vickiboykis.com

Data science is different now

20–25 minutes

Woman holding a balance, Vermeer 1664

What do you think of when you read the phrase 'data science'? It's probably some combination of keywords like statistics, machine learning, deep learning, and 'sexiest job of the 21st century'. Or maybe it's an image of a data scientist, sitting at her computer, putting together stunning visuals from well-run A/B tests. Either way, it's glamorous, smart, and sophisticated. This is the narrative that data science has been selling since I entered the field almost ten years ago.

I started out as a data analyst.

While I was still mostly waiting for SQL queries to finish and cleaning extremely dirty and sad Excel files, I was reading Hacker News posts about mining massive datasets, Facebook's hot new data science team, and Hal Varian, and dreaming.

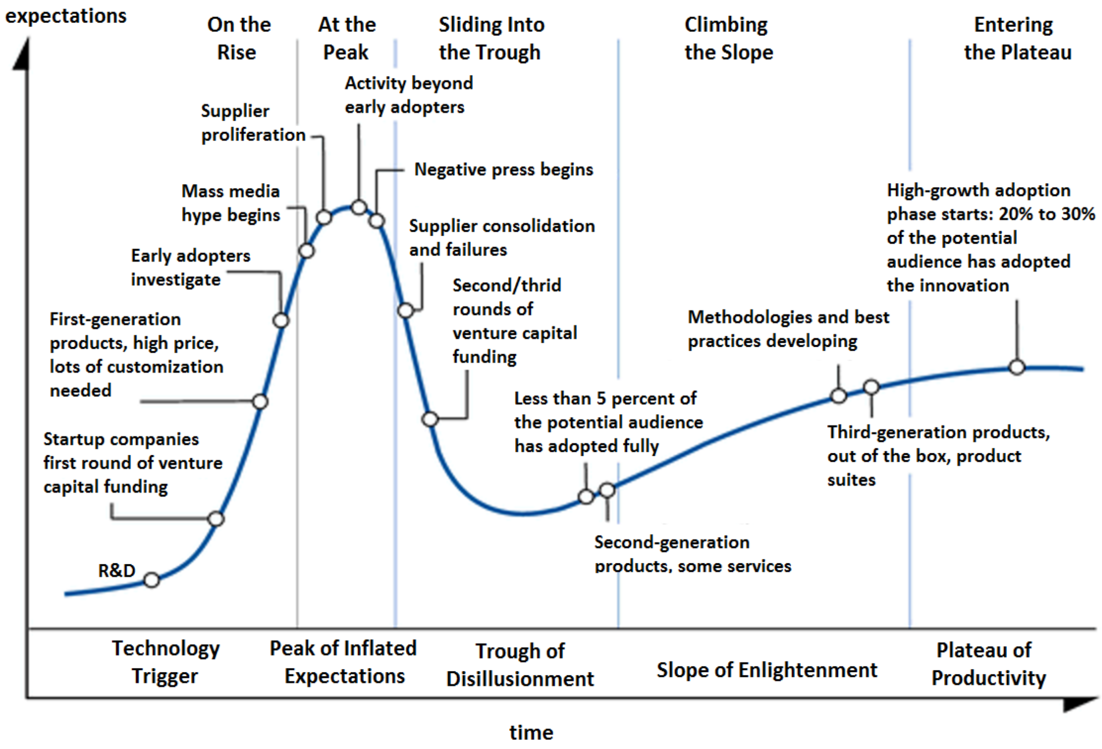
In 2012, I lucked out by being put on an analytics/engineering team that was transitioning some of its ETL processes from Oracle to Hadoop to keep up with data throughput.

I volunteered to be the first of the analysts to work with Pig and Hive, mainly because I was too impatient to wait for engineering work to be completed before having access to my data. But also, I was starstruck by this glowy, mysterious aura around data scientists - people who performed cool experiments, presented cool analyses, and got to have a MacBook for work.

I wanted to be one of those people! So, I learned Python online, brushed up on all of the statistics I'd taken in undergrad, and made a lot of command line mistakes in working with HDFS. In those early years, there was no real formalized way to learn "data science," other than to see what everyone else was doing, go to meetups, and try to read the tea leaves from HR job descriptions.

After fumbling on my own for a very long time, I've now been established in "data science" for the past 6 years, and, to serve as the mentor that I didn't have, I've been answering emails and having coffee meetings with people looking for advice to get into data science.

