What They Don't Tell You About Data Science 1: You Are a Software Engineer First

Posted by nadbor Dec 5th, 2017 9:18 pm

This is the first of a series of posts about things I wish someone had told me when I was first considering a career in data science. Part 2

A popular meme places data science at the intersection of hacking, statistics and domain knowledge. It isn’t exactly untrue but it may give an aspiring data scientist the mistaken impression that those three areas are equally important. They’re not.

I’m leaving domain knowledge out of this discussion because, while it’s absolutely necessary to have it to get anything done at all, it usually doesn’t have to be very deep and you’re almost always expected to pick it up on the job.

First of all, hacking is something that we do every day while we can go months or years without touching any statistics. Of course, statistics and probability are baked into much of the software we use but we no more need to think about them daily than a pilot needs to think about the equations of aerodynamics.

Secondly, on those rare occasions when you do come up with some brilliant probabilistic model or business insight, it will still have to be implemented as a piece of software before it creates any value. And make no mistake - it will be implemented by you or not at all. A theoretical data scientist who dictates equations to engineers for implementation is not - and will never be - a thing.

Data science is a subset of software engineering. You design and implement software. It's a peculiar kind of software and the design process is unusual but ultimately this is what you do. It is imperative that you get good at it.

Your colleagues will cut you a lot of slack with respect to programming on account of you bringing other skillsets to the table. As a result it is entirely possible for someone to be doing data science for years without picking up good engineering practices and modern technologies. Don't let this happen to you.

The purely technological part of data science - installing things, getting things in and out of databases, version control, db and cluster administration etc. - may seem like a boring chore to you (I know it did to me) - best left to vanilla engineers who are into this stuff. This type of thinking is a mistake. Becoming better at engineering will:

- cause you to spend less time on the routine data preparation tasks and let you focus on models (have the data cleaned and ready in a week rather than a month)
- allow you to iterate more rapidly, test more ideas in the same amount of time
- give you access to new datasets (data too big for your laptop? No problem, you can spin up a spark cluster and munge it in minutes)
- … and modeling techniques (new crazy model described on arXiv? Or a cutting edge library released? You will skim the docs and get it working in no time.)
- make it more likely that your code will end up in production (because you write it production-ready)
- open doors to more interesting jobs

That doesn't mean that you have to be an expert coder to start working as a data scientist. You don't even have to be an expert coder to start working as a coder. But you do need to have the basics and be willing to learn.

A trained engineer with no knowledge of statistics is one online course away from being able to perform a majority of data science jobs. A trained statistician with no tech skills won't be able to do any data science at all. They may still be a useful person to have around (as a data analyst maybe) but would be completely unable to do any data science on their own.

Why do we even have data scientists then? Why aren't vanilla engineers taking all the data science jobs?

Data science may not require much in terms of hard maths/stats knowledge but it does require that you're interested in data and models. And most engineers simply aren't. The good ones are too busy and too successful as it is to put any serious effort into learning something else. And the mediocre simply lack the kind of curiosity that makes someone excited about reinforcement learning or tweaking a shoe reccomender.

Moreover, *there is* a breed of superstar software engineers doing drive-by data science. I know a few engineers each of whom can run circles around your average data scientist. They can read all the latest papers on a given AI/ML topic, then implement, test and productionise a state of the art recommender/classifier/whatever - all without breaking a sweat - and then move on to non-data related projects where they can make more impact. One well known example of such a person is Erik Bernhardsson - the author of Annoy and Luigi.

These people don't call themselves 'data scientists' because they don't have to - they already work wherever they want, on whatever projects they want, making lots of money - they don't need the pretense. No, 'data scientist' is a term invented so all the failed scientists - the bored particle physicists and disenchanted neurobiologists - can make themselves look useful to employers.

There is no denying, that

> *“I’m a data scientist with a strong academic background”*

Does sound more employable than

> *“I’m have wasted 10 best years of my life on theoretical physics but I also took a Python course online, can I have jobs now plz”*

I’m being facetious here but of course I do think a smart science grads can be productive data scientists. And they will become immensely more productive if they make sure to steer away from ‘academic with a python course’ and towards ‘software professional who can also do advanced maths’.