celebrate the individual, smaller accomplishments along the way.

So, yes, it is quite obviously possible to fail and it is not particularly hard to do. But then again, it is also not particularly hard to avoid most of these issues.

## How to succeed at this

## Pick a course and stick to it

There are tons of great resources out there that offer a comprehensive introduction to beginning Machine Learning concepts. One well known option is Andrew Ng's course on Coursera. Another option is offered by fast.ai. There isn't a right or wrong decision —both have excellent reputations, with each course offering a teaching style that is slightly different than the other.<sup>1</sup>

The salient point here is that you should make every effort to stick to a single course of study. Switching courses when you are presented with something you don't understand is unlikely to produce meaningful progress in the average case over a statistically significant period of time. The solution is not “just one more book from Amazon” or “a different, less technical tutorial.” At some point, you simply have to buckle down, grit your teeth, and fight your way up and to the right of the learning curve.

To be eminently clear, I am *not* suggesting that you don't supplement your own study efforts with outside resources or otherwise consult third party materials when you have a problem. I *am* suggesting that at some point, as you get deeper and deeper, you will run into concepts that are difficult to understand simply because they are, in fact, difficult concepts. Embrace the difficulty as a sign that you are pushing yourself past your comfort zone, which is where (in my experience) all things that are worth discussing in this world happen. Changing courses or books, or video series just isn't a viable long-term learning strategy and is indicative of a mindset that is looking for a “silver bullet.” As stated above, there are no silver bullets, no shortcuts — there is just hard work.

## Pick a framework and stick to it

Now is not the time to compare and contrast TensorFlow with PyTorch with MXNet, etc. As a complete beginner, you don't have sufficient knowledge or understanding of either the problem space or the implementation of any of these frameworks for any such decision to be something other than a regurgitation of another's thought process.

Pick a mainstream framework with a large and active development community and stick to that. At this point, focus on learning over-arching concepts and avoid the

rabbit hole that is the particulars of a given framework that may or may not be relevant a year from now.

## Remember that “it’s just like French”

If you ever were required to study a foreign language, it’s quite possible that you have run into the concept of gendered nouns. To grossly over-simplify, in many languages (of which English is *not* one), certain nouns are either masculine or feminine. By way of example, take this excerpt from this article (my emphases added).

---

*There are some nouns that express entities with gender for which there is only one form, which is used regardless of the actual gender of the entity, for example, **the word for person; personne; is always feminine, even if the person is male, and the word for teacher; professeur; is always masculine even if the teacher is female.***

---

The point is that, just as trying to understand the logic of why a person is always feminine regardless of gender is a fruitless exercise (the answer is inevitably “it just is — accept it as an article of truth and move on”), trying to understand everything about Machine Learning during the first pass is simply not a reasonable expectation. There are certain things that you will simply have to take on faith for the time being.

That is not to say that you don’t note the question for further research at some later point in time, but trying to learn everything about everything that you don’t understand each time you confront something you don’t understand is merely a clever variation of the “**You are intending to learn all of Machine Learning**” discussed above.

Despite what your better angels may be telling you, your drive to try to delve into the details is *not* helping you learn. In fact, quite the opposite, as these are rabbit holes that just potentially delay the time until you obtain a solid grasp and foundation on the larger picture that provides the necessary context for more meaningful exploration.

## Establish a toolset and a repeatable workflow

I don’t have anything particularly interesting (and certainly nothing original) to add about the specifics of any one tool or workflow, other than to say it is important to

have a set of tools that you know, are comfortable with, and that are reasonably reliable.

There are multiple, well-documented solutions on everything from Google Cloud to AWS to Azure to Jupyter Notebooks to plugins for Vim, Emacs, VSCode, et al. Setting up an industry-grade Machine Learning pipeline for a single developer's use is, as of the latter half of 2019, very affordable and very well-documented.<sup>2</sup>

## **Get a decent piece of note-taking software**

I don't believe this gets enough attention. You will be, even in the initial stages of this journey, bombarded with information, some of which you will be able to grasp and some of which you will not. As stated above, given the state of our knowledge, there are certain things that you will have to take on faith for the time being. But you should absolutely noting those concepts that you do not understand so that you can eventually, when you have more context, do the deep dive that is truly required.

I personally use Coggle, but at the risk of boring you with repetition, it is not really about the tool you pick. Text files, Markdown files, Jupyter Notebooks, OneNote, Evernote, etc. will all work. My personal preference stems from the fact that I prefer "mind map"-style diagrams. As opposed to a hierarchical format (as in an outline, for instance), which I often find do not line up with the mental model of the material in my mind. A mind-map allows me to connect nodes to other nodes in fairly arbitrary patterns, which allows me to document things in a way that is more natural to my brain.

## **Learn to accept failure as a normal, resting state**

If you are transitioning to Machine Learning from something other than Software Engineering, this might seem non-intuitive. Put simply (and more than a little tongue-in-cheek), an overwhelming majority of the job could be colloquially described as "getting your code to compile", which, by definition, means that the majority of time, your code is in a non-working, failing state.

Of course, as human beings, we do everything in our power possible to exacerbate that feeling of inadequacy in others and ourselves by making sure that errors are shown in bold text, in bright red, accompanied by iconography that indicates that you are engaged in wrongdoing, etc.



The reality is that there is an aspect of this work that, like Software Engineering (and I feel comfortable saying this after having observed more than a few Data Scientists at work), represents a somewhat tedious cycle of experimentation, bug-identification and fixing, followed by further experimentation. If you don't have the tenacity, the drive to build, and a natural curiosity that is capable of powering you through those moments, this might not be the discipline for you.

## Closing Thoughts

Becoming even a Machine Learning Beginner is far from a simple undertaking, but then, nothing worth doing ever is.

I will leave you with a simple thought experiment — *if (i) Data Science is the fastest growing job in the United States and (ii) only people who have a Data Science background can fill that job, how can (i) be true? How can it possibly be the fastest growing job?*

In other words, for the first (I believe, spurious) assertion to be true, wouldn't there have to be scores of people just sitting around who, for some reason, just happen to have the perfect skill set required by an industry that didn't meaningfully exist a decade ago. Does that seem likely?

Or is it more likely that the more reasonable explanation — that otherwise capable individuals like yourselves and like me are cross-training and developing new skills in this area precisely because these skills are, in fact, accessible?

I leave that as an exercise to the curious reader.

## Footnotes

N.B. — I am not associated and/or affiliated any of the organizations mentioned in this article other than my current employer, the Center for Clinical Data Science. I do not receive any discounts or other benefits from any of the companies mentioned (other than my employer) that are not publicly available to anyone with a valid credit card as of the time of this writing.

[1] For the sake of complete disclosure, I am working through the fast.ai coursework. After briefly evaluating both the fast.ai coursework and Andrew Ng's course offering, I decided that the fast.ai approach would personally serve me better. I like the "top-down" approach in which, very generally speaking, you start by "doing" and then move to "understanding".

[2] I personally use Google Cloud Platform and their AI Notebooks service, but I have also used, and was happy with, AWS SageMaker.

[Data Science](#)   [Machine Learning](#)   [Artificial Intelligence](#)

**Medium**

[About](#) [Help](#) [Legal](#)