Since 2012, the data science industry has moved extremely quickly. It's gone through almost every stage in the Gartner hype cycle.


expectations
On the Rise
At the Peak
Sliding Into the Trough
Climbing the Slope
Entering the Plateau
Activity beyond early adopters
Supplier proliferation
Negative press begins
Mass media hype begins
Supplier consolidation and failures
High-growth adoption phase starts: 20% to 30% of the potential audience has adopted the innovation
Early adopters investigate
Second/thrid rounds of venture capital funding
Methodologies and best practices developing
First-generation products, high price, lots of customization needed
Less than 5 percent of the potential audience has adopted fully
Third-generation products, out of the box, product suites
Startup companies first round of venture capital funding
Second-generation products, some services
R&D
Technology Trigger
Peak of Inflated Expectations
Trough of Disillusionment
Slope of Enlightenment
Plateau of Productivity
time

We've been through the early adoption phase, the negative press around AI and bias, the second and third rounds of venture capital for companies like Facebook, and are now at the point of high-growth adoption: where banks, healthcare companies, and other Fortune 100 companies that move five years behind the market are also hiring for data science in machine learning.

A lot has changed. Big Data (remember Hadoop? and Pig?) is out. R has seen a meteoric rise in adoption. Python was written up in the Economist. Then the cloud changed everything all over again.

Unfortunately, what has not changed is the mass media hype around the field of data science, which has trumpeted data scientist as the 'sexiest career of the 21st century' so many times, that there is now what I believe to be an important problem that we as a community need to talk about. That problem is an oversupply of junior data scientists hoping to enter the industry, and mismatched expectations on what they can hope to find once they do get that coveted title of "data scientist."

Glut of new data scientists

First, let's talk about the oversupply of junior data scientists. The continuing media hype cycle around data science has enormously exploded the amount of junior talent available on the market over the past five years.

This is purely anecdotal evidence, so take it with a large grain of salt. But, based on my own participation as a resume screener, mentor to data scientists leaving boot camps, interviewer, interviewee, and from conversations with friends and colleagues in similar positions, I've developed an intuition that the number of candidates per any given data science position, particularly at the entry level, has grown from 20 or so per slot, to 100 or more. I was talking to a friend recently who had to go through 500 resumes for a single opening.

This is not abnormal. More anecdotal evidence comes from job openings like this one, from machine learning's godfather, Andrew Ng, whose AI startup demanded 70-80 hours a week. He was flooded with applications, after blithely noting that previously many people had tried to volunteer for free. As of this latest writing, they ran out of space in their current office.

It's very, very hard to estimate the true gap between market demand and supply, but here's a starting point.

> A study of job ads from April found more than 10,000 vacancies in the US for people with AI or machine-learning skills.

The article goes on to note,

> More than 100,000 people have started a deep learning course offered by Fast.ai, a startup focused on widening use of AI.

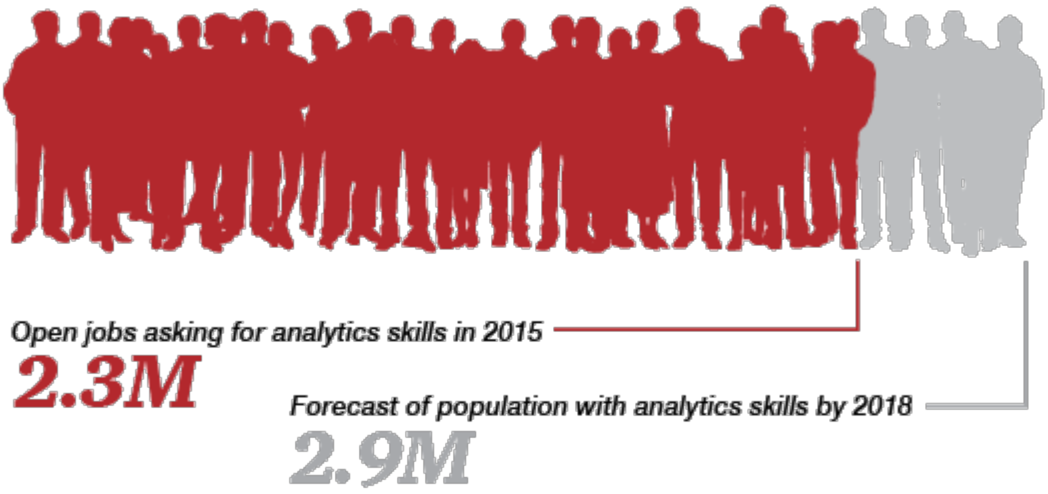
Assuming an average MOOC completion rate of around 7%, that would mean 7,000 people are available to fill those 10,000 jobs. For a single year. But, how about next year. Are we assuming a steady rate of data science job creation? If anything, the data science job market as such looks set to shrink, in line with my personal expectations.

