Harvard Data Science Review • 1.2

What Does It Take to be a Successful Data Scientist?

Michael R. Berthold

Published on: Nov 01, 2019

Updated on: Nov 08, 2019

DOI: 10.1162/99608f92.e0eaabfc

ABSTRACT

Given recent claims that data science can be fully automated or made accessible to nondata scientists through easy-to-use tools, I describe different types of data science roles within an organization. I then provide a view on the required skill sets of successful data scientists and how they can be obtained, concluding that data science requires both a profound understanding of the underlying methods as well as exhaustive experience gained from real-world data science projects. Despite some easy wins in specific areas using automation or easy-to-use tools, successful data science projects still require education and training.

Keywords: data science, analytics, practitioner, education, insights, discovery

Data scientists are rare, that's not new. Lots of educational programs are popping up to train more to meet the demand. Universities are creating data science departments, centers, or even entire divisions and schools. Online universities offer courses left and right. Even commercial providers present data science certifications in just a few weeks or months (or sometimes over a weekend).

But what is the right approach to earning your stripes and calling yourself a successful data scientist?

1. Theory or Practice?

At some point in the past years, there was hope that a single, simple solution could enable everybody to become a data scientist—if we just gave them the right tools. But similar to a doctor needing to know how the human body functions, a data scientist needs to understand the state-of-the-art models and algorithms to be able to make educated choices and recommendations. We are, after all, talking about data scientists here, not just users of black boxes that were designed by successful data scientists. A doctor isn't turning us into a doctor by telling us what medicine to take either.

But is a theoretical education sufficient? My answer here is no. Data science is as much about knowing the tool as it is about having experience applying it to real-world problems, about having that 'gut feeling' that raises your eyebrows when the results are suspiciously positive (or just weird). I have seen this countless times with students in our data science classes. Early on, when aspiring data scientists start working on practical exercises, no matter how smart they are, they present results that are totally off. Once asked 'Are you sure this makes sense?' they realize and begin to question their results, but this is learned behavior. These are often things as simple as questioning a 98% accuracy on a credit churn benchmark. Rather than wondering if this could point to a data pollution issue (the testing data containing some information about the outcome), the student proudly presents their 25% margin over their fellow students.

Becoming a successful data scientist requires both knowing the theory and having the experience to know how to get to, and when to trust, your results. The big question is can we teach 'real-world experience' during our courses as well.

2. Playing Is Training Enough?

Many wannabe data scientists claim they gained that real-world experience from working on online data analysis challenges—Kaggle or others. But that's only partly true because these challenges focus on a small, important, but fairly static part of the job. Some data scientist trainers have started building practical exercises, modeling some of those other real-world traps. KNIME, for instance, can be used to create data in addition to analyzing it. We use this for our own teaching courses to create real-world, look-alike databases about artificial customers with given distributions and dependencies to marital status, income, shopping behaviors, preferences, and other features. The data generation modules also allow us to inject outliers, anomalies, and other patterns that break standard analysis methods if not detected earlier. But this is still very similar to learning how to drive on a playground; it doesn't prepare you for driving in downtown Manhattan. Somehow, we can't prepare for real life in the privacy of our home or classroom.

Let's dive a bit deeper into what a data scientist actually does. Many articles have already covered the horizontal spread of activities: everything from data sourcing, blending, and transforming all the way to creating interactive, analytical applications or otherwise deploying models into production (and I am not even touching upon monitoring and continuously updating those production models). Lots of those online challenges ignore these surrounding activities and focus solely on the modeling part. But that's not the only problem. Let's also consider the vertical spread of tasks: *Why* do we need data science?

3. Data Science?

Data science is needed for different types of activities, and those require increasingly sophisticated skills and expertise from the data scientists, too.

Novice

This is the easiest setup that we can, at least partially, practice for in isolation. The problem and goal are well-defined, the data is mostly in good shape (and exists!), and the goal is to optimize a model to provide better outcomes. Examples are tasks such as predicting churn of customers and placing online advertisements. These are projects that essentially just support and confirm what the business stakeholder knows and put this knowledge into practice.

