1

Introduction: Becoming a Unicorn

"Data science" is a very popular term these days, and it gets applied to so many things that its meaning has become very vague. So I'd like to start this book by giving you the definition that I use. I've found that this one gets right to the heart of what sets it apart from other disciplines. Here goes:

> Data science means doing analytics work that, for one reason or another, requires a substantial amount of software engineering skills.

Sometimes, the final deliverable is the kind of thing a statistician or business analyst might provide, but achieving that goal demands software skills that your typical analyst simply doesn't have. For example, a dataset might be so large that you need to use distributed computing to analyze it or so convoluted in its format that many lines of code are required to parse it. In many cases, data scientists also have to write big chunks of production software that implement their analytics ideas in real time. In practice, there are usually other differences as well. For example, data scientists usually have to extract features from raw data, which means that they tackle very open-ended problems such as how to quantify the "spamminess" of an e-mail.

It's very hard to find people who can construct good statistical models, hack quality software, and relate this all in a meaningful way to business problems. It's a lot of hats to wear! These individuals are so rare that recruiters often call them "unicorns."

The message of this book is that it is not only possible but also relatively straightforward to become a "unicorn." It's just a question of acquiring the particular balance of skills required. Very few educational programs teach all of those skills, which is why unicorns are rare, but that's mostly a historical accident. It is perfectly reasonable for a single person to have the whole palette of abilities, provided they're willing to ignore the traditional boundaries between different disciplines.

This book aims to teach you everything you'll need to know to be a competent data scientist. My guess is that you're either a computer programmer

The Data Science Handbook, First Edition. Field Cady.

looking to learn about analytics or more of a mathematician trying to bone up on their coding. You might also be a businessperson who needs the technical skills to answer your business questions or simply an interested layman. Whoever you are though, this book will teach you the concepts you need.

This book is not comprehensive. Data science is too big an area for any person or book to cover all of it. Besides, the field is changing so fast that any "comprehensive" book would be out-of-date before it came off the presses. Instead, I have aimed for two goals. First, I want to give a solid grounding in the big picture of what data science is, how to go about doing it, and the foundational concepts that will stand the test of time. Second, I want to give a "complete" skill set, in the sense that you have the nuts-and-bolts knowledge to go out and do data science work (you can code in Python, you know the libraries to use, most of the big machine learning models, etc.), even if particular projects or companies might require that you pick up a new skill set from somewhere else.

1.1 Aren't Data Scientists Just Overpaid Statisticians?

Nate Silver, a statistician famous for accurate forecasting of US elections, once famously said: "I think data scientist is a sexed-up term for statistician." He has a point, but what he said is only partly true. The discipline of statistics deals mostly with rigorous mathematical methods for solving well-defined problems. Data scientists spend most of their time getting data into a form where statistical methods could even be applied. This involves making sure that the analytics problem is a good match to business objectives, extracting meaningful features from the raw data and coping with any pathologies of the data or weird edge cases. Once that heavy lifting is done, you can apply statistical tools to get the final results, although, in practice, you often don't even need them. Professional statisticians need to do a certain amount of preprocessing themselves, but there is a massive difference in degree.

Historically, data science emerged as a field independently from statistics. Most of the first data scientists were computer programmers or machine learning experts who were working on Big Data problems. They were analyzing datasets of the kind that statisticians don't touch: HTML pages, image files, e-mails, raw output logs of web servers, and so on. These datasets don't fit the mold of relational databases or statistical tools, so for decades, they were just piling up without being analyzed. Data science came into being as a way to finally milk them for insights.

In 20 years, I suspect that statistics, data science, and machine learning will blur into a single discipline. The differences between them are, after all, really

just a matter of degree and/or historical accident. But in practical terms, for the time being, solving data science problems requires skills that a normal statistician does not have. In fact, these skills, which include extensive software engineering and domain-specific feature extraction, constitute the overwhelming majority of the work that needs to be done. In the daily work of a data scientist, statistics plays second fiddle.

1.2 How Is This Book Organized?

This book is organized into three sections. The first, The Stuff You'll Always Use, covers topics that, in my experience, you will end up using in almost any data science project. They are core skills, which are absolutely indispensable for data science at any level.

The first section was also written with an eye toward people who need data science to answer a specific question but do not aspire to become full-fledged data scientists. If you are in this camp, then there is a good chance that Part I of the book will give you everything you need.

The second section, Stuff You Still Need to Know, covers additional core skills for a data scientist. Some of these, such as clustering, are so common that they almost made it into the first section, and they could easily play a role in any project. Others, such as natural language processing, are somewhat specialized subjects that are critical in certain domains but superfluous in others. In my judgment, a data scientist should be conversant in all of these subjects, even if they don't always use them all.

The final section, Stuff That's Good to Know, covers a variety of topics that are optional. Some of these chapters are just expansions on topics from the first two sections, but they give more theoretical background and discuss some additional topics. Others are entirely new material, which does come up in data science, but which you could go through a career without ever running into.

1.3 How to Use This Book?

This book was written with three use cases in mind:

1) You can read it cover-to-cover. If you do that, it should give you a self-contained course in data science that will leave you ready to tackle real problems. If you have a strong background in computer programming, or in mathematics, then some of it will be review.
2) You can use it to come quickly up to speed on a specific subject. I have tried to make the different chapters pretty self-contained, especially the chapters after the first section.

3) The book contains a lot of sample codes, in pieces that are large enough to use as a starting point for your own projects.

1.4 Why Is It All in Python™, Anyway?

The example code in this book is all in Python, except for a few domain-specific languages such as SQL. My goal isn't to push you to use Python; there are lots of good tools out there, and you can use whichever ones you want.

However, I wanted to use one language for all of my examples. This keeps the book readable, and it also lets readers follow the whole book while only knowing one language. Of the various languages available, there are two reasons why I chose Python:

1) Python is the most popular language for data scientists. R is its only major competitor, at least when it comes to free tools. I have used both extensively, and I think that Python is flat-out better (except for some obscure statistics packages that have been written in R and that are rarely needed anyway).
2) I like to say that for any task, Python is the second-best language. It's a jack-of-all-trades. If you only need to worry about statistics, or numerical computation, or web parsing, then there are better options out there. But if you need to do all of these things within a single project, then Python is your best option. Since data science is so inherently multidisciplinary, this makes it a perfect fit.

As a note of advice, it is much better to be proficient in one language, to the point where you can reliably churn out code that is of high quality, than to be mediocre at several.

1.5 Example Code and Datasets

This book is rich in example code, in fairly long chunks. This was done for two reasons:

1) As a data scientist, you need to be able to read longish pieces of code. This is a nonoptional skill, and if you aren't used to it, then this will give you a chance to practice.
2) I wanted to make it easier for you to poach the code from this book, if you feel so inclined.

You can do whatever you want with the code, with or without attribution. I release it into the public domain in the hope that it can give some people a small leg up. You can find it on my GitHub page at www.github.com/field-cady.

The sample data that I used comes in two forms:

1) Test datasets that are built into Python's scientific libraries
2) Data that is pulled off the Internet, from sources such as Yahoo and Wikipedia. When I do this, the example scripts will include code that pulls the data.

1.6 Parting Words

It is my hope that this book not only teaches you how to do nut-and-bolts data science but also gives you a feel of how exciting this deeply interdisciplinary subject is. Please feel free to reach out to me at www.fieldcady.com or field.cady@gmail.com with comments, errata, or any other feedback.