Looking at a larger study, LinkedIn says there are 151,717 people with data science skills missing in the market. Although it's unclear whether this directly means data scientists or just people with some subset of those skills, let's assume that it's the former. So, there are 150,000 vacancies for data scientists in the country.

Given that there are 100,000 that have started a data science course, let’s assume again that 7,000 of these finish.

But, neither of those numbers is taking into account all of the programs and avenues for creating new data science candidates: MOOCs outside of Fast.ai like Coursera, over 10 nationwide bootcamps like Metis and General Assembly that have cohorts of 25 people every 12 weeks, remote degrees from places like UCLA, on-site undergraduate degrees in analytics and data science, YouTube, and more. There are also a large amount of PhDs who, unable to find jobs in an extremely tight job market, are migrating from academia to data science.

Here’s a third corroborating account,which noted that, in 2015, there were 40k job openings for data scientists. It estimated in general that the market supply for analytics skills (again, a much larger swath than data science, but still a point of comparison), would overcrowd the market by 2018.



Notes: US data only.
Source: Burning Glass Technologies analysis of 26.9 million US job postings from 2015. McKinsey Global Institute, Big Data: The next frontier for innovation, competition, and productivity (June 2011).

The amount of junior talent entering data science programs. Combine this with the hundreds of bootcamps putting on data science curricula, and, as someone looking for an industry to enter, you’re looking at a perfect storm.

On top of the gut feel that I have from working in the industry and talking to 100+ people who also do, these two tweets finally convinced me that there is a true data science supply bubble. First, this intro class tweet:

and UVA starting up a data science school.

Since academia is typically a lagging indicator in adoption to new trends in the work place, it's been long enough that it's truly worrying for junior data scientists, all of who are hoping to find data science positions. It can be very hard for someone with a new degree in data science to find a data science position, given how many new people they're competing with in the market.

This wasn't the case even three, four years ago, but now that data science has changed from a buzzword to something even larger companies outside of the Silicon Valley bubble hire for, positions have not only become more codified, but with more rigorous entry requirements that will prefer people with previous data science experience every time. Data science interviews are still very hard to get right, and still a complete mismatch for jobs.

As many blog posts point out, you won't necessarily land your dream job on the first try. As a result, the market can be very hard, and very discouraging for the flood of beginners.

Data science as a misleading job req

The second issue is that once these junior people get to the market, they come in with an unrealistic set of expectations about what data science work will look like. Everyone thinks they're going to be doing machine learning, deep learning, and Bayesian simulations.

This is not their fault; this is what data science curriculums and the tech media emphasize. Not much has changed since I first glanced, starry-eyed, at Hacker News logistic regression posts many, many moons ago.

The reality is that “data science” has never been as much about machine learning as it has about cleaning, shaping data, and moving it from place to place.

A recent, extremely non-scientific survey I did confirms this:

as do many, many tweets from industry experts:

(and me):

Along with data cleaning, what’s become more clear as the hype cycle continues its way to productivity is that data tooling and being able to put models into production has become even more important than being able to build ML algorithms from scratch on a single machine, particularly with the explosion of the availability of cloud resources.

What is becoming clear is that, in the late stage of the hype cycle, data science is asymptotically moving closer to engineering, and the skills that data scientists need moving forward are less visualization and statistics-based, and more in line with traditional computer science curricula:

> Concepts like unit testing and continuous integration rapidly found its way into the jargon and the toolset commonly used by data scientist and numerical scientist working on ML engineering.

This has led to several things, including, first, the rise of the “machine learning engineer” job title as one that carries more prestige and higher earnings potential over the last 3-4 years,

Second, it’s led to a severe degree of job title deflation for data scientists, where because of the prestige of the data scientist job title, companies like Lyft will hire for data science job titles, but with data analyst skillsets, resulting in an even more skewed picture of what constitutes a “data science” job, and exactly how many of them are available to new entrants.