In order to tackle these types of problems, a data scientist needs to understand the ins and outs of models and algorithms and must be able to adjust the many little knobs to optimize performance. This is a task that can be somewhat automated, and experiments show that automation can often beat a not-so-experienced data scientist when it comes to model automation on standard tasks.

But even at this base level, our data scientist needs some experience to be able to ensure the goal is properly translated into a metric to be optimized as well as the ability to ensure the data isn't polluted. Classic examples of junior mistakes are using an optimization metric that ignores different costs for different types of errors or not realizing that the data used for training isn't unbiased (e.g., training your model on existing customers isn't a good basis for making recommendations about whether someone completely new may or may not be a good customer).

Apprentice

In reality, this job is usually much less well-defined. The business owner knows what they want to optimize, but they don't have a clear problem formulation, and way too often, they don't have the right data. Stereotypical statements for this setup are project descriptions of the type 'We have this data, please answer that question!' Examples can range from predicting machine failures ('We measure all those things, just tell us a day before the machine breaks.') to predicting customer satisfaction ('We send out a survey every month, just tell me who will cancel their contract tomorrow.').

Here our data scientist needs experience communicating with stakeholders and domain experts to identify the data to be collected and to find and train the right models to provide the answers to the right question. This also involves a lot of nontheoretical but practical work around data blending and transformation and ensuring proper model deployment and monitoring. In training, we can help the data scientist by providing blueprints for similar applications, but automation often fails because the data types aren't quite covered or the model optimization routines miss the mark just a bit. This is also an issue with the maturity of the field: We haven't yet encountered problems of all types, and many of these types of projects require a touch of creativity in their solution. An automated solution or a solution created by an inexperienced data scientist may seem to provide the right type of answer, but it will often be a long shot from providing the best possible answer.

Expert

The last type of data science activity is actually the truly interesting one. The goal is to create new insights that will then trigger new analytical activities and may completely change how things are done in the future. Setups of this kind are often initially poorly described ('I don't know what the solution looks like, but I'll know it when I see it!'), and the data scientist's job is to support this type of explorative hypothesis generation. In the past, we were restricted to simple, interactive data visualization environments, but today, an experienced data scientist can help to quickly try out different types of pattern discovery algorithms or predictive models and refine that setup given user feedback. Typically a lot of this feedback will be of the type 'We know this' or 'We don't care about that,' which will lead to continued refinement. The true breakthrough, however, is often initiated by comments of the type 'This is weird, I wonder ...,' triggering a new hypothesis about underlying dependencies.

For this type of activity, our data scientist needs experience dealing with open-ended—often research type—questions and the ability to quickly iterate over different types of analysis methods and models. It requires out-of-the-box thinking and the ability to move beyond an existing blueprint, and, of course, it requires learning from past experiences. In this type of scenario, often the type of insights generated yesterday aren't interesting today because the past insights did advance and change the knowledge of both the data scientist and the domain expert!

Presumably, this segmentation is a bit blurry; some apprentices will never aspire to become an expert, having job requirements that are well-defined and can be solved using standard techniques. And obviously, this will change over time with the data science field maturing. From what we see at KNIME (our built-in recommendation engine relies on anonymous tool usage information), the famous 90-9-1 doesn't quite apply here, but it is still only a fairly small percentage of our users (<10%) that regularly use nodes that we'd refer to as expert modules. The vast majority of our users start with one of the example workflows (which, in turn, rely on expert nodes) or use relatively standard modules themselves. This is also a view validated by conversations with our larger customers: Many of the users there rely on workflows as templates to start from instead of creating complex workflows from scratch.

4. Where To?

Data science, like computer science, requires a mix of theory and practice. Similar to how we now run software projects as part of most computer science curriculum, we should add practical projects to data science curricula. But like successful programmers, successful data scientists will require years of practical, real-world experience before being able to tackle real problems independently.

For some of the easier tasks, we can put junior data scientists to work or potentially even automate (parts of) the process. But for the truly interesting discipline of data science—the one that helps us advance our knowledge and understanding of how things work—we require true master data scientists with deep theoretical understanding, lots of experience, and the ability to think beyond the obvious.