We, as senior practitioners, journalists, managers, industry conference speakers, HR managers writing job reqs, just have not done a great job keeping up with this extremely important piece of the puzzle.

Advice to new data scientists

So, in the spirit of continuing to offer advice to beginners, here is the email I would send to anyone else who asked me how to get into data science in 2019.

It’s a two-step plan:

```
1. Don't shoot for a data science job
2. Be prepared for most of your data scientist
work to not be data science. Adjust your
skillset for that.
```

These are really discouraging! But, let me elaborate on both of them, and hopefully they’ll seem less bleak.

Don’t get into data science

Given that there are 50, sometimes 100, sometimes 200 people for each junior role, don’t compete with those people. Don’t do a degree in data science, don’t do a bootcamp (as a side note, most of the bootcamps I’ve seen have been ineffective and crunched way too much information in a short number of time for candidates to effectively get a feel for data science, but that’s a different, separate blog post).

Don’t do what everyone else is doing, because it won’t differentiate you. You’re competing against a stacked, oversaturated industry and just making things harder for yourself. In that same PWC report that I referenced earlier, the number of data science positions is estimated at 50k. The number of data engineering postings is 500k. The number of data analysts is 125k.

It’s much easier to come into a data science and tech career through the “back door”, i.e. starting out as a junior developer, or in DevOps, project management, and, perhaps most relevant, as a data analyst, information manager, or similar, than it is to apply point-blank for the same 5 positions that everyone else is applying to. It will take longer, but at the same time as you’re working towards that data science job, you’re learning critical IT skills that will be important to you your entire career.

Like what, for example?

Learn the skills needed for data science today

Here are some problems you’ll actually have to deal with in the data space:

1. Creating Python packages
2. Putting R in production
3. Optimizing Spark jobs so they run more efficiently
4. Version controlling data
5. Making models and data reproducible
6. Version controlling SQL
7. Building and maintaining clean data in data lakes
8. Tooling for time series forecasting at scale

9. Scaling sharing of Jupyter notebooks
10. Thinking about systems for clean data
11. Lots of JSON

Although there are plenty of lovely statistical problems to think about in data science, none of these links deal with them. While tuning models, visualization, and analysis make up some component of your time as a data scientist, data science is and has always been primarily about getting clean data in a single place to be used for interpolation.

What do all these blog posts have in common? Good generalist engineering skills with a data background.

How do you prepare to solve these problems and be ready for the workforce? Learn these three skills, which all are foundational, and build on each other, from easiest, to hardest.

The really key thing about all of these skills is that they are also fundamental and critical to software development outside of data science, meaning that, in case you can't find a data science job, you can transition quickly to software development, or devops. I consider this flexibility just as important as training for a specific data-related gig.

1. Learn SQL.

First, I recommend learning SQL for everyone, regardless of whether their ambition is to be a data engineer, ML expert, or AI superwhiz.

SQL is not sexy, and it's not a solution to the list of problems I just listed. But for all intents and purposes, in order to understand how to access data, chances are extremely high that you'll come across a database somewhere that will require you to write some SQL queries and get an answer.

SQL is so great and so popular that even NoSQL and key-value store solutions are reimplementing it. Just check out Presto, Athena, which is powered by Presto, BigQuery, KSQL, Pandas, Spark, and many, many more. If you find yourself overwhelmed by the sheer amount of data tooling out there, chances are, there is a SQL for you. And, once you understand the SQL paradigm, chances are it'll be easier to understand other query languages, which opens up an entire new universe.

The next step, after you learn SQL well, is to understand a bit about how databases work and why so you can learn to optimize your queries. You're not going to be a database developer, but again, a lot of the concepts will carry over into your other programming life.

2.Learn a programming language extremely well and learn programming concepts.

Isn't SQL a programming language? It is, but it's declarative. You specify the outputs you want (i.e. which columns from your table you want to pull), but not how those columns are actually returned to you. SQL abstracts a lot of what's going on under the covers of a database.

You want a procedural language, one where you have to specify how and where the data is selected from. Most modern languages are procedural: Java, Python, Scala, R, Go, etc.

There is a lot of debate about which language to choose for data science, and I won't prescribe one for any given circumstance, except to say that in my career, Python has served me extremely well. It's easy enough to get started in as a beginner, is (arguably) the most popular programming language in the data sphere right now, and can be used for a number of different things, from putting together a model in scikit learn, to accessing the AWS API, to building a web application, to cleaning data, to creating deep learning models. For statistical depth, R for sure has Python beat.

But again, my advice is not to go for statistical depth, but general programming breadth.

There are some tasks Python is not great for: large-scale applications,packaging dependencies, and some specific numerical work, particularly time series and a bunch of features that come in R out of the box, similar to what statsmodels offers, but at a much more granular level.

If you don't pick Python, that's fine. But you should pick a language that will, again, allow you to be flexible outside of a data science environment, for example if your first job is as a data analyst, a QA analyst, a junior in DevOps, or any number of different ways that will allow you to get your foot in the door.

Once you pick your language and figure out how to use it, start to learn the paradigms behind it and how it relates to the computer science ecosystem at large.

How do you do OOP in your language? What does OOP even mean? How do you optimize your code? How do your language's dependencies work? How do you package your code in your given language, how do you do version control, continuous integration, how do you deploy model artifacts? Where is your language's community, where are its meetups?

Get to know your language well. Get to know its warts and its best parts. Build something fun in your language.

And then, when you're at a good point and feel confident enough to continue, but only then, learn your second language. It will teach you so much more about the wider world of language design, algorithms, and patterns.

3.Learn how to work in the cloud.

Now that you know how to program, it's time to take those skills and theory to the cloud.

The cloud is everywhere these days, and chances are you will have to work in the cloud in one of your next jobs. It's much easier to come in with a head start, particularly as more and more machine learning paradigms move to cloud vendors these days (SageMaker, Cloud AI, and Azure Machine Learning), there are off-the shelf templates to implement algorithms, and more of your company's data starts to be stored there.

Chances are you'll be working with AWS, the industry leader, but more and more places are adopting Google Cloud, and many more conservative, traditional businesses that already do business with Microsoft will have Azure. I recommend doing a survey of all three, and then picking one you're most interested in to get comfortable with. Cloud design paradigms are similar in that you have to understand how to glue services together, how to logically partition your part of the cloud away from other servers on the cloud, and how to work with lots and lots of JSON.

A cool thing is that all three vendors now offer certifications of their offerings. I'm generally not a big believer in certifications as a signifier of knowledge, but what you'll learn going through the certs is how the cloud works, which is another component of engineering: networks.

Learn about the three general offerings and build something fun in the cloud before you get to your next job.

The piece that is largely missing here, is of course, the “soft skills” - knowing what to build when, knowing how to communicate in the workplace, understanding what other people want. This piece is just as important as the technical, and there are lots of blog posts devoted to it, but for the sake of keeping this post under 4,000 words, I’ll skip it and let you do your own reading.

Last Steps

Now, *deep breath* I think you’re ready.

If all or any of that sounds interesting to you, you’re ready to be a data scientist, or machine learning engineer, or cloud expert, or AI wizard in 2019.

Remember that the ultimate goal in following this advice is to beat the hordes doing data science degrees, bootcamps, and working through tutorials.

You want to get your foot in the door, get a data-adjacent position, and move towards whatever your dream job is, while finding out as much as you can about the tech industry in general.

My last piece of general advice is a small pep talk:

This stuff is really hard for everyone, and there are a million things it seems like you have to know. Don’t get discouraged.

Don’t get paralysis by analysis. Pick a small piece of something and start there. Do something small. Learn something small, build something small. Tell other people. Remember that your first job in data science will probably not be as a data scientist.

One of my favorite books ever is Bird by Bird, by Anne Lamott.

It’s about how to write. The story she tells in the book, of how the book got its title, is a book report her brother had to write.

“Thirty years ago my older brother, who was ten years old at the time, was trying to get a report on birds written that he’d had three months to write. [It] was due the next day. We were out at our family cabin in Bolinas, and he was at the kitchen table close to tears, surrounded by binder paper and pencils and unopened books on birds, immobilized by the hugeness of the task ahead. Then my father sat down beside him, put his arm around my brother’s shoulder, and said. ‘Bird by bird, buddy. Just take it bird by bird.’”

And he got it done.

Don’t let the hype overwhelm you. Don’t get clouded by the buzzwords or the images of hipsters with MacBooks. Concentrate on a single bird, and build from there.

Go forth, and good luck!